NEURAL NETWORKS AND DEEP LEARNING CST 395 CS 5TH SEMESTER HONORS COURSE- Dr Binu V P, 9847390760

CS 5th Semester Honors course for the Computer Science at KTU- Dr Binu V P

Gradient Descent, Stochastic Gradient Descent, Batch Gradient Descent

September 22, 2022

Gradient Descent

Gradient descent is an optimization algorithm often used for finding the weights or coefficients of machine learning algorithms, such as artificial neural networks and logistic regression.

It works by having the model make predictions on training data and using the error on the predictions to update the model in such a way as to reduce the error.

The goal of the algorithm is to find model parameters (e.g. coefficients or weights) that minimize the error of the model on the training dataset. It does this by making changes to the model that move it along a gradient or slope of errors down toward a minimum error value. This gives the algorithm its name of “gradient descent.”

Types of Gradient Descent

Gradient descent can vary in terms of the number of training patterns used to calculate error; that is in turn used to update the model.

The number of patterns used to calculate the error includes how stable the gradient is that is used to update the model. We will see that there is a tension in gradient descent configurations of computational efficiency and the fidelity of the error gradient.

The three main flavors of gradient descent are **batch, stochastic, and mini-batch.**

Stochastic Gradient Descent

Stochastic gradient descent, often abbreviated SGD, is a variation of the gradient descent algorithm that calculates the error and updates the model for each example in the training dataset.

The update of the model for each training example means that stochastic gradient descent is often called an online machine learning algorithm.

Upsides

The frequent updates immediately give an insight into the performance of the model and the rate of improvement.

This variant of gradient descent may be the simplest to understand and implement, especially for beginners.

The increased model update frequency can result in faster learning on some problems.

The noisy update process can allow the model to avoid local minima (e.g. premature convergence).

Downsides

Updating the model so frequently is more computationally expensive than other configurations of gradient descent, taking significantly longer to train models on large datasets.

The frequent updates can result in a noisy gradient signal, which may cause the model parameters and in turn the model error to jump around (have a higher variance over training epochs).

The noisy learning process down the error gradient can also make it hard for the algorithm to settle on an error minimum for the model.

Batch Gradient Descent

Batch gradient descent is a variation of the gradient descent algorithm that calculates the error for each example in the training dataset, but only updates the model after all training examples have been evaluated.

One cycle through the entire training dataset is called a **training epoch.** Therefore, it is often said that batch gradient descent performs model updates at the end of each training epoch.

Upsides

Fewer updates to the model means this variant of gradient descent is more computationally efficient than stochastic gradient descent.

The decreased update frequency results in a more stable error gradient and may result

in a more stable convergence on some problems.
The separation of the calculation of prediction errors and the model update lends the algorithm to parallel processing based implementations.

Downsides
The more stable error gradient may result in premature convergence of the model to a less optimal set of parameters.
The updates at the end of the training epoch require the additional complexity of accumulating prediction errors across all training examples.
Commonly, batch gradient descent is implemented in such a way that it requires the entire training dataset in memory and available to the algorithm.
Model updates, and in turn training speed, may become very slow for large datasets.

Mini-Batch Gradient Descent

Mini-batch gradient descent is a variation of the gradient descent algorithm that splits the training dataset into small batches that are used to calculate model error and update model coefficients.

Implementations may choose to sum the gradient over the mini-batch which further reduces the variance of the gradient.

Mini-batch gradient descent seeks to find a balance between the robustness of stochastic gradient descent and the efficiency of batch gradient descent. It is the most common implementation of gradient descent used in the field of deep learning.

Upsides
The model update frequency is higher than batch gradient descent which allows for a more robust convergence, avoiding local minima.
The batched updates provide a computationally more efficient process than stochastic gradient descent.
The batching allows both the efficiency of not having all training data in memory and algorithm implementations.
Downsides
Mini-batch requires the configuration of an additional “mini-batch size” hyperparameter for the learning algorithm.
Error information must be accumulated across mini-batches of training examples like batch gradient descent.

How to Configure Mini-Batch Gradient Descent

Mini-batch gradient descent is the recommended variant of gradient descent for most applications, especially in deep learning.

Mini-batch sizes, commonly called “batch sizes” for brevity, are often tuned to an aspect of the computational architecture on which the implementation is being executed. Such as a power of two that fits the memory requirements of the GPU or CPU hardware like 32, 64, 128, 256, and so on.

Batch size is a slider on the learning process.Small values give a learning process that converges quickly at the cost of noise in the training process.
Large values give a learning process that converges slowly with accurate estimates of the error gradient.

A good default for batch size might be 32.
It is a good idea to review learning curves of model validation error against training time with different batch sizes when tuning the batch size.
Tune batch size and learning rate after tuning all other hyperparameters.

Popular posts from this blog

NEURAL NETWORKS AND DEEP LEARNING CST 395 CS 5TH SEMESTER HONORS COURSE NOTES - Dr Binu V P, 9847390760

October 03, 2022

About Me Syllabus Question Paper Dec 2022 Module 1 (Basics of Machine Learning) Overview of Machine Learning Machine Learning Algorithm Linear Regression Capacity, Overfitting and Underfitting Regularization Hyperparameters and Validatio …

READ MORE

Machine Learning Algorithm

October 01, 2022

A machine learning algorithm is an algorithm that is able to learn from data. But what do we mean by learning? Mitchell (1997) provides the definition "A computer program is said to learn from experience E with respect to some class of tasks T and …

READ MORE

Syllabus

October 01, 2022

Syllabus Module - 1 (Basics of Machine Learning) Machine Learning basics - Learning algorithms - Supervised, Unsupervised, Reinforcement, overfitting, Underfitting, Hyper parameters and Validation sets, Estimators -Bias and Variance. Challenges …

READ MORE

DR.BINU V P-9847390760

Associate Professor and Head Dept of Computer Science-IHRD Karunagappally.Love to share the thoughts and views .I play lot of Games and basically an Athlet in my school and college days.I never keep my studies as a second option.Stood first In MTech and BTech.I did my resea …

Archive

Report Abuse