IEE 577: Data Science for Decision System Analytics **Spring 2021**

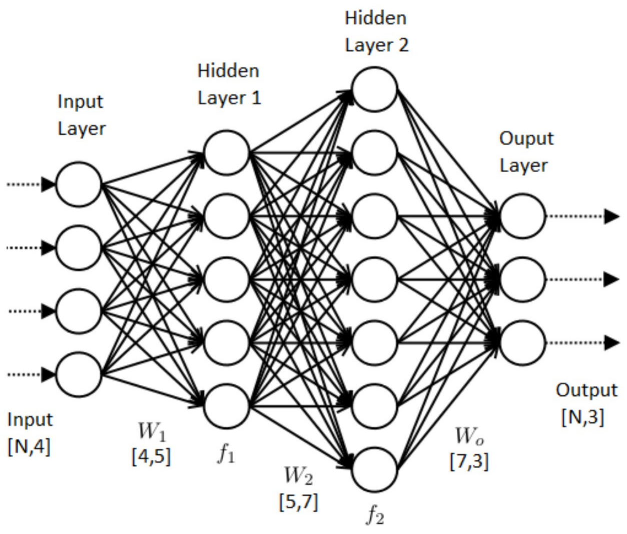
Assignment 1 — Feb 11

Instructor: Hao Yan *Scribe: Vivek Nagarajan*

Submission Guideline For submission of the homework, please submit the report and the code. The code should be in either the notebook file (.ipynb) or python file (.py) if you prefer to directly write the python outside the notebook. Even though you provide the notebook, it is required to put the necessary plots/tables inside the report to answer the homework questions. Also, please make sure your code can run through before the submission.

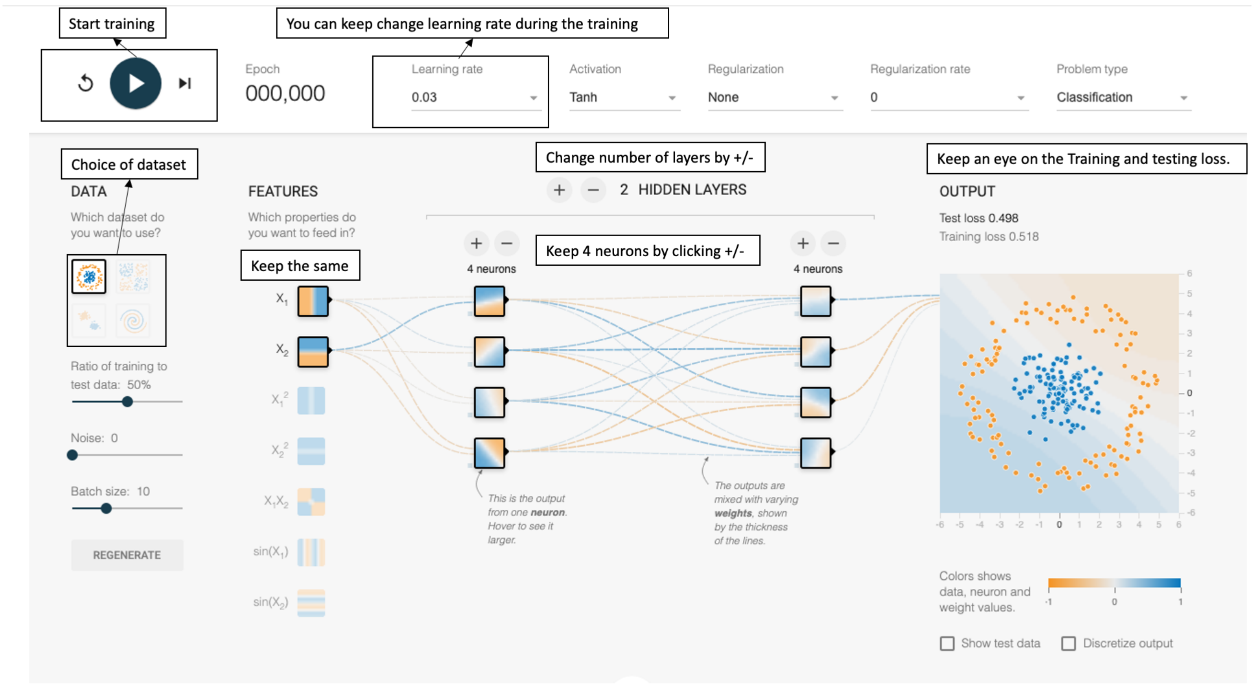
Problem 1: Exploring Model Complexity on Training and Testing (40 points)

The goal of this problem is to understand the model complexity using a pretty complicated neural network model. Neural network model is a complicated machine learning model. In general, if there are more neurons in each layer, the model is more complicated (flexible). Similarly, if there are more layers in the neural networks, the model is also more complicated.



We will explore the use of `https://playground.tensorflow.org/` for the training of the neural network. You can select the number of neurons and number of hidden layers for classification of 4 different datasets. Here is the guideline of using the Tensorflow playground.

1. **(20 points) Depth of network**: First, please try to complete Table 2.1 by trying on selecting two out of four datasets with different neural networks to test the effect


Start training

You can keep change learning rate during the training

Epoch 000,000

Learning rate 0.03

Activation Tanh

Regularization None

Regularization rate 0

Problem type Classification

Choice of dataset

DATA

Which dataset do you want to use?

Ratio of training to test data: 50%

Noise: 0

Batch size: 10

REGENERATE

FEATURES

Which properties do you want to feed in?

Keep the same

Change number of layers by +/-

2 HIDDEN LAYERS

Keep 4 neurons by clicking +/-

4 neurons

4 neurons

This is the output from one neuron. Hover to see it larger.

The outputs are mixed with varying weights, shown by the thickness of the lines.

Keep an eye on the Training and testing loss.

OUTPUT

Test loss 0.498

Training loss 0.518

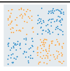
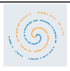
Colors shows data, neuron and weight values.

Show test data

Discretize output

Figure 1.1. Guide of Using the Tensorflow Playground

Table 1.1. Training Loss and Testing Loss for Neural Network with Different Layers. Please keep 4 neurons for each layer.

Dataset	Loss	1 hidden layers	2 hidden layers	4 hidden layers
	Training			
	Testing			
	Training			
	Testing			

of different number of layers (depth). Please keep the number of neurons as 4 for each layer. During the training, you may find the training loss is oscillating, in this case, please continue the training by reducing the learning rate from 0.03 to 0.01 or even smaller to stabilize the training. Please train the model until the training loss become stable. Please try to explain the behavior.

2. **(20 points) Width of network**: First, please try to complete Table 2.2 by trying on selecting two out of four datasets with different neural networks to test the effect of different number of neurons. We will keep the models with 3 hidden layers. Please try to explain the behavior.

Table 1.2. Training Loss and Testing Loss for Neural Network with Different Layers. Please keep 3 layers for the network.

Dataset	Loss	2 Neurons	4 Neurons	8 Neurons
	Training			
	Testing			
	Training			
	Testing			

Problem 2: Short Answers for Concepts (20 points)

This problem reviews basic concepts from probability.

1. (10 points) Suppose P(snow today) = 0.30, P(snow tomorrow) = 0.60, P(snow today and tomorrow) = 0.25. Given that it snows today, what is the probability it will snow tomorrow? Please show brief steps for calculation.
2. (5 points) What does it mean when a machine learning model is overfitting? Select the most appropriate explanation. Only the answer is needed.
   (a) A model that is overfitting cannot model the training data or generalize to new data.
   (b) A model that is overfitting can model the training data but not generalize to new data.
   (c) A model that is overfitting is a model that has become obsolete over time.
   (d) A model that is overfitting is a model that was trained on too much data
3. (5 points) Which of the following can help resolve the overfitting issue? Only the answer is needed.
   (a) Increase the model complexity or flexibility
   (b) Increase the number of training samples
   (c) Adding more features or input variables to the model

Problem 3: Get Familiar with Python (40 points)

1. (20 pints) Please write a program that uses Numpy, creates two random 100×100 arrays A and B and add them together as $C = A + B$ in two different ways:
   - By using the for-loop method to compute each element of C by $C_{ij} = A_{ij} + B_{ij}$ for all $i, j = 1, \cdots, 100$.
   - By using the numpy built-in function for matrix multiplication.

Please answer the following questions

(a) Write a small program for both methods and test each methods 100 times and record the timing results in a numpy ector of length 100.

(b) For each method, please compute the average and standard deviation of the running time.

Hint: you can use **timeit** function to get the time of a certain small program.

2. (20 pints) Please write a program that uses Numpy, creates two random 100×100 arrays A and B and multiple them together as $C = AB$ in two different ways:

   - By using the for-loop method to compute each element of C by $C_{ij} = \sum_k A_{ik}B_{kj}$ for all $i, j = 1, \cdots, 100$.
   - By using the numpy built-in function for matrix multiplication.

   Please answer the following questions

   (a) Write a small program for both methods and test each methods 100 times and record the timing results in a numpy ector of length 100.

   (b) For each method, please compute the average and standard deviation of the running time.

Problem 4: Bias and Variance Trade-off (Bonus 30 points)

1. (10 points) If X is a random variable, with mean $\mu = E[X]$, and $\sigma^2 = Var[X]$, please proved that
$$E[(X-a)]^2 = \sigma^2 + (\mu - a)^2$$
This formula can be used to understand the bias and variance trade-off in the simplest setting. Please also use this formula to solve the following optimization problem.
$$\min_a E[(X-a)]^2$$

2. (20 points) Here, we will try to derive the Bias and Variance Trade-off again. We would like to derive the trade-off from the model parameter perspective. More specifically, assume a regression problem as follows:
$$y = f(x;\theta) + \epsilon$$
We have collected a set of training data $D_{tr} = \{x_i, y_i\}$ and the model parameter is estimated as $\hat{\theta}$. Typically, Mean Squared Error (MSE) is used to evaluate the model parameter θ as $\text{MSE}(\theta) = E_\theta[(\hat{\theta} - \theta)^2]$. Please prove that
$$\text{MSE}(\theta) = \text{Var}(\hat{\theta}) + \text{Bias}^2(\hat{\theta})$$
where $\text{Var}(\hat{\theta}) = E_{\hat{\theta}}(\hat{\theta} - E_{\hat{\theta}}(\hat{\theta}))^2$, and $\text{Bias}(\hat{\theta}) = E(\hat{\theta}) - \theta$.