

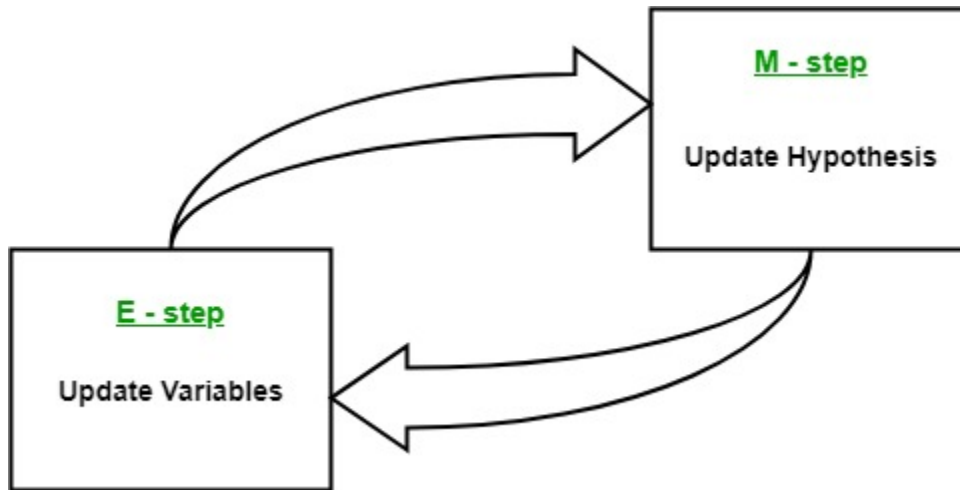
Expectation-Maximization Algorithm

Introduction

- The expectation-Maximization algorithm can be used to handle situations where **variables are partially observable**. When certain variables are observable, we can use those instances to learn and estimate their values. Then, we can predict the values of these variables in instances when it is not observable.
- EM algorithm is applicable **to latent variables, which are variables that are not directly observable but are inferred from the values of other observed variables**. By leveraging the known general form of the probability distribution governing these latent variables, the EM algorithm can predict their values.
- The EM algorithm serves as the foundation for many unsupervised clustering algorithms in the field of machine learning. It provides a framework to find the **local maximum likelihood parameters of a statistical model and infer latent variables in cases where data is missing or incomplete**.

Algorithm

- The Expectation-Maximization (EM) algorithm is an iterative optimization method that combines different [unsupervised machine learning](#) algorithms to find maximum likelihood or maximum posterior estimates of parameters in statistical models that involve unobserved latent variables.
- The EM algorithm is commonly used for latent variable models and can handle missing data. It consists of an **estimation step (E-step)** and a **maximization step (M-step)**, forming an iterative process to improve model fit.
- In the E step, **the algorithm computes the latent variables** i.e. expectation of the log-likelihood using the current parameter estimates.
- In the M step, **the algorithm determines the parameters that maximize the expected log-likelihood obtained in the E step**, and corresponding model parameters are updated based on the estimated latent variables.



- In the E step, the algorithm computes the latent variables i.e. expectation of the log-likelihood using the current parameter estimates.
- In the M step, the algorithm determines the parameters that maximize the expected log-likelihood obtained in the E step, and corresponding model parameters are updated based on the estimated latent variables.

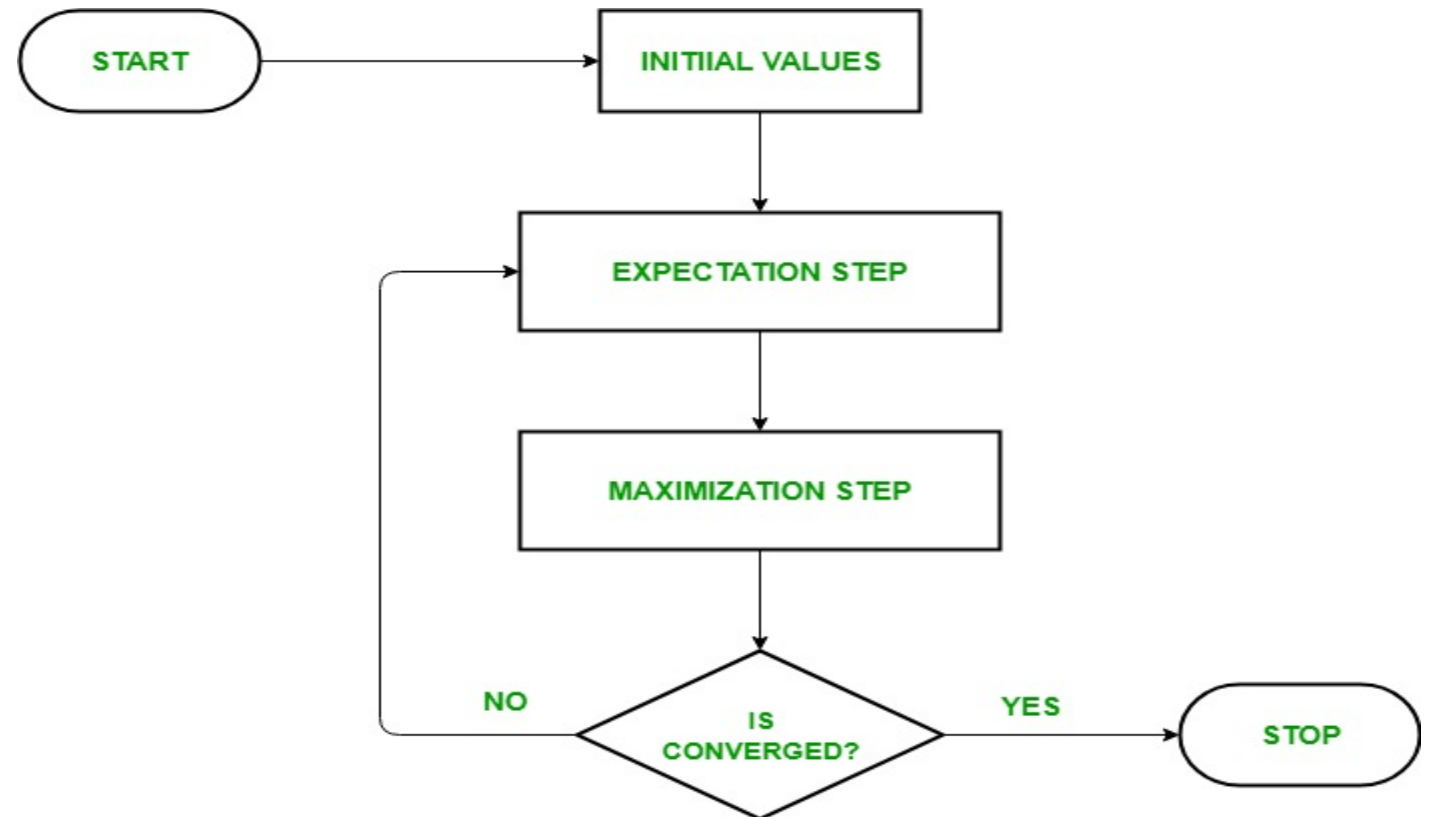
By iteratively repeating these steps, the EM algorithm seeks to maximize the likelihood of the observed data. It is commonly used for unsupervised learning tasks, such as clustering, where latent variables are inferred and has applications in various fields, including machine learning, computer vision, and natural language processing.

Key Terms in Expectation-Maximization (EM) Algorithm

- **Latent Variables:** Latent variables are unobserved variables in statistical models that can only be inferred indirectly through their effects on observable variables. They cannot be directly measured but can be detected by their impact on the observable variables.
- **Likelihood:** It is the probability of observing the given data given the parameters of the model. In the EM algorithm, the goal is to find the parameters that maximize the likelihood.
- **Log-Likelihood:** It is the logarithm of the likelihood function, which measures the goodness of fit between the observed data and the model. EM algorithm seeks to maximize the log-likelihood.
- **Maximum Likelihood Estimation (MLE):** MLE is a method to estimate the parameters of a statistical model by finding the parameter values that maximize the likelihood function, which measures how well the model explains the observed data.
- **Posterior Probability:** In the context of Bayesian inference, the EM algorithm can be extended to estimate the maximum a posteriori (MAP) estimates, where the posterior probability of the parameters is calculated based on the prior distribution and the likelihood function.
- **Expectation (E) Step:** The E-step of the EM algorithm computes the expected value or posterior probability of the latent variables given the observed data and current parameter estimates. It involves calculating the probabilities of each latent variable for each data point.
- **Maximization (M) Step:** The M-step of the EM algorithm updates the parameter estimates by maximizing the expected log-likelihood obtained from the E-step. It involves finding the parameter values that optimize the likelihood function, typically through numerical optimization methods.
- **Convergence:** Convergence refers to the condition when the EM algorithm has reached a stable solution. It is typically determined by checking if the change in the log-likelihood or the parameter estimates falls below a predefined threshold.

How Expectation-Maximization (EM) Algorithm Works

- The essence of the Expectation-Maximization algorithm is to use the available observed data of the dataset to estimate the missing data and then use that data to update the values of the parameters.



1. Initialization:

1. Initially, a set of initial values of the parameters are considered. A set of incomplete observed data is given to the system with the assumption that the observed data comes from a specific model.

2. E-Step (Expectation Step):

In this step, we use the observed data in order to estimate or guess the values of the missing or incomplete data. It is basically used to update the variables.

1. Compute the posterior probability or responsibility of each latent variable given the observed data and current parameter estimates.
2. Estimate the missing or incomplete data values using the current parameter estimates.
3. Compute the log-likelihood of the observed data based on the current parameter estimates and estimated missing data.

3. M-step (Maximization Step):

In this step, we use the complete data generated in the preceding “Expectation” – step in order to update the values of the parameters. It is basically used to update the hypothesis.

1. Update the parameters of the model by maximizing the expected complete data log-likelihood obtained from the E-step.
2. This typically involves solving optimization problems to find the parameter values that maximize the log-likelihood.
3. The specific optimization technique used depends on the nature of the problem and the model being used.

4. Convergence:

In this step, it is checked whether the values are converging or not, if yes, then stop otherwise repeat *step-2* and *step-3* i.e. “Expectation” – step and “Maximization” – step until the convergence occurs.

1. Check for convergence by comparing the change in log-likelihood or the parameter values between iterations.
2. If the change is below a predefined threshold, stop and consider the algorithm converged.
3. Otherwise, go back to the E-step and repeat the process until convergence is achieved.

Applications of the EM algorithm

- It can be used to fill in the missing data in a sample.
- It can be used as the basis of unsupervised learning of clusters.
- It can be used for the purpose of estimating the parameters of the Hidden Markov Model (HMM).
- It can be used for discovering the values of latent variables.

Advantages of EM algorithm

- It is always guaranteed that likelihood will increase with each iteration.
- The E-step and M-step are often pretty easy for many problems in terms of implementation.
- Solutions to the M-steps often exist in the closed form.

Disadvantages of EM algorithm

- It has slow convergence.
- It makes convergence to the local optima only.
- It requires both the probabilities, forward and backward (numerical optimization requires only forward probability).

Problems with solution links

THANK YOU