Lesson 17 (Neural Network Playground–1)

Over-Fitting Using Feature Engineering: In this lesson we see what happens when feature engineering is used to commit the cardinal sin of machine learning—over-fitting.

1. Go to `playground.tensorflow.org` and set-up configuration 1 shown in the table below.

configuration	1	2	3	4
data	2-cluster			
train-test split	50%			
noise	0			
batch size	10			
features	x_1, x_2			
hidden layers	0			
nodes	0			
learning rate	0.03			
activation function	linear			
regularization	none			
regularization rate	0			
problem type	classification			
discretize output	yes			

2. Click the `REGENERATE` button in the bottom left corner to generate a new data set. Then begin training the network by clicking the play button in the upper left corner.

   (a) What is the final training error?

   (b) What is the final testing error?

   (c) What are the values of the weights for this trained logistic regression network?

3. Increase `Noise` to 50.
   Click the `REGENERATE` button in the bottom left corner to generate a new data set. Then begin training the network by clicking the play button in the upper left corner.

   (a) What is the final training error?

   (b) What is the final testing error?

   (c) What are the values of the weights for this trained logistic regression network?

4. Do **not** click the `REGENERATE` button. Use the same data set used in part 3 above. Click the rewind button (upper left corner) to reset the network. Try to reduce testing error by *engineering* new features to include: x_1^2, x_2^2, x_1x_2, $\sin(x_1)$ and $\sin(x_2)$. Then begin training the network by clicking the play button in the upper left corner.

   (a) What is the final training error?

   (b) What is the final testing error?

   (c) What are the values of the weights for this trained logistic regression network?