Machine Learning Refined	Spring 2025
Topic: Fundamentals of Numerical Optimization — 3/1 2025	
FAUST	*Scribe: David Zhang*

1 Overview

Main discussion points

- Gradient Descent
- Newton's Method

1.1 Calculus-defined Optimality:

Definition 1. *Linear approximation of a function g at a point v is defined as:*

$$h(w) = g(v) + g'(v)(w - v)$$

where g(v) is the function tangent at v which also contains first derivative information.

Definition 2. *The quadratic approximation of a function g at a point v is defined as:*

$$h(w) = g(v) + g'(v)(w - v) + \frac{1}{2}(w - v)^T g''(v)(w - v)$$

In general we write the linear approximatino as

$$h(\mathbf{w}) = g(\mathbf{v}) + \nabla g(\mathbf{v})^T(\mathbf{w} - \mathbf{v})$$

where $\nabla g(\mathbf{v})$ is the gradient of g at $\mathbf{v}$.

$$\nabla g(\mathbf{v}) = \begin{bmatrix} \frac{\partial g}{\partial w_1}(\mathbf{v}) \\ \vdots \\ \frac{\partial g}{\partial w_d}(\mathbf{v}) \end{bmatrix}$$

Finding a minimum would be when $\nabla g(\mathbf{v}) = \mathbf{0}_{N \times 1}$. These are also called stationary points.

Ideally we would want the function to be convex, so that we can find the global minimum.

In N dimensions, the quadratic funciton in $\mathbf{w}$ is defined as:

$$h(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{Q}\mathbf{w} + \mathbf{r}^T\mathbf{w} + d$$

.

1.2 Numerical Methods for Optimization

$$\mathbf{w}^* = \underset{w}{\operatorname{argmin}}\, g(w)$$

All numerical optimization schemes for minimization work as follows:

- Start at some initial point $\mathbf{w}^{(0)}$
- Update the point iteratively
- Stop when some stopping criterion is met

Definition 3. *Stopping Condition*

- *When a pre-specified number of iterations are complete*
- *When the gradient is small enough within an epsilon threshold*

1.3 Gradient Descent

From the first order Taylor Series Approximation centered at $\boldsymbol{w}^0$:

$$h(\boldsymbol{w}) \approx g(\boldsymbol{w}^{(0)}) + \nabla g(\boldsymbol{w}^{(0)})^T(\boldsymbol{w} - \boldsymbol{w}^{(0)})$$

Through simple calculus, the steepest descent direction is given as

$$\boldsymbol{w}^k = \boldsymbol{w}^{k-1} - \nabla_k g(\boldsymbol{w}^{k-1})$$

1.4 Newton's Method

Newton's method is a second order optimization method. The idea is to use the second order Taylor Series Approximation centered at $\boldsymbol{w}^{(0)}$:

$$h(\boldsymbol{w}) \approx g(\boldsymbol{w}^{(0)}) + \nabla g(\boldsymbol{w}^{(0)})^T(\boldsymbol{w} - \boldsymbol{w}^{(0)}) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}^{(0)})^T \boldsymbol{Q}(\boldsymbol{w} - \boldsymbol{w}^{(0)})$$

Newton's method is often more efficient, but is constrained by the fact that the Hessian matrix must be positive definite.

To do this we can use the first order condition by setting the gradient of h to zero and solving for w. This gives the $N \times N$ system of linear equations

$$\nabla^2 g(\boldsymbol{w}^0)\boldsymbol{w} = \nabla^2 g(\boldsymbol{w}^0)\boldsymbol{w}^0 - \nabla(\boldsymbol{w}^0)$$

Definition 4. $\nabla^2 g(\boldsymbol{w}) = \frac{1}{2}(\boldsymbol{Q} + \boldsymbol{Q}^T)$

Where the LHS is the Hessian of the matrix.

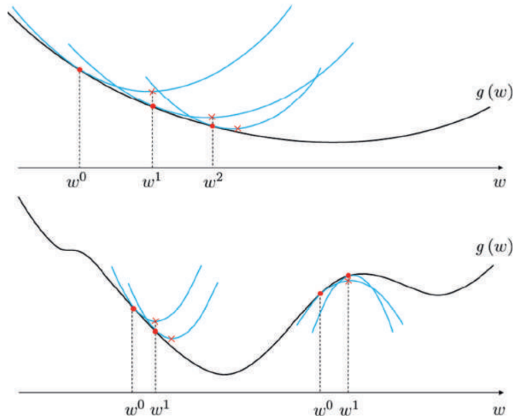

Figure 1: Newton's method illustration

Algorithm 1 Newton's Method

Initialize $\mathbf{w}^{(0)}$
for $k = 1, 2, \ldots$ until convergence **do**
  $\mathbf{w}^{(k)} \leftarrow \mathbf{w}^{(k-1)} - \nabla^2 g\big(\mathbf{w}^{(k-1)}\big)^{-1} \nabla g\big(\mathbf{w}^{(k-1)}\big)$
end for

References

[1] *Jeremy Wattós, Reza Borhanié Aggelos K. Katsaggelos,* Machine Learning Refined: Foundations, Algorithms, and Applications*, Northwestern University.*