

Machine Learning in Finance - Homework 2 – Due 09.17.2024

Textbook reading:

Chapters 3

- **Upload Adam_search.ipynb to colab**
- **The path of convergence is curved. Why is this the case?**
The learning rate 0.1 had led to overshoot of the target.

- **How much faster/slower is the processing when using the following hardware:**

– CPU, TPU4, TPU v2-8

| Hardware \ Runtime | CPU | TPU4 | TPU v2-8 |
|--------------------|-------|-------|----------|
| Trial 1 | 5.833 | 0.081 | 1.277 |
| Trial 2 | 1.578 | 1.298 | 1.128 |
| Trial 3 | 7.684 | 0.053 | 1.089 |
| Trial 4 | 1.606 | 1.478 | 1.126 |
| Average | 3.623 | 0.728 | 1.155 |
| Standard Deviation | 3.517 | 0.766 | 0.083 |

Table 1. Runtime by hardware

- **What parameters will straighten this path and why does it do so?**
By decreasing the learning rate to 0.04, it can straighten up the path.
- **Make a case that the SGD optimizer might be a better choice for this problem.**
The function $f(x,y) = x^2 + y^2$ is smooth and well-behaved with a single minimum at (0,0). So the adaptive learning rate feature from Adam may have added unnecessary complexity to the optimization process. SGD's constant learning rate, on the other hand, is well suited for this problem whose gradient is constant.
- **How does the performance of SGD compare to Adam?**
By changing the optimizer to
`optimizer = optim.SGD([params], lr=0.1)`
I repeated the experiment. It took fewer than 20 iterations to converge. Total runtime was 0.036 seconds when using TPU4.
- **Starting at (2.5,-1.0) run the same routine on the following function**

```
x**2 + y**2 + 1.1 * (x+3)**2 + 3 * (y-1)**2
```
- **What is the minimum value found by the algorithm?**
5.472 at (-1.6272, 0.7607)

- **What Adam parameters are the most crucial in creating a direct line from initial point to optimal?**

The exponential decay rates, or betas. By setting them to 0, The convergence path becomes straight, although it bounces around the minimum (0,0).

- **What parameters make this path more convoluted?**

Learning rate and betas both can make the path more convoluted, specifically, higher initial learn rate, higher beta1, and smaller beta2.

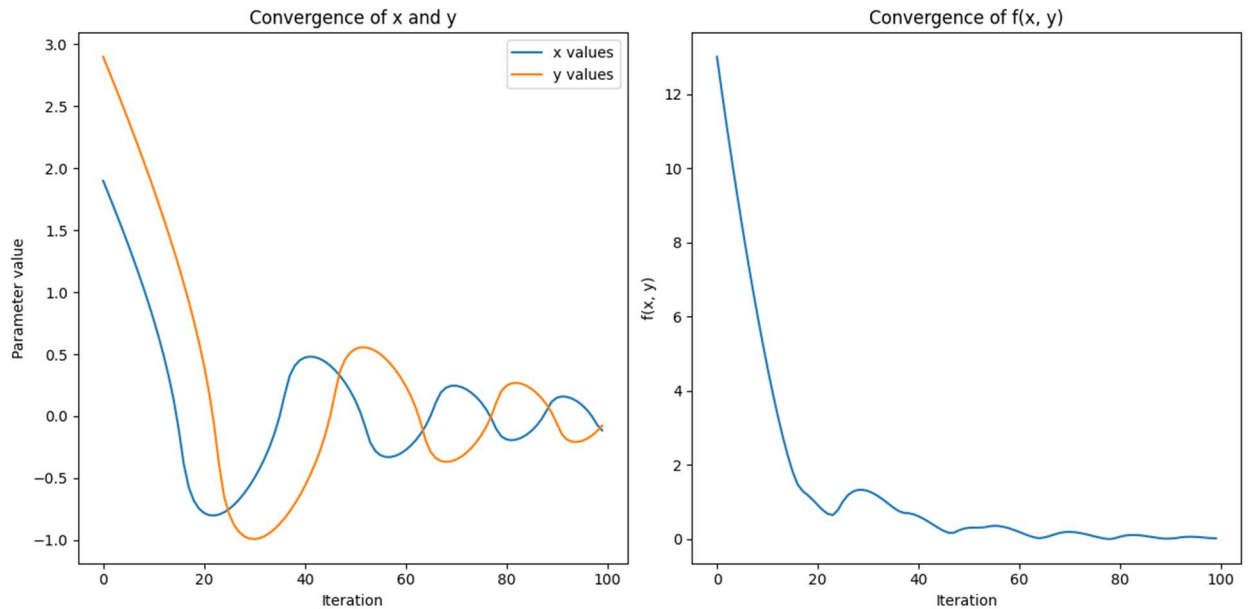


Figure 1. betas = (0.9,0.5)

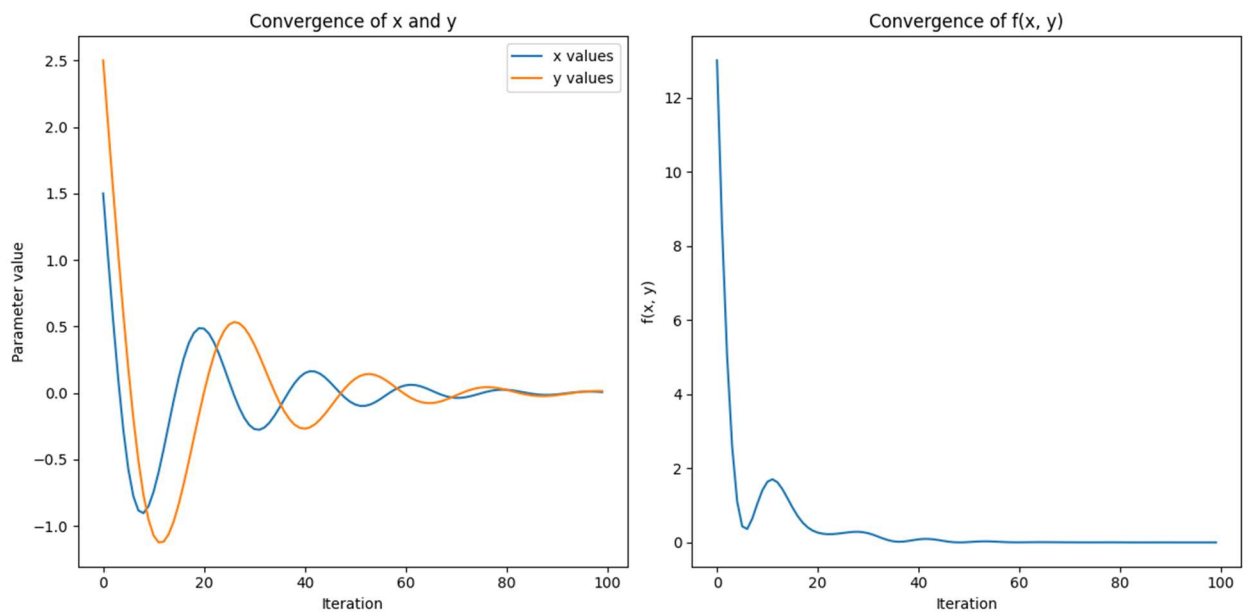


Figure 2. lr = 0.5

Note that you can access the GPU or TPU in colab by selection the following for the menu in Colab:
Edit > Notebook settings