Nicolas Carpenter
CMPUT 466: Machine Learning

Coding Assignment #1

Question 1.

A) The bias and weight terms for x2y and y2x for variables var1 = 1, var2 = 0.3, degree = 45 are:
w_x2y = 0.5244587225052052 b_x2y = 0.005071660776465743
w_y2x = 0.5449400592153517 b_y2x = 0.011800263839855787

B) Figure 1 shows the regression plots with var2 equaling 0.8, 0.3, 0.1 (left to right) where var1 and degree remain constant at 1 and 45 respectively.

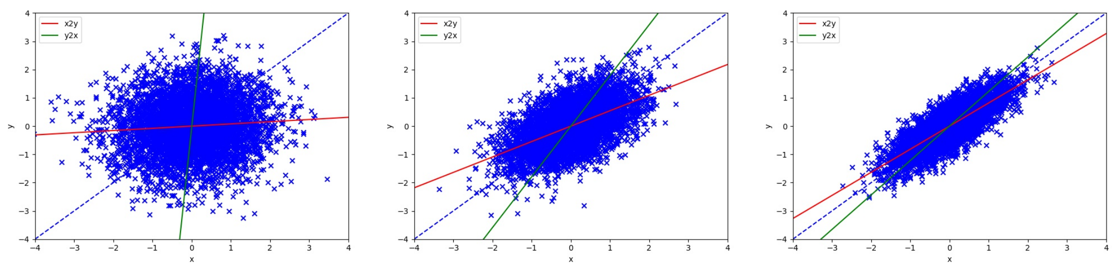

FIGURE 1. Regression plots: varied var2.

C) As var2 moved from 0.8 down to 0.1 the data became less varied about the horizontal line rotated by 45 degrees. This led to both x2y and y2x, for smaller values of var2, to approach the dashed line that the data is mostly generated around. Also, the weights of both x2y and y2x increased as var2 became smaller.

D) My experimental protocol to check the effects degree on the x2y and y2x regressions was to keep var1 and var2 constant as I chose degrees slightly higher and lower than the original degree of 45. This was to see how both regressions changed after crossing this original diagonal, where both x2y and y2x are equidistant to the data generation line (DGN). Then to check the effects as the degree moves further away from the original diagonal I chose a degree of 0.

Figure 2 shows the regression plots with degree equaling 60, 30, 0 (left to right) where var1 and var2 remain constant at 1 and 0.1 respectively.

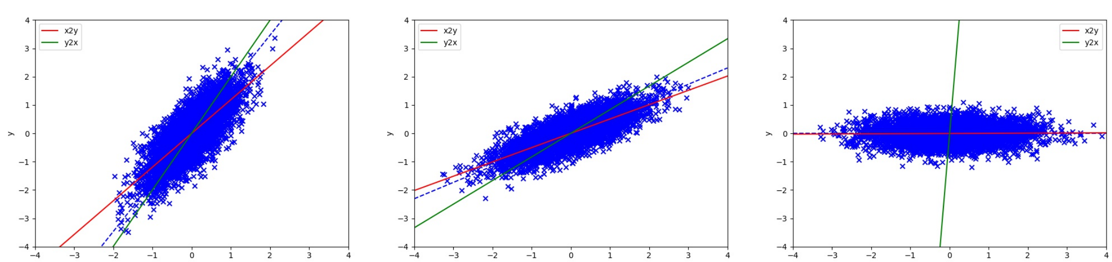

FIGURE 2. Regression plots: varied degree.

With this protocol, I was able to determine that as the degree approaches 0, x2y regression approaches the DGL and y2x regression approaches the ⊥DGL. As the degree approaches 90, vise versa.

Question 2.

A) Mini-batch GD: alpha = 0.001, batch_size = 10, MaxIter = 100

Epoch with best validation performance	: epoch 100
Epoch 100 validation performance (risk)	: 0.5462778014875631
Epoch 100 test performance (risk)	: 0.5122943852532776

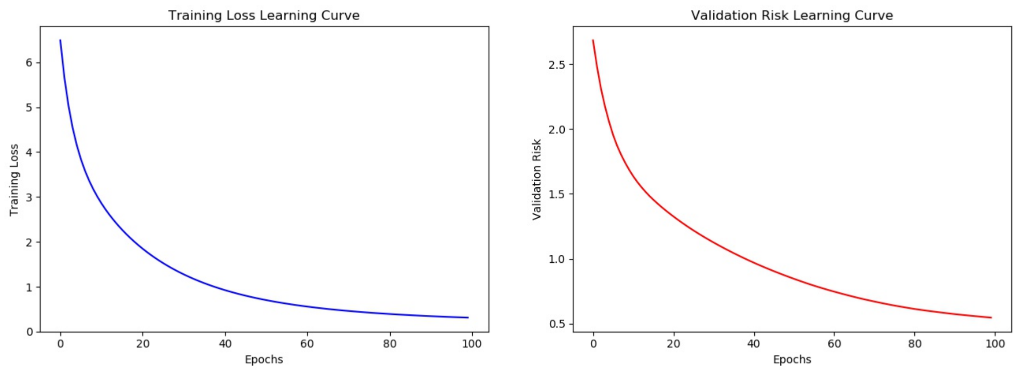

FIGURE 3. Training loss and validation risk learning curves.

B) Mini-batch GD: alpha = 0.001, batch_size = 10, MaxIter = 100, decay $\in \{3, 1, 0.3, 0.1, 0.03, 0.01\}$

Epoch with best validation performance	: epoch 98
Epoch 98 validation performance (risk)	: 0.347937084874763O5
Epoch 98 test performance (risk)	: 0.3254641989474234

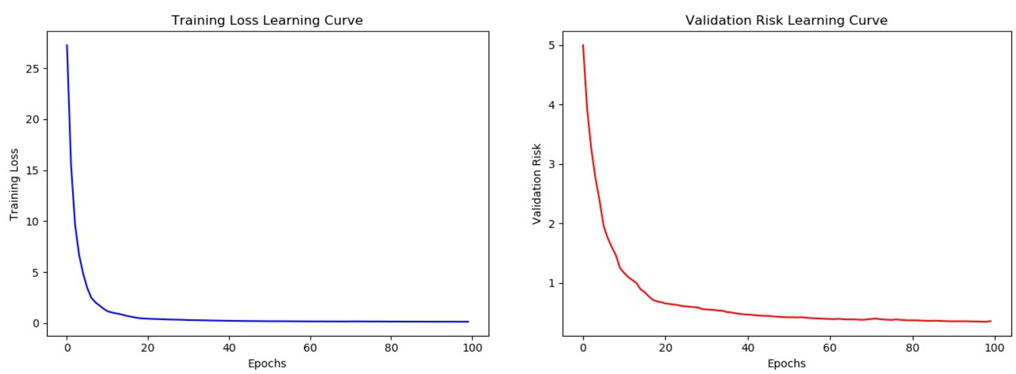

FIGURE 4. Training loss and validation risk learning curves.

C) Do larger learning rates perform better than a smaller ones in terms of validation risk? We can investigate this relationship by plotting the learning curves of gradient descents with different learning rates. I chose learning rates 0.0001, 0.001, 0.01, and 0.1 to showcase the transition between fast and slow rates.

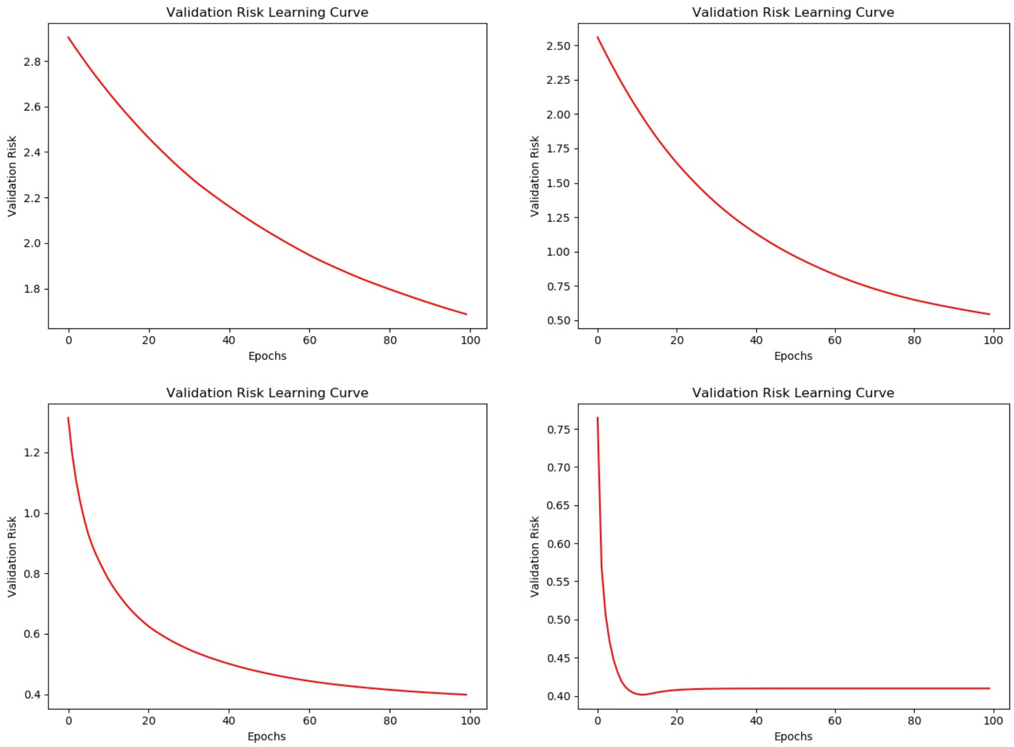

FIGURE 5. Validation risk learning curves: learning rates 0.0001, 0.001, 0.01, 0.1 from top left to bottom right.

From these plots we can see that the validation risk decreased quicker the larger the learning rate. Since we kept the number of iterations the same this resulted in the slower learning rates getting cut off before getting close to the optimum. However, with the largest learning rate we its lowest validation risk was on its 12th epoch and then did not decreasing any further as it oscillated around the optimum. As a result learning_rate = 0.01 performed the best.

Based off of these results we can conclude that the best learning rate depends on the number of iterations; with more iterations smaller learning rates can more precisely approach the optimum than large ones, while with fewer iterations large learning rates descend quickly enough to be more effective than small ones.