Lesson 6: Least Mean Square Algorithm

Table of Contents

6.1 Introduction

In this lesson, we will explore various techniques for solving problems related to signal processing and adaptive filtering. First, we will examine the Wiener-Hopf equation, which is a powerful tool for solving linear filtering problems. Next, we will delve into the steepest descent method, which is a method for minimizing a function by taking the derivative with respect to the variables and moving in the opposite direction of the gradient. We will also explore the least mean squares (LMS) algorithm, which is a widely used algorithm for adaptive filtering. Finally, we will discuss the learning curve, which is a graphical representation of the performance of a model as the amount of training data increases. Together, these techniques will give us a comprehensive understanding of how to approach and solve problems in signal processing and adaptive filtering.

6.2 Wiener-Hopf equation

The Wiener–Hopf method has been motivated by interdisciplinary interests ever since its inception. It resulted from a collaboration between Norbert Wiener, who worked on stochastic processes, and Eberhard Hopf, who worked on partial differential equations (PDEs). The method was first described in their joint article [1] where they study a convolution-type integral equation for $f(x)$ on a semi-axis

$$\int_0^\infty k(x-t)f(t)\,\mathrm{d}t = g(x), \quad x>0,$$

with $k(x)$ and $g(x)$ given.

The study of this equation was principally motivated by an interest of one of the authors, in a differential equation governing the radiation equilibrium of stars . The Wiener–Hopf equation arises from extending the integral equation into $x<0$,

$$\int_0^\infty k(x-t)f(t)\,\mathrm{d}t = \begin{cases} g(x), & \text{for } x>0, \\ h(x), & \text{for } x\le 0, \end{cases}$$

for some function $h(x)$. Note that, although $h(x)$ is unknown, it is not independent as it is uniquely defined by the left-hand side of once $f(x)$ is determined. Applying the Fourier transform to results in an equation of type (the interaction of a Fourier transform with its convolution has to be used, and analytic properties of half-range Fourier transforms employed[1]). This laid the foundation for the study of scalar Wiener–Hopf equations.

6.3 Steepest descent method/Gradient descent method

<<Insert content>>

6.4 Least mean squares (LMS) algorithm

The Least Mean Square (LMS) algorithm is a type of supervised learning algorithm that is commonly used to train artificial neural networks, particularly adaptive filters. The LMS algorithm is used to adjust the weights of a neural network in order to minimize the mean square error between the predicted output and the actual output. The LMS algorithm works by iteratively updating the weights of the neural network based on the current input and output. The update is done by calculating the error between the predicted output and the actual output, and then adjusting the weights in the opposite direction of the error gradient. This process is repeated for multiple iterations, until the error is minimized.

One of the main advantages of the LMS algorithm is its simplicity and ease of implementation. It is also highly robust and can work well even in the presence of noise. Additionally, it does not require the computation of the second derivative of the error function, which makes it computationally efficient. However, LMS algorithm may converge to a suboptimal solution if the input data is not uniformly distributed and the step size used for weight update is not carefully chosen. It also requires a large number of iterations to converge, which can be time-consuming.

6.5 Learning curve

A learning curve is a graphical representation of how a model or algorithm's performance improves with increasing amounts of training data. It is commonly used to evaluate machine learning algorithms, particularly in supervised learning. The x-axis typically represents the number of training examples, while the y-axis represents a measure of performance, such as accuracy or error.

A key characteristic of a learning curve is the steady-state error value, which is the error value that the algorithm converges to as the number of training examples increases. The steady-state error value is an important measure of the performance of a learning algorithm, as it represents the error that the algorithm is able to achieve with a large amount of training data.

For a Least Mean Squares (LMS) algorithm, the steady-state error is always larger than the stationary error for a given Wiener filter. The difference between these two errors is called the residual error, and is represented as Jex = J(¥) - Jmin, where J(¥) is the steady-state error, Jmin is the minimum error, and Jex is the residual error.

6.6 Lesson 6 Questions

1. What is the Wiener-Hopf equation and what types of problems is it used to solve?
2. How does the Wiener-Hopf equation decompose an integral equation into two separate integral equations?
3. What are the advantages of using the Wiener-Hopf equation?
4. What is the steepest descent method and what types of problems is it used to solve?
5. How does the steepest descent method find the minimum of a given function?
6. What are the advantages and disadvantages of using the steepest descent method?
7. What is the LMS algorithm and what types of problems is it used to solve?

8. How does the LMS algorithm find the optimal filter coefficients?
9. What are the advantages and disadvantages of using the LMS algorithm?
10. What is a learning curve and what types of problems is it used to evaluate?
11. What is the steady-state error value, and why is it an important measure of the performance of a learning algorithm?
12. How can a learning curve be used to identify the best model or algorithm?