Dimensionality Reduction using Principal Component Analysis

Structure of this Module

Principal Component Analysis

TOPICS
Introduction to PCA
PCA Process Steps
Data Standardization
Finding Covariance Matrix of our Dataset
Eigenvectors and Eigenvalues
Recast Data using new PCAs
Explained Variance Ratio and Scree Plot

PCA - Definition

PCA is an unsupervised dimensionality reduction algorithm to find a more meaningful basis or coordinate system (axis) for our data and works based on a covariance matrix to find the strongest features of your samples.

It is used **when we need to tackle the ’curse of dimensionality’** among data, i.e. where having too many dimensions (features) in your data causes noise and difficulties (it can be sound, picture, or context).

This specifically gets worst when features have ***different scales*** (e.g. weight, length, area, speed, power, temperature, volume, time, cell number, etc.)

- ***Higher Dimensional Data cases training process to be slow.***
- ***Higher Dimensional Data cannot be Visualized.***

Higher number of dimensions can affect a model’s accuracy since there is more data that needs to be generalized.

Dimensionality Reduction is way to reduce the complexity of a model and avoid overfitting.

Categories of Dimensionality Reduction:

- ***Feature Selection***, we select a subset of the original features.
- ***Feature Extraction***, we derive information from the feature set to construct a new feature subspace.

(PCA is a Feature Extraction method)

When is PCA Used

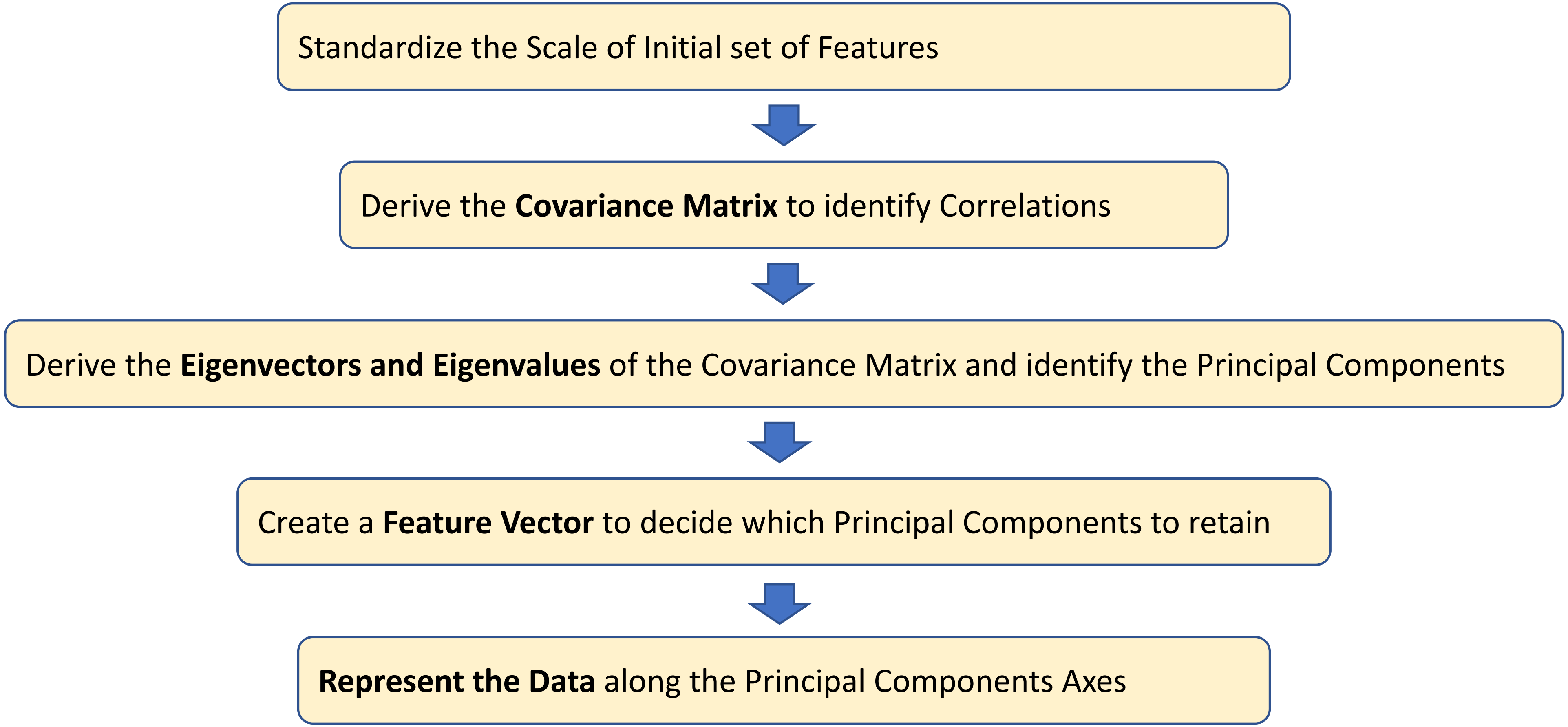
Better Perspective and Less Complexity: When we need a more realistic perspective and we have many features on a given data set and specifically when we have this intuitive knowledge that we don’t need this much number of features. Intuitively, modelling is much easier in 2D or 3D, rather than in 100D, isn’t it?

Better Visualization: When we cannot get a good visualization due to a high number of dimensions we use PCA to reduce it into a shadow of 2D or 3D features.

Reduce Size: When we have too much data due to too many features and we are going to use process-intensive algorithms (like many supervised algorithms) on the data so we need to get rid of redundancy.

Different Perspective: Maybe we don’t have any of these motivations but merely need to improve our knowledge of the data. PCA can give us the best linearly independent and different combinations of features so you can use them to describe your data differently.

PCA Process Steps



Standardize the Scale of Initial set of Features

Derive the **Covariance Matrix** to identify Correlations

Derive the **Eigenvectors and Eigenvalues** of the Covariance Matrix and identify the Principal Components

Create a **Feature Vector** to decide which Principal Components to retain

Represent the Data along the Principal Components Axes

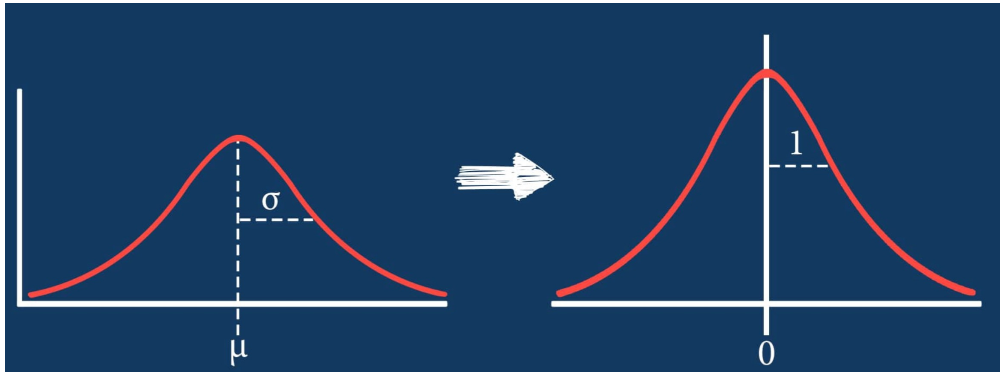
Standardize Dataset Features

Standardization is a step that is performed as part of the Pre-processing before a number of Machine Learning process. The goal of Standardization is to bring uniformity in the scales of Continuous Numeric Variables with a mean of 0.

Difference in scales in the values of different features in datasets are natural due the use of different units. However, Machine Learning models such as Gradient Descent, K-Means, KNN, ANN, PCA, etc are sensitive to varying scales with the dataset used in learning. Hence Standardization is performed.

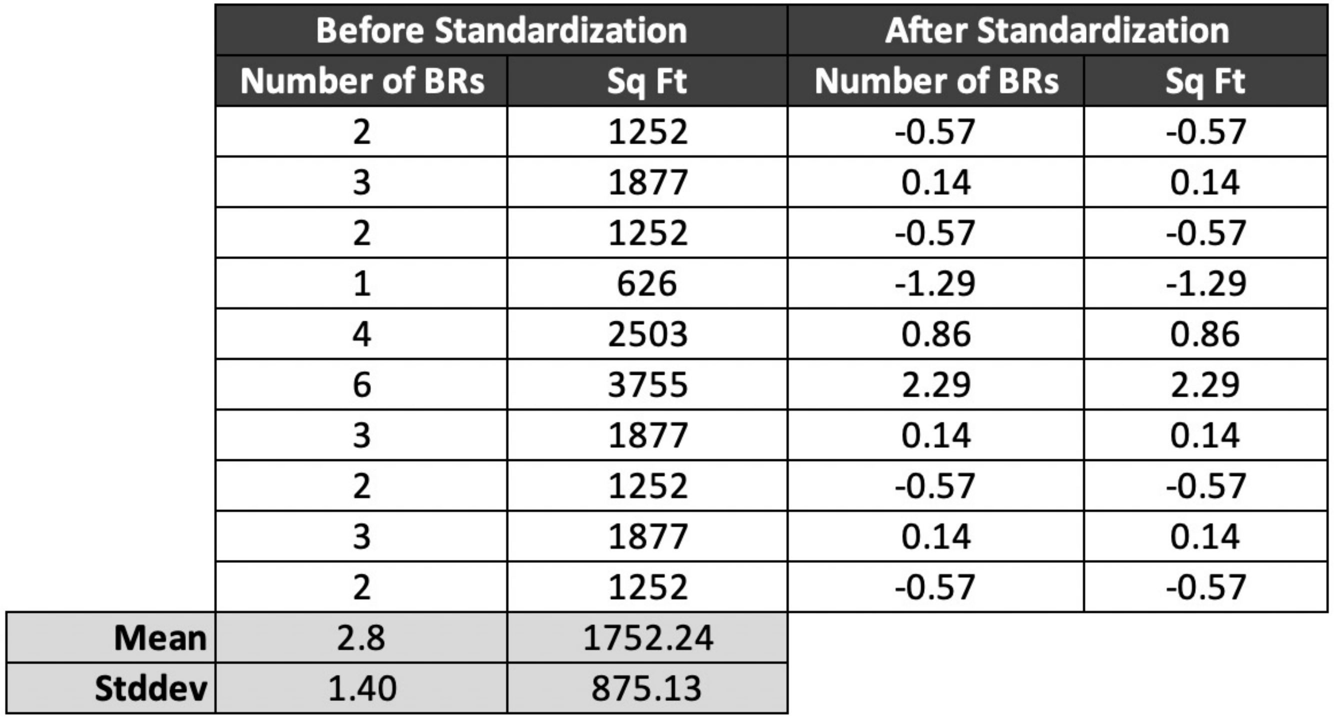
The z-score method is the most popular for standardising data.

It is done by subtracting the mean and dividing by the standard deviation for each value of each feature.



$$x_{\text{stand}} = \frac{x - \text{mean}(x)}{\text{standard deviation }(x)}$$

Standardize Dataset Features



	Before Standardization		After Standardization	
	Number of BRs	Sq Ft	Number of BRs	Sq Ft
	2	1252	-0.57	-0.57
	3	1877	0.14	0.14
	2	1252	-0.57	-0.57
	1	626	-1.29	-1.29
	4	2503	0.86	0.86
	6	3755	2.29	2.29
	3	1877	0.14	0.14
	2	1252	-0.57	-0.57
	3	1877	0.14	0.14
	2	1252	-0.57	-0.57
Mean	2.8	1752.24		
Stddev	1.40	875.13		

<< Example of how a Standardization of a couple of Features will look like.

Derive Covariance Matrix

The **covariance matrix** is a *p* × *p* symmetric matrix (where *p* is the number of dimensions) that has **as entries the covariances associated with all possible pairs of the initial variables**.

While the **Variance** (square of Standard Deviation) is an indication of how much variability is existing within a particular feature, **Covariance** is a value that is associated with a pair of features and is an indication of **co-variability between these two features**.

A data set with 2 dimensions/features *x and y*, the covariance matrix is a 2×2 matrix of this from:

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

Similarly, a data set with 3 dimensions/features *x, y and z*, the covariance matrix is a 3×3 matrix of this from:

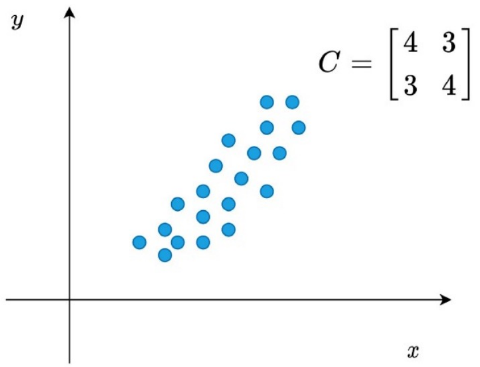
$$\begin{array}{cc} & \begin{array}{ccc} \text{x} & \text{y} & \text{z} \end{array} \\ \begin{array}{c} \text{x} \\ \text{y} \\ \text{z} \end{array} & \begin{bmatrix} var(x) & cov(x,y) & cov(x,z) \\ cov(x,y) & var(y) & cov(y,z) \\ cov(x,z) & cov(y,z) & var(z) \end{bmatrix} \end{array}$$

Calculation of Covariance Values

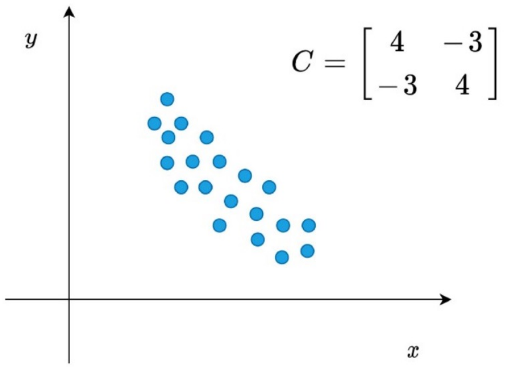
In the covariance formula, the values of both variables are multiplied by taking the difference from the mean.

$$cov(x, y) = \frac{\sum_i^n (x_i - \mu) \cdot (y_i - \mu)}{N}$$

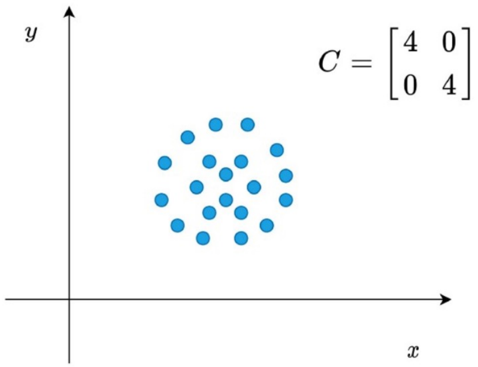
Positive Covariance



Negative Covariance

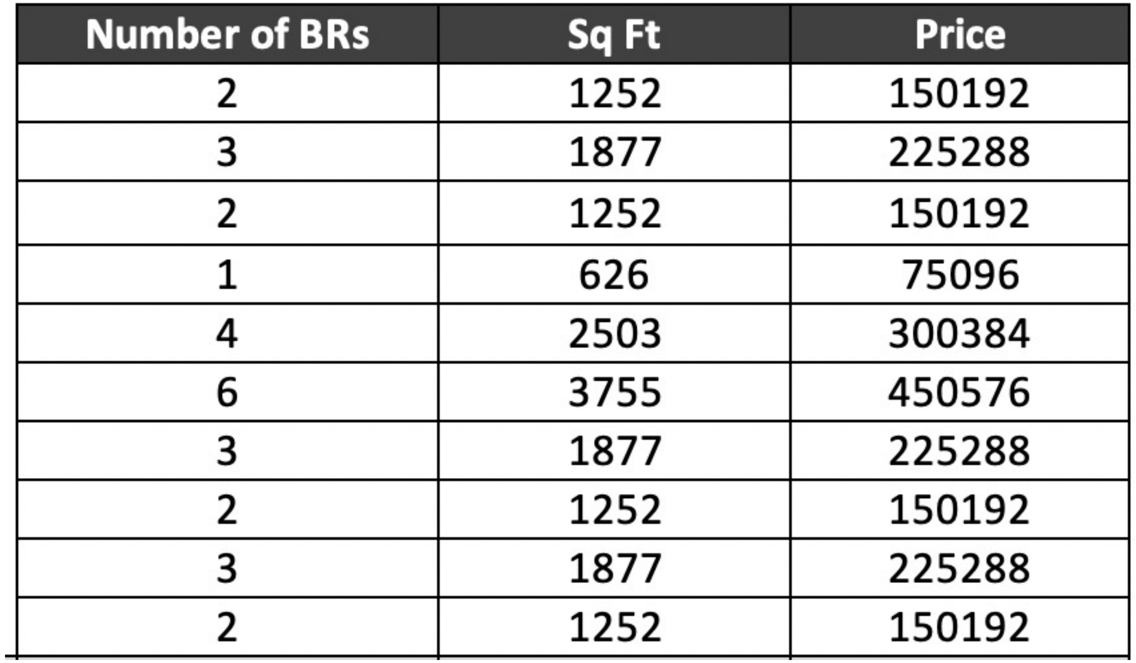


Zero Covariance

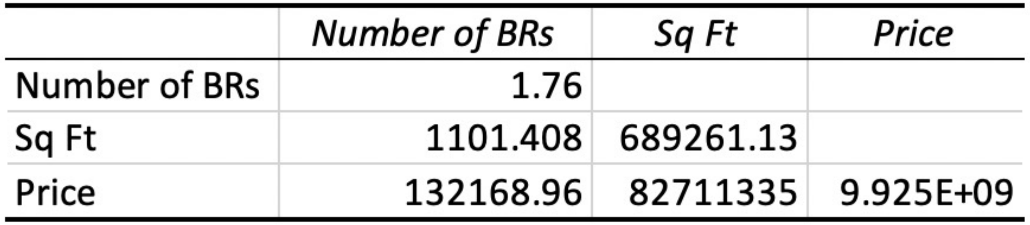


Calculation of Covariance Values

In the covariance formula, the values of both variables are multiplied by taking the difference from the mean.



Number of BRs	Sq Ft	Price
2	1252	150192
3	1877	225288
2	1252	150192
1	626	75096
4	2503	300384
6	3755	450576
3	1877	225288
2	1252	150192
3	1877	225288
2	1252	150192



	Number of BRs	*Sq Ft*	*Price*
Number of BRs	1.76		
Sq Ft	1101.408	689261.13	
Price	132168.96	82711335	9.925E+09

The above co-variance matrix is calculated automatically using excel Data Analysis – Covariance features.

Note that the cov(x,y) = cov(y,x)

Eigenvectors and Eigenvalues

- **Eigenvectors of the Covariance Matrix:** ***Directions of the axes (or new basis vector)*** *where there is the* ***most variance*** **(most information)** and that we call **Principal Components**.

- **Eigenvalues:** The **coefficients attached to Eigenvectors**, which give the ***amount of variance carried*** *in each Principal Component*.

By **ranking our eigenvectors** in order of their eigenvalues, highest to lowest, we get the principal components in order of significance.

The process of finding the Eigenvector and Eigenvalues is called the **Eigen-Decomposition**.

Basically, for any square matrix A, its **eigenvectors** are all the vectors v which satisfy the following equation:

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

λ is some constant also known as the **eigenvalue** for that particular eigenvector.

Eigenvectors and Eigenvalues

Let’s assume that we have this square matrix A: $A = \begin{bmatrix} 2 & 0 \\ 0 & 5 \end{bmatrix}$

Its eigenvectors and corresponding eigenvalues are given by as follows

$$\begin{bmatrix} 2 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

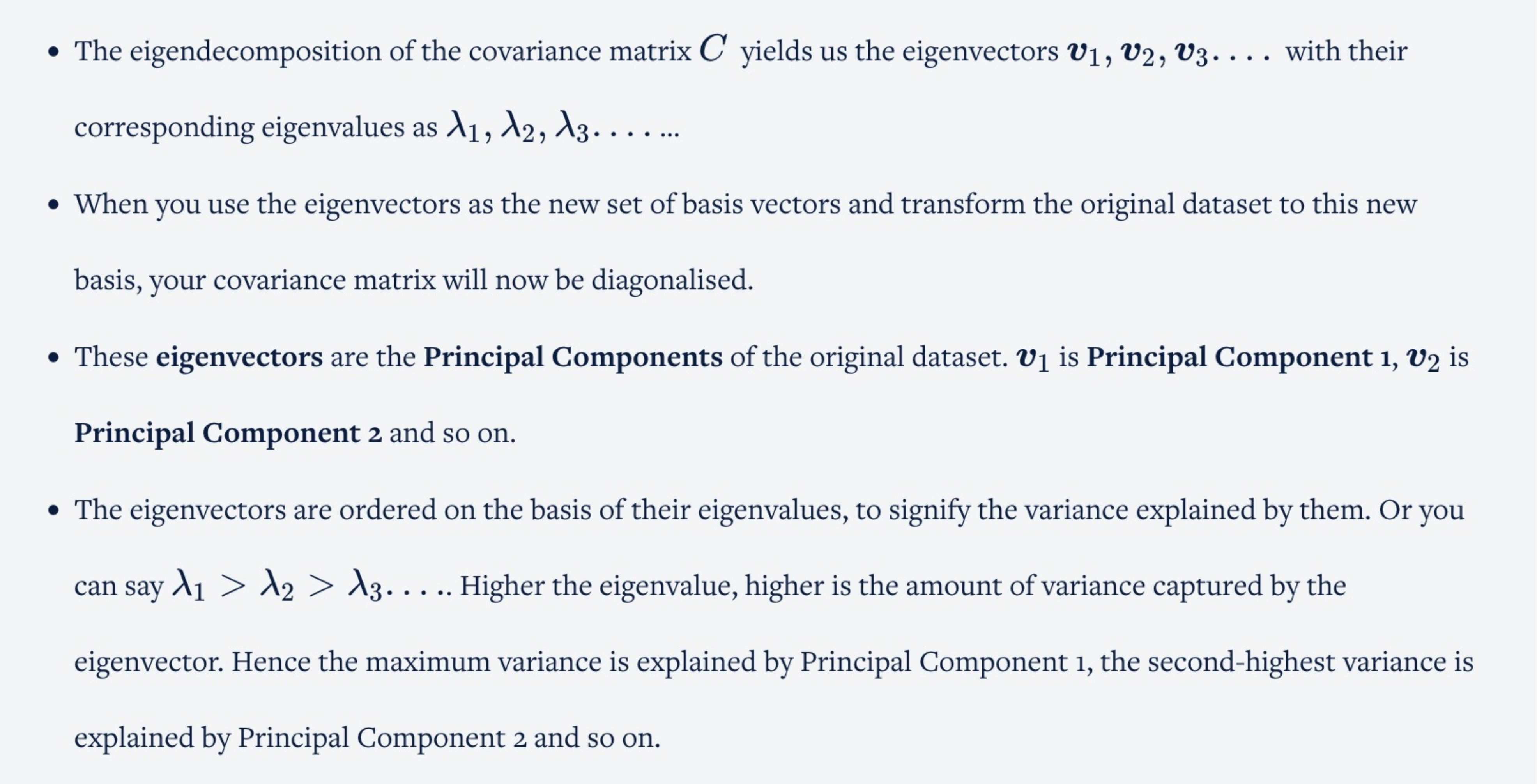
$$\begin{bmatrix} 2 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 5 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Here A has 2 sets of eigenvectors/eigenvalues -

$$\boldsymbol{v}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \lambda_1 = 2 \ and \ \boldsymbol{v}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \lambda_2 = 5$$

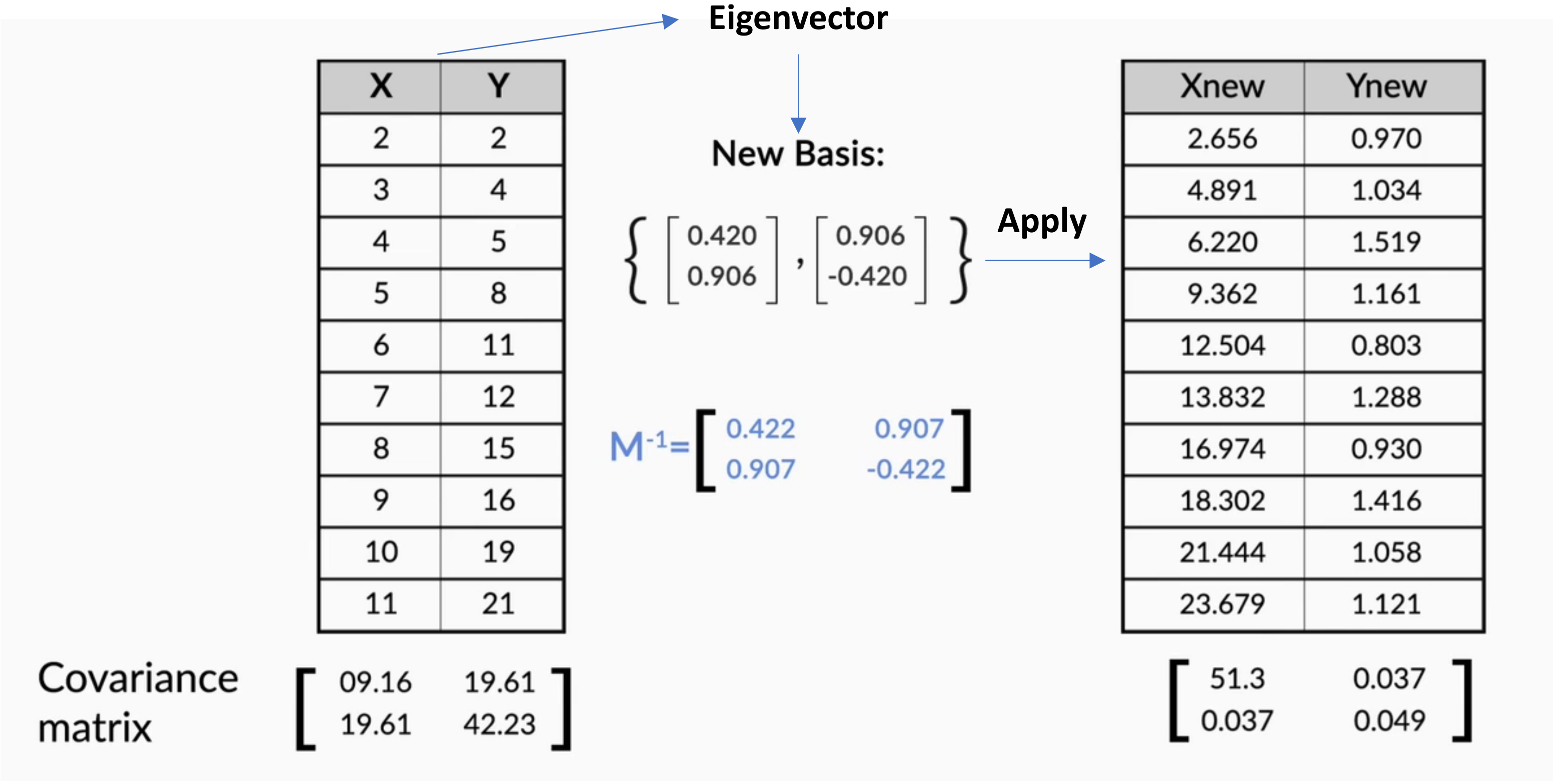
Our goal is to find a new set of basis vectors where the covariance matrix gets diagonalised.
It turns out that these new set of basis vectors are in fact the eigenvectors of the Covariance Matrix.
Also, these eigenvectors are the Principal Components of our original dataset. In other words, these eigenvectors are the directions that capture maximum variance.

Eigenvectors and Eigenvalues



- The eigendecomposition of the covariance matrix C yields us the eigenvectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3 . . . .$ with their corresponding eigenvalues as $\lambda_1, \lambda_2, \lambda_3 . . . . .$...
- When you use the eigenvectors as the new set of basis vectors and transform the original dataset to this new basis, your covariance matrix will now be diagonalised.
- These **eigenvectors** are the **Principal Components** of the original dataset. $\boldsymbol{v}_1$ is **Principal Component 1**, $\boldsymbol{v}_2$ is **Principal Component 2** and so on.
- The eigenvectors are ordered on the basis of their eigenvalues, to signify the variance explained by them. Or you can say $\lambda_1 > \lambda_2 > \lambda_3 . . . .$.. Higher the eigenvalue, higher is the amount of variance captured by the eigenvector. Hence the maximum variance is explained by Principal Component 1, the second-highest variance is explained by Principal Component 2 and so on.

Diagonalization



Eigenvector

X	Y
2	2
3	4
4	5
5	8
6	11
7	12
8	15
9	16
10	19
11	21

Covariance matrix $\begin{bmatrix} 09.16 & 19.61 \\ 19.61 & 42.23 \end{bmatrix}$

New Basis:

$$\left\{ \begin{bmatrix} 0.420 \\ 0.906 \end{bmatrix}, \begin{bmatrix} 0.906 \\ -0.420 \end{bmatrix} \right\}$$

$$M^{-1} = \begin{bmatrix} 0.422 & 0.907 \\ 0.907 & -0.422 \end{bmatrix}$$

Apply

Xnew	Ynew
2.656	0.970
4.891	1.034
6.220	1.519
9.362	1.161
12.504	0.803
13.832	1.288
16.974	0.930
18.302	1.416
21.444	1.058
23.679	1.121

$$\begin{bmatrix} 51.3 & 0.037 \\ 0.037 & 0.049 \end{bmatrix}$$

Correlation Exists

Correlation Eliminated

New Basis:
Unit of Transformation similar to transforming scaler Units.

Explained Variance Ratio & Ranking PCAs

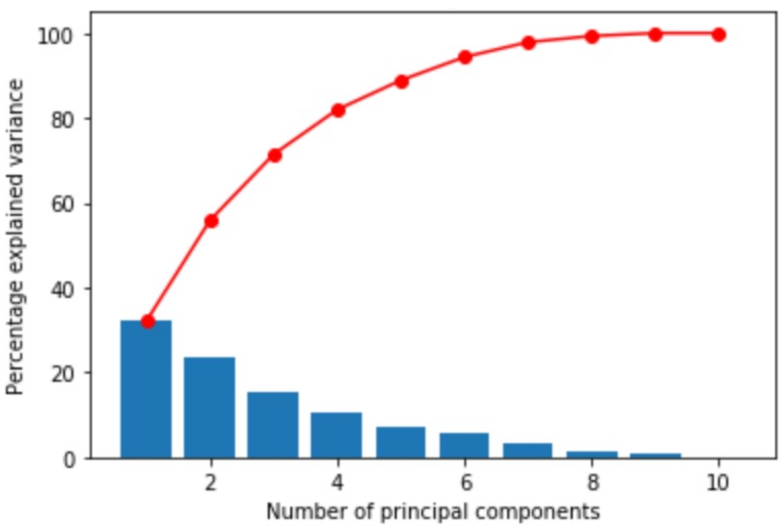
The **Eigenvalues** are indicators of **the variance explained by that particular eigenvector**. So higher is the eigenvalue, higher is the variance explained by that eigenvector and hence that direction is more important for us.

Our goal was to find a new set of ***basis vectors*** where the covariance matrix gets diagonalised. The new set of basis vectors are in fact the **eigenvectors** of the Covariance Matrix. Therefore, these eigenvectors are the ***Principal Components*** of our original dataset. In other words, these eigenvectors are the directions that capture maximum variance.

When these Principal Components are listed based on the higher to lower values of their corresponding Eigenvalues, we get our list of Principal Components. With Eigenvalues indicating the amount of variance explained by them, we are then to decide how many of the top ones we should choose in order to bring down the number of dimensions retaining enough information.

E.g. If the original Dataset had 70 dimensions, and the top 6 Principal Components explain about 95% of the variance in the dataset, it might be enough to choose only the top 6 PCAs for our further Machine Learning processes.

Scree Plot



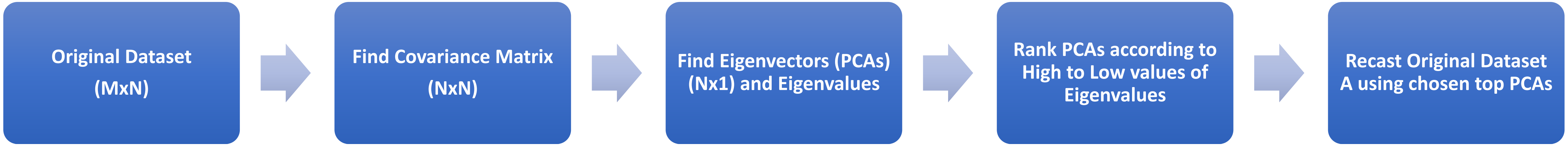
Represent the data using the chosen PCAs

The last step of Principal Component Analysis is to recast the Original Dataset using the new Principal Components found (say, we have chosen top 6).

This process will give us the transformed dataset with 6 Features that retain 95% of the variance information from the original dataset.

This transformed dataset may then be used for further ML.

PCA Process Revisited



Original Dataset (MxN)

Find Covariance Matrix (NxN)

Find Eigenvectors (PCAs) (Nx1) and Eigenvalues

Rank PCAs according to High to Low values of Eigenvalues

Recast Original Dataset A using chosen top PCAs

Project

Python Demo