Optimal Filtering
VI: Steepest Descent

ECE416 Adaptive Algorithms

Prof. Fred L. Fontaine

Department of Electrical Engineering
The Cooper Union

Spring 2022

Linear Regression Problem

Let us go back to the general linear regression problem, of which the Wiener filter and linear prediction are special cases:

Given $\mathbf{u}$, d with $R = E\left(\mathbf{u}\,\mathbf{u}^H\right)$, $\mathbf{p} = E\left(\mathbf{u}\,d^*\right)$, $\sigma_d^2 = E\left(|d|^2\right)$, find $\mathbf{w}$ to achieve MMSE (minimum mean-square error) for:

$$e = d - \mathbf{w}^H \mathbf{u}$$

The solution is:

$$\mathbf{w}_0 = R^{-1}\mathbf{p}$$

and the cost function $J\left(\mathbf{w}\right) = E\left(|e|^2\right)$ can be expressed as:

$$J\left(\mathbf{w}\right) = J_{\text{min}} + \left(\mathbf{w} - \mathbf{w}_0\right)^H R\left(\mathbf{w} - \mathbf{w}_0\right)$$

where:

$$J_{\text{min}} = \sigma_d^2 - \mathbf{w}_0^H R \mathbf{w}_0$$

Steepest Gradient Descent (SGD)

Also called *steepest descent*.

Instead of directly inverting R, we want to iterate towards a solution.

Given an initial condition $\mathbf{w}[0]$, we propose an update:

$$\mathbf{w}[n+1] = \mathbf{w}[n] - \frac{\mu}{2} \left. \nabla J \right|_{\mathbf{w}=\mathbf{w}[n]}$$

where μ is called the *step-size*.

Steepest Gradient Descent (SGD)

Goals:

- Determine conditions on μ for the algorithm to converge.
- When we have convergence, does it converge to $\mathbf{w}_0$?
- What is the rate of convergence?

Steepest Descent for Linear Regression Problem

$$\nabla J = 2R\left(\mathbf{w} - \mathbf{w}_0\right) = 2R\mathbf{w} - 2\mathbf{p}$$

This gives:

$$\mathbf{w}\left[n+1\right] = \left(I - \mu R\right)\mathbf{w}\left[n\right] + \mu\mathbf{p}$$

Eigenmodes

If λ is an eigenvalue of R, then $1 - \mu\lambda$ is an eigenvalue of $I - \mu R$, and they have the same eigenvectors.

Therefore, we can do an eigendecomposition: take orthonormal eigenvectors $\{\mathbf{q}_m\}_{1 \le m \le M}$ associated with the eigenvalues $\{\lambda_m\}_{1 \le m \le M}$.

Denote $w_m[n] = \mathbf{q}_m^H \mathbf{w}[n]$, $p_m = \mathbf{q}_m^H \mathbf{p}$. Then:

$$w_m[n+1] = (1 - \mu\lambda_m) w_m[n] + \mu p_m$$

This is a simple first order IIR filter with a pole at $1 - \mu\lambda_m$.

Eigenmodes

We can justify the eigendecomposition matricially as follows. With $R = Q\Lambda Q^H$, we can write:

$$I - \mu R = I - \mu Q\Lambda Q^H = Q\left(I - \mu\Lambda\right)Q^H$$

Then:

$$\mathbf{w}\left[n+1\right] = Q\left(I - \mu\Lambda\right)Q^H\mathbf{w}\left[n\right] + \mu\mathbf{p}$$

Multiply on the left by $Q^H = Q^{-1}$:

$$Q^H\mathbf{w}\left[n+1\right] = \left(I - \mu\Lambda\right)\left(Q^H\mathbf{w}\left[n\right]\right) + \mu\left(Q^H\mathbf{p}\right)$$

The elements of the vectors $Q^H\mathbf{w}$, $Q^H\mathbf{p}$ are the eigencomponents w_m, p_m, and $I - \mu\Lambda = diag\left\{1 - \mu\lambda_m\right\}$.

Condition for Stability

Stability requires:

$$|1 - \mu\lambda_m| < 1$$

Recall that all eigenvalues are real, as is μ. Thus:

$$\begin{aligned} -1 &< 1 - \mu\lambda_m < 1 \\ 0 &< \mu\lambda_m < 2 \end{aligned}$$

Assuming R is invertible, $\lambda_m > 0$ and:

$$0 < \mu < \frac{2}{\lambda_m}$$

As this must be true for all eigenvalues, we get:

$$0 < \mu < \frac{2}{\lambda_{\max}}$$

Step Size Choice

It is usually desirable to avoid negative poles, which yield oscillation, so $\mu \leq 1/\lambda_{\text{max}}$. On the other hand, larger μ (as we shall see) yields faster convergence, so the most common choice is:

$$\mu = \frac{1}{\lambda_{\text{max}}}$$

For the moment, however, just assume $\mu \leq 1/\lambda_{\text{max}}$, so all the poles are in the range $0 \leq 1 - \mu\lambda < 1$.

Time Constants

By matching $\alpha^n = e^{-t/\tau}$, we can associate a time constant τ with a pole α. Here, τ has units of normalized discrete time (i.e., sample time $T = 1$).

The time constant τ_m for eigenmode corresponding to λ_m is:

$$\tau_m = \frac{1}{\left|\ln\left(1 - \mu\lambda_m\right)\right|}$$

The slowest rate of convergence (largest τ_m) corresponds to the case where $1 - \mu\lambda_m$ is closest to 1, i.e., for $\lambda_{\min}$.

Rate of Convergence

The eigenmode with the smallest eigenvalue converges the most slowly!

$$\tau_{\max} = \frac{1}{|\ln(1 - \mu\lambda_{\min})|}$$

Increasing μ improves the rate of convergence! If we select $\mu = 1/\lambda_{\max}$, that gives us the best result and then:

$$\tau_{\max} = \frac{1}{\left|\ln\left(1 - \frac{\lambda_{\min}}{\lambda_{\max}}\right)\right|}$$

Rate of Convergence and Condition Number

But the condition number of R is:

$$\chi\left(R\right) = \frac{\lambda_{\max}}{\lambda_{\min}}$$

Therefore, the larger the condition number, the slower the rate of convergence! If R is ill-conditioned, steepest descent can take a long time to converge!

Convergence to Optimal Solution

Assuming we have stability, we have confidence that $\mathbf{w}[n]$ converges to a steady-state value.

At steady-state, set $\mathbf{w}_\infty = \mathbf{w}[n+1] = \mathbf{w}[n]$ to get:

$$\begin{aligned} \mathbf{w}_\infty &= (I - \mu R)\,\mathbf{w}_\infty + \mu \mathbf{p} \\ \mathbf{w}_\infty &= R^{-1}\mathbf{p} = \mathbf{w}_0 \end{aligned}$$

This suggest $\mathbf{w}[n] \longrightarrow \mathbf{w}_0$ regardless of the initial condition! The time constants and hence rate of convergence do not depend on the initial condition, either. We will confirm this another way.

Matrix Analysis of Steepest Descent

We can also get a direct solution formula by unraveling the recursion:

$$\mathbf{w}\left[n\right] = \left(I - \mu R\right)^{n} \mathbf{w}\left[0\right] + \mu \sum_{k=0}^{n-1} \left(I - \mu R\right)^{k} \mathbf{p} \text{ for } n \geq 1$$

If $0 < \mu < 2/\lambda_{\max}$, all the eigenavlues of $I - \mu R$ lie inside the unit circle, so we can apply $\left(I - A\right)^{-1} = \sum_{n=0}^{\infty} A^{n}$. Then:

$$\mu \sum_{k=0}^{\infty} \left(I - \mu R\right)^{k} = \mu \left(I - \left(I - \mu R\right)\right)^{-1} = R^{-1}$$

Also $\left(I - \mu R\right)^{n} \longrightarrow \mathbf{0}$. Therefore:

$$\mathbf{w}\left[n\right] \longrightarrow R^{-1}\mathbf{p} = \mathbf{w}_0$$

regardless of $\mathbf{w}_0$.

The Problem with Steepest Descent

The gradient involves R, $\mathbf{p}$, statistical parameters we do not generally know precisely.

This is because the MMSE cost function involves an *ensemble* average: $E\left(|e|^2\right)$.

From Steepest Descent to LMS

Later we will obtain an approximate solution by estimating the gradient from our data, and this will lead to *least-mean square (LMS)* adaptive algorithm. As such, LMS is not actually an "optimal" algorithm, only one inspired by an optimization problem. In fact, in that respect, it is a fairly poor estimate.

Nevertheless it is a core adaptive algorithm, and turns out to be optimal in a completely different respect: it is optimally *robust* (more precisely, it is H^{∞}-optimal).

RLS: An Exact Solution to a Deterministic Optimization Problem

On the other hand, we can replace $E\left(|e[n]|^2\right)$ with a time-average of $|e[n]|^2$, which will lead to the *recursive least-square (RLS)* algorithm as an *exact* solution to a *deterministic* optimization problem.

RLS and Kalman Filters

We will end up articulating the RLS algorithm as a special case of a Kalman filter.

A Kalman filter estimates the state of a dynamical system from measured output.

The connection to RLS is to create a system model where the ideal tap weight vector we seek is the (unknown) state!

OPTIMAL FILTERING
VI: STEEPEST DESCENT

ECE416 ADAPTIVE ALGORITHMS