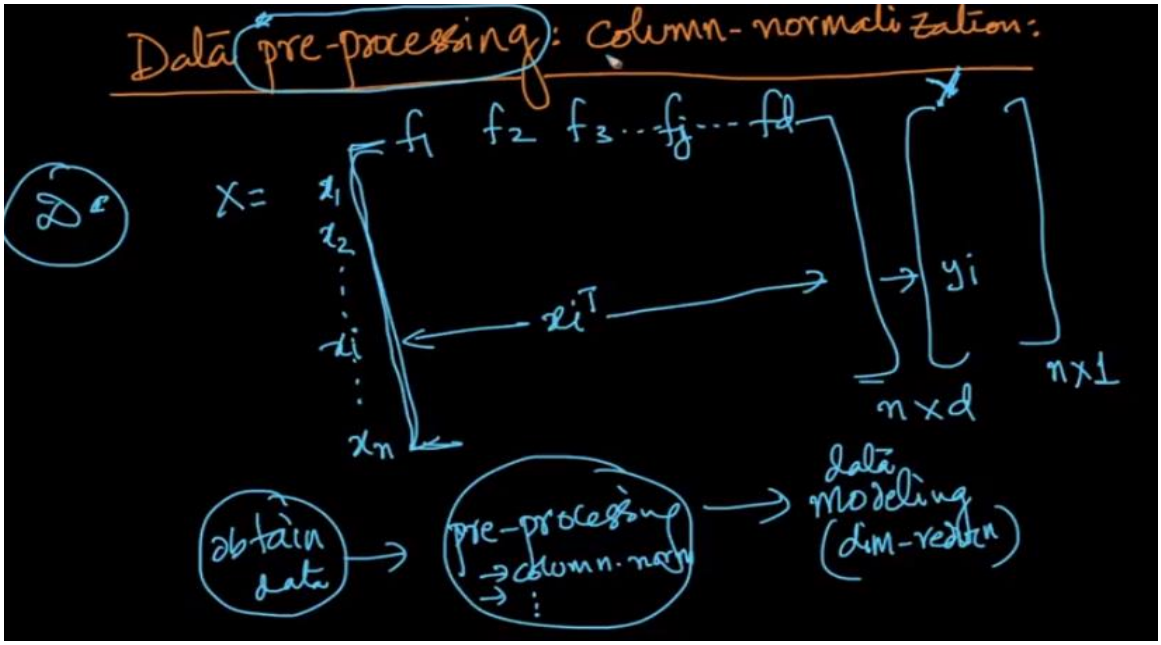
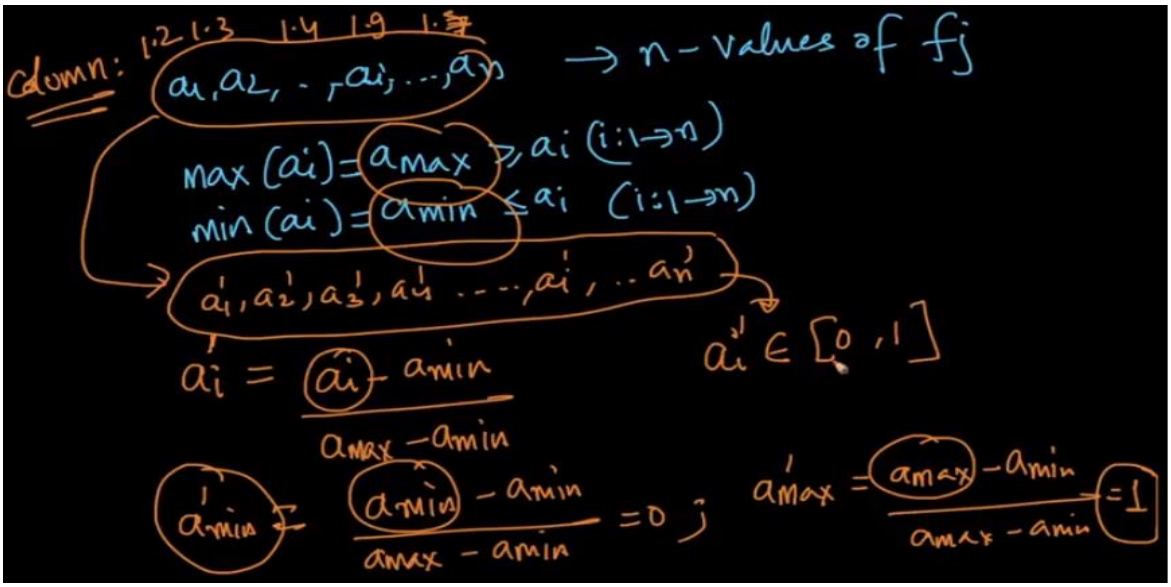
Dimensionality Reduction & Visualization

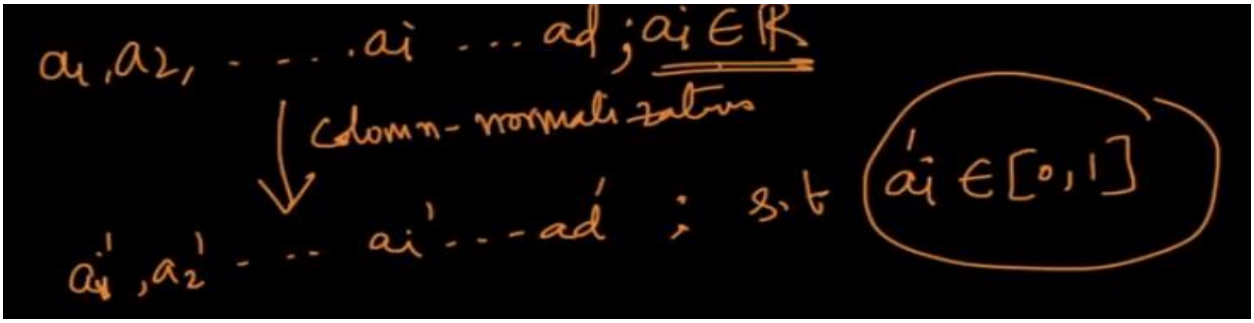
Column Normalization: Pre-process data for dimensionality reduction



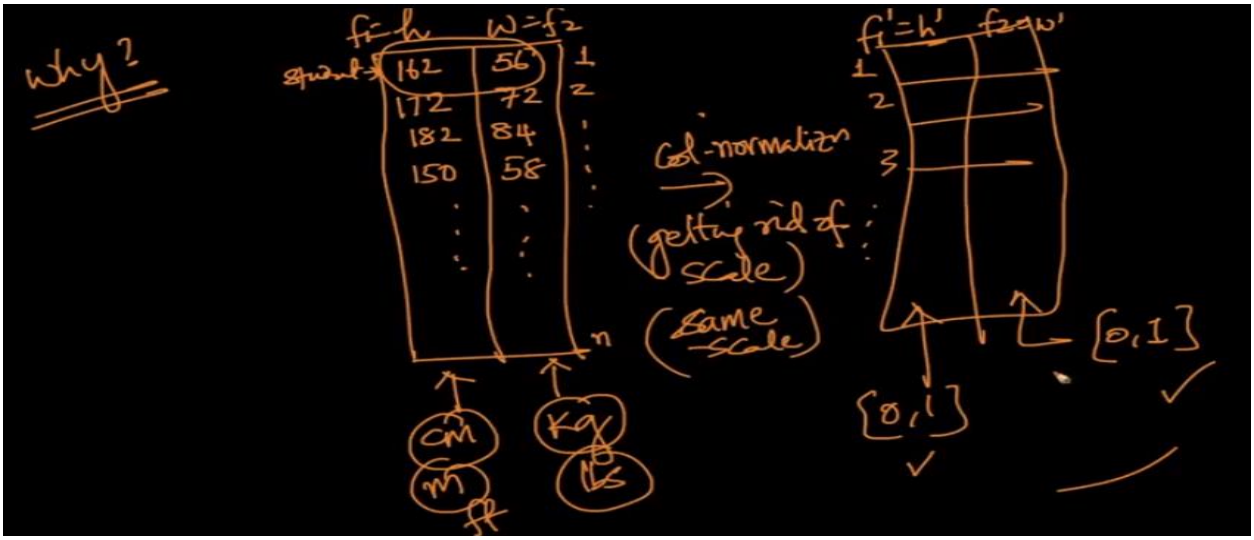
Column normalization technique – range [0,1] for feature[j]



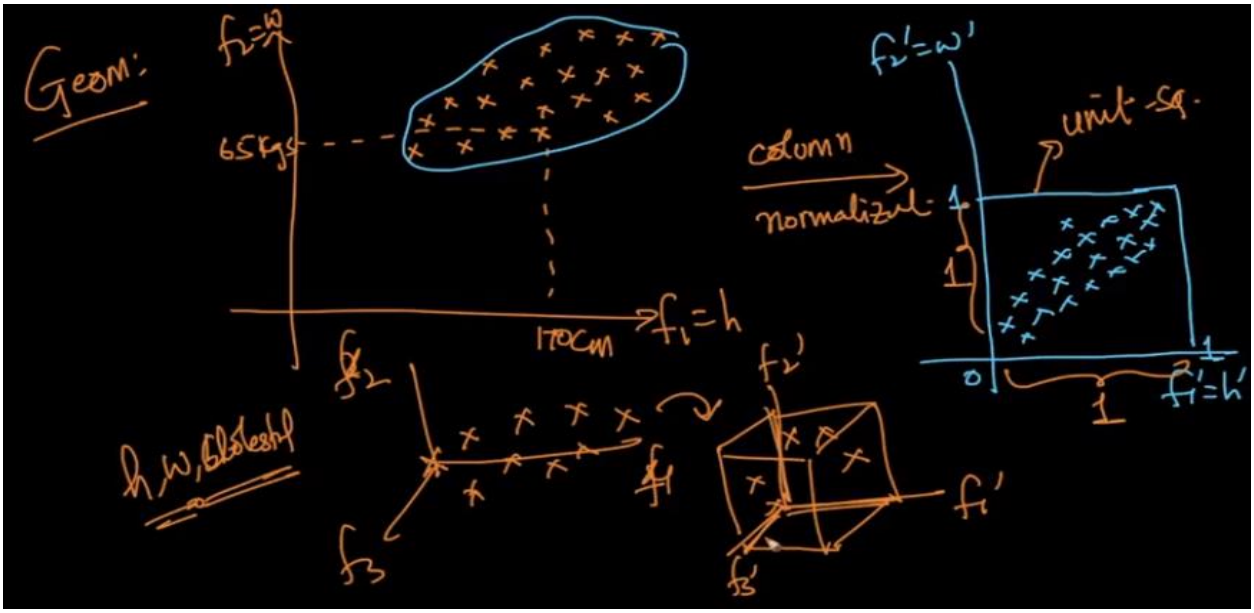
Feature transformed to new real values, ranging from [0,1] – do it for all the features.



Reasoning – Need of column normalization (Getting rid of feature scale)

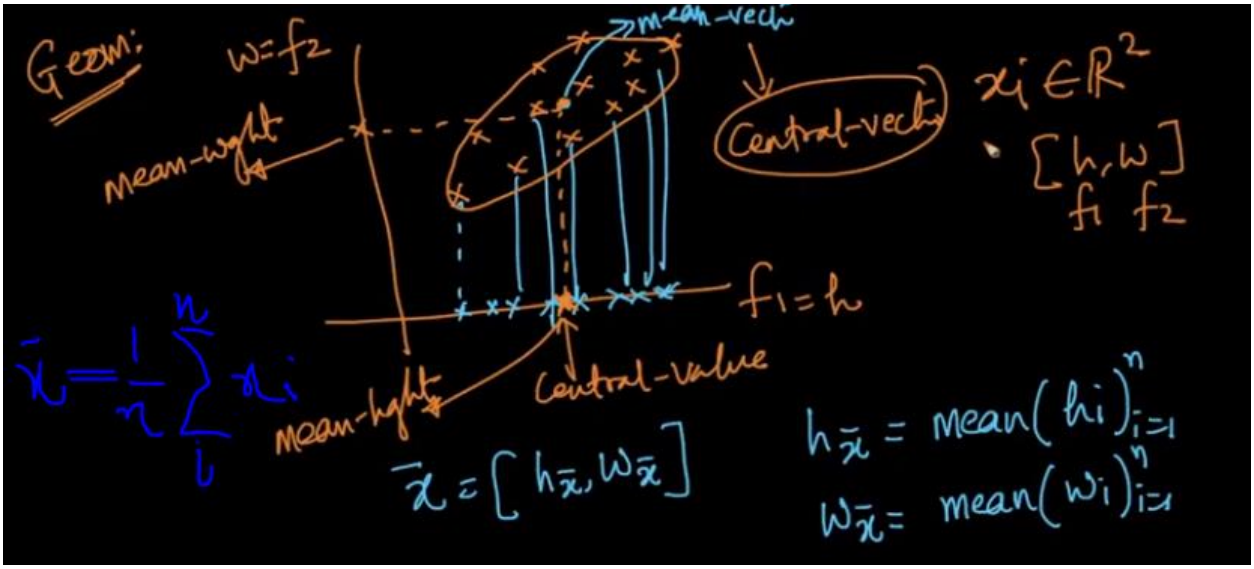


Geometric interpretation – for two features, data fits into unit square.

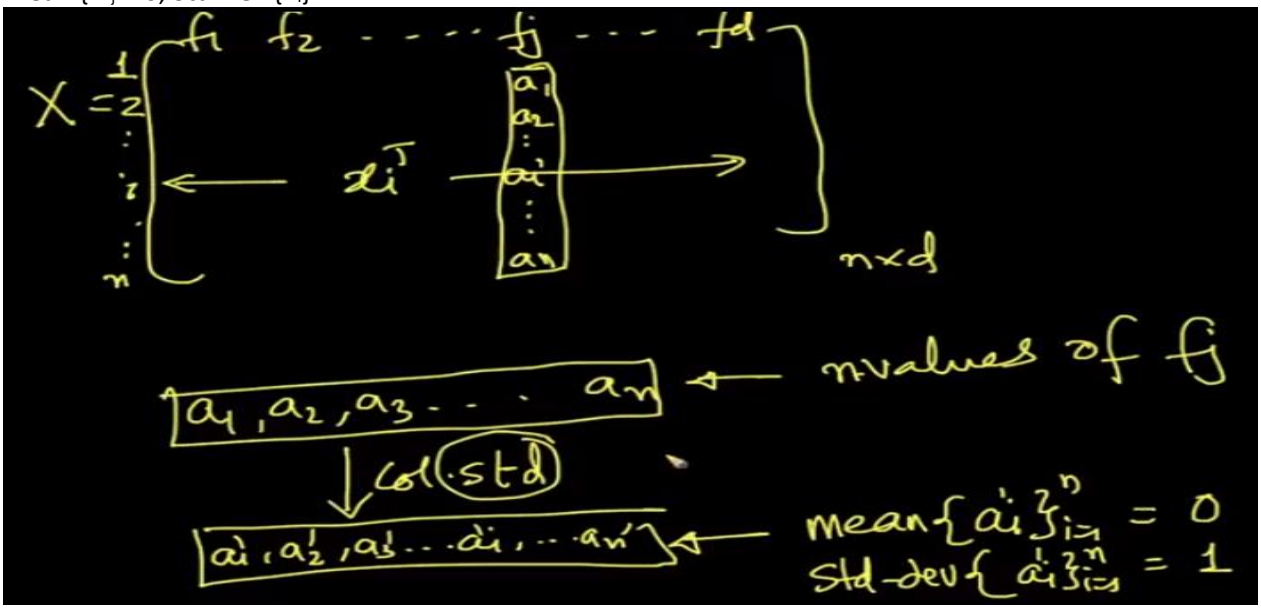


Terminologies: -

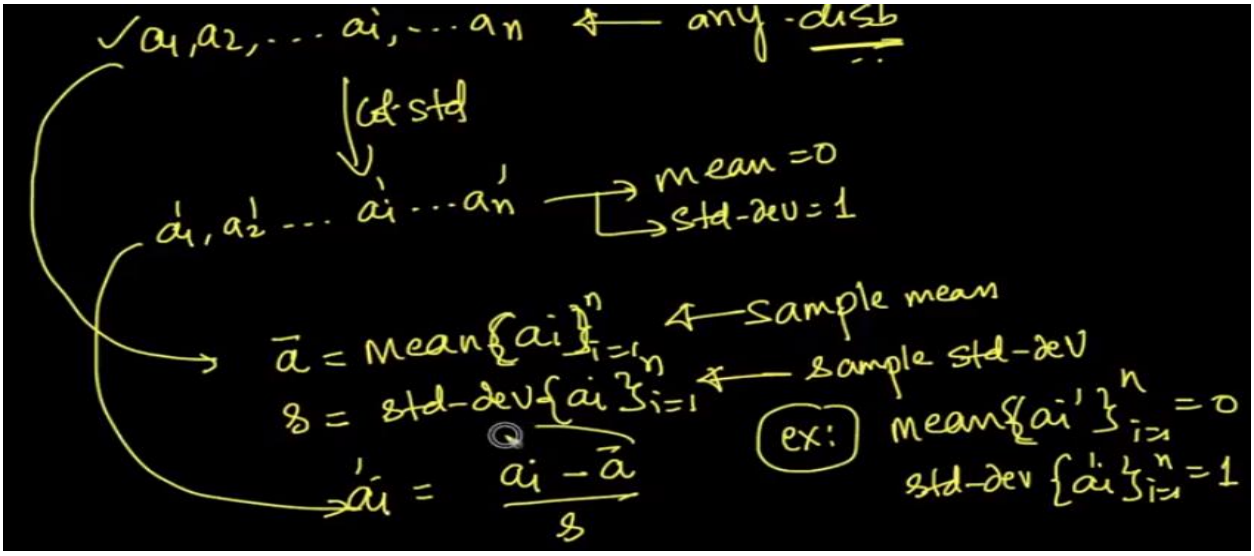
Mean Vector: Central value of the data



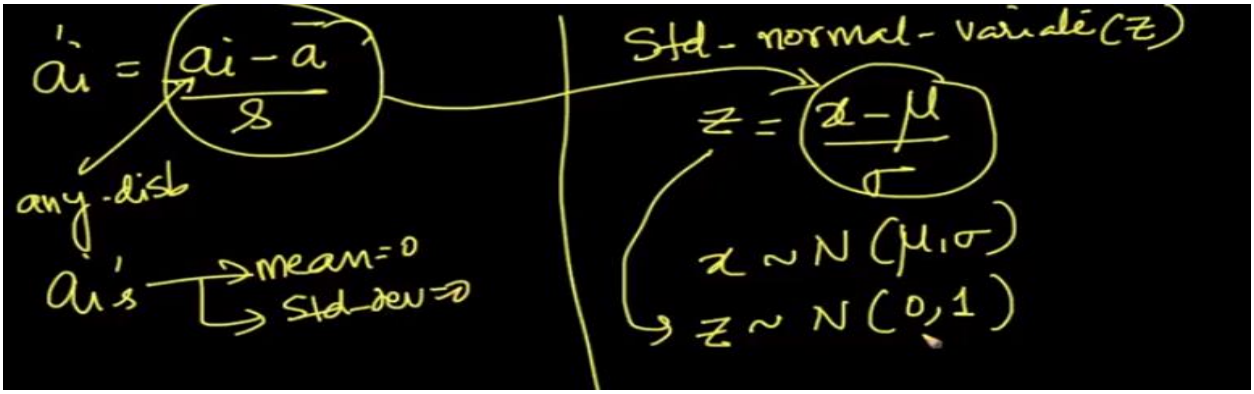
Column Standardization: - More often used than normalization
Mean {X_j} = 0, Std-Dev {X_i} = 1



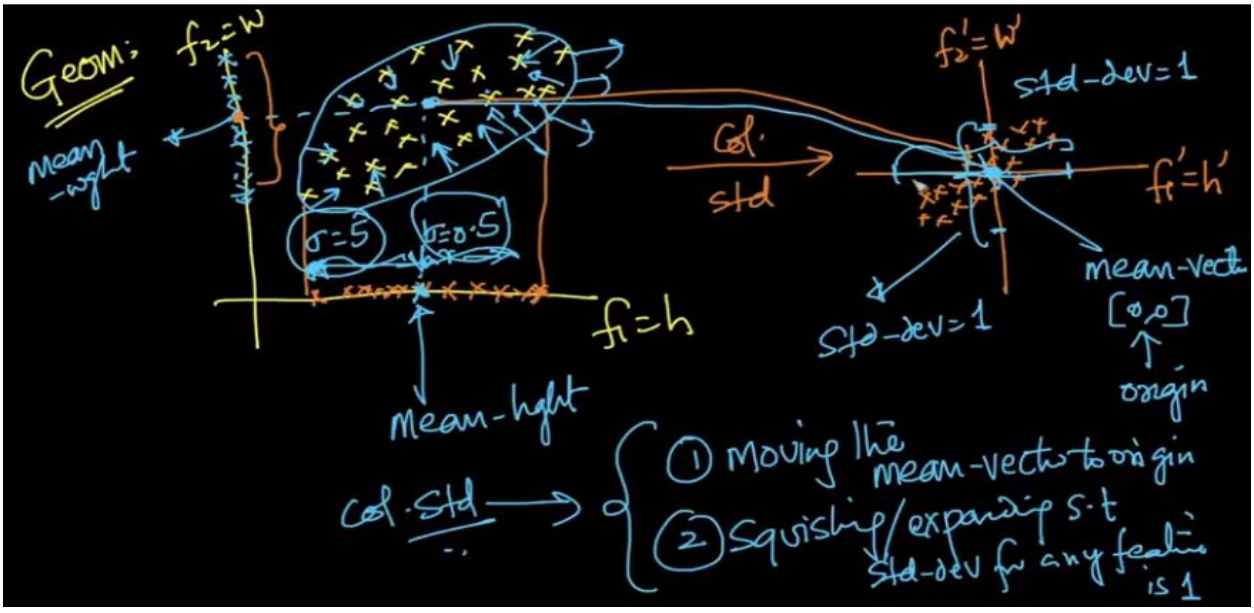
Transform a_i -> a_i' such that a_i' = (a_i – mean {a_i}) / std-dev {a_i} and a_i' has mean – 0 and std-dev = 1



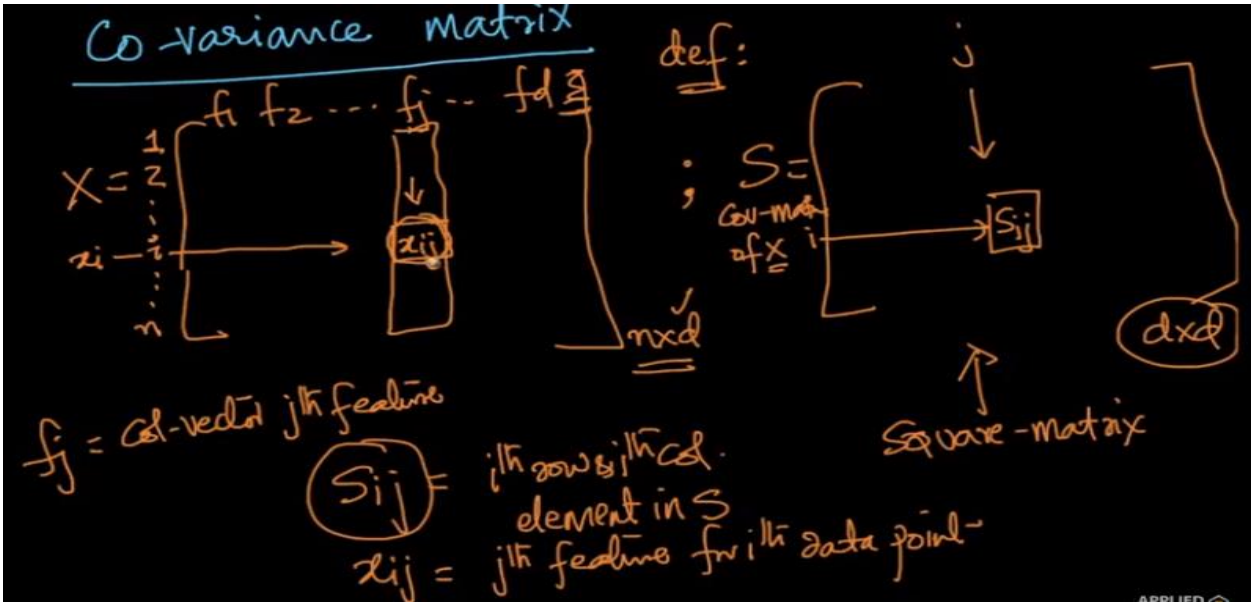
Comparison with standard normal variate

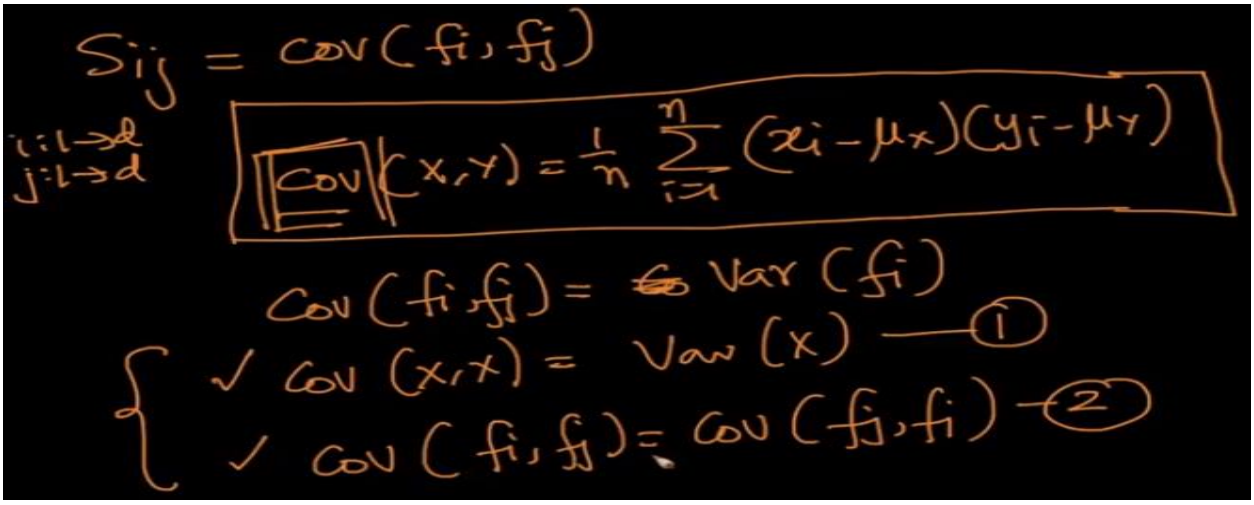


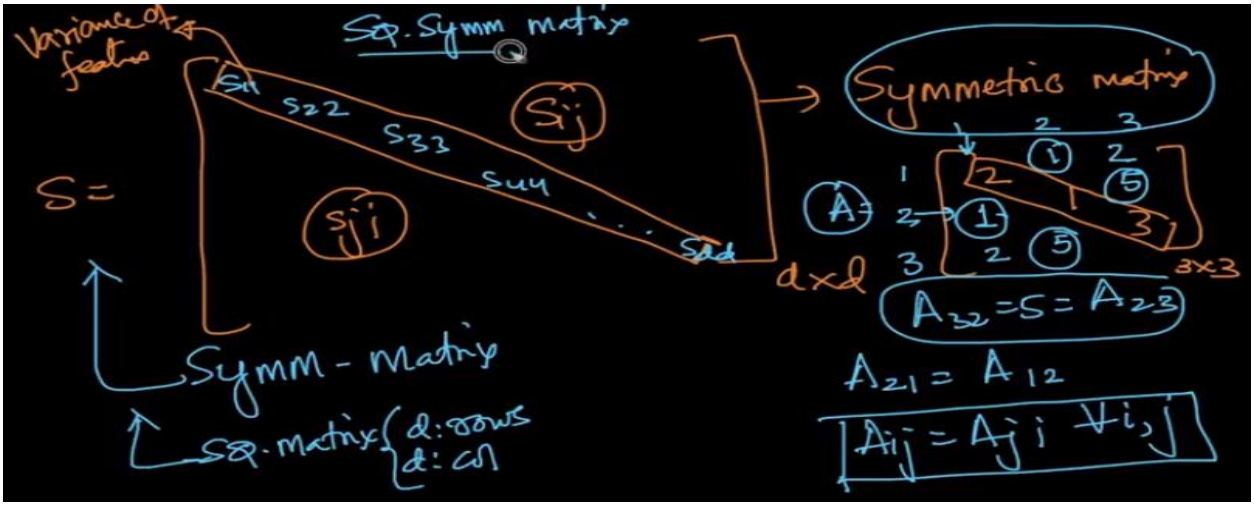
Geometric intuition of column standardization :-



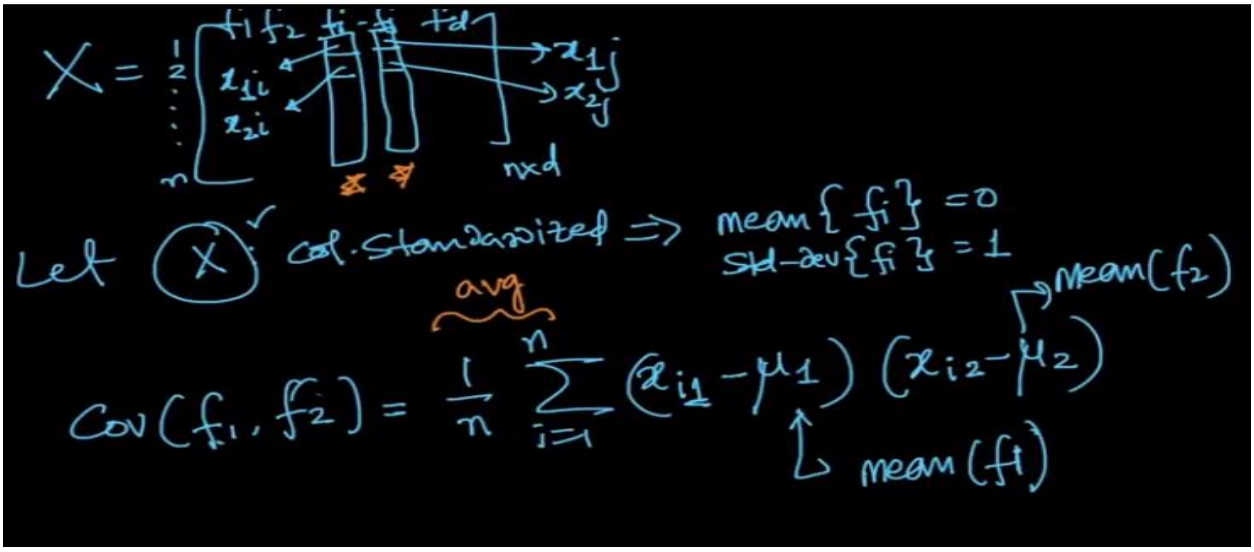
Covariance Matrix: - {symmetric and square matrix of d-dimensions}

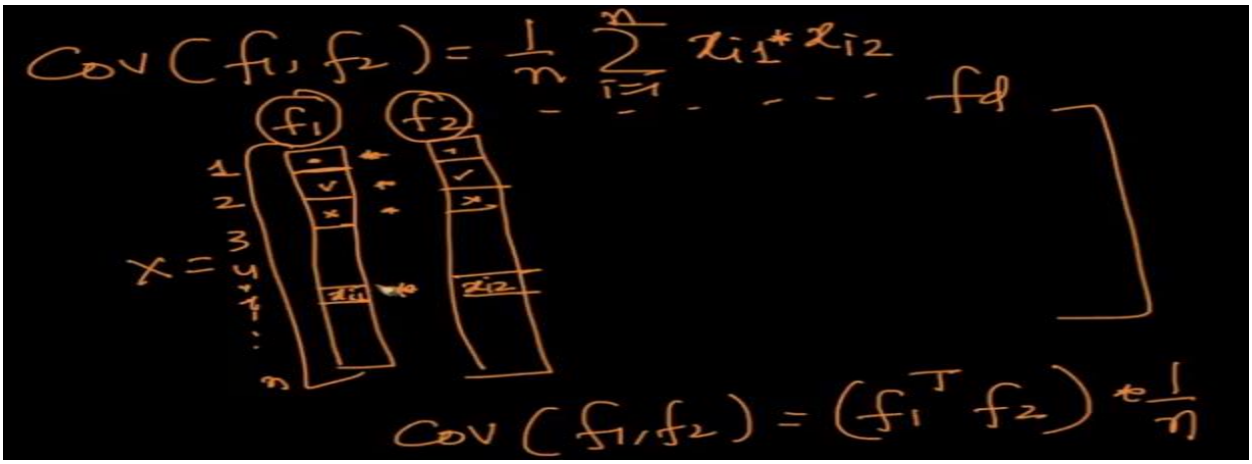


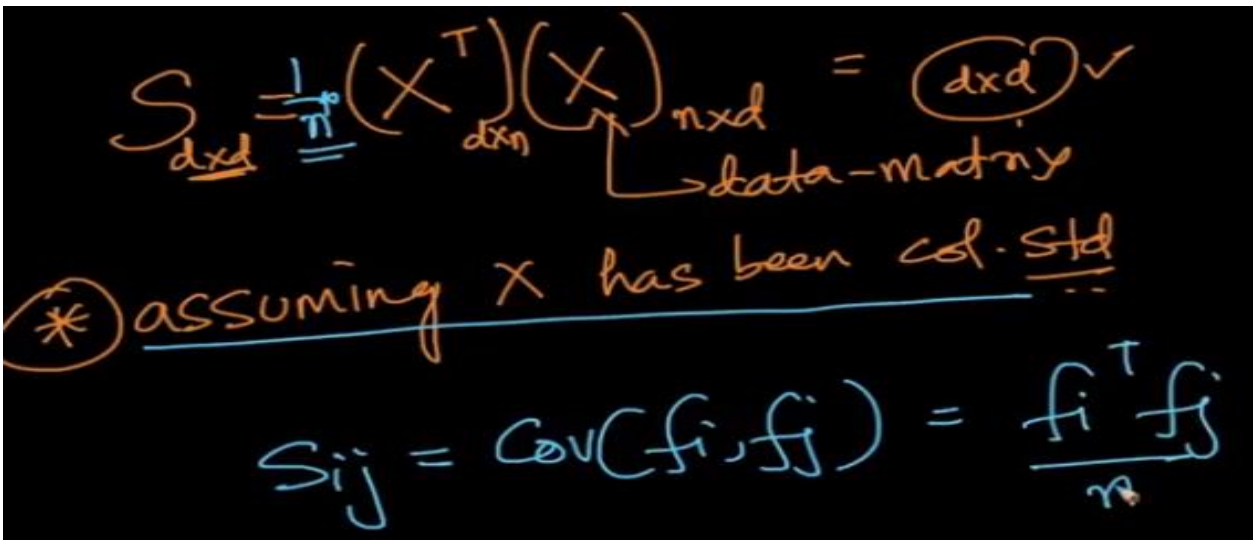




Covariance matrix of column standardize dataset.

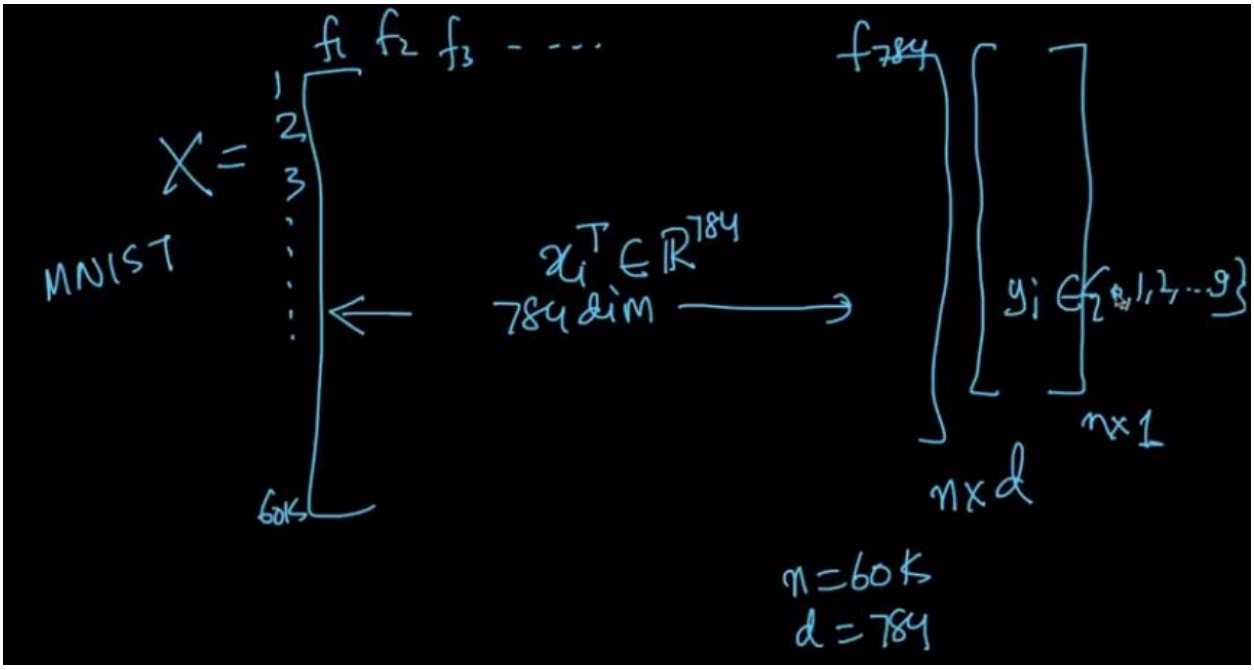






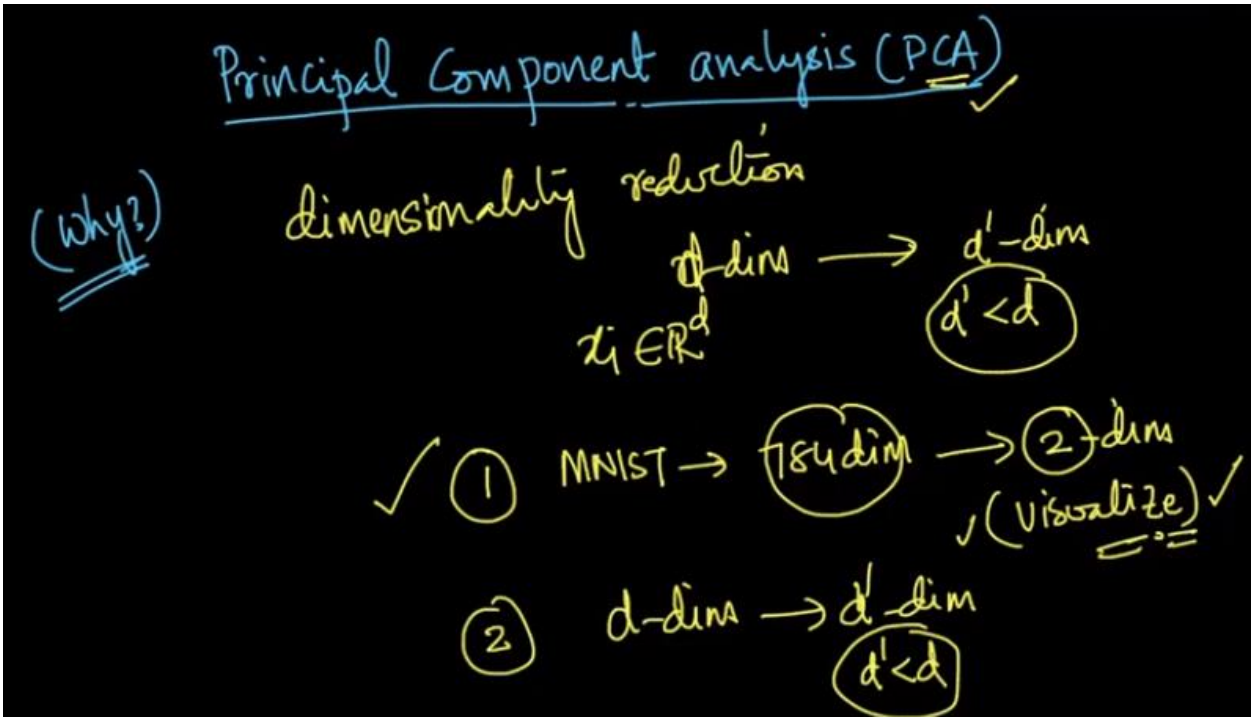
Principle Component Analysis: - (Example MNIST Dataset)

Data Representation: - N (60K) features flattened to 784 dimensions



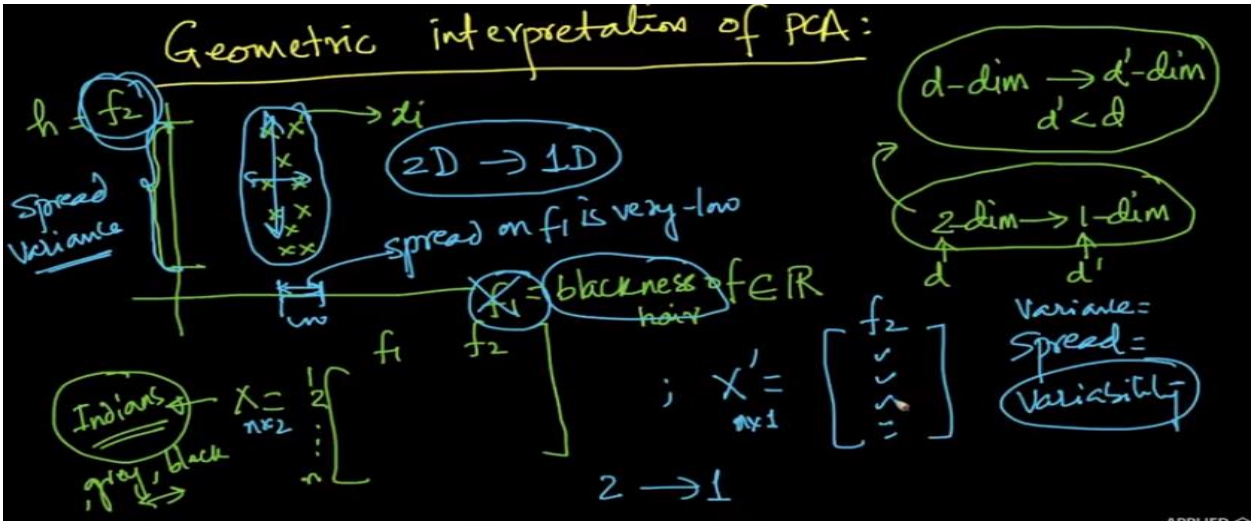
Difficult to visualize for classification task – where PCA comes into the picture.

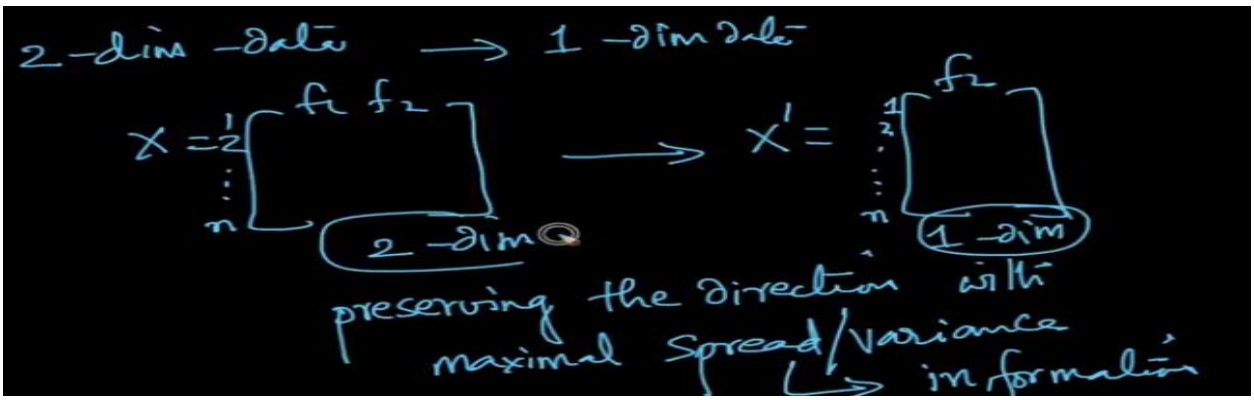
PCA – Method to reduce the dimensions of the dataset.



Geometric Interpretation of PCA: -

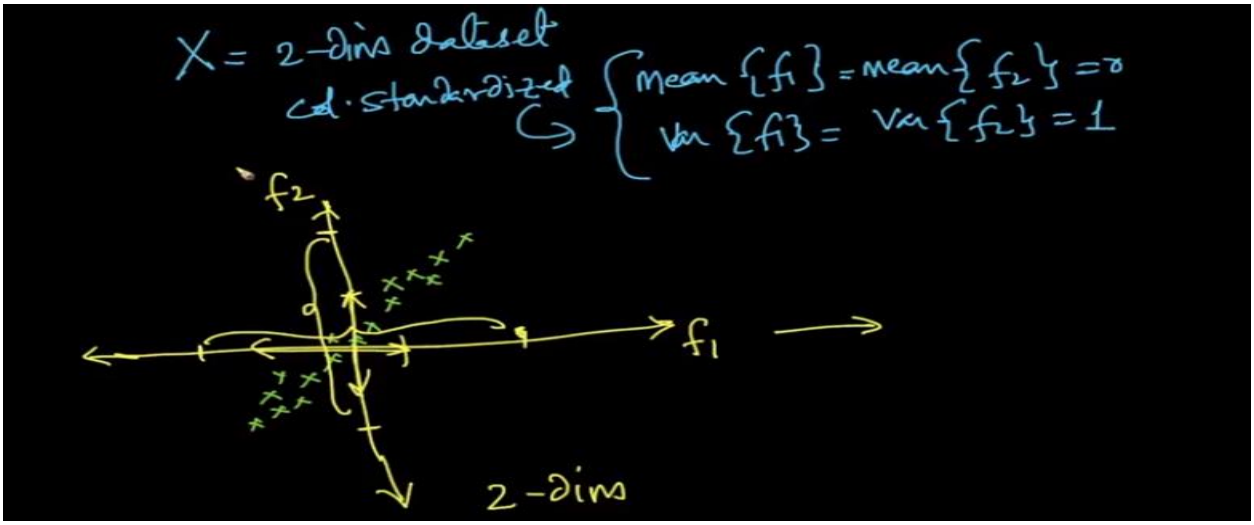
Dropping the feature which has less variance (drop by intuition but has problem of losing information)



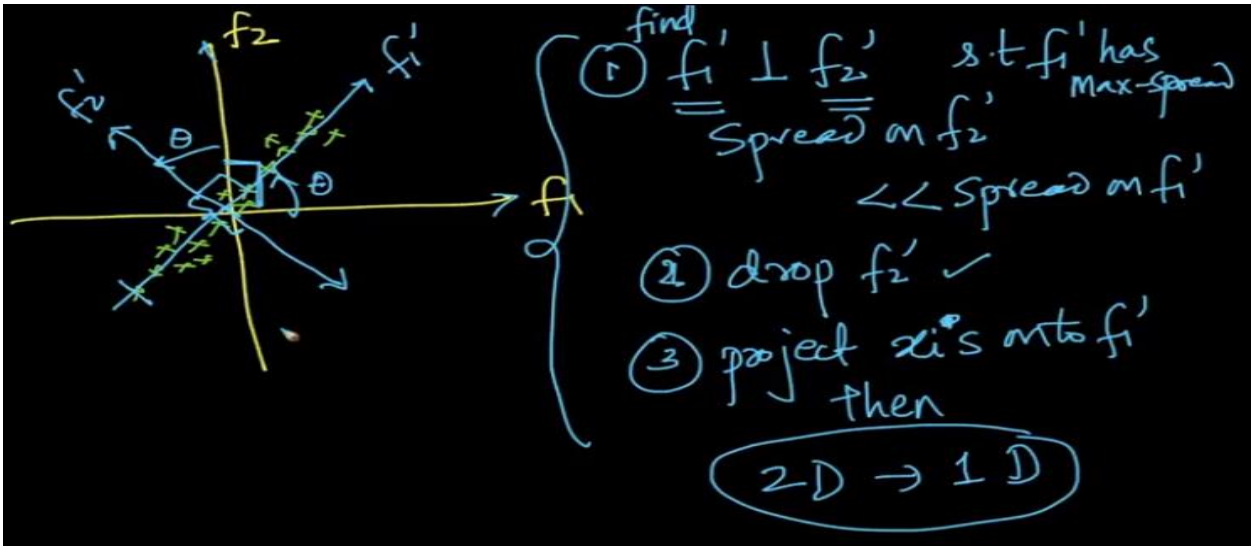


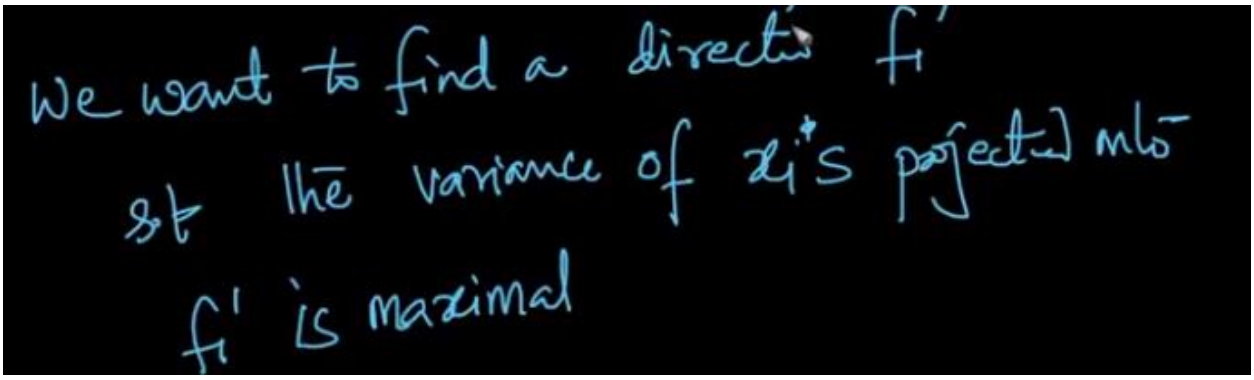
Feature f2 has most information and f1 has less. So f1 can be dropped directly.

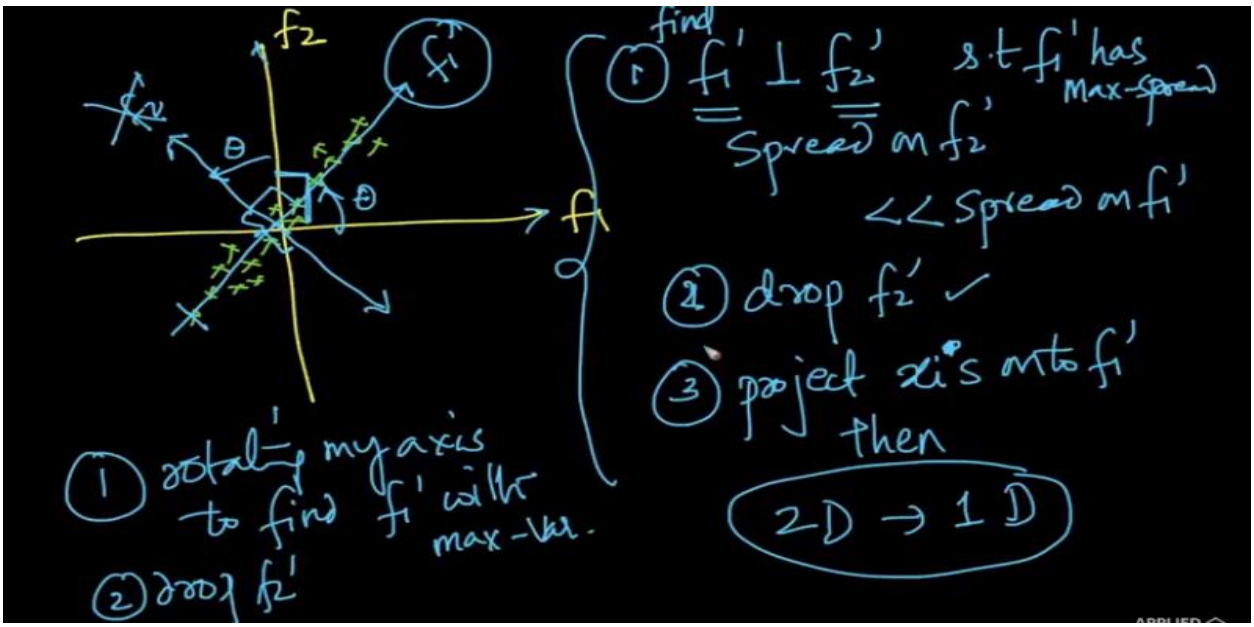
Feature have enough spread in both the directions. Dropping a feature based on intuition is not good.



Find method for maximizing the variance of data in one direction.

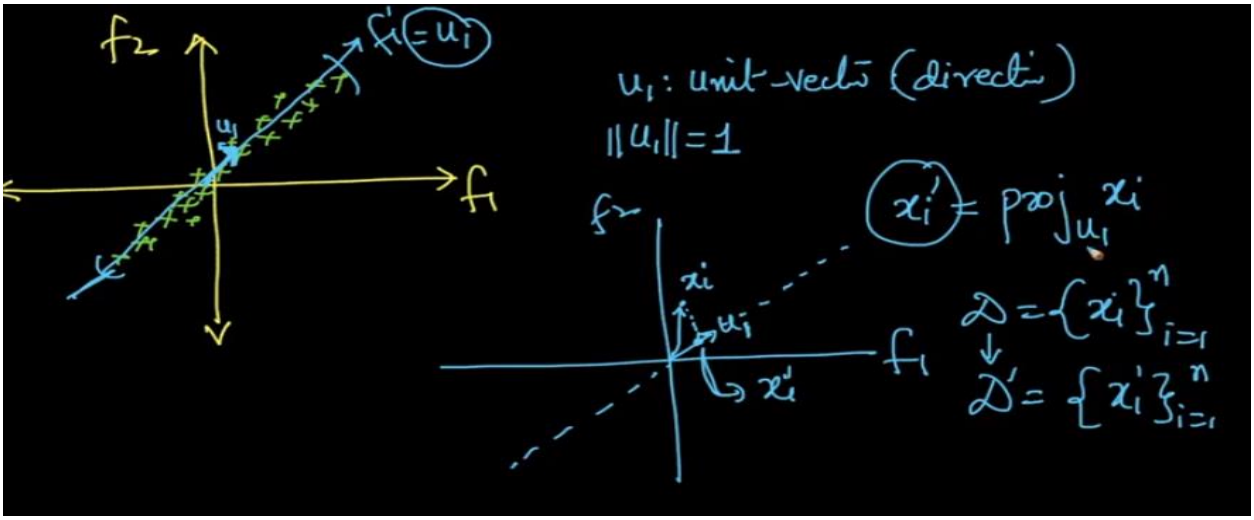


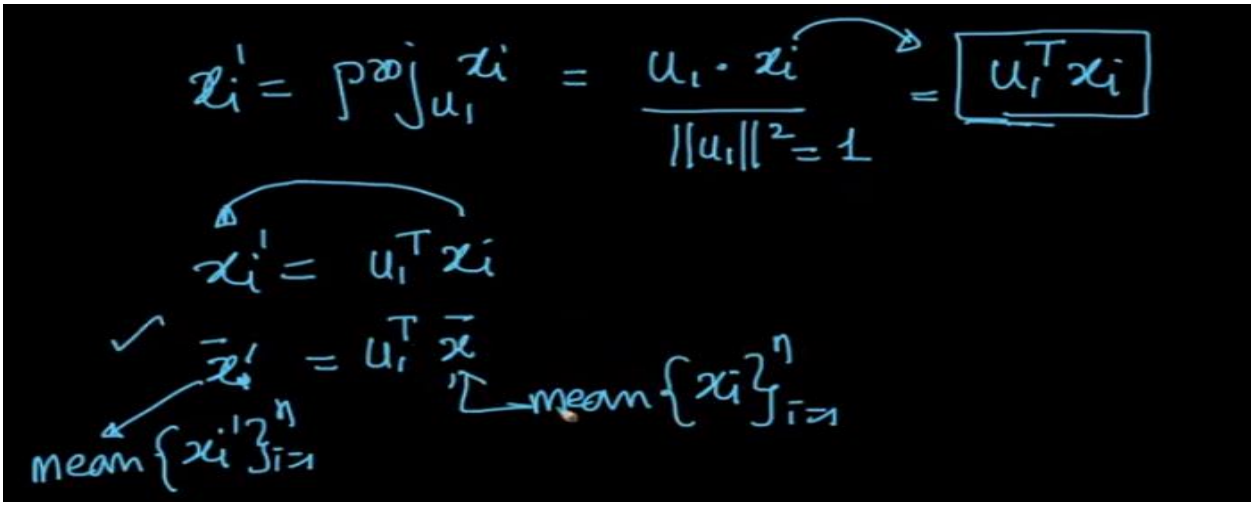


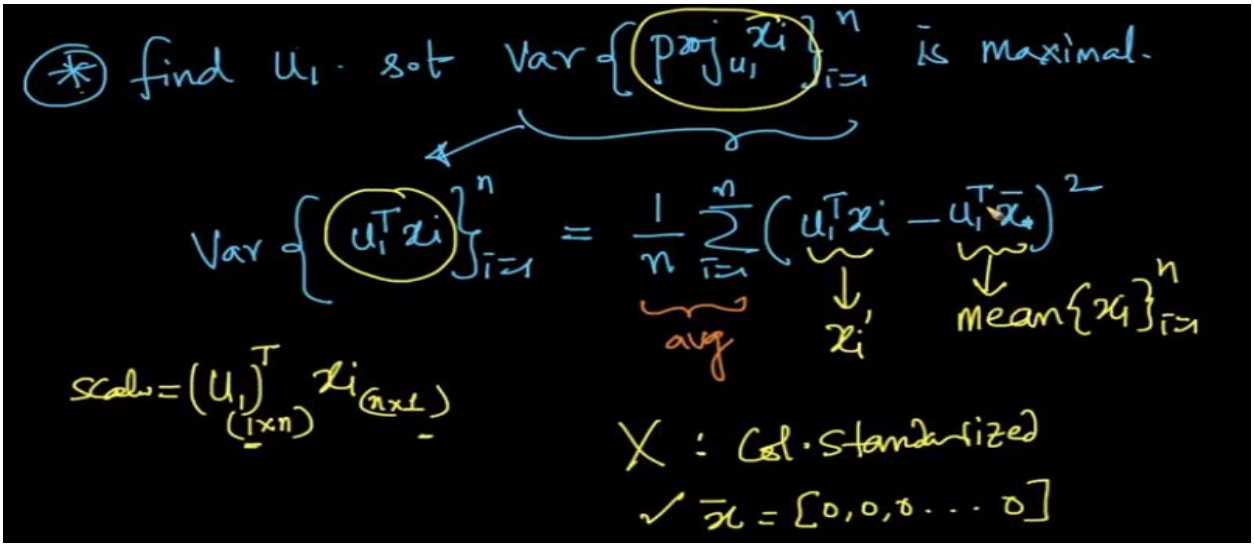


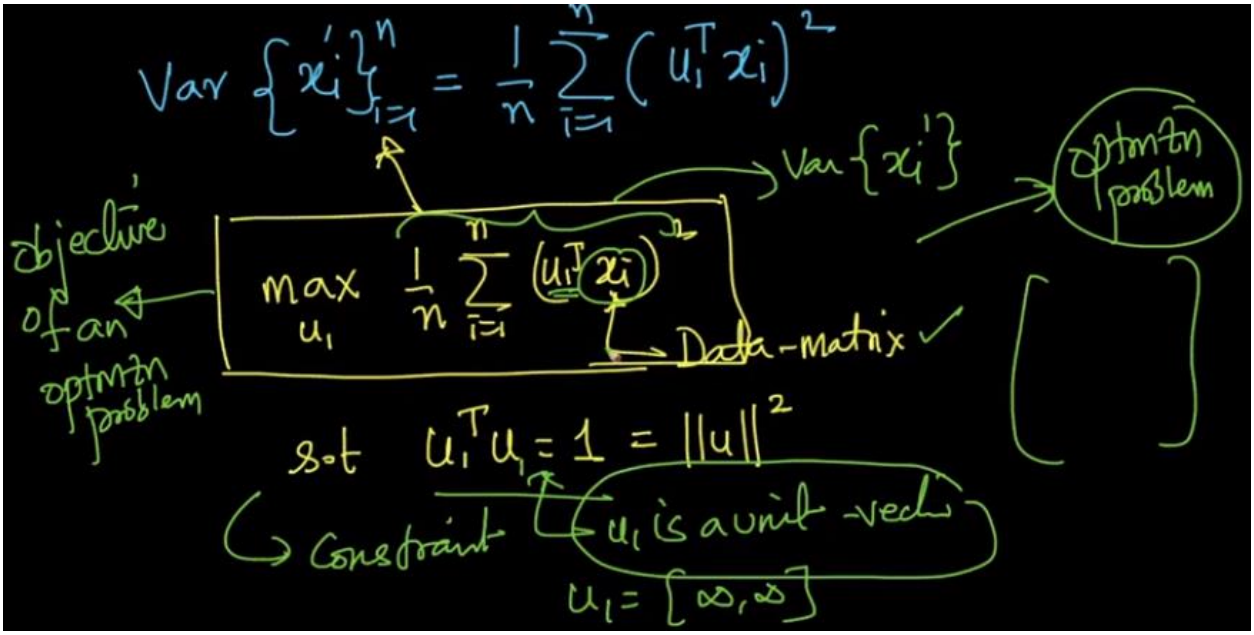
Mathematical Interpretation: - as optimization problem

(Maximizing Variance): Find unit vector U_1 such that projection of X_i on U_1 is maximized.

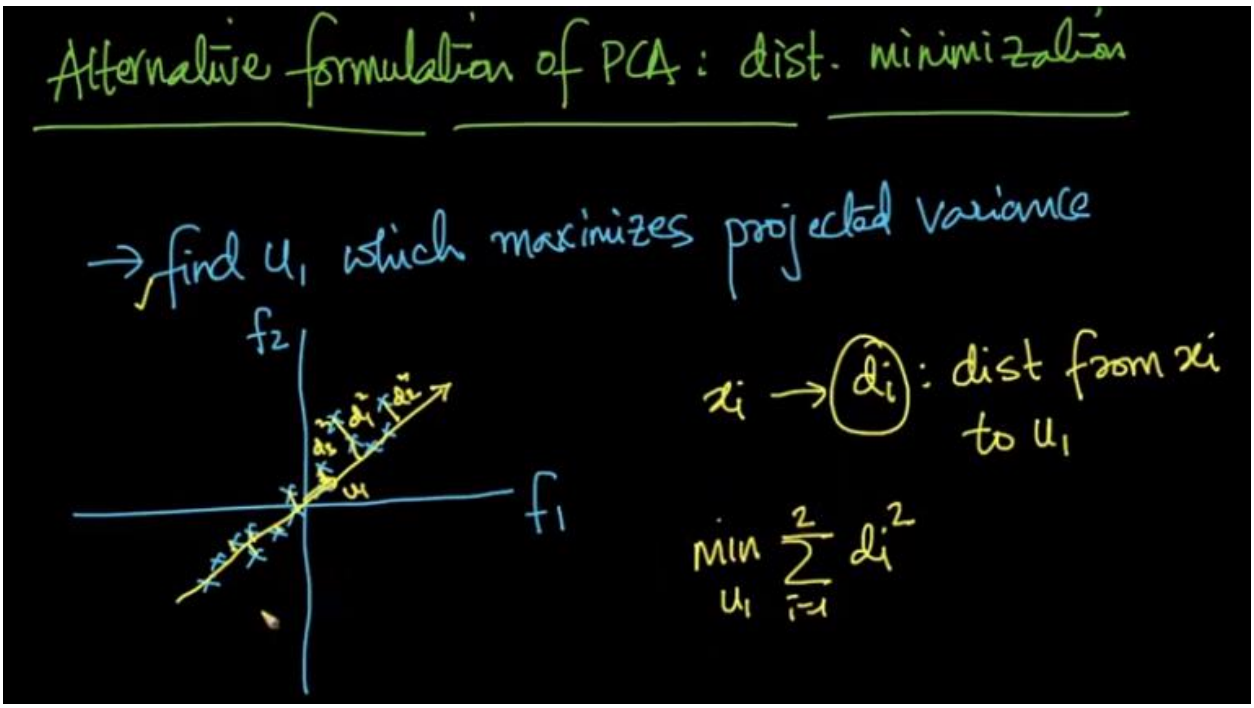


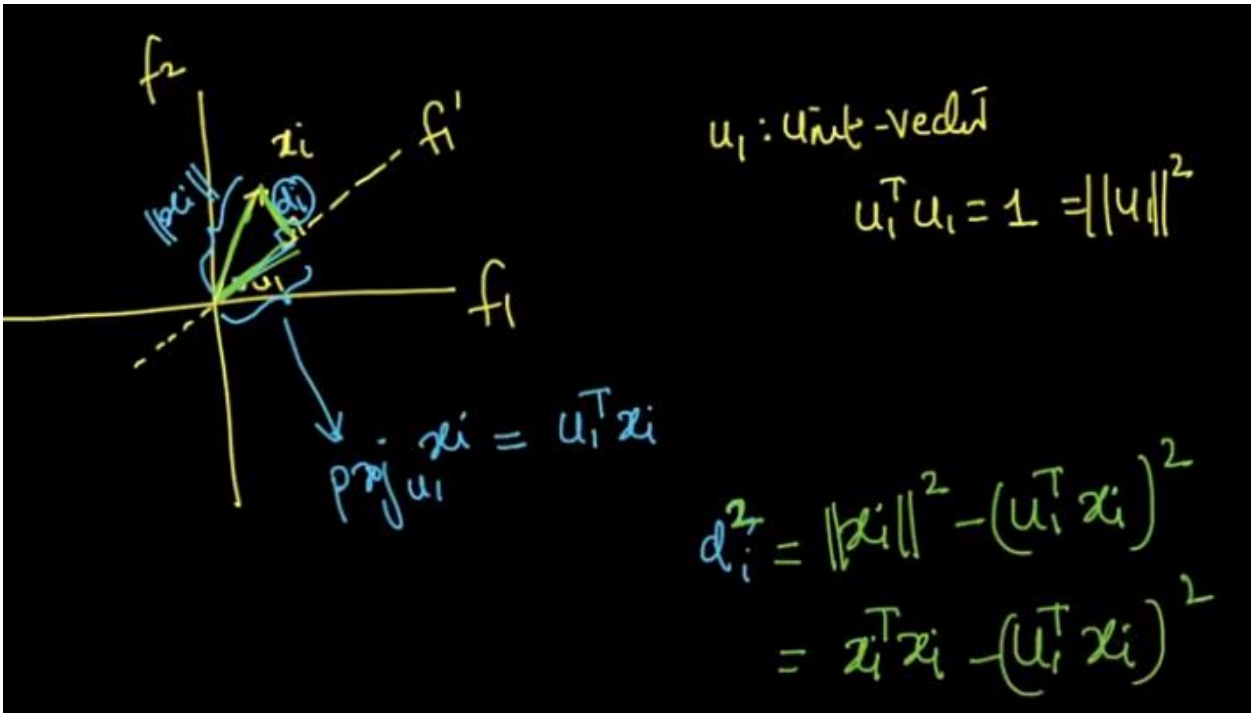






Distance Minimization: Alternate optimization statement for PCA





u_1 : unit-vector

$u_1^T u_1 = 1 = \|u_1\|^2$

$$d_i^2 = \|x_i\|^2 - (u_1^T x_i)^2$$

$$= x_i^T x_i - (u_1^T x_i)^2$$

dist min PCA

$$\min_{u_1} \sum_{i=1}^{n} \left(x_i^T x_i - (u_1^T x_i)^2 \right)$$

d_i^2

s.t $u^T u = 1$

x_i

$$X = \begin{bmatrix} \leftarrow x_i^T \rightarrow \end{bmatrix}$$

$$\max_{u_1} \frac{1}{n} \sum_{i=1}^{n} (u_1^T x_i)^2$$

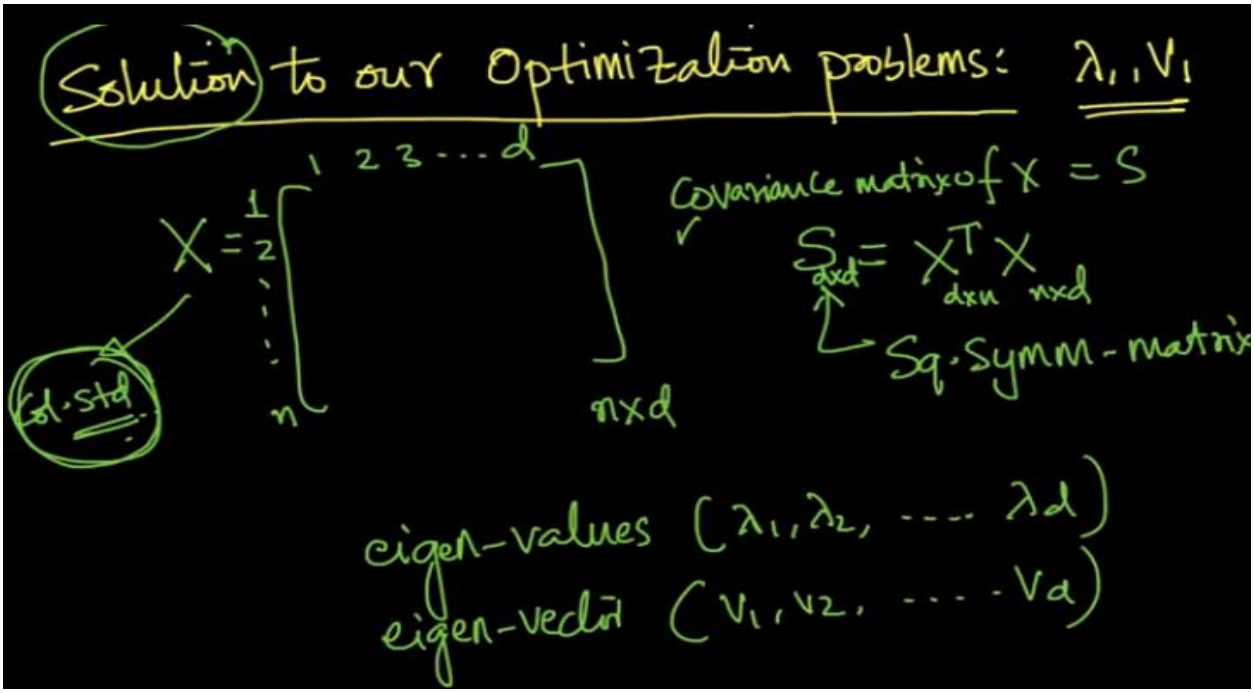
s.t $u^T u = 1$

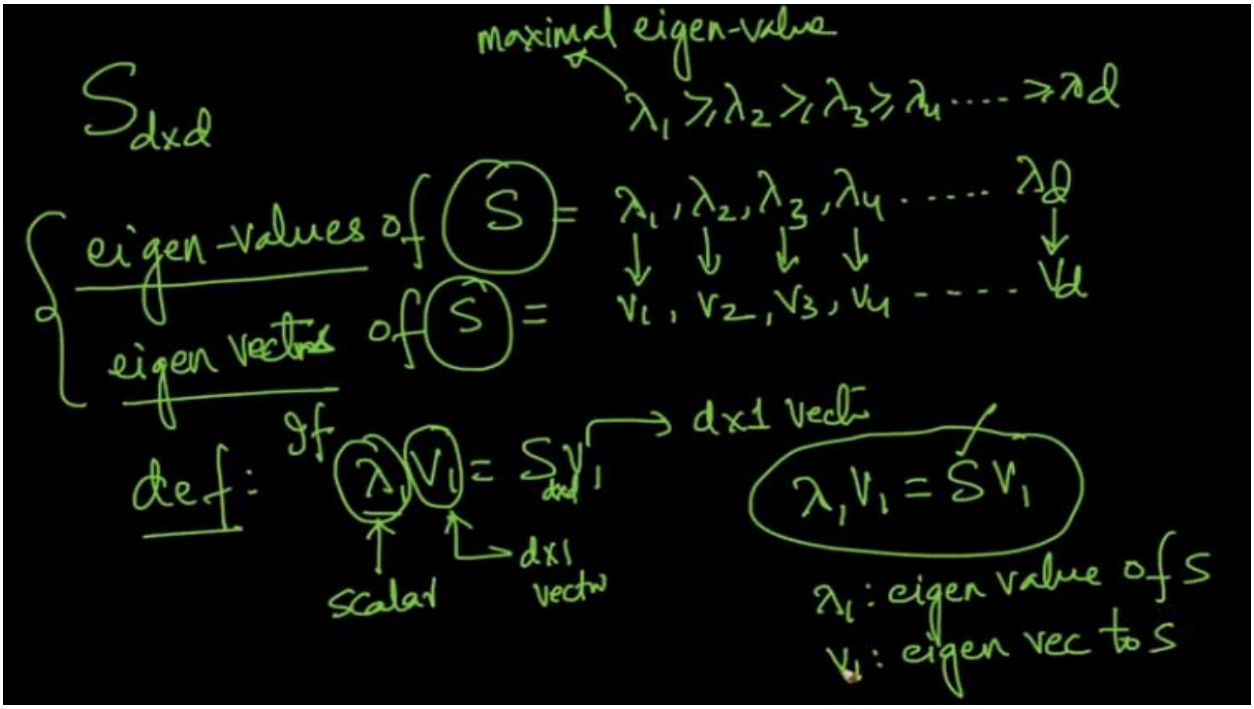
} - Variance maximization PCA

Solution to optimization problem:

Eigen vector (direction of the maximal variance)

Eigen values (percentage of variance explained)





$\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots \geq \lambda_d$

$v_1, v_2, v_3, \dots v_d$

$S_{d \times d}$

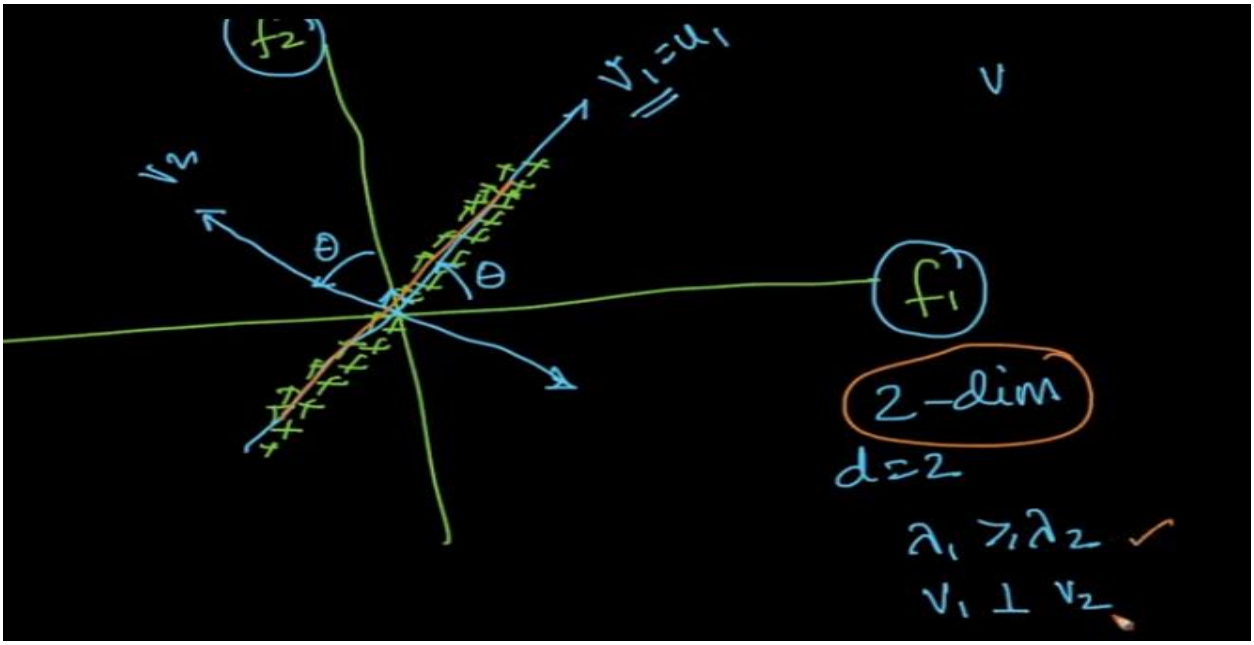
$v_i \perp v_j$: $v_i^T v_j = 0 = v_i \cdot v_j = 0$

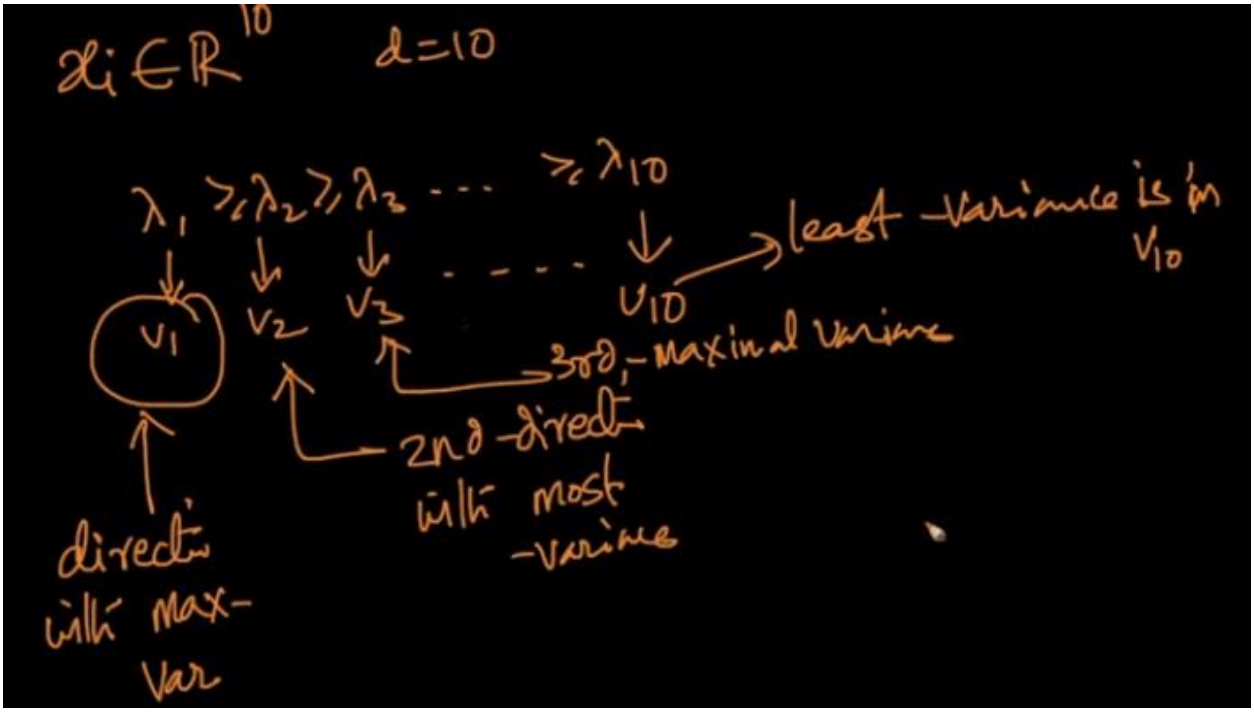
$u_1 = v_1$ = eigen-vector of $S (= X^T X)$ corr. to largest eigen-value $(= \lambda_1)$

u_1 : max-variance direction

$X = \begin{bmatrix} & \checkmark & \end{bmatrix}_{n \times d}$

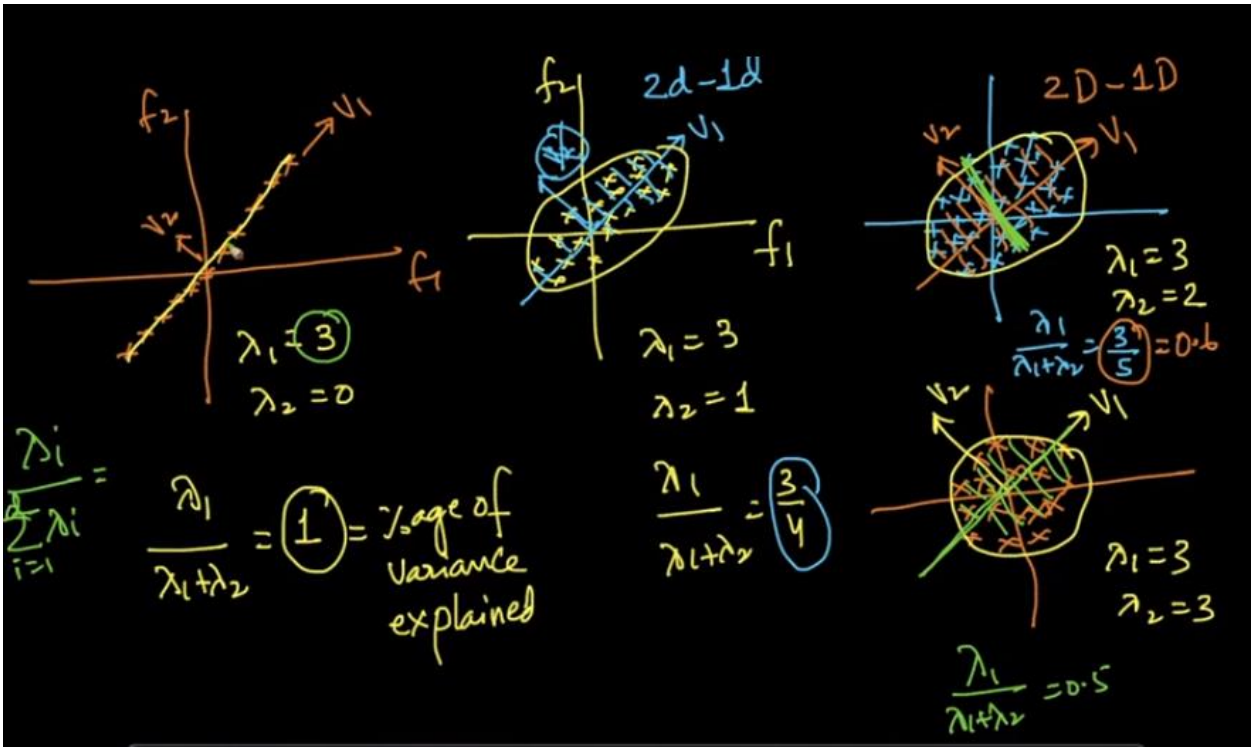
1. Col. std of X is done
2. $S_{d \times d} = X^T X$
3. eigen(S) → eigen values & vectors of S
   $\lambda_1 \geq \lambda_2 \geq \dots \lambda_d$
   $v_1, v_2, \dots v_d$
4. $u_1 = v_1$ (Why?)





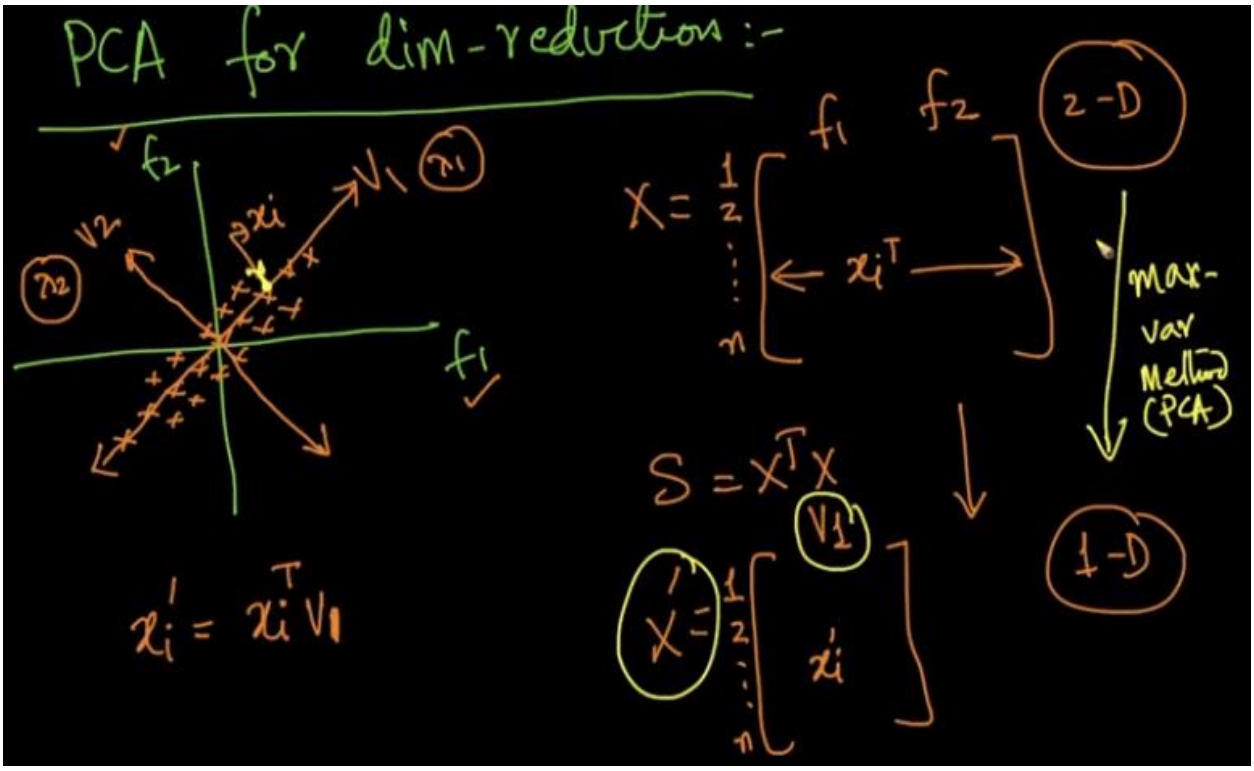
Projecting the data and variance explained: -

Eigen vector explained the direction of maximum variance and lambda (eigen values) explains the percentage of variance on the projected eigen vector (v1)

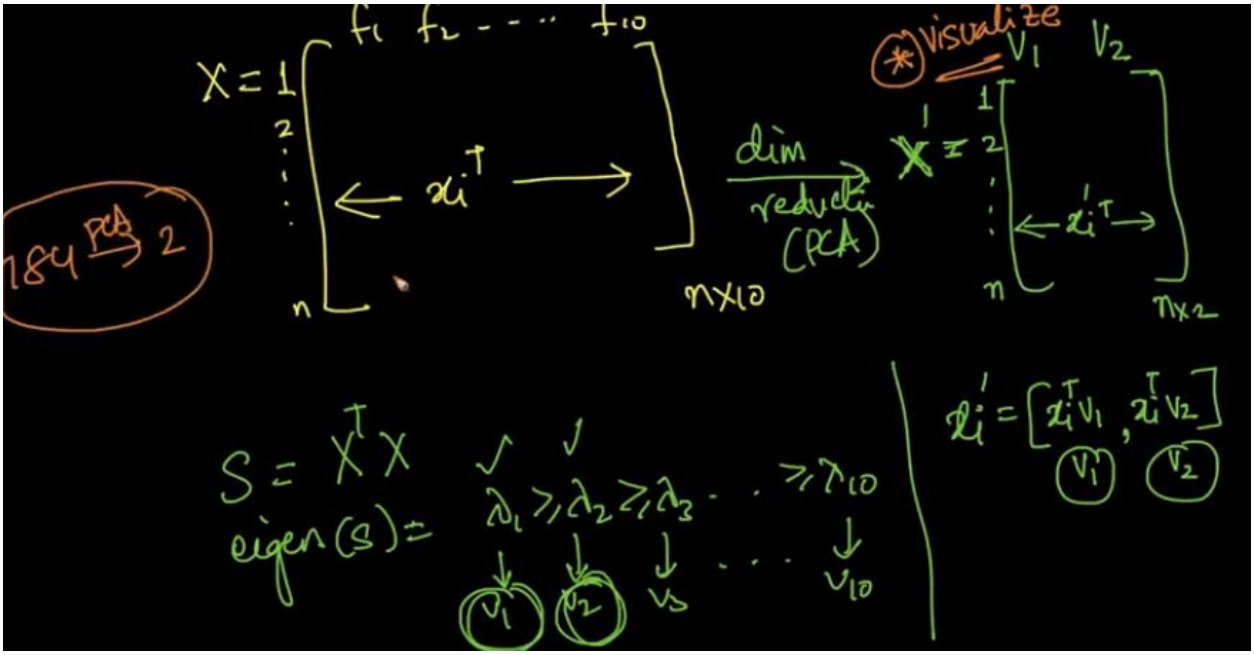


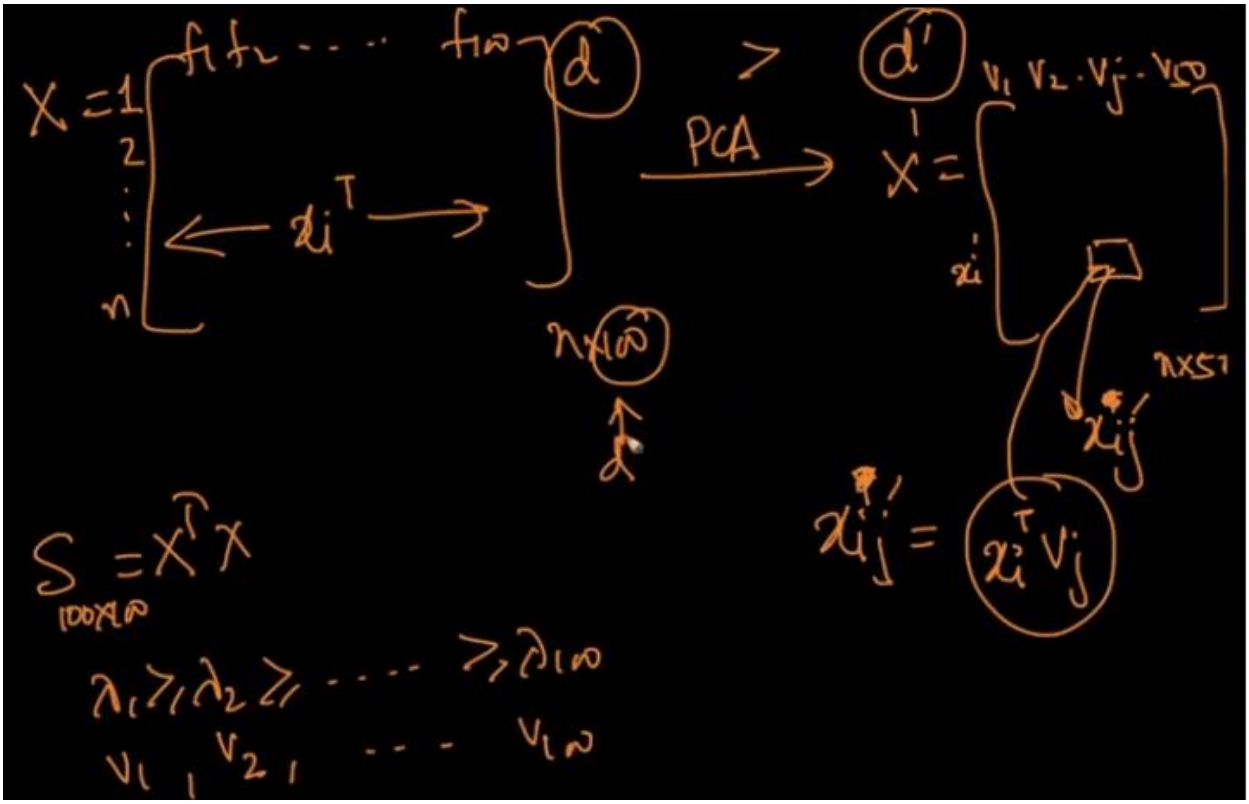
Idea to reduce dimension using PCA: -

(A) Take maximum variance eigen vector as feature ($X_i' = X_i^T V1$)

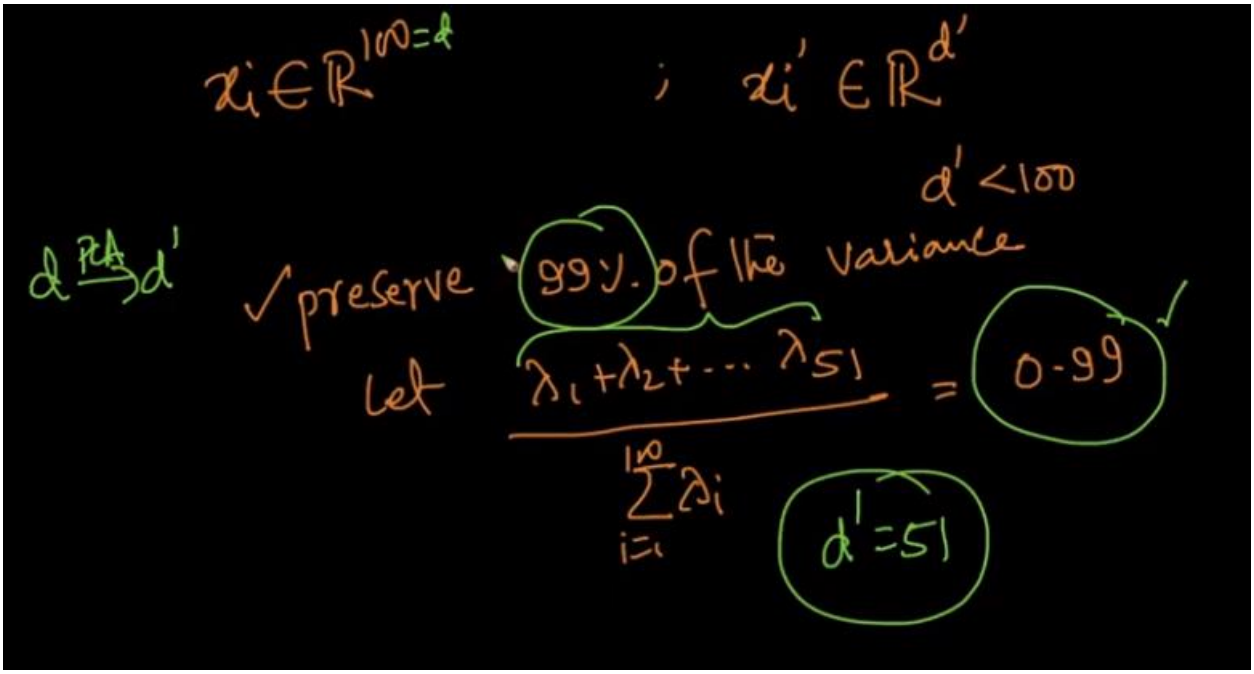


(B) Take maximum variance eigen vector as feature ($X_i' = X_i^T V1 , X_i^T V2$)

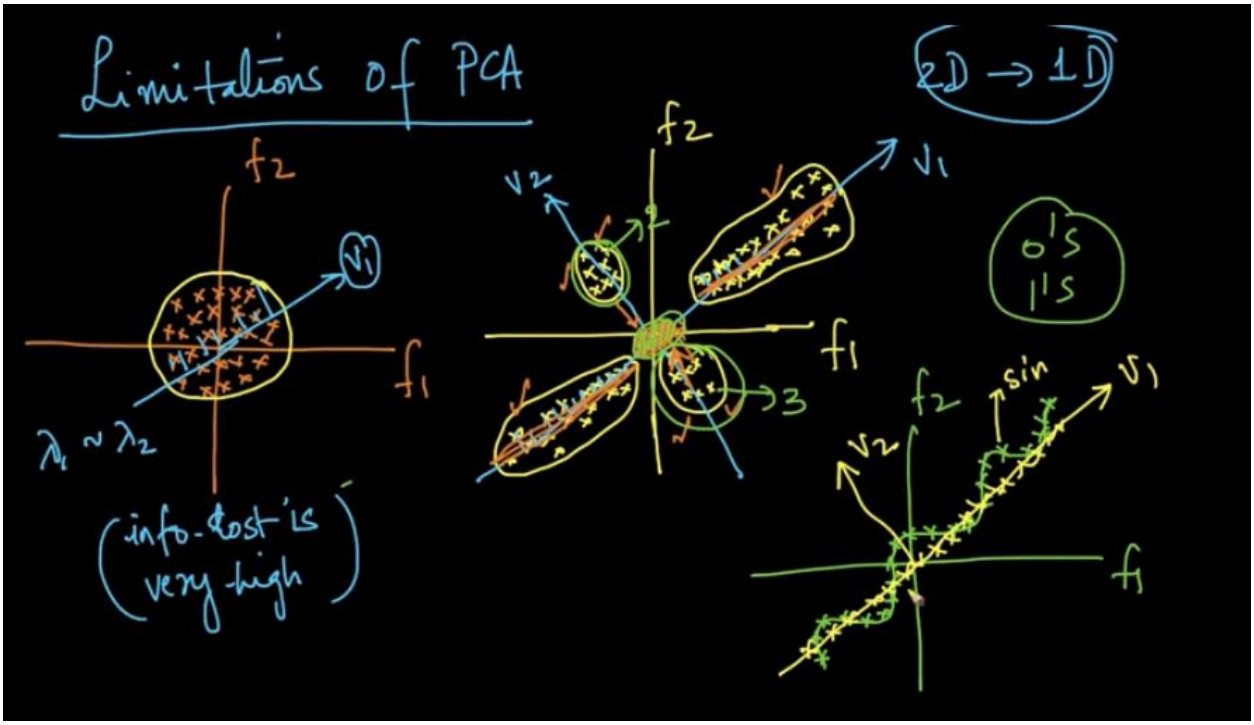




Eigen values (measure of maximum variance) – select feature to retain the amount of variance.

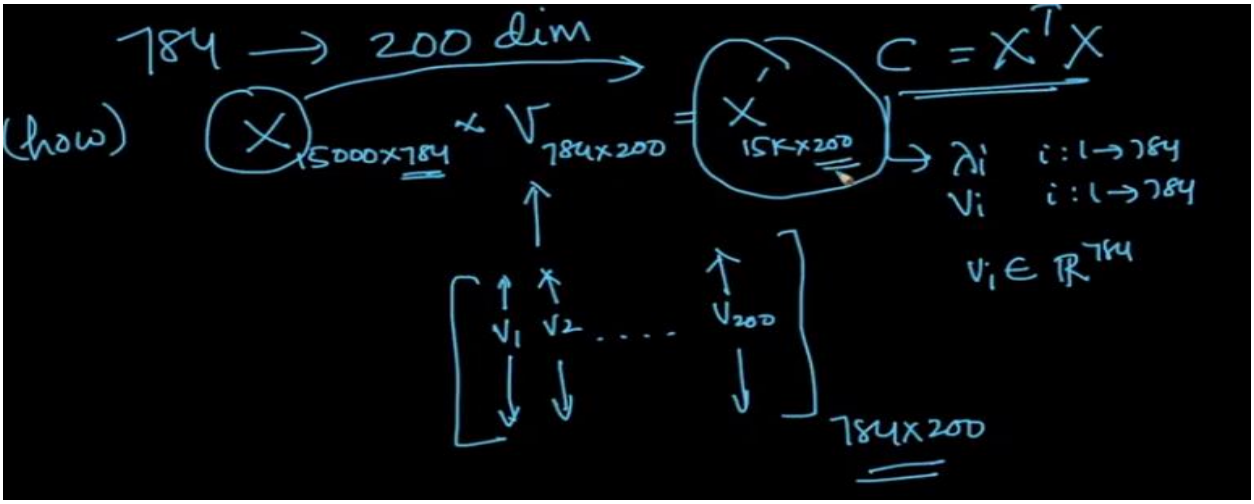


Limitations of PCA: - information loss for types of data spread

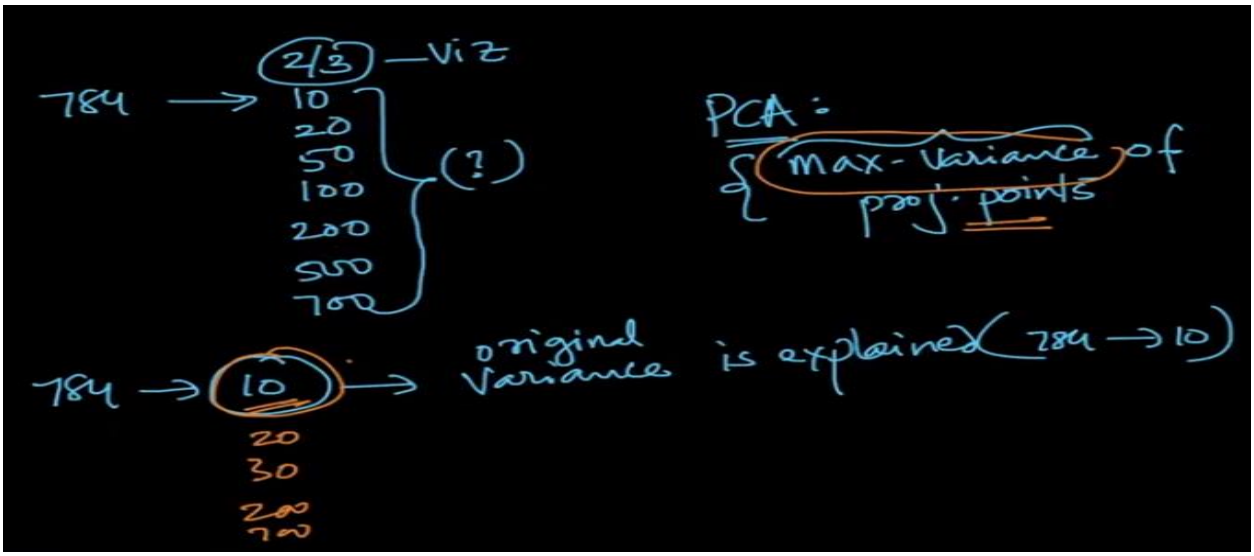


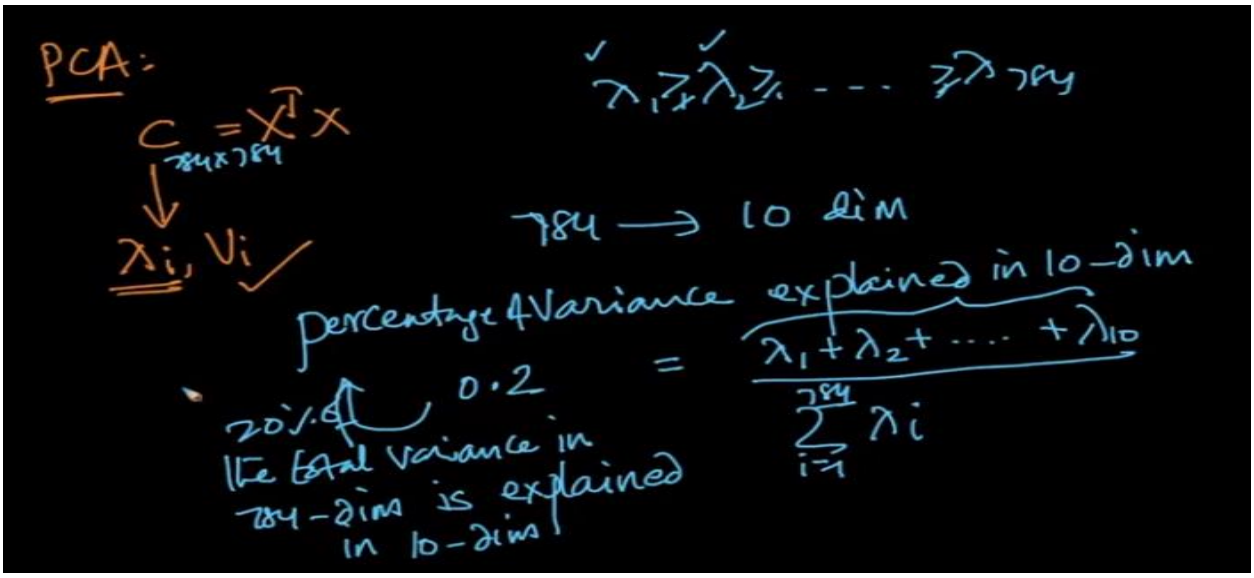
Machine Learning Model (Dimensionality Reduction)

Convert into lower dimension

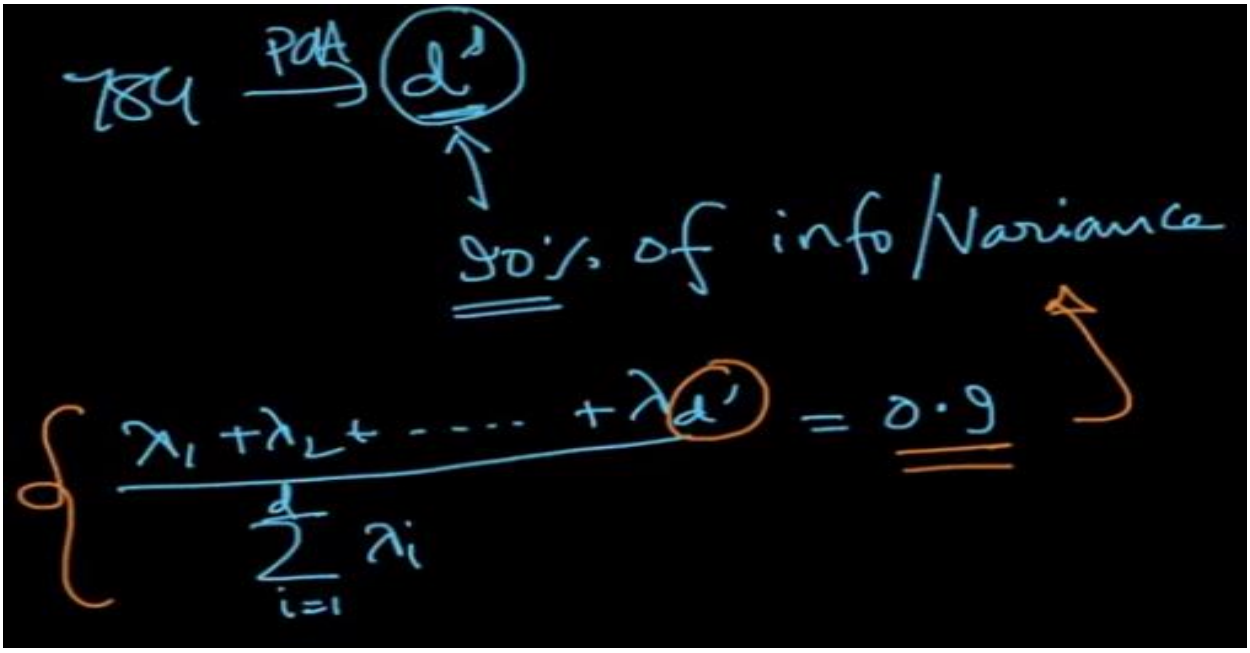


What dimension to choose – maximum variance retained

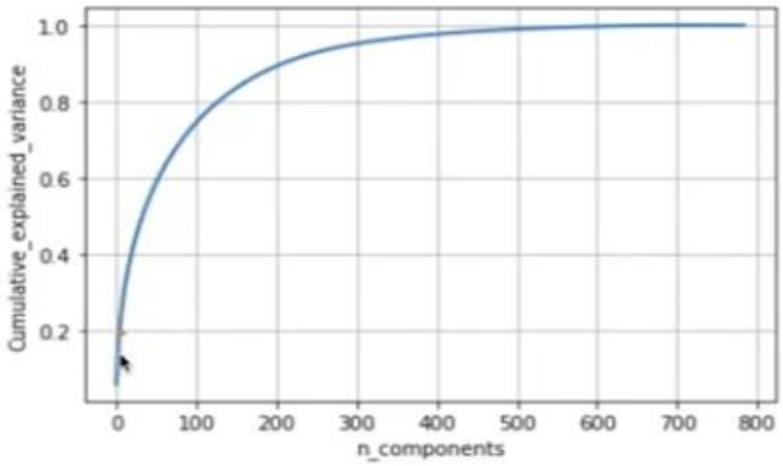




Find d’ which retains 90% variance explained.



```
# PCA for dimensionality redcution (non-visualization)

pca.n_components = 784
pca_data = pca.fit_transform(sample_data)

percentage_var_explained = pca.explained_variance_ / np.sum(pca.explained_v

cum_var_explained = np.cumsum(percentage_var_explained)

# Plot the PCA spectrum
plt.figure(1, figsize=(6, 4))

plt.clf()
plt.plot(cum_var_explained, linewidth=2)
plt.axis('tight')
plt.grid()
plt.xlabel('n_components')
plt.ylabel('Cumulative_explained_variance')
plt.show()
```



90% variance explained by 200 principle components.