Chapter 12: Variations on Backpropagation

Brandon Morgan

1/22/2021

E12.4

We are given the following quadratic function:

$$F(X) = \frac{1}{2}X^T \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} X + \begin{bmatrix} 4 & -4 \end{bmatrix} X$$

1

Using momentum, $\gamma = 0.75$, and a learning rate $\alpha = 1$, perform two iterations of steepest descent with the initial point $x_0^T = [0, 0]$.

To start, we need to find the gradient, which can be calculated by $\nabla F(X) = AX + d$ for a quadtratic function:

$$\nabla F(X) = \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} X + \begin{bmatrix} 4 \\ -4 \end{bmatrix}$$

Iteration 1

Now, we evaluate our gradient at the initial point $x_0^T = [0, 0]$:

$$g_0 = \nabla F(X) = \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 4 \\ -4 \end{bmatrix} = \begin{bmatrix} 4 \\ -4 \end{bmatrix}$$

Now, using momentum, our new point will be: $x_1 = \gamma x_0 - (1 - \gamma)\alpha g_0$:

$$x_1 = 0.75 \begin{bmatrix} 0 \\ 0 \end{bmatrix} - (1 - 0.75)(1) \begin{bmatrix} 4 \\ -4 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Iteration 2

Now, we evaluate our gradient at the new $x_1^T = [-1, 1]$:

$$g_0 = \nabla F(X) = \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \begin{bmatrix} 4 \\ -4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Because our gradient is equal to zero, we have found a stationary point at $x^T = [-1, 1]$ after one iteration using momentum.

If we did not use momentum, our new point for iteration 2 would have been:

$$x_1 = x_0 - \alpha g_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 1 \begin{bmatrix} 4 \\ -4 \end{bmatrix} = \begin{bmatrix} 4 \\ -4 \end{bmatrix}$$

2

From P12.2, our momentum variation of steepest descent can be written as

$$x_{k+1} = Wx_k + v$$

where $W = \begin{bmatrix} 0 & I \\ -\gamma I & T \end{bmatrix}$, where $T = [(1+\gamma)I - (1-\gamma)\alpha A]$.

It was then shown that for the algorithm to be stable, the magnnitude of each eigen value of the matrix W must be less than 1.

We can create the W matrix as below:

```
gamma = 0.75
learning_rate = 1
A = matrix(c(3, -1, -1, 3), ncol=2, byrow=TRUE)
I = diag(2)
T=(1+gamma)*I-(1-gamma)*learning_rate*A
zeroes = matrix(0, ncol=2, nrow=2)
W = matrix(0, ncol=4, nrow=4)
W[1:2,1:2]=zeroes
W[1:2,3:4]=I
W[3:4,1:2]=-gamma*I
W[3:4,3:4]=T
W
```

```
##       [,1]  [,2] [,3] [,4]
## [1,]  0.00  0.00 1.00 0.00
## [2,]  0.00  0.00 0.00 1.00
## [3,] -0.75  0.00 1.00 0.25
## [4,]  0.00 -0.75 0.25 1.00
```

```
eigen(W)
```

```
## eigen() decomposition
## $values
## [1] 0.625+0.5994789i 0.625-0.5994789i 0.375+0.7806247i 0.375-0.7806247i
##
## $vectors
##                      [,1]                 [,2]                   [,3]
## [1,] 0.5345225+0.000000i 0.5345225+0.000000i  0.5345225+0.0000000i
## [2,] 0.5345225+0.000000i 0.5345225+0.000000i -0.5345225+0.0000000i
## [3,] 0.3340766+0.320435i 0.3340766-0.320435i  0.2004459+0.4172615i
## [4,] 0.3340766+0.320435i 0.3340766-0.320435i -0.2004459-0.4172615i
```

```
##                         [,4]
## [1,]  0.5345225-0.0000000i
## [2,] -0.5345225+0.0000000i
## [3,]  0.2004459-0.4172615i
## [4,] -0.2004459+0.4172615i
```

As we can see above, the eigenvalues of our matrix is complex, where it was shown if the eigen values are complex, then their magnitude will be $\sqrt{\gamma} = \sqrt{0.75} = 0.866 < 1$; therefore the algorithm is stable with this learning rate and momentum.

3

If the momentum was zero:

```
gamma = 0
W[1:2,1:2]=zeroes
W[1:2,3:4]=I
W[3:4,1:2]=-gamma*I
W[3:4,3:4]=T
W
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    0    0 1.00 0.00
## [2,]    0    0 0.00 1.00
## [3,]    0    0 1.00 0.25
## [4,]    0    0 0.25 1.00
```

```
eigen(W)
```

```
## eigen() decomposition
## $values
## [1] 1.25 0.75 0.00 0.00
##
## $vectors
##           [,1]       [,2] [,3] [,4]
## [1,] 0.4417261 -0.5656854    1    0
## [2,] 0.4417261  0.5656854    0    1
## [3,] 0.5521576 -0.4242641    0    0
## [4,] 0.5521576  0.4242641    0    0
```

Then we can see from above that the eigen values are all was less than 1, therefore the algorithm would be stable.