EECS 545 Machine Learning: Homework #2

Due on February 8, 2022 at 12pm

Professor Honglak Lee Section A

Yuang Huang
yahuang@umich.edu

Problem 1

Logistic regression

Solution

Part a: Hessian H

$$l(\mathbf{w}) = \sum_{i=1}^{N} y^{(i)} \log h(\mathbf{x}^{(i)}) + (1 - y^{(i)}) \log(1 - h(\mathbf{x^{(i)}})), \tag{1}$$

where $h(\mathbf{x}) = \sigma(\mathbf{w^T x}) = \frac{1}{1+\exp(-\mathbf{w^T x})}$ and we denote that $pred = \mathbf{w^T x}$.
Then we assume that:

$$l_i(\mathbf{w}) = y^{(i)} \log \sigma(pred^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma(pred^{(i)})), \tag{2}$$

where we know that $\frac{\partial pred}{\partial \mathbf{w}} = \mathbf{x^T}$ and $\frac{\partial pred}{\partial \mathbf{w^T}} = \mathbf{x}$.
It can be shown that:

$$\begin{aligned} \nabla l_i(\mathbf{w}) &= \frac{y^{(i)} x^{(i)}}{\sigma(pred^{(i)})} - \frac{(1 - y^{(i)}) x^{(i)}}{(1 - \sigma(pred^{(i)}))} \\ &= y^{(i)} x^{(i)} (1 - \sigma(pred^{(i)})) - (1 - y^{(i)}) x^{(i)} \sigma(pred^{(i)}) \\ &= x^{(i)} y^{(i)} - x^{(i)} \sigma(pred^{(i)}) \end{aligned} \tag{3}$$

Then we can be write:

$$\begin{aligned} H^{(i)} = \nabla^2 l_i(\mathbf{w}) &= -x^{(i)} x^{(i)^T} \frac{1}{1 + \exp(pred^{(i)})} \frac{\exp(pred^{(i)})}{1 + \exp(pred^{(i)})} \\ &= -x^{(i)} x^{(i)^T} \sigma(pred^{(i)})(1 - \sigma(pred^{(i)})) \end{aligned} \tag{4}$$

so the Hessian H is written by:

$$H = -\mathbf{x R x^T} \tag{5}$$

where R is the diagnal matrix that the diagnal elements are $\sigma(pred^{(i)})(1 - \sigma(pred^{(i)}))$. Thus,

$$\begin{aligned} \mathbf{z^T} H \mathbf{z} &= -\mathbf{z x R x^T z^T} \\ &= -||\mathbf{z^T R X}||^2 \leq 0. \end{aligned} \tag{6}$$

So it is shown that Hessian H is negative semi-definite and thus l is concave and has no local maxima other than the global one.

Problem 1 continued on next page...

Part b:

Since Hessian $H = -\mathbf{x}\mathbf{R}\mathbf{x}^{\mathbf{T}}$, and R depends on w (and vice versa), we get iterative reweighted least squares (IRLS)

$$\begin{aligned} R_{ii} &= \sigma(pred^{(i)})(1-\sigma(pred^{(i)})) \\ &= h^{(n)}(1-h(n)), \end{aligned} \tag{7}$$

where

$$\begin{aligned} \mathbf{w}^{(new)} &= \left(\mathbf{\Phi}^{\mathbf{T}}\mathbf{R}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathbf{T}}\mathbf{R}\mathbf{z} \\ \mathbf{z} &= \mathbf{\Phi}\mathbf{w}^{(old)} - \mathbf{R}^{-1}(\mathbf{h}-\mathbf{y}). \end{aligned} \tag{8}$$

Initialize Newton's method with $\mathbf{w} = \mathbf{0}$, after iterations, $\mathbf{w}$ are shown as followed:

- $w_0 = -1.84922892$
- $w_1 = -0.62814188$
- $w_2 = 0.85846843$

So the slope term of the decision boundary is $-\frac{w_1}{w_2} = 0.73170061$ and the itercept term of the decision boundary is $-\frac{w_0}{w_2} = 2.15410241$.

Part c:

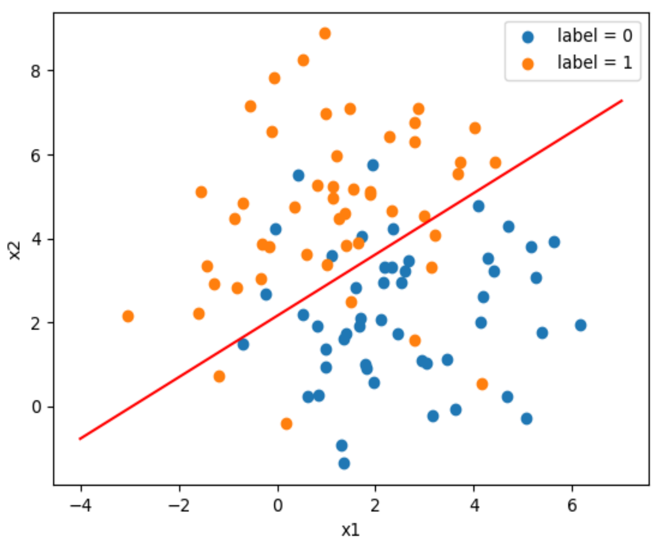

Figure 1: The training data and the decision boundary fit by logistic regression

As shown in Fig. 1, we can see the training data and the decision boundary fit by logistic regression. It obviously has a good classification effect.

Problem 2

Softmax Regression via Gradient Ascent

Solution
Part a:

$$\nabla_{\mathbf{w_m}} l(\mathbf{w}) = \sum_{i=1}^{N} \phi(\mathbf{x}^{(i)}) \left[\mathbf{I}(y^{(i)} = m) - \frac{\exp(\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}))}{1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)}))} \right], \tag{9}$$

and we know :

$$\begin{aligned} p(y = k|\mathbf{x}, \mathbf{w}) &= \frac{\exp(\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}))}{1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)}))}, \\ l(\mathbf{w}) &= \sum_{i=1}^{N} \sum_{k=1}^{K} \log(\left[p(y^{(i)} = k|\mathbf{x}^{(i)}, \mathbf{w})\right]^{\mathbf{I}(y^{(i)=k})}). \end{aligned} \tag{10}$$

with (9) and (10), we can get:

$$\begin{aligned} \nabla_{\mathbf{w_m}} l(\mathbf{w}) &= \nabla_{\mathbf{w}} \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbf{I}(y^{(i)} = k) \left[\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}) - \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \\ &= \sum_{i=1}^{N} \nabla_{\mathbf{w}} \sum_{k=1}^{K} \mathbf{I}(y^{(i)} = k) \left[\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}) - \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \\ &= \sum_{i=1}^{N} \Big(\nabla_{\mathbf{w}} \sum_{k \neq m}^{K} \mathbf{I}(y^{(i)} = k) \left[\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}) - \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \\ &+ \nabla_{\mathbf{w}} \mathbf{I}(y^{(i)} = m) \left[\mathbf{w}_m^T \phi(\mathbf{x}^{(i)}) - \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \Big) \\ &= \sum_{i=1}^{N} \Big(- \nabla_{\mathbf{w}} \sum_{k \neq m}^{K} \mathbf{I}(y^{(i)} = k) \left[\log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \\ &+ \mathbf{I}(y^{(i)} = m) \left[\phi(\mathbf{x}^{(i)}) - \nabla_{\mathbf{w}} \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})) \right]) \Big) \\ &= \sum_{i=1}^{N} \Big(\mathbf{I}(y^{(i)} = m) \phi(\mathbf{x}^{(i)}) - \nabla_{\mathbf{w}} \log(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)}))) \Big) \\ &= \sum_{i=1}^{N} \phi(\mathbf{x}^{(i)}) \Big(\mathbf{I}(y^{(i)} = m) - \nabla_{\mathbf{w}} \frac{\exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)}))}{(1 + \sum_{j=1}^{K-1} \exp(\mathbf{w}_j^T \phi((\mathbf{x})^{(i)})))} \Big) \\ &= \sum_{i=1}^{N} \phi(\mathbf{x}^{(i)}) \Big(\mathbf{I}(y^{(i)} = m) - p(y^{(i)} = m|\mathbf{x}^{(i)}, \mathbf{w}) \Big) \end{aligned} \tag{11}$$

Problem 2 continued on next page...

Part b:

After 500 iterations, the accuracy of the predictions from my model is 0.98 and the accuracy of the predictions from sklearn's LogisticRegression is 0.92.

The predictions from sklearn's LogisticRegression:
[2. 3. 1. 1. 1. 2. 2. 1. 1. 3. 2. 3. 1. 2. 3. 2. 1. 1. 2. 1. 3. 2. 3. 3. 1. 3. 3. 3. 3. 3. 2. 3. 2. 1. 3. 2. 1. 2. 3. 1. 1. 1. 1. 2. 1. 2. 2. 2. 3. 3.]

The predictions from my model (500iter):
[2. 3. 1. 1. 1. 2. 3. 1. 1. 3. 2. 3. 1. 2. 3. 2. 1. 1. 2. 1. 3. 2. 3. 3. 1. 3. 3. 3. 3. 3. 2. 3. 2. 1. 3. 2. 1. 2. 3. 1. 1. 1. 1. 2. 1. 2. 3. 3. 3. 3.]

The result of the predictions from sklearn's LogisticRegression:
[True True True True True True False True True True True True True True True True True True True True True True True True True True True False True True True True True True True True True True True True True True True True True True False False True True]

The predictions from my model (500iter):
[True True True True True True True True True True True True True True True True True True True True True True True True True True True False True True True True True True True True True True True True True True True True True True True True True True]

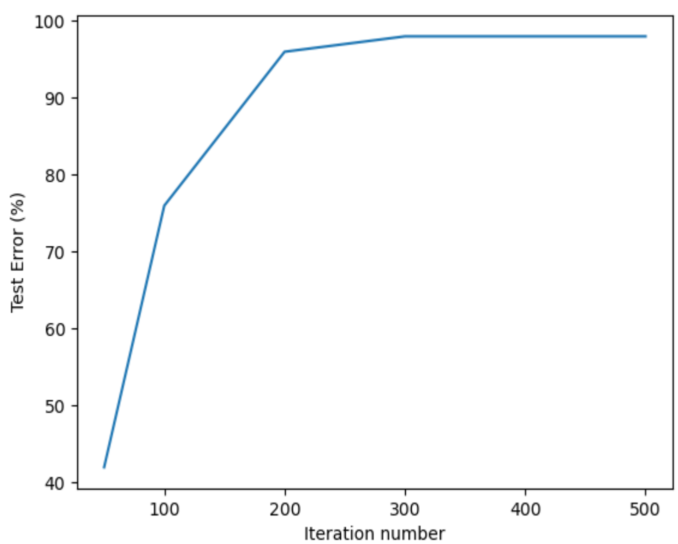

Figure 2: the test of error with respect to size of training sets

As shown in Fig. 2, we can see the the test of error with respect to iteration numbers. When the iteration number is 200, the performance is better than sklearn's LogisticRegression.

Problem 3

Gaussian Discriminate Analysis

Solution
Part a:

$$
\begin{aligned}
p(y=1|\mathbf{x}^{(i)}) &= \frac{p(\mathbf{x}^{(i)}|y=1)p(y=1)}{p(\mathbf{x}^{(i)})} \\
&= \frac{p(\mathbf{x}^{(i)}|y=1)p(y=1)}{p(\mathbf{x}^{(i)}|y=1)p(y=1)+p(\mathbf{x}^{(i)}|y=0)p(y=0)} \\
&= \frac{\frac{1}{(2\pi)^{\frac{M}{2}}|\Sigma|^{\frac{1}{2}}}\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))\phi}{\frac{1}{(2\pi)^{\frac{M}{2}}|\Sigma|^{\frac{1}{2}}}\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))\phi+\frac{1}{(2\pi)^{\frac{M}{2}}|\Sigma|^{\frac{1}{2}}}\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_0)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_0))(1-\phi)} \\
&= \frac{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))\phi}{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))\phi+\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_0)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_0))(1-\phi)} \\
&= \frac{1}{1+\frac{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))\phi}{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_0)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_0))(1-\phi)}}
\end{aligned} \tag{12}
$$

Then, we can assum that:

$$
\begin{aligned}
\log\frac{p(y=1|\mathbf{x}^{(i)})}{p(y=0|\mathbf{x}^{(i)})} &= \log\frac{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))}{\exp(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_0)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_0))}+\log\frac{p(y=1)}{p(y=0)} \\
&= (-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_1)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_1))-(-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_0)^T\Sigma^{-1}(\mathbf{x}^{(i)}-\mu_0))+\log\frac{p(y=1)}{p(y=0)} \\
&= (\mu_1-\mu_0)^T\Sigma^{-1}\mathbf{x}^{(i)}-\frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1+\frac{1}{2}\mu_0^T\Sigma^{-1}\mu_0+\log\frac{\phi}{1-\phi}
\end{aligned} \tag{13}
$$

where $(\mu_1-\mu_0)^T\Sigma^{-1}\mathbf{x}^{(i)}$ is w_1 and $-\frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1+\frac{1}{2}\mu_0^T\Sigma^{-1}\mu_0+\log\frac{\phi}{1-\phi}$ is w_0. Then add an extra coordinate $x_0=1$ to $\mathbf{x}$ and refine $\mathbf{x}$. Thus, with (12) and (13) we will get:

$$
\begin{aligned}
p(y=1|\mathbf{x}^{(i)}) &= \frac{1}{1+\exp(-\log\frac{p(y=1|\mathbf{x}^{(i)})}{p(y=0|\mathbf{x}^{(i)})})} \\
&= \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x}^{(i)})}
\end{aligned} \tag{14}
$$

so the posterior distribution of the label (y) at $\mathbf{x}$ takes the form of a logistic function, and can be written as:

$$
p(y=1|\mathbf{x};\phi,\Sigma,\mu_0,\mu_1) = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})} \tag{15}
$$

Problem 3 continued on next page...

Part b:

Considering that part b is a special case of part c, we will directly consider M both part b and part c here.

The log-likelihood of the data is:

$$
\begin{aligned}
l(\phi, \mu, \Sigma) &= \log(\Pi_{i=1}^{N} p(\mathbf{x}^{(i)}, y^{(i)}; \phi, \mu, \Sigma)) \\
&= \log(\Pi_{i=1}^{N} p(\mathbf{x}^{(i)}|y^{(i)}; \phi, \mu, \Sigma) p(y^{(i)}; \phi)) \\
&= \log(\Pi_{i=1}^{N} p(\mathbf{x}^{(i)}|y^{(i)}; \phi, \mu, \Sigma)) + \log \Pi_{i=1}^{N} p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{N} (\log(\frac{1}{(2\pi)^{\frac{M}{2}} |\Sigma|^{\frac{1}{2}}}) - \frac{1}{2}(\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} (\mathbf{x}^{(i)} - \mu_{y^{(i)}}) \\
&+ y^{(i)} \log \phi + (1 - y^{(i)}) \log(1 - \phi))
\end{aligned}
\tag{16}
$$

Then take the partial derivative of ϕ to function l:

$$
\begin{aligned}
\frac{\partial l}{\partial \phi} &= \sum_{i=1}^{N} [\frac{y^{(i)}}{\phi} - \frac{1 - y^{(i)}}{1 - \phi}] \\
&= \frac{\sum_{i=1}^{N} y^{(i)}}{\phi} - \frac{N - \sum_{i=1}^{N} y^{(i)}}{1 - \phi},
\end{aligned}
\tag{17}
$$

let the partial derivative equal to zero:

$$
\begin{aligned}
&\frac{\sum_{i=1}^{N} y^{(i)}}{\phi} - \frac{N - \sum_{i=1}^{N} y^{(i)}}{1 - \phi} = 0 \\
&\Rightarrow \frac{\sum_{i=1}^{N} y^{(i)}}{\phi} = \frac{N - \sum_{i=1}^{N} y^{(i)}}{1 - \phi} \\
&\Rightarrow \phi = \frac{1}{N} \sum_{i=1}^{N} 1\left\{y^{(i)} = 1\right\}
\end{aligned}
\tag{18}
$$

Then take the partial derivative of μ_0 to function l:

$$
\begin{aligned}
\nabla_{\mu_0} l &= \nabla_{\mu_0} [\sum_{i:y^{(i)}=0} -\frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)^T \Sigma^{-1} (\mathbf{x}^{(i)} - \mu_0)] \\
&= \sum_{i:y^{(i)}=0} [\Sigma^{-1}\mathbf{x}^{(i)} - \Sigma^{-1}\mu_0],
\end{aligned}
\tag{19}
$$

let the partial derivative equal to zero:

$$
\mu_0 = \frac{\sum_{i=1}^{n} 1\left\{y^{(i)} = 0\right\} \mathbf{x}^{(i)}}{\sum_{i=1}^{N} 1\left\{y^{(i)} = 0\right\}}.
\tag{20}
$$

Similarly, μ_1 can be written as:

$$
\mu_1 = \frac{\sum_{i=1}^{n} 1\left\{y^{(i)} = 1\right\} \mathbf{x}^{(i)}}{\sum_{i=1}^{N} 1\left\{y^{(i)} = 1\right\}}.
\tag{21}
$$

Problem 3 continued on next page...

Then take the partial derivative of Σ to the function l, and for this situation, we only consider M equals to 1:

$$\begin{aligned}
\nabla_\Sigma l &= \nabla_\Sigma[\sum_{i=1}^{N} \log(\frac{1}{(2\pi)^{\frac{M}{2}}|\Sigma|^{\frac{1}{2}}}) - \frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)^T \Sigma^{-1}(\mathbf{x}^{(i)} - \mu_0)] \\
&= \nabla_\Sigma[\sum_{i=1}^{N} \log(\frac{1}{(2\pi)^{\frac{M}{2}}}) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)^T \Sigma^{-1}(\mathbf{x}^{(i)} - \mu_0)] \\
&= \sum_{i=1}^{N}[-\frac{1}{2\Sigma} + \frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)(\mathbf{x}^{(i)} - \mu_0)^T \frac{1}{\Sigma^2}],
\end{aligned} \tag{22}$$

let the partial derivative equal to zero:

$$\begin{aligned}
&\sum_{i=1}^{N}[-\frac{1}{2\Sigma} + \frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)(\mathbf{x}^{(i)} - \mu_0)^T \frac{1}{\Sigma^2}] = 0 \\
&\Rightarrow \sum_{i=1}^{N}[\frac{1}{2}(\mathbf{x}^{(i)} - \mu_0)(\mathbf{x}^{(i)} - \mu_0)^T \frac{1}{\Sigma^2}] = \sum_{i=1}^{N} \frac{1}{2\Sigma} = \frac{N}{2\Sigma} \\
&\Rightarrow \Sigma = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}^{(i)} - \mu_0)(\mathbf{x}^{(i)} - \mu_0)^T.
\end{aligned} \tag{23}$$

Part c:

Considering that part b is a special case of part c, we have proved the general case in part b.

For ϕ, it is the same as b. For μ, it is also similar in b,

$$\mu_t = \frac{\sum_{i=1}^{n} 1\left\{y^{(i)} = t\right\} \mathbf{x}^{(i)}}{\sum_{i=1}^{N} 1\left\{y^{(i)} = t\right\}}. \tag{24}$$

Problem 4

Naive Bayes for classifying SPAM

Solution
Part a:
The error of the model is 1.625%.

Part b:
The index of the 5 tokens that are most indicative of the the SPAM class:
[1368 393 1356 1209 615]
The 5 tokens that are most indicative of the the SPAM class:
$['valet' \quad 'ebai' \quad 'unsubscrib' \quad 'spam' \quad 'httpaddr']$

Part c:
MATRIX.TRAIN.50 Error: 3.8750%
MATRIX.TRAIN.100 Error: 2.6250%
MATRIX.TRAIN.200 Error: 2.6250%
MATRIX.TRAIN.400 Error: 1.8750%
MATRIX.TRAIN.800 Error: 1.7500%
MATRIX.TRAIN.1400 Error: 1.6250%

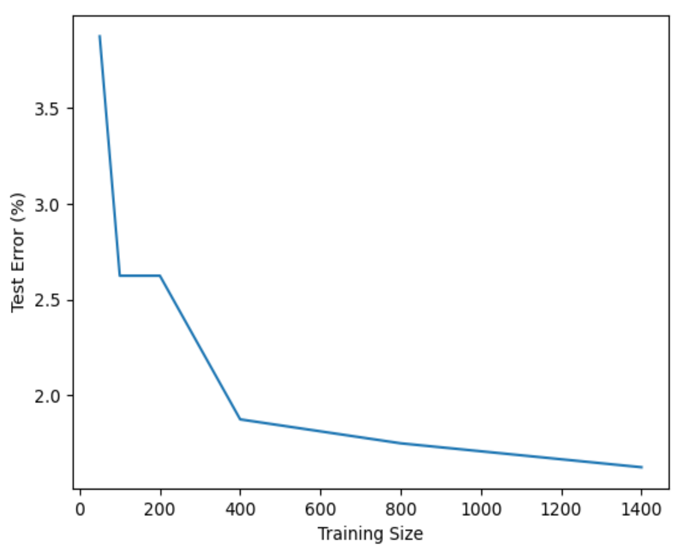

Figure 3: The test of error with respect to size of training sets

As shown in Fig. 3, we can see the the test of error with respect to size of training sets. It is obviously that the 1400 training set size gives us the best classification error.