Dimensionality Reduction

As a result of the curse of dimensionality, too many features can lead to worse performance. Often, data can be represented by fewer dimensions (features). There are two ways to reduce dimensionality:

1. Select a subset of features that are "important"
2. Combine features using linear/non-linear transformations

Principal Component Analysis

Principal component analysis (PCA) is a dimensionality reduction technique that creates new features by applying linear transformations on a combination of the original features. The resulting features are called principal components, on which the original data will be projected.

Each principal component is a vector and are orthogonal to each other. Their relative importance is determined by comparing how much of the variance within the original data they each explain/preserve.

Singular value decomposition (SVD) enables us to find this information by obtaining the diagonal matrix. Its non-zero entries along the diagonal represent each eigenvector(principal component)'s eigenvalue/length. The **bigger** the value, the **more important** the vector.

*Note: It is important to scale the data **prior to** applying PCA because distance is an important aspect considered for the algorithm.

Syntax

The syntax consists of importing the class containing the method:

```
from sklearn.decomposition import PCA
```

creating the instance of the class:

```
pca=PCA(n_components=3)
```

and fit the instance and create a transformed version of the data:

```
X_trans=pca.fit_transform(X_train
```