AA 222 - Project 1 README

Philip Phan

April 9, 2021

1 Description of Methods

For Project 1, I implemented RMSProp and Adam as my optimization methods. RMSProp was used to optimize the Himmelblau and Powell functions, while Adam was used to optimize the Rosenbrock and secret functions.

1.1 RMSProp

RMSProp is a first order method that is an extension of the *adaptive subgradient* method, or *Adagrad*. Both methods work by adapting a learning rate for each component of x.

The issue with Adagrad is the learning rate decreases as the method is run, often becoming infinitesimally small before it converges. RMSProp extends Adagrad to avoid this issue. RMSProp maintains a decaying average of squared gradients.

The equations for the method are provided below:

$$\hat{\mathbf{s}}^{(k+1)} = \gamma \hat{\mathbf{s}}^{(k)} + (1 - \gamma)(\mathbf{g}^{(k)} \odot \mathbf{g}^{(k)}) \tag{1}$$

where the decay $\gamma \in [0, 1]$ is usually close to 0.9. Note that $\odot$ represents the element-wise product.

$\hat{\mathbf{s}}$ is them substituted into the following update equation:

$$x_i^{(k+1)} = x_i^{(k)} - \frac{\alpha}{\epsilon + \sqrt{\hat{\mathbf{s}}_i^{(k+1)}}} \mathbf{g}_i^{(k)} \tag{2}$$

In the equations above, x_i represent the position in the function, and $\mathbf{g}^{(k)}$ is the gradient.

1.2 Adam

The adaptive moment estimation method, or *Adam*, adapts learning rates to each parameter, similar to RMSProp. It stores both an exponentially decaying squared gradient like RMSProp, and an exponentially decaying gradient like in momentum based methods.

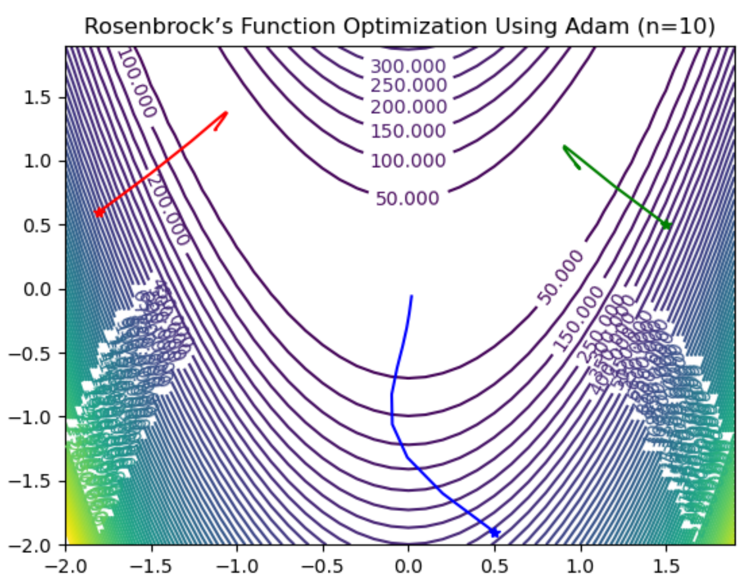
The equations applied during each iteration are provided below:

$$
\begin{aligned}
\text{biased decaying momentum:} \quad & \mathbf{v}^{(k+1)} = \gamma_v \mathbf{v}^{(k)} + (1-\gamma_v)\mathbf{g}^{(k)} \\
\text{biased decaying squared gradient:} \quad & \mathbf{s}^{(k+1)} = \gamma_s \mathbf{s}^{(k)} + (1-\gamma_s)(\mathbf{g}^{(k)} \odot \mathbf{g}^{(k)}) \\
\text{corrected decaying momentum:} \quad & \hat{\mathbf{v}}^{(k+1)} = \mathbf{v}^{(k+1)}/(1-\gamma_v^k) \\
\text{corrected squared gradient:} \quad & \hat{\mathbf{s}}^{(k+1)} = \mathbf{s}^{(k+1)}/(1-\gamma_s^k) \\
\text{next iterate:} \quad & \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha \hat{\mathbf{v}}^{(k+1)} / \left(\epsilon + \sqrt{\hat{\mathbf{s}}^{(k+1)}}\right)
\end{aligned}
\tag{3}
$$

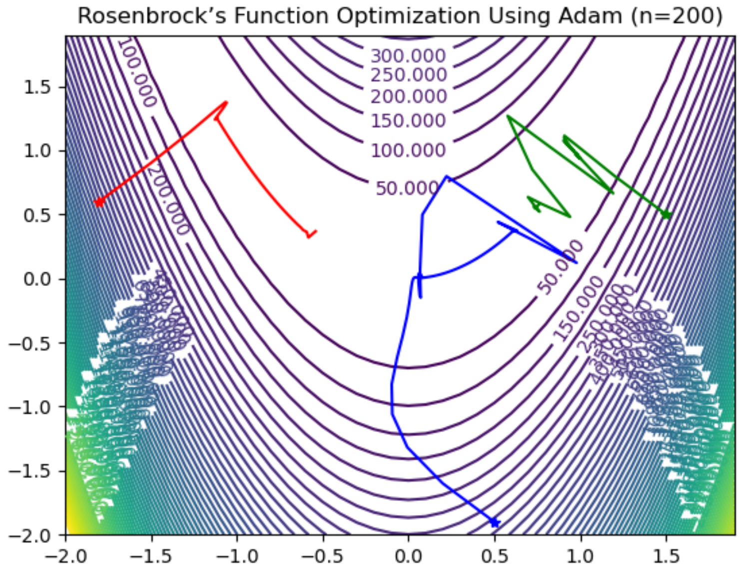
In the equations above, $\mathbf{x}$ represent the position in the function, and $\mathbf{g}^{(k)}$ is the gradient. γ_v and γ_s are set parameters.

2 Rosenbrock's Function Optimization

Below is a plot showing the path taken by the optimization algorithm (Adam) from three different points, plotted over the contours of Rosenbrock's function. Note that the paths only contain ten iterations.

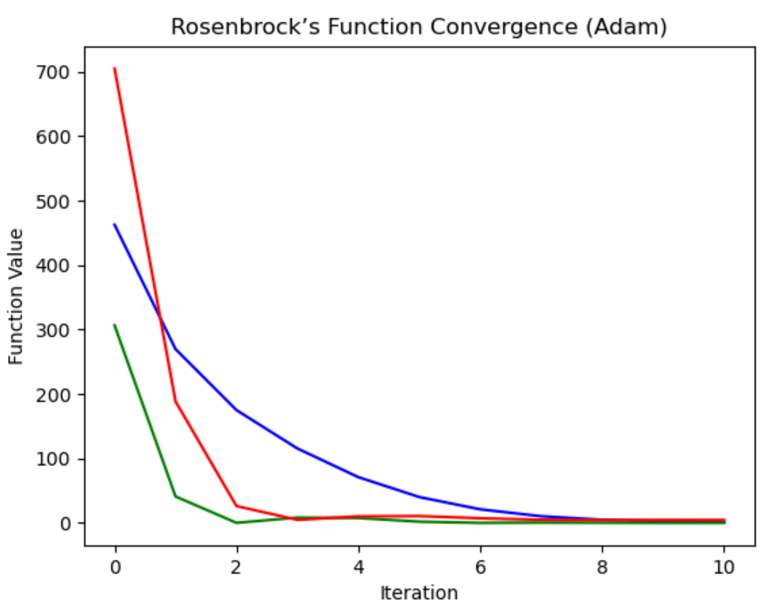


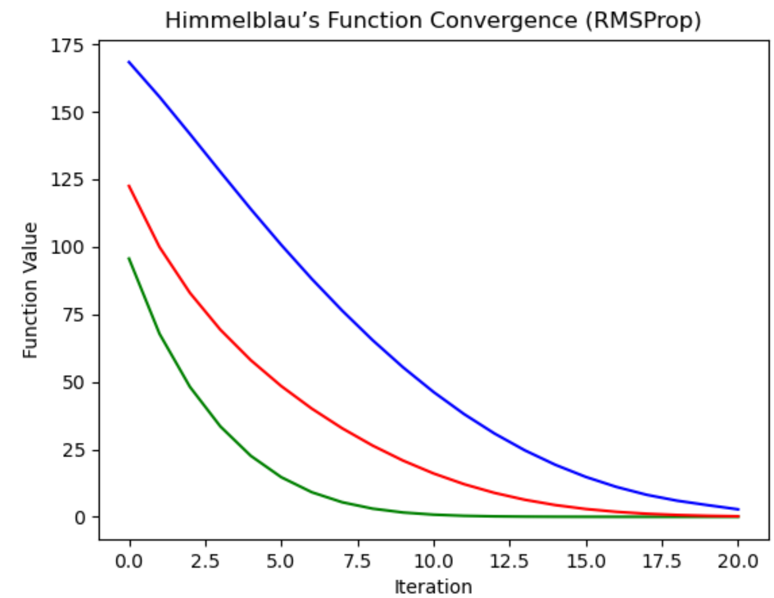
If the iterations are allowed to go to large numbers (n = 200), we can see the function begins to struggle in the function's valley.

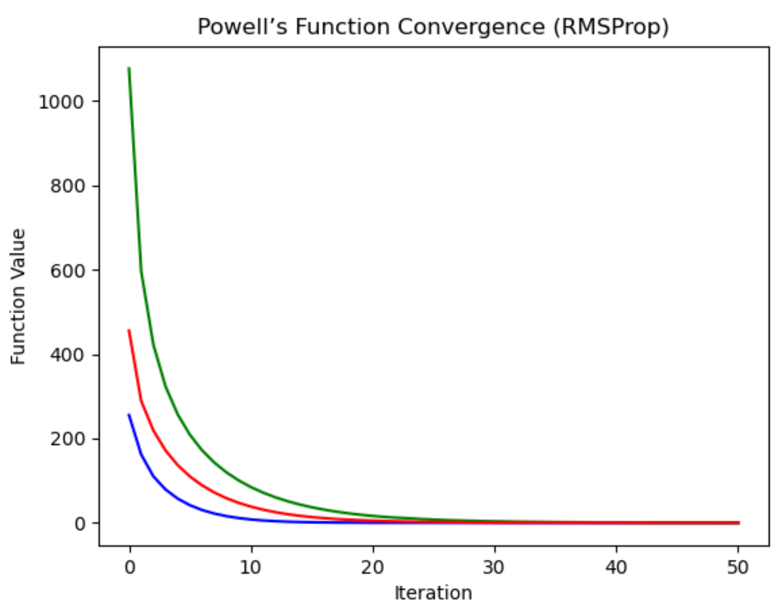


3 Convergence plots

Convergence plots for Rosenbrock's function, Himmelblau's function, and Powell's function are provided below. Three different starting points were chosen for each function.




Himmelblau's Function Convergence (RMSProp)
175
150
125
100
75
50
25
0
Function Value
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Iteration


Powell's Function Convergence (RMSProp)
1000
800
600
400
200
0
Function Value
0
10
20
30
40
50
Iteration