Stochastic Gradient Descent for Latent Factor Model - Pseudocode

Kaiqi Zhao

October 2022

1 Overview

Gradient descent is used to find the local optima for an objective function. The objective function can be considered as a hyperplane in the high-dimensional optimization space. The idea is to start with a random location in the space. At each step, take the steepest direction to move until reaching the local minima.

Stochastic gradient descent is to approximate the gradient at each step to allow smaller memory requirements and easier implementation. The approximation may result in "noisy" updates. Please refer to the lecture notes for more about the theory.

2 Pseudocode

Here is a very simple version of an SGD optimizer for the latent factor model. It stops after max_epoch number of epoches have been completed. Note that only the training dataset D_{train} is used for the optimization/training, and the number of factors is k.

Algorithm 1 SGD Optimizer for Latent Factor Model

Initialize user and item profile vectors $\mathbf{Q} \in \mathbb{R}^{n \times k}$, $\mathbf{P} \in \mathbb{R}^{m \times k}$
for $epoch = 0$ to $max\_epoch - 1$ **do**
  for each (i, j, r_{ij}) in D_{train} **do** ▷ For each rating of user i, item j
    $\hat{r}_{ij} \leftarrow \mathbf{Q}[i, :].dot(\mathbf{P}[j, :])$ ▷ Get rating prediction
    for $f = 0$ to $k - 1$ **do** ▷ Update for the f-th factor
      $\nabla_{q_{if}} L_{ij} \leftarrow -2 * (r_{ij} - \hat{r}_{ij}) * \mathbf{P}[j, f] + 2 * \lambda_1 * \mathbf{Q}[i, f]$
      $\nabla_{p_{jf}} L_{ij} \leftarrow -2 * (r_{ij} - \hat{r}_{ij}) * \mathbf{Q}[i, f] + 2 * \lambda_2 * \mathbf{P}[j, f]$
      $\mathbf{Q}[i, f] \leftarrow \mathbf{Q}[i, f] - \eta * \nabla_{q_{if}} L_{ij}$
      $\mathbf{P}[j, f] \leftarrow \mathbf{P}[j, f] - \eta * \nabla_{p_{jf}} L_{ij}$
    end for
  end for
end for