Bradley Johnson

Jeffery Dirden

ITAI 1378

Professor McManus

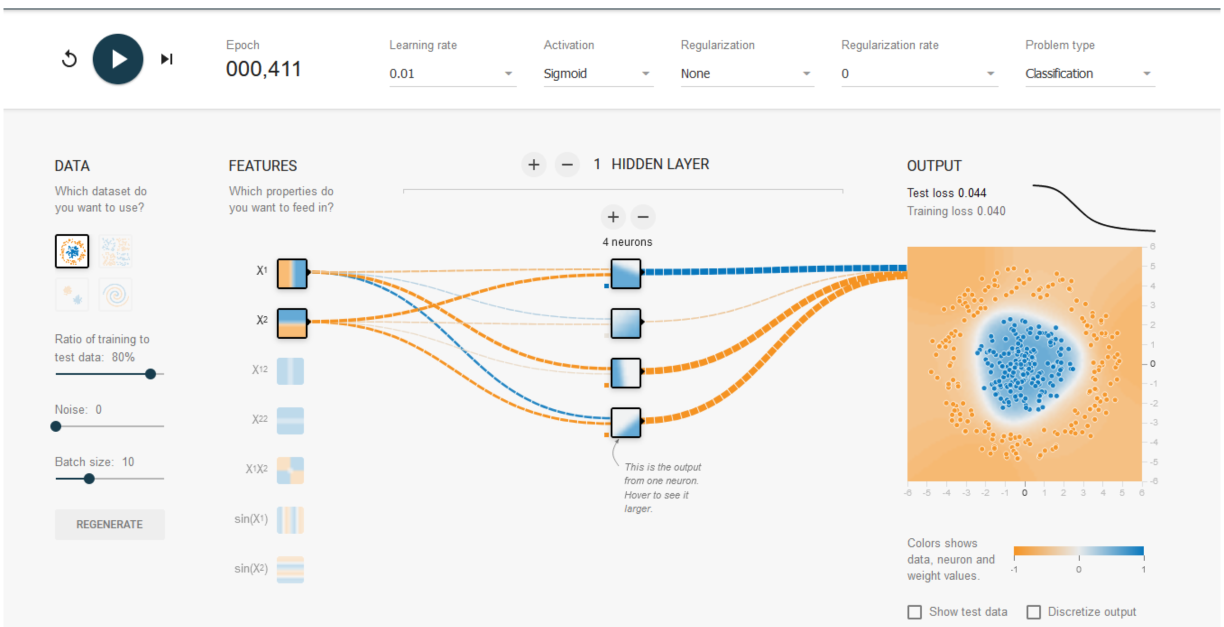
2 October 2024

Assignment A06

Neural networks are a type of artificial intelligence that attempt to simulate how the human brain works. They are composed of layers of neurons, which are interconnected units that process information. The key components of a neural network include input layers, hidden layers, and output layers. Information flows through these layers, where the neurons apply mathematical functions to make predictions or classifications. The significance of neural networks lies in their ability to recognize patterns in complex data, making them essential for tasks like image recognition and natural language processing. Activation functions, neurons, learning rates, and the structure of hidden layers all contribute to how effectively a neural network can learn from data. These components play a crucial role in how the network processes information and adapts to various types of data. Understanding how each parameter influences performance helps in designing more effective neural networks.

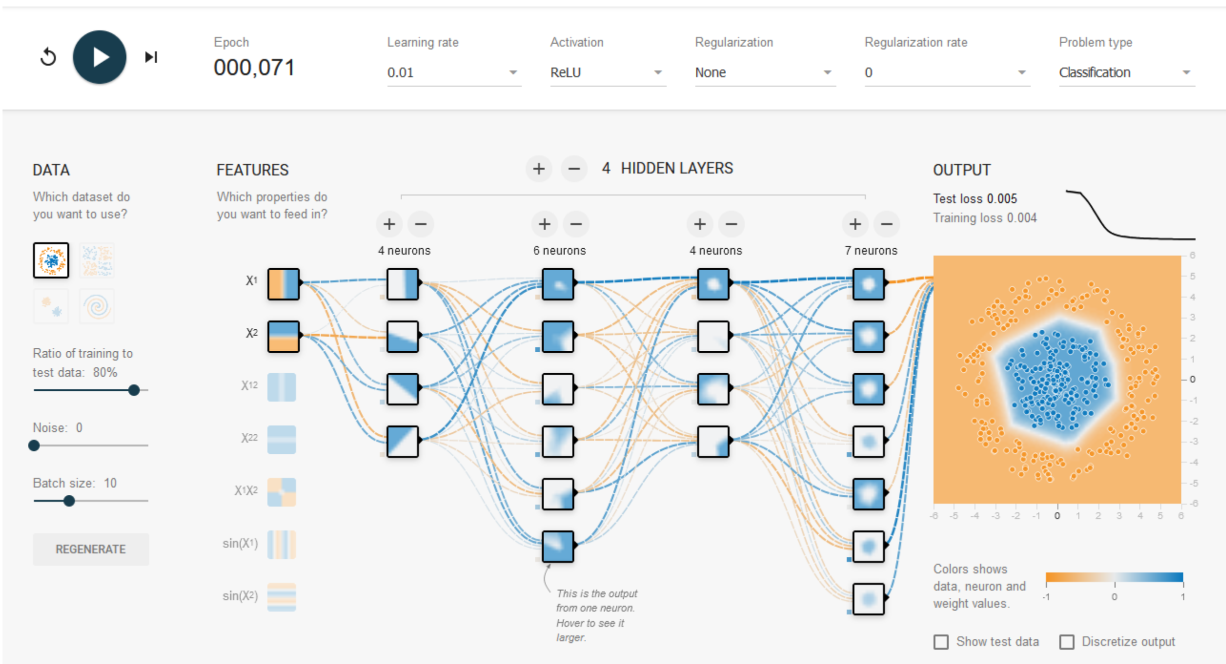
Task 1 - Activation Functions

In this task, I used a neural network with one hidden layer and tested two activation functions: ReLU and Sigmoid. Activation functions help the network make decisions by transforming input data into output. ReLU activates neurons only if the input is positive, which led to faster learning and better accuracy in my experiment. Sigmoid, which outputs values between 0 and 1, slowed the learning process because it tends to saturate, making it harder for the network to learn complex patterns.



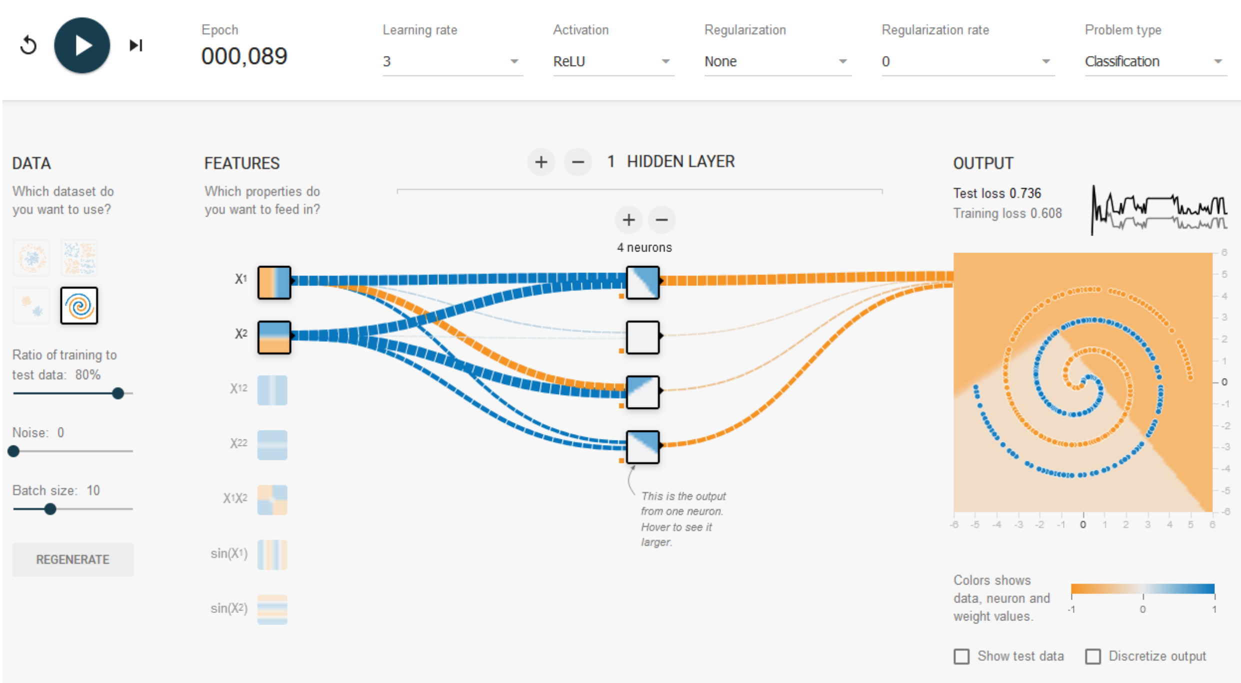
Task 2 - Hidden Layer Neurons

I explored how the number of neurons and hidden layers affect the network’s performance. Neurons process data, and hidden layers help the network learn more complex features. Increasing the number of neurons allowed the network to capture more detailed patterns, improving performance. Adding more hidden layers made the network better at learning complex patterns, but too many layers caused overfitting, where the network struggled with new data.



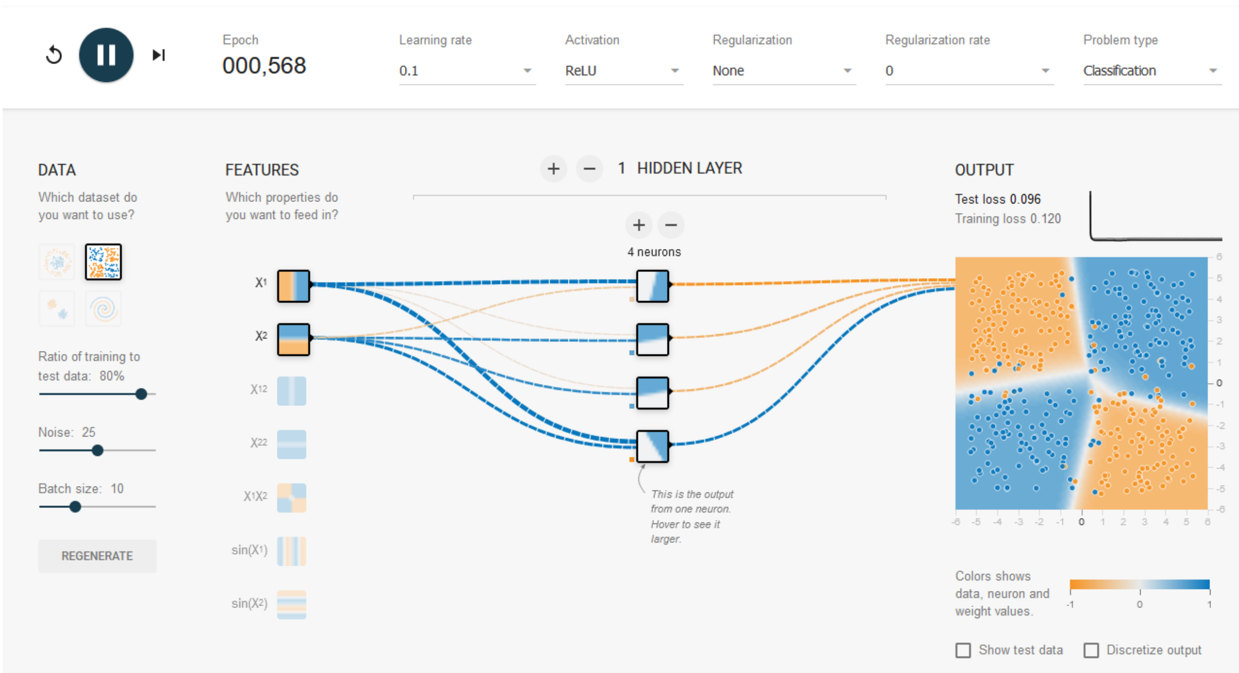
Task 3 - Learning Rate

The learning rate determines how quickly a network updates its weights during training. I experimented with high and low learning rates. A high learning rate allowed the network to converge quickly but sometimes missed the best solution, leading to less accurate predictions. A lower learning rate resulted in better accuracy, but the network took longer to train and sometimes got stuck in suboptimal solutions.



Task 4 - Data Noise

By adjusting the "Noise" slider, I introduced random variations into the data to simulate real-world imperfections. As noise increased, the network had more difficulty generalizing to new data because it focused too much on the irrelevant details. This experiment showed how important clean data is for building effective neural networks.



Task 5 - Dataset Exploration

I tested the network on different datasets in TensorFlow Playground, including spirals, circles, and linear data. The spiral dataset was more complex, requiring more neurons and layers for the network to learn the pattern. It took longer to train and needed careful tuning of other parameters. The linear dataset was much simpler, and the network performed well with fewer neurons and layers, showing that simpler data doesn’t require complex architectures.

Observations and Explanations

Each change to the neural network's parameters had noticeable effects. ReLU activation function resulted in faster and more efficient learning compared to Sigmoid. Increasing the number of neurons and layers improved performance, but too many layers led to overfitting, especially with complex data. The learning rate required careful balancing—too high caused instability, while too low slowed the learning process. Adding noise to the data affected the network’s ability to generalize, highlighting the importance of clean data. More complex datasets needed more sophisticated networks, showing that selecting the right model is essential for the task.

Practical Implications

Understanding the role of activation functions, neuron counts, learning rates, and data noise is essential for real-world applications. In practice, tuning these parameters can improve the accuracy of models used for tasks like image classification, fraud detection, or language translation.

Conclusion

Overall, this hands-on experience with TensorFlow Playground has deepened my understanding of neural networks and their applications, and I now appreciate the complexity of building effective models. Through these tasks, I gained valuable insights into how neural networks function and how small changes in parameters can greatly impact performance. One of the main challenges was finding the right balance in parameter settings, particularly with the learning rate and the number of neurons.

References

https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle&regDataset=reg-plane&learningRate=0.03&regularizationRate=0&noise=0&networkShape=4,2&seed=0.92466&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false

https://wandb.ai/ayush-thakur/dl-question-bank/reports/ReLU-vs-Sigmoid-Function-in-Deep-Neural-Networks--VmlldzoyMDk0MzI

https://www.nvidia.com/en-us/glossary/tensorflow/