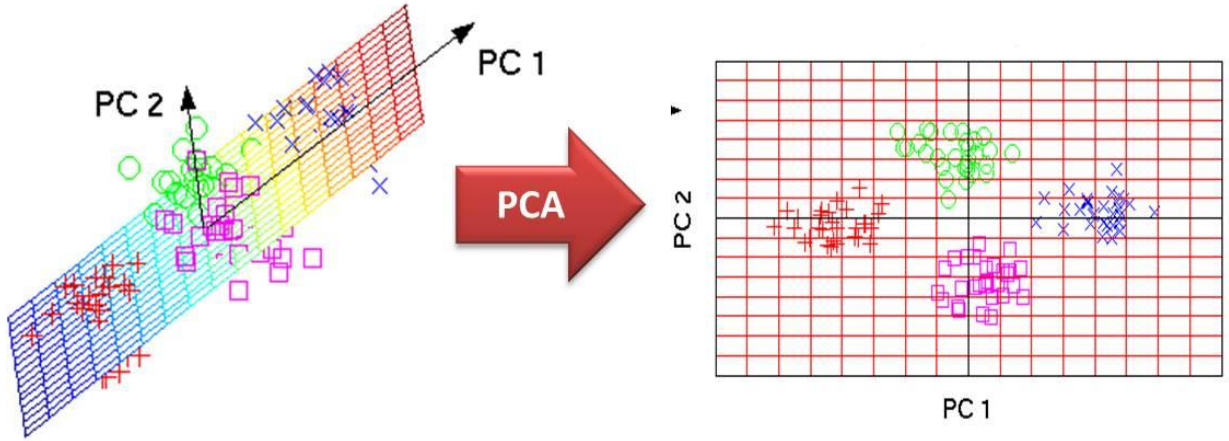
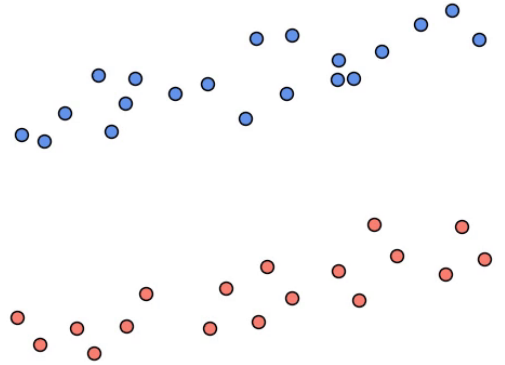
Dimensionality Reduction
&
Principal Component Analysis



There are two techniques to make dimensionality reduction

- **Feature Extraction**
- **Feature Selection**

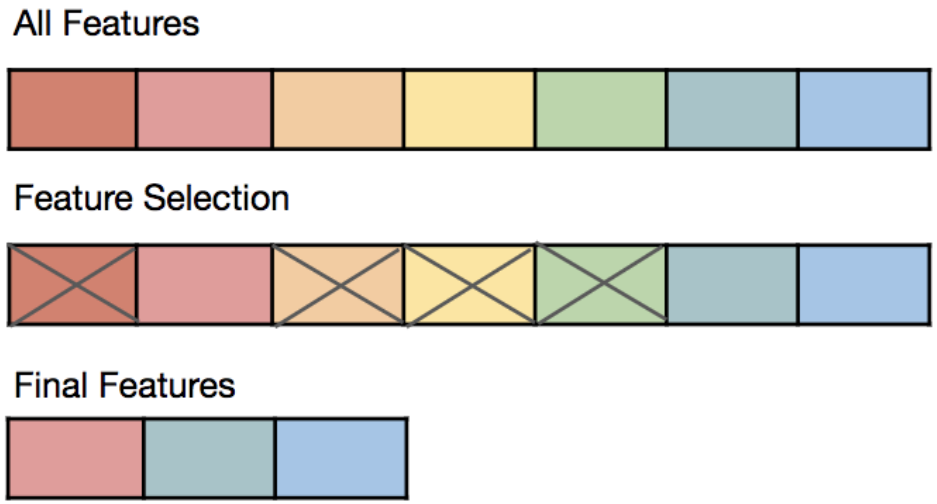


Curse of Dimensionality

Increasing the number of features does not always improve accuracy. When data does not have enough features, the model is likely to **underfit**, and when data has too many features, it is likely to **overfit.**

Feature Selection

Feature selection methods are intended to reduce the number of input variables to those that are believed to be most useful to a model in order to predict the target variable.



Feature Extraction

In feature extraction, a set of new features are found. That is found through some mapping from the existing features.

Linear Feature Extraction

•**Principal Component Analysis (PCA)**: It seeks a projection that preserves as much information as possible in the data.
•**Linear Discriminant Analysis (LDA)**:- It seeks a projection that best discriminates the data.

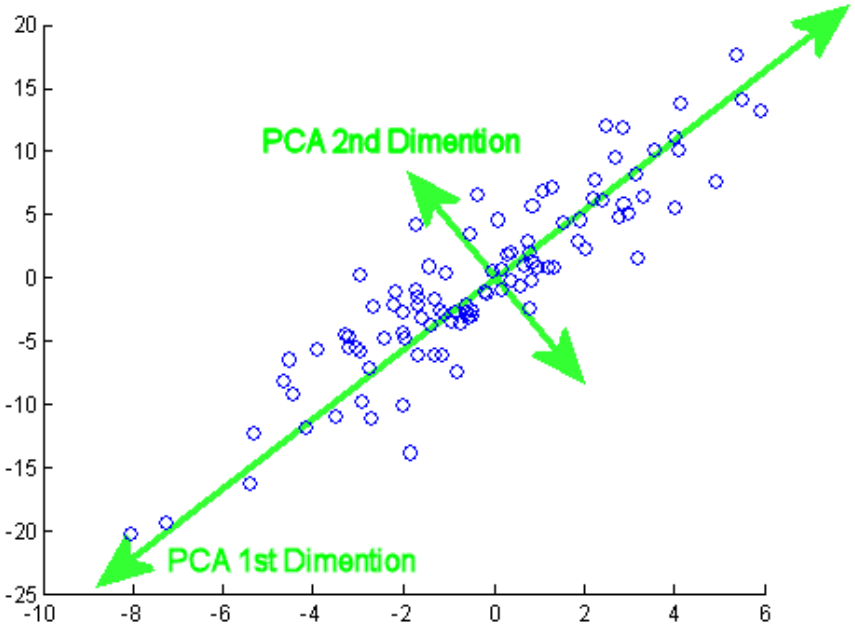
We can use (PCA) for the following purposes:

- To reduce the number of dimensions in the dataset.
- To find patterns in the high-dimensional dataset
- To visualize the data of high dimensionality
- To captures as much of the original variance in the data as possible

Principal Components Analysis

Principal components analysis is a dimensionality reduction technique that enables you to identify correlations and pattern in a data set so that it can be transformed into a data set of significantly lower dimension without loss of any important information.

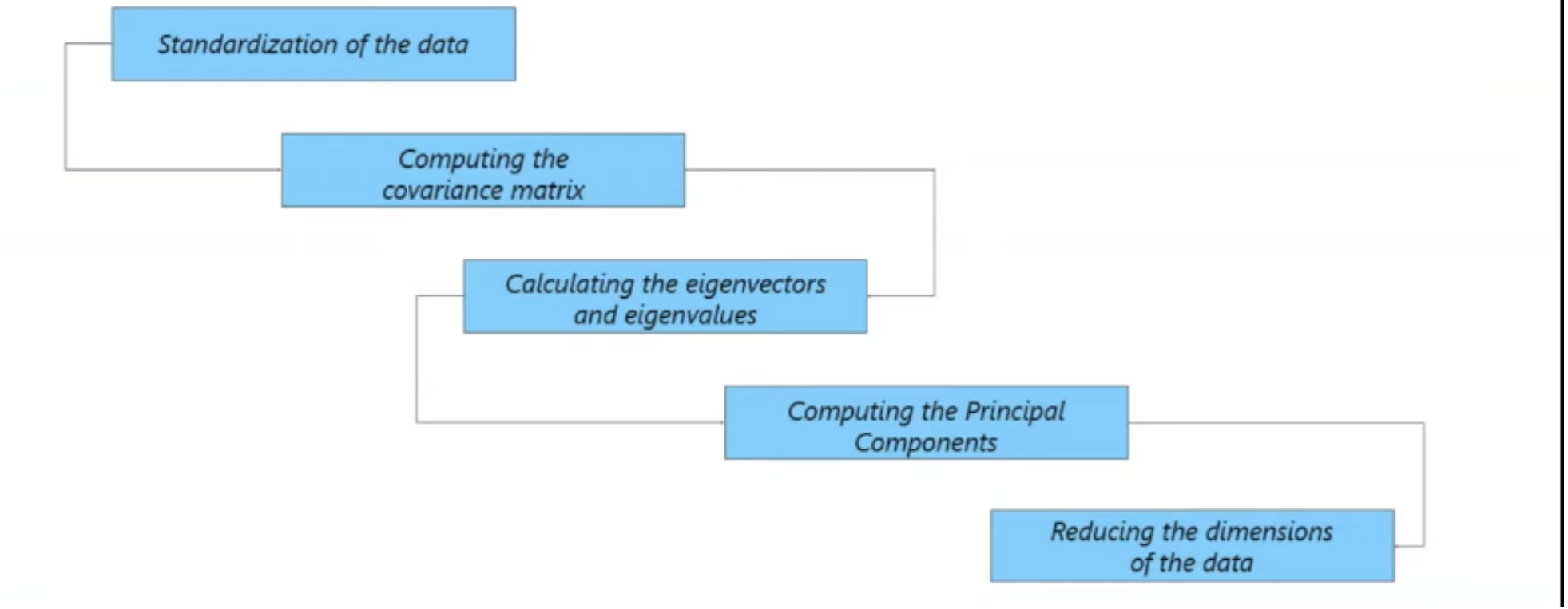
Solving overfitting problem (so many features)



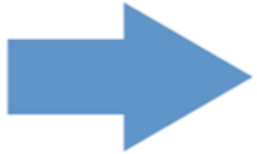
Steps involved in PCA

1. Standardize the PCA.
2. Calculate the covariance matrix.
3. Find the eigenvalues and eigenvectors for the covariance matrix.
4. Plot the vectors on the scaled data.

STEP BY STEP PCA



Univ	SAT	Top10	Accept	SFRatio	Expenses	GradRate
Brown	1310	89	22	13	22704	94
CalTech	1415	100	25	6	63575	81
CMU	1260	62	59	9	25026	72
Columbia	1310	76	24	12	31510	88
Cornell	1280	83	33	13	21864	90
Dartmouth	1340	89	23	10	32162	95
Duke	1315	90	30	12	31585	95
Georgetown	1255	74	24	12	20126	92
Harvard	1400	91	14	11	39525	97
JohnsHopkins	1305	75	44	7	58691	87
MIT	1380	94	30	10	34870	91
Northwestern	1260	85	39	11	28052	89
NotreDame	1255	81	42	13	15122	94
PennState	1081	38	54	18	10185	80



PC_1	PC_2	PC_3	PC_4	PC_5	PC_6

The i^{th} principal component is a weighted average of original measurements / columns:

$$PC_i = a_{i1}X_1 + a_{i2}X_2 + a_{i3}X_3.... + a_{in}X_n$$

Weights (a_{ij}) are chosen such that:

1. PCs are ordered by their variance (PC_1 has largest variance, followed by PC_2, PC_3, and so on)
2. Pairs of PCs have correlation = 0
3. For each PC, sum of squared weights = 1 (Unit Vector)

What do we need:

$$cov(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(Xi - \bar{x})(Yi - \bar{y})$$

1) Covariance Matrix

2) Eigenvectors and eigen values of Covariance matrix

Covariance

- Exact value is not as important as its sign.
- A positive value of covariance indicates that both dimensions increase or decrease together, e.g., as the number of hours studied increases, the grades in that subject also increase.
- A negative value indicates while one increases the other decreases, or vice-versa, e.g., active social life vs. performance in ECE Dept.
- If covariance is zero: the two dimensions are independent of each other, e.g., heights of students vs. grades obtained in a subject.

Eigenvalue Problem

- The eigenvalue problem is any problem having the following form:

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

A: m x m matrix

v: m x 1 non-zero vector

λ: scalar

- Any value of λ for which this equation has a solution is called the eigenvalue of A and the vector **v** which corresponds to this value is called the eigenvector of **A**.

Eigenvalue Problem

- Going back to our example:

$$\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \times \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 12 \\ 8 \end{bmatrix} = 4 \times \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

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

- Therefore, (3,2) is an eigenvector of the square matrix **A** and 4 is an eigenvalue of **A**

The explained variance variable is now a float type array which contains variance ratios for each principal component. The values for the explained variance variable looks like this:

It can be seen that first principal component is responsible for 72.22% variance. Similarly, the second principal component causes 23.9% variance in the dataset. Collectively we can say that (72.22 + 23.9) 96.21% percent of the classification information contained in the feature set is captured by the first two principal components.

0.722265
0.239748
0.0333812
0.0046056

Pc1 = Most significant information

Pc2 = second most significant information

Advantage

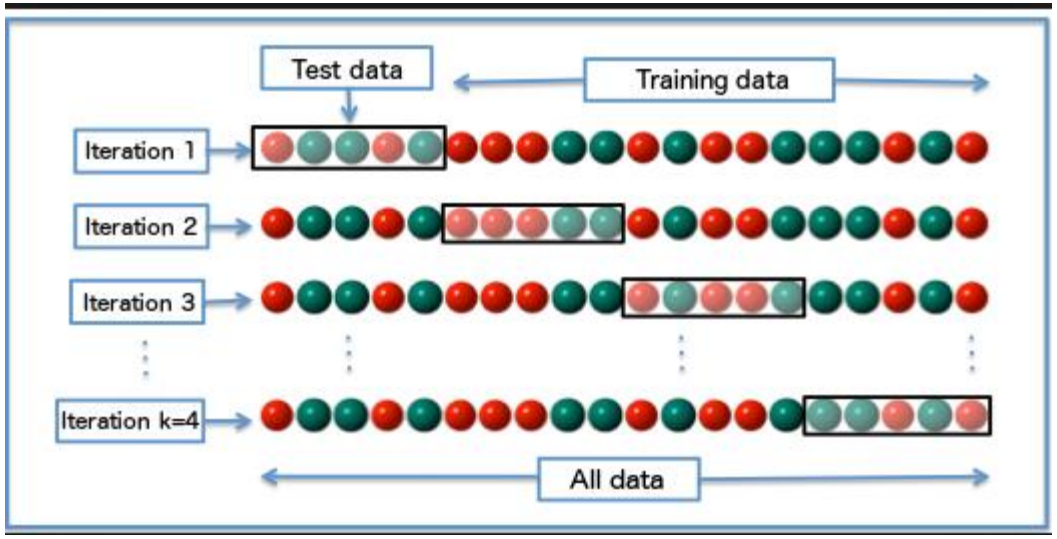
- Removes Correlated Features
- Improves Algorithm Performance
- Improves Algorithm Performance
- Improves Visualization

Disadvantage

- Information Loss
- Data standardization is must before PCA
- Principal Component Analysis assumes that the relationships between variables are linear. However, if there are non-linear relationships between variables, Principal Component Analysis may not work well.

Train_Test Split approach.

K-Folds Cross Validation:



•In this method, the training dataset will be split into multiple ‘k’ smaller parts/sets. Hence the name ‘k’-fold.
•The current training dataset would now be divided into ‘k’ parts, out of which one dataset is left out and the remaining ‘k-1’ datasets are used to train the model.
•This is done multiple number of times. The number of times that it has to be done is mentioned by the user in the code.
•The one that was kept out of the training is used as a ‘validation dataset’. This can be used to tune hyperparameters and see how the model performs and change the values accordingly, to yield better results.
•Even though the size of the dataset isn’t reduced considerably, it was reduced to a certain extent. This method also makes sure that the model remains robust and generalizes well on the data.