CSCI 5521 Homework 1

John Nguyen

October 2, 2019

Problem 1

For this problem, I will use the notation $(x_i, y_i)_{i=1,\ldots,n}$ instead of using the notation $(x^t, r^t)_{t=1,\ldots,n}$.

(i). We want to find the values of w_1^*, w_0^* such that:

$$E(w_1, w_0|\mathcal{Z}_{train}) = \frac{1}{N}\sum_{t=1}^{N}(y_t - (w_1 x_t + w_0))^2$$

Notice that by expanding the the square, we find that:

$$E(w_1, w_0|\mathcal{Z}_{train}) = \frac{1}{N}\sum_{t=1}^{N}(y_t - (w_1 x_t + w_0))^2 = \frac{1}{N}\sum_{t=1}^{N}(2w_0 x_t w_1 + x^2 w_1^2 - 2x_t y_t w_1 - 2w_0 y_t + w_0^2 + y_t^2)$$

Notice that $E(w_1, w_0|\mathcal{Z}_{train})$ is a convex upward (in fact, it is quadratic with positive coefficients) function with respect to both w_1 and w_0. Thus it is sufficient to take the partial derivatives of E with respect to w_1 or w_0 and then solve for w_1 or w_0 respectively. We will solve for the desired w_0 first:

$$\begin{aligned}
\frac{\partial E}{\partial w_0} &= \frac{1}{N}\sum_{t=1}^{N}(2x_t w_1 - 2y_t + 2w_0) \\
&= \frac{1}{N}\sum_{t=1}^{N} 2x_t w_1 - \frac{1}{N}\sum_{t=1}^{N} 2y_t + \frac{1}{N}\sum_{t=1}^{N} 2w_0 \\
&= 2w_1\overline{x} - 2\overline{y} + 2w_0 \\
&= 0
\end{aligned}$$

where $\overline{x} = \frac{1}{N}\sum_{t=1}^{N} x_t$ and $\overline{y} = \frac{1}{N}\sum_{t=1}^{n} y_t$. We now solve for w_0:

$$w_0 = \overline{y} - w_1\overline{x}$$

So $w_0 = \overline{y} - w_1\overline{x}$. Notice that the desired value of w_0 is dependent on the desired value of w_1. We will now calculate the desired value of w_1:

$$\begin{aligned}
\frac{\partial E}{\partial w_1} &= \frac{1}{N}\sum_{t=1}^{N}(2w_0 x_t + 2x^2 w_1 - 2x_t y_t) \\
&= \frac{1}{N}\sum_{t=1}^{N} 2w_0 x_t + \frac{1}{N}\sum_{t=1}^{N} 2x^2 w_1 - \frac{1}{N}\sum_{t=1}^{N} 2x_t y_t
\end{aligned}$$

We now substitute the formula we found for the desired w_0 value, $w_0 = \overline{y} - w_1\overline{x}$:

$$\begin{aligned}
\frac{1}{N}\sum_{t=1}^{N} 2w_0x_t + \frac{1}{N}\sum_{t=1}^{N} 2x_t^2w_1 - \frac{1}{N}\sum_{t=1}^{N} 2x_ty_t &= \frac{1}{N}\sum_{t=1}^{N} 2(\overline{y} - w_1\overline{x})x_t + \frac{1}{N}\sum_{t=1}^{N} 2x_t^2w_1 - \frac{1}{N}\sum_{t=1}^{N} 2x_ty_t \\
&= \frac{1}{N}\sum_{t=1}^{N} 2\overline{y}x_t - \frac{1}{N}\sum_{t=1}^{N} 2w_1\overline{x}x_t + \frac{1}{N}\sum_{t=1}^{N} 2x_t^2w_1 - \frac{1}{N}\sum_{t=1}^{N} 2x_ty_t \\
&= 2\frac{1}{N}\overline{y}\sum_{t=1}^{N} x_t - 2\frac{1}{N}w_1\overline{x}\sum_{t=1}^{N} x_t + 2\frac{1}{N}w_1\sum_{t=1}^{N} x_t^2 - 2\frac{1}{N}\sum_{t=1}^{N} x_ty_t \\
&= 2\overline{yx} - 2w_1\overline{x}^2 + \frac{2}{N}w_1\sum_{t=1}^{N} x_t^2 - \frac{2}{N}\sum_{t=1}^{N} x_ty_t
\end{aligned}$$

We now solve for w_1 :

$$w_1 = \frac{\sum_{t=1}^{N} x_ty_t - N\overline{xy}}{\sum_{t=1}^{N} x_t^2 - N\overline{x}^2}$$

Thus the optimal values of (w_1, w_0) are: $(w_1 = \frac{\sum_{t=1}^{N} x_ty_t - \overline{xy}N}{\sum_{t=1}^{N} x_t^2 - N\overline{x}^2}, w_0 = \overline{y} - w_1\overline{x})$, i.e. $(w_1 = \frac{\sum_{t=1}^{N} x_ty_t - \overline{xy}N}{\sum_{t=1}^{N} x_t^2 - N\overline{x}^2}, w_0 = \overline{y} - (\frac{\sum_{t=1}^{N} x_ty_t - \overline{xy}N}{\sum_{t=1}^{N} x_t^2 - N\overline{x}^2})\overline{x})$.

(ii). While we could use the method in (i), it would be cumbersome. We will calculate the optimal values in a different way.

Consider the error function:

$$\begin{aligned}
E(v_2, v_1, v_0|\mathcal{Z}_{train}) &= \frac{1}{N}\sum_{t=1}^{N}(y_t - (v_2x_t^2 + v_1x_t + v_0))^2 \\
&= \frac{1}{N}\sum_{t=1}^{N}(v_2^2x_t^4 + 2v_2v_1x_t^3 + 2v_2v_0x_t^2 - 2v_2x_t^2y_t + v_1^2x_t^2 + 2v_1v_0x_t - 2v_1x_ty_t + v_0^2 - 2v_0y_t + y_t^2)
\end{aligned}$$

If we consider E only as a function of v_2, notice that we obtain a quadratic function:

$$E(v_2, v_1, v_0|\mathcal{Z}_{train}) = \frac{1}{N}\left[v_2^2\sum_{t=1}^{N} x_t^4 + 2v_2\sum_{t=1}^{N} x_t^2(v_1x_t + v_0 - y_t) + \sum_{t=1}^{N}(v_1x_t + c - y_t)^2\right]$$

Since this is a quadratic with a positive coefficient on the highest term, this function is concave upwards and attains its minimum value when $v_2 = \frac{-b}{2a}$ where $b = 2\sum_{t=1}^{N}(v_1x_t + v_0 - y_t)$ and $a = \sum_{t=1}^{N} x_t^4$:

$$v_2 = \frac{-2\sum_{t=1}^{N}(v_1x_t^3 + v_0x_t^2 - y_tx_t^2)}{2\sum_{t=1}^{N} x_t^4} = \frac{-\sum_{t=1}^{N}(v_1x_t^3 + v_0x_t^2 - y_tx_t^2)}{\sum_{t=1}^{N} x_t^4}$$

We then cross multiply to obtain the formula:

$$v_2 \sum_{t=1}^{N} x_t^4 = -\sum_{t=1}^{N}(v_1 x_t^3 + v_0 x_t^2 - x_t^2 y_t) = \sum_{t=1}^{N} -v_1 x_t^3 - \sum_{t=1}^{N} v_0 x_t^2 + \sum_{t=1}^{N} y_t x_t^2$$

We rearrange the terms to get the formula:

$$v_2 \sum_{t=1}^{N} x_t^4 + v_1 \sum_{t=1}^{N} x_t^3 + v_0 \sum_{t=1}^{N} x_t^2 = \sum_{t=1}^{N} x_t^2 y_t$$

Which is now a linear equation. We now do the same process, except consider E to be a function of v_1. So we obtain the quadratic:

$$v_1^2 \sum_{t=1}^{N} x_t^2 + 2v_1 \sum_{t=1}^{N}(v_2 x_t^3 + v_0 x_t - x_t y_t) + \sum_{t=1}^{N}(v_2 x_t^2 + v_0 - y_t)^2$$

which has its minimum value when $v_1 = \frac{\sum_{t=1}^{N}(v_2 x_t^3 + v_0 x_t - x_t y_t)}{\sum_{t=1}^{N} x_t^2}$. Again, by cross multiplying, we obtain the linear equation:

$$v_2 \sum_{t=1}^{N} x_t^3 + v_1 \sum_{t=1}^{N} x_t^2 + v_0 \sum_{t=1}^{N} x_t = \sum_{t=1}^{N} x_t y_t$$

We again, use the same process, considering E as a function of v_0 to obtain the quadratic:

$$v_0^2 \sum_{t=1}^{N} 1 + 2v_0 \sum_{t=1}^{N}(v_2 x_t^3 + v_1 x_t - y_t) + \sum_{t=1}^{N}(v_2 x_t^2 + v_1 x_t - y_t)^2$$

which obtains its minimum value when $v_0 = \frac{\sum_{t=1}^{N}(v_2 x_t^2 + v_1 x_t - y_t)}{N}$. By cross multiplying, we obtain the linear formula:

$$v_2 \sum_{t=1}^{N} x_t^2 + v_1 \sum_{t=1}^{N} x_t + v_0 n = \sum_{t=1}^{N} y_t$$

So we now have a system of 3 equations with 3 unknowns:

$$\begin{gathered}
v_2 \sum_{t=1}^{N} x_t^4 + v_1 \sum_{t=1}^{N} x_t^3 + v_0 \sum_{t=1}^{N} x_t^2 = \sum_{t=1}^{N} x_t^2 y_t \\
v_2 \sum_{t=1}^{N} x_t^3 + v_1 \sum_{t=1}^{N} x_t^2 + v_0 \sum_{t=1}^{N} x_t = \sum_{t=1}^{N} x_t y_t \\
v_2 \sum_{t=1}^{N} x_t^2 + v_1 \sum_{t=1}^{N} x_t + v_0 n = \sum_{t=1}^{N} y_t
\end{gathered}$$

We can represent this system using matrices:

$$\begin{bmatrix} \sum_{t=1}^{N} x_t^4 & \sum_{t=1}^{N} x_t^3 & \sum_{t=1}^{N} x_t^2 \\ \sum_{t=1}^{N} x_t^3 & \sum_{t=1}^{N} x_t^2 & \sum_{t=1}^{N} x_t \\ \sum_{t=1}^{N} x_t^2 & \sum_{t=1}^{N} x_t & n \end{bmatrix} \begin{bmatrix} v_2 \\ v_1 \\ v_0 \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^{N} x_t^2 y_t \\ \sum_{t=1}^{N} x_t y_t \\ \sum_{t=1}^{N} y_t \end{bmatrix}$$

Therefore, to solve for the optimal values of v_2, v_1 and v_0, we simply compute:

$$\begin{bmatrix} v_2 \\ v_1 \\ v_0 \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^{N} x_t^4 & \sum_{t=1}^{N} x_t^3 & \sum_{t=1}^{N} x_t^2 \\ \sum_{t=1}^{N} x_t^3 & \sum_{t=1}^{N} x_t^2 & \sum_{t=1}^{N} x_t \\ \sum_{t=1}^{N} x_t^2 & \sum_{t=1}^{N} x_t & n \end{bmatrix}^{-1} \begin{bmatrix} \sum_{t=1}^{N} x_t^2 y_t \\ \sum_{t=1}^{N} x_t y_t \\ \sum_{t=1}^{N} y_t \end{bmatrix}$$

(iii). Professor Gopher's claim is correct. Recall the error functions we are working with: $E(w_1, w_0|\mathcal{Z}_{train}) = \frac{1}{N}\sum_{t=1}^{N}(y_t - (w_1 x_t + w_0))^2$ and $E(v_2, v_1, v_0|\mathcal{Z}_{train}) = \frac{1}{N}\sum_{t=1}^{N}(y_t - (v_2 x_t^2 + v_1 x_t + v_0))^2$. In both error functions, the training data $\mathcal{Z}_{train} = \{x_1, x_2, x_3, ..., x_N\}$ are a set of fixed values.

Since w_1^* and w_0^* are the optimal coefficients for minimizing the linear regression error function, E, we can say that for all $w_1, w_0 \in \mathbb{R}$, $E(w_1^*, w_0^*) \leq E(w_1, w_0)$.

On the other hand, the quadratic regression model generated in 1(ii) uses the optimal values v_2^*, v_1^* and v_0^* which minimize the quadratic regression error function $E(v_2, v_1, v_0)$. For this reason, we can say that for all $v_2, v_1, v_0 \in \mathbb{R}$, we have the inequality: $E(v_2^*, v_1^*, v_0^*) \leq E(v_2, v_1, v_0)$.

Let $v_2 = 0$, then the quadratic regression error function becomes the linear regression error function:

$$E(v_2 = 0, v_1, v_0|\mathcal{Z}_{train}) = \frac{1}{N}\sum_{t=1}^{N}(y_t - ((0)x_t^2 + v_1 x_t + v_0))^2 = \frac{1}{N}\sum_{t=1}^{N}(y_t - (v_1 x_t + v_0))^2 = E(w_1 = v_1, w_0 = v_0|\mathcal{Z}_{train})$$

Since v_1, v_0, w_1 and w_0 are all arbitrary real numbers, we find that $E(v_2 = 0, v_1, v_0|\mathcal{Z}_{train}) = E(w_1, w_0|\mathcal{Z}_{train})$ by substituting the changing dummy variable names, v_1 to w_1 and v_0 to w_0.

Since v_2^*, v_1^* and v_0^* are the optimal values for quadratic regression, we find that $E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{train}) \leq E(v_2 = 0, v_1, v_0)$ for any choice of v_1 and v_0. Choose $v_1 = w_1^*$ and $v_0 = w_0^*$. We then obtain the following inequality:

$$E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{train}) \leq E(v_2 = 0, v_1 = w_1^*, v_0 = w_0^*|\mathcal{Z}_{train}) = E(w_1^*, w_0^*|\mathcal{Z}_{train})$$

So $E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{train}) \leq E(w_1^*, w_0^*|\mathcal{Z}_{train})$, as desired.

(iv). False; the quadratic model may be overfitting.

In order to prove Professor Gopher's claim false, it is sufficient to prove there exists a test set $\mathcal{Z}_{test}$ such that $E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{test}) > E(w_1^*, w_0^*|\mathcal{Z}_{test})$.

Let w_1^* and w_0^* the optimal coefficients found in 1(i) and let v_2^*, v_1^* and v_0^* be the optimal coefficients found in 1(ii). Suppose $v_2^* x^2 + v_1^* x + v_0 \neq w_1^* x + w_0$. So there exists some value $x' \in \mathbb{R}$ such that $v_2^* x'^2 + v_1^* x' + v_0 \neq w_1^* x' + w_0^*$. Let $\mathcal{Z}_{test} = \{(x', y')\}$, where $y' = w_1^* x' + w_0^*$. We now compute the testing error of the linear and quadratic models:

$$
\begin{aligned}
E(w_1^*, w_0^*|\mathcal{Z}_{test}) &= \frac{1}{1}\sum_{t=1}^{1}(y_t - (w_1^* x_t + w_0))^2 \\
&= (y' - (w_1^* x' + w_0))^2 \\
&= (y' - w_1^* x' - w_0)^2 \\
&= ((w_1^* x' + w_0) - w_1^* x' - w_0)^2 \\
&= (w_1^* x' + w_0 - w_1^* x' - w_0)^2 \\
&= 0^2 \\
&= 0
\end{aligned}
$$

$$
\begin{aligned}
E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{test}) &= \frac{1}{1}\sum_{t=1}^{1}(y_t - (v_2^* x_t^2 + v_1^* x_t + v_0))^2 \\
&= (y' - (v_2^* x'^2 + v_1^* x' + v_0))^2 \\
&= (y' - v_2^* x'^2 - v_1^* x' - v_0)^2 \\
&= ((w_1^* x' + w_0^*) - v_2^* x'^2 - v_1^* x' - v_0)^2 \\
&= (w_1^* x' + w_0^* - v_2^* x'^2 - v_1^* x' - v_0)^2
\end{aligned}
$$

Recall that $w_1^* x' + w_0 \neq v_2^* x'^2 + v_1^* x' + v_0$. Therefore, $w_1^* x' + w_0^* - v_2^* x'^2 - v_1^* x' - v_0 = \delta \neq 0$. Therefore:

$$E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{test}) = (w_1^* x' + w_0^* - v_2^* x'^2 - v_1^* x' - v_0)^2 = \delta^2$$

Since $\delta \neq 0$, we know $\delta^2 > 0$. So:

$$E(w_1^*, w_0^*|\mathcal{Z}_{test}) = 0 < \delta^2 = E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{test})$$

Now, suppose $v_2^* x^2 + v_1^* x + v_0 = w_1^* x + w_0^*$. Then for any set of points $\mathcal{Z}$, $E(w_1^*, w_0^*|\mathcal{Z}) = E(v_2^*, v_1^*, v_0^*|\mathcal{Z})$.

We have proven Professor Gopher's claim false by finding a test set $\mathcal{Z}_{test}$ such that $E(w_1^*, w_0^*|\mathcal{Z}_{test}) < E(v_2^*, v_1^*, v_0^*|\mathcal{Z}_{t}$

Problem 2

(i). Let A be the matrix defined in the problem. Then:

$$
\begin{aligned}
&\text{tr}(A) = 701 \\
&\text{tr}(A^T) = 701 \\
&\text{tr}(A^T A) = 394807 \\
&\text{tr}(AA^T) = 394807
\end{aligned}
$$

These values can be calculated by running the following code:

```
import numpy as np

A = [[1, 1, 1, 1, 1],
     [1, 2, 3, 4, 5],
     [1, 3, 9, 27, 81],
     [1, 4, 16, 64, 256],
     [1, 5, 25, 125, 625]]
A_t = np.transpose(A)
AA_t = A @ A_t
A_tA = A_t @ A

print(np.trace(A))
print(np.trace(A_t))
print(np.trace(A_tA))
print(np.trace(AA_t))
```

(ii). $\det(A)$ is the volume of the parallelepiped formed by the row vectors a_i^T, where a_i is the i^{th} column of A. Since the row vectors are 5 dimensional, the shape would actually be some polytope.

(iii). We calculate A^{-1} using numpy by executing the following code:

```
import numpy as np

A = [[1, 1, 1, 1, 1],
     [1, 2, 3, 4, 5],
     [1, 3, 9, 27, 81],
     [1, 4, 16, 64, 256],
     [1, 5, 25, 125, 625]]

np.linalg.inv(A)
```

Notice that there was no error. Therefore A is invertible, so $\det(A) > |0|$.

(iv). From 2(iii), we know A is invertible, so A has full rank. Since $A \in \mathbb{R}^{5\times 5}$, it must be that $\text{rank}(A) = 5$.

Problem 3

(i). (a). See p3main.py

(b).

Error Rates for LinearSVC with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.51	0.14	0.47	0.18	0.12	0.10	0.52	0.60	0.04	0.50	0.32	0.21

Error Rates for LinearSVC with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.16	0.22	0.00	0.51	0.16	0.25	0.14	0.28	0.04	0.10	0.19	0.14

Error Rates for LinearSVC with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.11	0.07	0.12	0.09	0.06	0.04	0.03	0.06	0.12	0.12	0.08	0.03

Error Rates for SVC with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.67	0.76	0.86	0.86	0.86	0.90	0.52	0.56	0.88	0.62	0.75	0.14

Error Rates for SVC with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.18	0.29	0.00	0.65	0.43	0.69	0.14	0.14	0.04	0.04	0.26	0.24

Error Rates for SVC with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.61	0.48	0.59	0.51	0.51	0.56	0.58	0.50	0.44	0.58	0.53	0.05

Error Rates for LogisticRegression with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.12	0.16	0.14	0.08	0.18	0.06	0.32	0.24	0.06	0.22	0.16	0.08

Error Rates for LogisticRegression with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.08	0.12	0.00	0.16	0.16	0.12	0.10	0.12	0.04	0.06	0.09	0.05

Error Rates for LogisticRegression with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.09	0.05	0.11	0.08	0.06	0.03	0.02	0.04	0.14	0.06	0.07	0.04

(ii). (a) See p3main.py

(b)

Error Rates for LinearSVC with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.31	0.27	0.21	0.18	0.33	0.21	0.42	0.16	0.20	0.22	0.25	0.08

Error Rates for LinearSVC with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.26	0.12	0.16	0.13	0.13	0.23	0.20	0.13	0.15	0.23	0.17	0.05

Error Rates for LinearSVC with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.09	0.08	0.08	0.07	0.08	0.09	0.09	0.09	0.10	0.09	0.09	0.01

Error Rates for SVC with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.52	0.42	0.44	0.44	0.49	0.45	0.39	0.49	0.41	0.45	0.45	0.04

Error Rates for SVC with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.27	0.25	0.27	0.27	0.27	0.25	0.24	0.25	0.26	0.26	0.26	0.01

Error Rates for SVC with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.78	0.80	0.77	0.86	0.84	0.84	0.83	0.81	0.75	0.83	0.81	0.03

Error Rates for LogisticRegression with Boston50											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.17	0.17	0.13	0.17	0.14	0.16	0.14	0.17	0.17	0.20	0.16	0.02

Error Rates for LogisticRegression with Boston75											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.11	0.08	0.11	0.10	0.11	0.10	0.09	0.09	0.09	0.13	0.10	0.01

Error Rates for LogisticRegression with Digits											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.07	0.07	0.07	0.07	0.07	0.08	0.06	0.06	0.07	0.06	0.07	0.01

Problem 4

(a). See p4main.py

(b).

Error Rates for LinearSVC with X1											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.11	0.09	0.18	0.11	0.12	0.14	0.07	0.06	0.16	0.15	0.12	0.04

Error Rates for LinearSVC with X2											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.05	0.00	0.07	0.00	0.03	0.02	0.01	0.01	0.04	0.03	0.03	0.02

Error Rates for SVC with X1											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.91	0.91	0.91	0.90	0.90	0.91	0.91	0.91	0.89	0.91	0.90	0.10

Error Rates for SVC with X2											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.91	0.91	0.91	0.90	0.90	0.91	0.91	0.91	0.83	0.87	0.89	0.03

Error Rates for LogisticRegression with X1											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.09	0.07	0.16	0.10	0.10	0.03	0.07	0.06	0.14	0.13	0.09	0.04

Error Rates for LogisticRegression with X2											
Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	SD
0.05	0.00	0.06	0.01	0.03	0.02	0.01	0.01	0.04	0.03	0.02	0.02