Machine Learning from Data CSCI 4100
Assignment 7
Jae Park (RIN: 661994900)

1.

a)

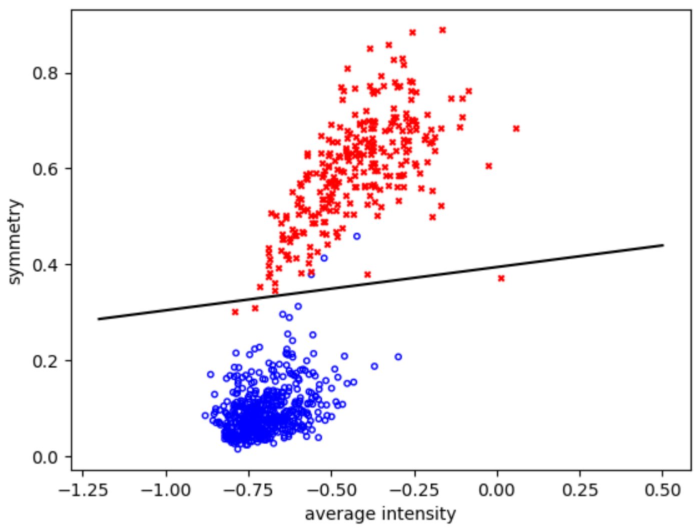

Figure 1: Output from training data

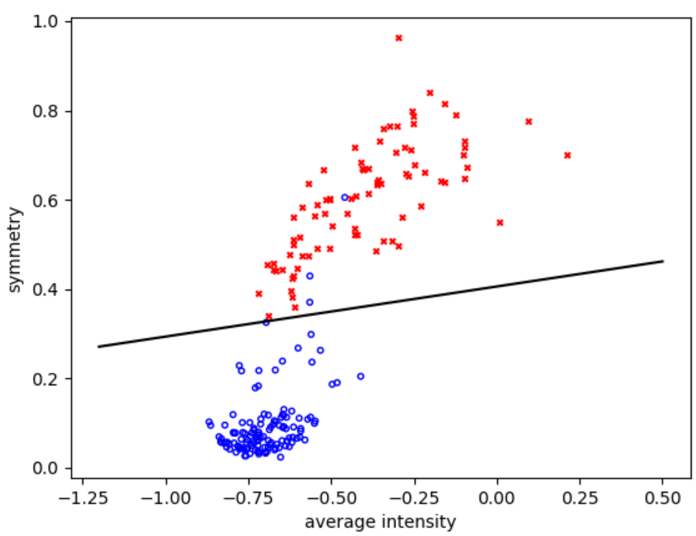

Figure 2: Output from test data

b)
E_{in} from train data: 0.008917197 (0.89%)

E_{test} from test data: 0.014423076 (1.44%)

c)
For ZipDigits.train, N = 758, $\delta = 0.05$
We are using two features, so $d_{vc} = 2 + 1 = 3$

$$\begin{aligned}
E_{out}(g) &\leq E_{in}(g) + \sqrt{\frac{8}{N} \ln\left(\frac{4((2N)^{d_{vc}}+1)}{\delta}\right)} \text{ ... LFD (2.16) } pg.58 \\
&\leq E_{in}(g) + \sqrt{\frac{8}{758} \ln\left(\frac{4(1516^3+1)}{0.05}\right)} \\
&= 0.0089 + 0.5273 \\
&\therefore E_{out}(g) = 0.5362
\end{aligned}$$

For ZipDigits.test, N = 208, $\delta = 0.05$, $d_{vc} = 3$

$$\begin{aligned}
E_{out}(g) &\leq E_{test}(g) + \sqrt{\frac{1}{2N} \ln \frac{2M}{\delta}} \text{ ... LFD (2.1) } pg.40 \\
&\leq E_{test}(g) + \sqrt{\frac{1}{2 \cdot 208} \ln \frac{2 \cdot 1}{0.05}} \\
&= 0.0144 + 0.0941 \\
&\therefore E_{out}(g) = 0.1085
\end{aligned}$$

The bound based on E_{test} is much better.

d)

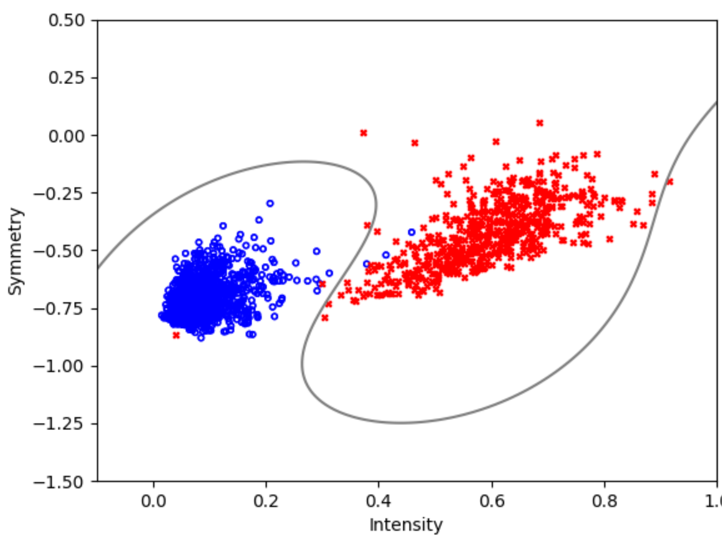

Figure 3: Output from training data

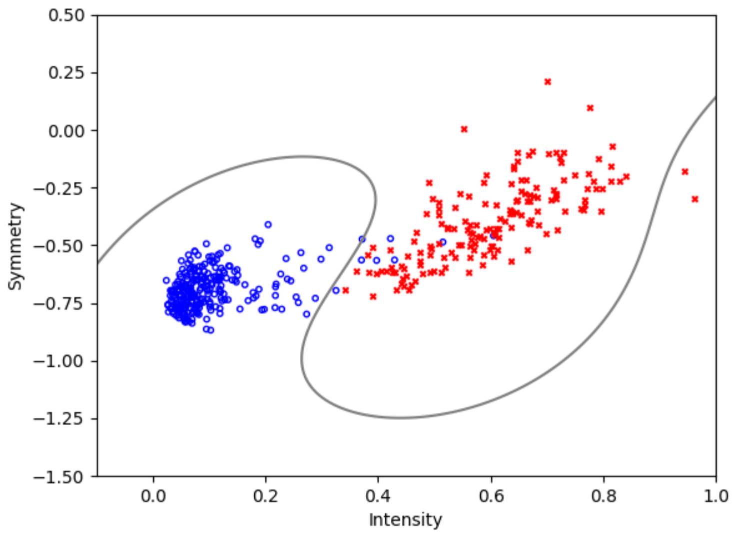

Figure 4: Output from test data

E_{in} from train data: 0.004484 (0.45%)

E_{test} from test data: 0.043269 (4.33%)

For ZipDigits.train,

$$E_{out}(g) \leq 0.004484 + 0.891404$$
$$\therefore E_{out}(g) = 0.895888$$

For ZipDigits.test,

$$E_{out}(g) \leq 0.043269 + 0.094167$$
$$\therefore E_{out}(g) = 0.137436$$

The bound based on E_{test} is better.

e) I would use the linear model **without** the 3rd order transformation. First, E_{test} was much higher with the transformation. Also, because of the increase in the VC dimension, the case for transformation has a bigger bound. Based on the data, the 3rd order transformation might be overfitting too much to the training data, perform worse on the test data, and therefore is an overkill.

2.

(a)

$$x_{new} = x_{prev} - \eta \cdot [2x + 4\pi\cos(2\pi x)\sin(2\pi y)]$$
$$y_{new} = y_{prev} - \eta \cdot [4y + 4\pi\sin(2\pi x)\cos(2\pi y)]$$

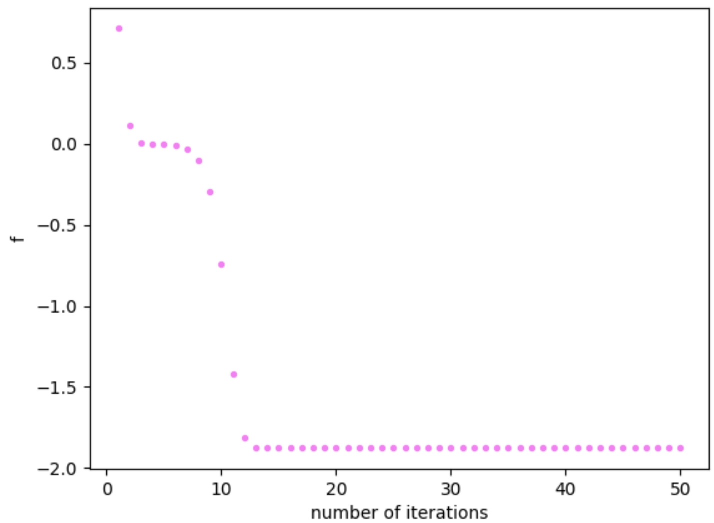

Figure 5: $\eta = 0.01$

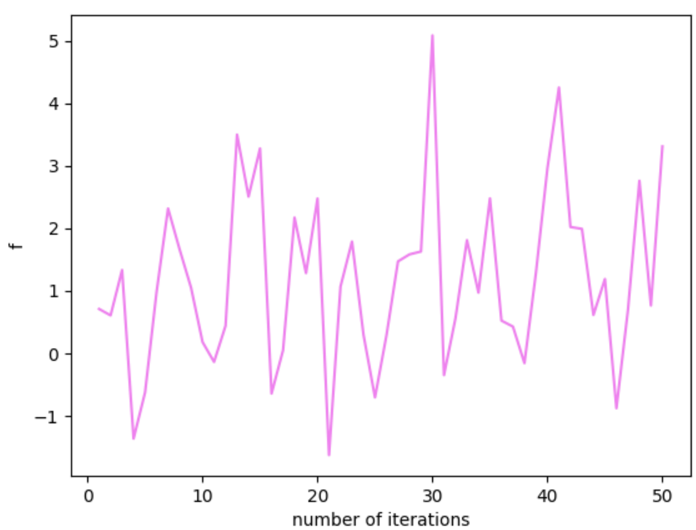

Figure 6: $\eta = 0.1$ $(doesn't\,converge\,but\,oscillates)$

(b)

Eta = 0.01			
(x, y)	x_final	y_final	f_final
(0.1, 0.1)	0.243804969	-0.237925821	-1.876687
(1, 1)	1.218070301	0.712811951	0.085168
(-0.5, -0.5)	-0.73137746	-0.237855363	-1.389056
(-1, -1)	-1.218070301	-0.712811951	0.085168
Eta = 0.1			
(x, y)	x_final	y_final	f_final
(0.1, 0.1)	0.100391674	-1.015751005	3.310198
(1, 1)	0.198723236	0.457110104	0.488978
(-0.5, -0.5)	0.172982164	-0.994336971	0.940948
(-1, -1)	-0.198723236	-0.457110104	0.488978

The table above illustrates that, depending on where in the function you start and depending on the step size, gradient descent will converge to different local minimas but not necessarily to the absolute minima.

Problem 3.16

a)

$$\begin{aligned}\text{cost(accept)} &= g(\mathbf{x}) \cdot 0 + (1 - g(\mathbf{x})) \cdot c_a \\ &= (1 - g(\mathbf{x}))c_a \\ \text{cost(reject)} &= g(\mathbf{x}) \cdot c_r + (1 - g(\mathbf{x})) \cdot 0 \\ &= g(\mathbf{x})c_r\end{aligned}$$

b)

$$\begin{aligned}\text{cost(reject)} &\geq \text{cost(accept)} \\ g(\mathbf{x})c_r &\geq (1 - g(\mathbf{x}))c_a \\ g(\mathbf{x})c_r &\geq c_a - g(\mathbf{x})c_a \\ g(\mathbf{x})(c_r + c_a) &\geq c_a \\ \therefore g(\mathbf{x}) &\geq \frac{c_a}{c_r + c_a} = \kappa.\end{aligned}$$

c)

$$\begin{aligned}\kappa_{supermarket} &= \frac{1}{1+10} = 0.09... \\ \kappa_{CIA} &= \frac{1000}{1+1000} = 0.99...\end{aligned}$$

The threshold κ values are in agreement with our understanding that, for the supermarket, a false reject is costly as the disappointed customer might never return. For the CIA, a false accept is a disaster and one must be avoided. False rejects can be tolerated in this case as the subjects are employees.