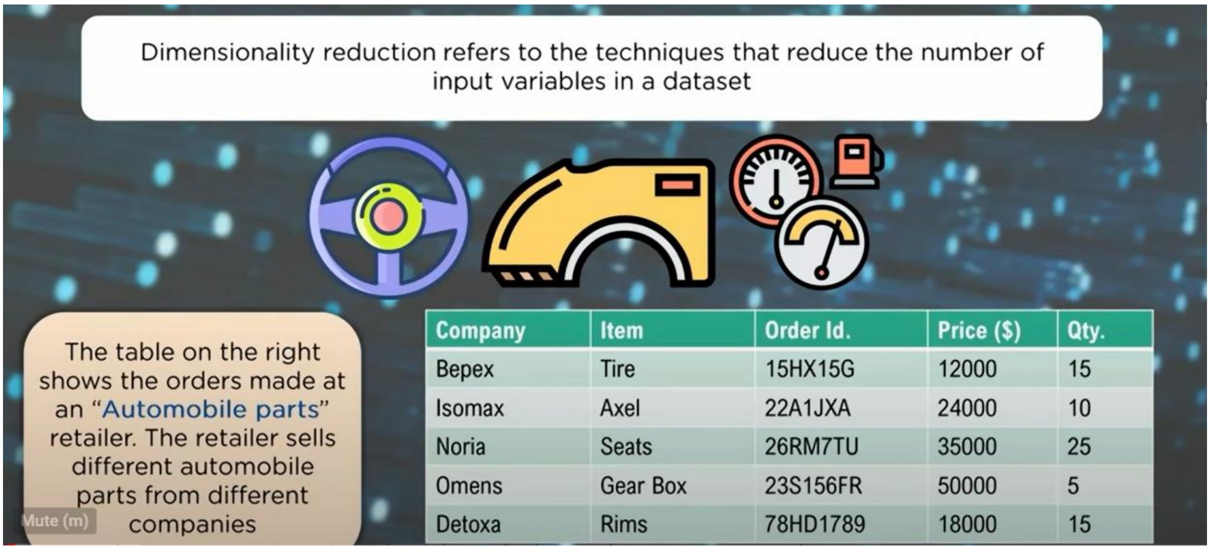
Principal Component Analysis is an **unsupervised learning** algorithm that is used for the **dimensionality reduction** in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the **Principal Components**.

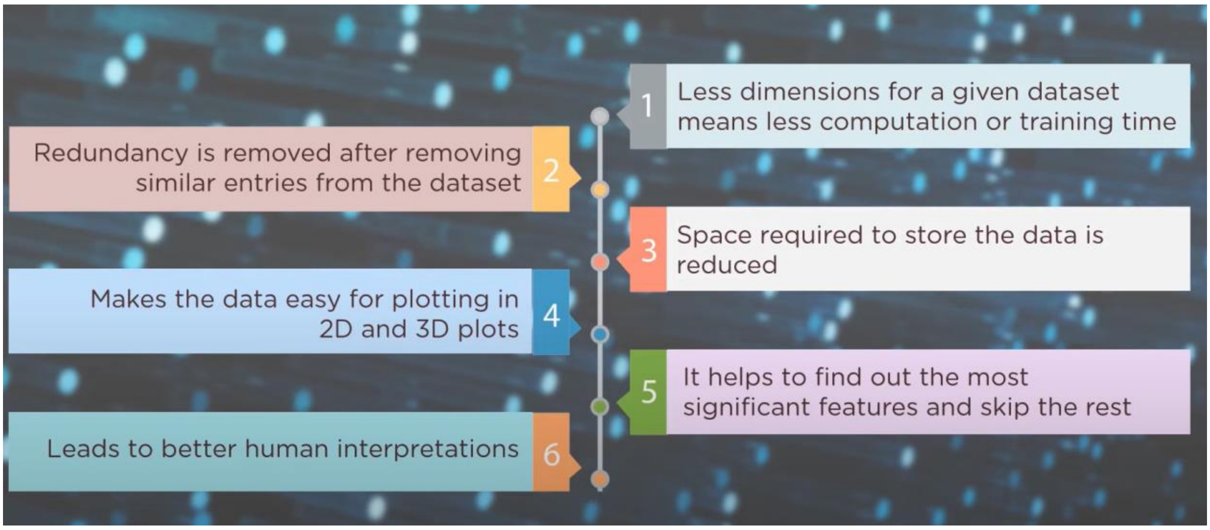
Some real-world applications of PCA are ***image processing, movie recommendation system etc.*** It is a feature extraction technique, so it contains the important variables and drops the least important variable.

DIMENSIONALITY REDUCTION

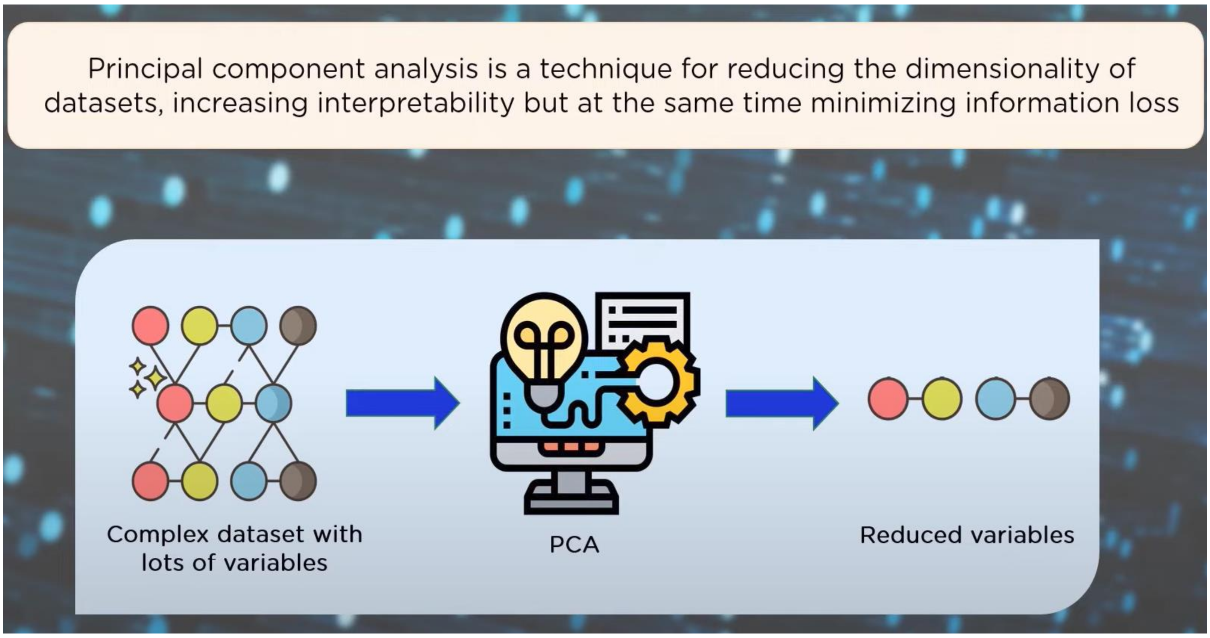


Company	Item	Order Id.	Price ($)	Qty.
Bepex	Tire	15HX15G	12000	15
Isomax	Axel	22A1JXA	24000	10
Noria	Seats	26RM7TU	35000	25
Omens	Gear Box	23S156FR	50000	5
Detoxa	Rims	78HD1789	18000	15

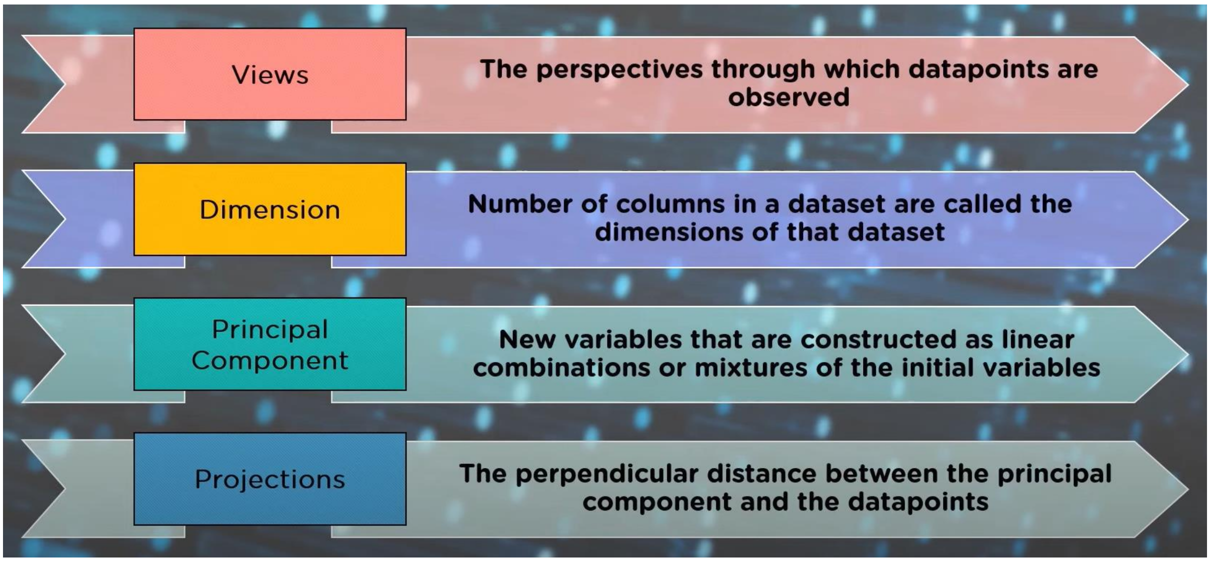
WHY DIMENSIONALITY REDUCTION



WHAT IS PRINCIPAL COMPONENT ANALYSIS



FEW TERMINOLOGIES

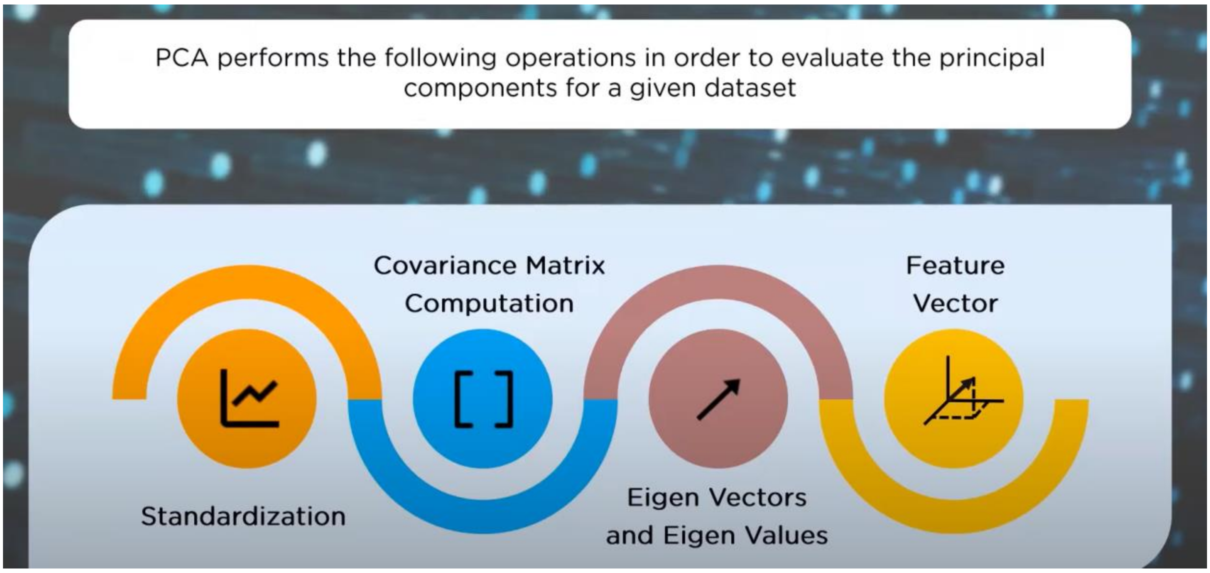


Principal Components in PCA

As described above, the transformed new features or the output of PCA are the Principal Components. The number of these PCs are either equal to or less than the original features present in the dataset. Some properties of these principal components are given below:

- The principal component must be the linear combination of the original features.
- These components are orthogonal, i.e., the correlation between a pair of variables is zero.
- The importance of each component decreases when going to 1 to n, it means the 1 PC has the most importance, and n PC will have the least importance.

HOW PCA WORKS?



Steps for PCA algorithm

1. **Getting the dataset**
   Firstly, we need to take the input dataset and divide it into two subparts X and Y, where X is the training set, and Y is the validation set.
2. **Representing data into a structure**
   Now we will represent our dataset into a structure. Such as we will represent the two-dimensional matrix of independent variable X. Here each row corresponds to the data items, and the column corresponds to the Features. The number of columns is the dimensions of the dataset.
3. **Standardizing the data**
   In this step, we will standardize our dataset. Such as in a particular column, the features with high variance are more important compared to the features with lower variance.
   If the importance of features is independent of the variance of the feature, then we will divide each data item in a column with the standard deviation of the column. Here we will name the matrix as Z.
4. **Calculating the Covariance of Z**
   To calculate the covariance of Z, we will take the matrix Z, and will transpose it. After transpose, we will multiply it by Z. The output matrix will be the Covariance matrix of Z.
5. **Calculating the Eigen Values and Eigen Vectors**
   Now we need to calculate the eigenvalues and eigenvectors for the resultant

covariance matrix Z. Eigenvectors or the covariance matrix are the directions of the axes with high information. And the coefficients of these eigenvectors are defined as the eigenvalues.

6. **Sorting the Eigen Vectors**
   In this step, we will take all the eigenvalues and will sort them in decreasing order, which means from largest to smallest. And simultaneously sort the eigenvectors accordingly in matrix P of eigenvalues. The resultant matrix will be named as P*.
7. **Calculating the new features Or Principal Components**
   Here we will calculate the new features. To do this, we will multiply the P* matrix to the Z. In the resultant matrix Z*, each observation is the linear combination of original features. Each column of the Z* matrix is independent of each other.
8. **Remove less or unimportant features from the new dataset.**
   The new feature set has occurred, so we will decide here what to keep and what to remove. It means, we will only keep the relevant or important features in the new dataset, and unimportant features will be removed out.

IMPORTANT PROPERTIES OF PCA

ADVANTAGES OF PCA

1. Removes correlated features – Such phenomenon is called collinearity.

2. Improves ML algorithm performance – With number of features being reduced with PCA, time taken to train model is significantly reduced.

3. Reduce overfitting

DISADVANTAGES OF PCA

1. Independent variables are now less interpretable
2. Information loss – If right number of components are not chosen, data loss may occur.
3. Feature scaling – Since PCA is variance maximising exercise, it requires features to be scaled prior to processing