1. Principal Component Analysis

Definition:

Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

It can also be defined as a statistical method that is used for feature extraction.

Example:

1PCA can be used for high-dimensional and correlated data.

Let's consider our data set is 2-dimensional with 2 variables x, y and that the eigenvectors and eigenvalues of the covariance matrix are as follows:

$$v1 = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix} \qquad \lambda_1 = 1.284028$$

$$v2 = \begin{bmatrix} -0.7351785 \\ 0.6778736 \end{bmatrix} \qquad \lambda_2 = 0.04908323$$

While ranking the eigenvalues in descending order, we get $\lambda1>\lambda2$, which means that the eigenvector that corresponds to the first principal component (PC1) is v1 and the one that corresponds to the second component (PC2) isv2.

After having the principal components, to compute the percentage of variance (information) accounted for by each component, we divide the eigenvalue of each component by the sum of eigenvalues. If we apply this on the example above, we find that PC1 and PC2 carry respectively 96% and 4% of the variance of the data.

Computing the eigenvectors and ordering them by their eigenvalues in descending order, allow us to find the principal components in order of significance.

2. Model Parameters and Hyperparameters

Definition:

Model Parameters: These are the parameters in the model that should be determined using the training data set. These are the fitted parameters.

Hyperparameters: These are adjustable parameters that should be tuned in order to obtain a model with optimal performance.

Example:

Suppose we want to build a simple linear regression model using an m-dimensional training data set. Then our model can be written as:

$$\hat{y}_i = \sum_{j=0}^{m} X_{ij} w_j$$

where **X** is the predictor matrix, and **W** are the weights.

Here w_0, w_1, w_2, …,w_m are the **model parameters**. If the model uses the gradient descent algorithm to minimize the objective function in order to determine the weights w_0, w_1, w_2, …,w_m, we can have an optimizer such as GradientDescent(eta, n_iter).

Here eta (learning rate) and n_iter (number of iterations) are the **hyperparameters** that would have to be adjusted in order to obtain the best values for the model parameters w_0, w_1, w_2, …,w_m.

3. Dimentionality Reduction

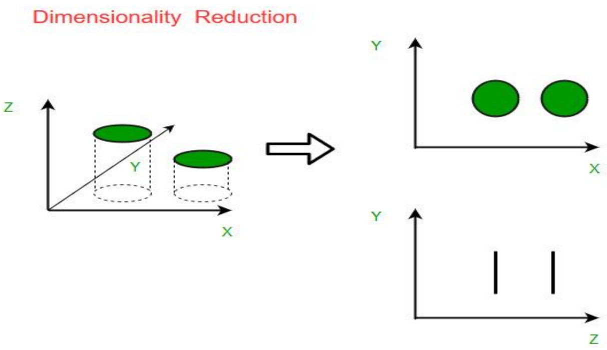
Definition:

Dimensionality reduction is the process of reducing the number of random variables under consideration, by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.

Example:

An example of dimensionality reduction can be discussed through a simple e-mail classification problem, where we need to classify whether the e-mail is spam or not. This can involve a large number of features, such as whether or not the e-mail has a generic title, the content of the e-mail, whether the e-mail uses a template, etc. However, some of these features may overlap. In another condition, a classification problem that relies on both humidity and rainfall can be collapsed into just one underlying feature, since both of the aforementioned are correlated to a high degree. Hence, we can reduce the number of features in such problems.

A 3-D classification problem can be hard to visualize, whereas a 2-D one can be mapped to a simple 2 dimensional space, and a 1-D problem to a simple line. The below figure illustrates this concept, where a 3-D feature space is split into two 1-D feature spaces, and later, if found to be correlated, the number of features can be reduced even further.



4. Reinforcement Learning

Definition:

It is about taking suitable action to maximize reward in a particular situation.

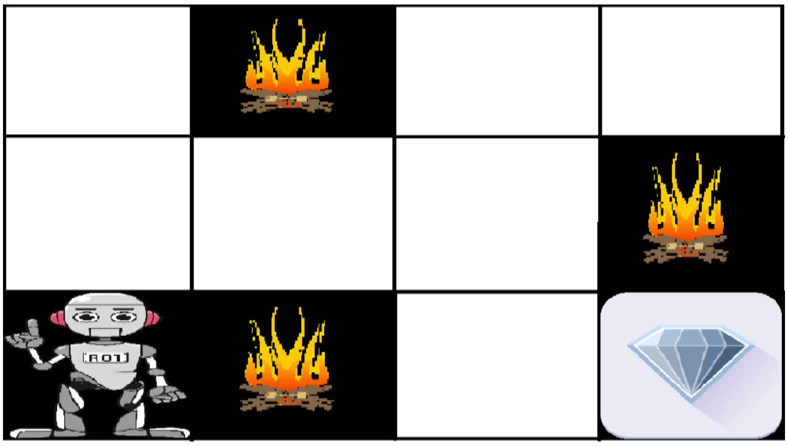
It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation.

In the absence of a training dataset, it is bound to learn from its experience.

Example:

The problem is as follows: We have an agent and a reward, with many hurdles in between. The agent is supposed to find the best possible path to reach the reward.

The following problem explains the problem more easily.



The above image shows the robot, diamond, and fire. The goal of the robot is to get the reward that is the diamond and avoid the hurdles that are fired. The robot learns by trying all the possible paths and then choosing the path which gives him the reward with the least hurdles.

Each right step will give the robot a reward and each wrong step will subtract the reward of the robot. The total reward will be calculated when it reaches the final reward that is the diamond.

5. Bias-Variance Tradeoff

Definition:

In statistics and machine learning, the bias–variance tradeoff is the property of a model that the variance of the parameter estimated across samples will be reduced by increasing the bias in the estimated parameters.

The bias error is an error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relations between features and target outputs (underfitting)

The variance is an error from sensitivity to small fluctuations in the training set. High variance may result from an algorithm modeling the random noise in the training data (overfitting).

Total Error = Bias + Variance

Examples:

1. Linear machine learning algorithms often have a high bias but a low variance.

2. Nonlinear machine learning algorithms often have a low bias but a high variance.

3. The k-nearest neighbour algorithm has low bias and high variance, but the trade-off can be changed by increasing the value of k which increases the number of neighbours that contribute to the prediction and in turn increases the bias of the model.

4. The support vector machine algorithm has low bias and high variance, but the trade-off can be changed by increasing the C parameter that influences the number of violations of the margin allowed in the training data which increases the bias but decreases the variance.