Gradient Descent and its applications in Machine learning

Gradient Descent is a common optimization algorithm used in machine learning. Its main purpose is to optimize a loss function, which measures how much a model's predictions deviate from the actual outputs. The idea is derived from calculus: the gradient of a function points in the steepest direction of rise, so moving in the negative direction lowers the value of the function. By repeatedly changing the parameters of the model slightly in the direction opposite to the gradient, the algorithm moves step by step towards values that minimize the error.

This method is significant in that the majority of machine learning models have very complicated functions with many parameters. Solving them directly to acquire the optimal parameters is infeasible or very costly. Gradient descent provides a practical way to modify parameters in increments until the model realizes the best fit of the data. It is effective, adaptable, and applicable to a wide range of problems, thus referred to as the "engine" of current artificial intelligence.

Applications of Gradient Descent in Machine Learning

1. **Linear Regression**

In linear regression, the goal is to find the best-fitting line that estimates continuous values from input features. Gradient descent adjusts the slope and the intercept of the line so that the gap between predicted values and actual values is minimized. This makes it possible for the algorithm to handle large data sets where solving equations directly is impossible.

2. **Logistic Regression**

Logistic regression is used for classification, for example, spam or not spam classification in an email. Gradient descent is used to adjust the model to reduce the classification errors by shifting the decision boundary. This renders the algorithm suitable for medical diagnostics, fraud detection, and customer behavior analysis.

3. **Neural Networks and Deep Learning**

The most common application of gradient descent is for neural network training. Every network may have millions of parameters (weights and bias), which cannot be manually optimized. Gradient descent with backpropagation allows the network to learn iteratively from data. It is impossible to train deep models for image processing, speech recognition, or language translation without gradient descent.

4. **Recommendation Systems**

Web services like Netflix, YouTube, or Amazon use recommendation systems to make personal experiences for users. Gradient descent minimizes matrix factorization methods where user preferences and product features are trained to be minimized through error prediction. This way, the system improves recommendations over time based on what a user does.

5. **Natural Language Processing (NLP)**

Gradient descent powers models that learn and generate human language. Word embedding techniques, such as Word2Vec, rely on loss functions optimized to represent word-to-word relationships. Modern-day transformer-based models, such as BERT or GPT, use gradient descent when trained on massive text corpora, with application areas like chatbots, translation, and sentiment analysis.

6. **Computer Vision**

Operations such as image classification, object detection, and analyzing medical images are based on convolutional neural networks. The models need gradient descent to adjust their millions of parameters. For example, in disease diagnosis using X-ray images, gradient descent makes the model more accurate with every training.

7. **Support Vector Machines (SVMs)**

While traditionally solved using mathematical programming, gradient descent can also be employed to maximize large-scale SVMs. This enables it to deal with very large datasets that are not efficiently solvable by traditional optimization methods.

References

- https://www.datacamp.com/tutorial/tutorial-gradient-descent
- https://www.geeksforgeeks.org/machine-learning/gradient-descent-algorithm-and-its-variants/
- https://www.udacity.com/blog/2025/02/gradient-descent-optimization-a-simple-guide-for-beginners.html
- https://www.ibm.com/think/topics/gradient-descent?