|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **W** | **B** | **Learning\_rate** | **Epochs** | **Error** |
| -2 | -2 | 1.0 | 100 | 0.017741009621761353 |
| -2 | -2 | 1.0 | 1000 | 2.4291865713546816e-11 |
| -2 | -2 | 1.0 | 2000 | 2.3947265822163888e-20 |
| -2 | -2 | 3.0 | 100 | 5.716362296559683e-05 |
| -2 | -2 | 5.0 | 100 | 7.114464162870803e-07 |
| -2 | -2 | 7.0 | 100 | 6.461558886290009e-09 |
| -2 | -2 | 10.0 | 100 | 1.844555021169613e-12 |
| -2 | -2 | 12.0 | 100 | 1.3090084531435982e-14 |
| -2 | -2 | 13.0 | 100 | 3.005477552584132e-13 |
| -2 | -2 | 14.0 | 100 | 1.64805147430089e-17 |
| 0 | 0 | 1.0 | 100 | 0.0013755069444627295 |
| 5 | 5 | 1.0 | 100 | 0.32453212800819825 |
| 10 | 10 | 1.0 | 100 | 0.3249997552707788 |
| 1.8 | 0 | 1.5 | 1000 | 1.0305100703007212e-16 |

**Conclusion:**

From the first 3 row , we can see that as we increase the epochs , the error is decreased. From row 4 to 10 we can see that if we change only learning rate then it affect the error too. As we increased learning rate from 1.0 to 12.0 and keeping weight, bias and epochs constant then error is decreased but if we change learning rate to 13.0 then error is increased and after that we change learning rate to 14.0 then error is still decreasing. So we can say that if we give learning rate too large then there might be overshoot and error become large. From last 4 row we can see that different weight and bias value at the time of initialize affect the error too. If we initialize weight and bias to very large value then it will take more epochs to converge.

It is depend upon the problem that how to initialize weight, bias, and what value for the learning rate and epochs. If we give too much value to the learning rate then it might be overshoot and if we give too much small value to learning rate then it will require more epochs to converge and it takes more time to train.