Wensong Hu
EECS
University of Michigan

February 13, 2024

HW2 — Classification

1 Logistic Regression

1.1 Find Hessian H of $l(w)$

From the log-likelihood funciton for logistic regression:

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

where $h(x) = \sigma(w^T x) = \frac{1}{1+exp(-w^T x)}$.

Calculate the gradient of the log loss function:

$$\nabla_w l(w) = \sum_{i=1}^{N} \nabla_w (y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))) \tag{1-2}$$

$$= \sum_{i=1}^{N} y^{(i)} \nabla \log h(x) + (1 - y^{(i)}) \nabla_w (\log(1 - h(x))) \tag{1-3}$$

$$= \sum_{i=1}^{N} y^{(i)} \frac{1}{h(x)} h'(x)(w^T x)' + (1 - y^{(i)}) \frac{1}{1 - h(x)} (1 - h(x))'(w^T x)' \tag{1-4}$$

$$= \sum_{i=1}^{N} y^{(i)} \frac{1}{\sigma(w^T x)} \sigma(w^T x)(1 - \sigma(w^T x))(w^T x)' + (1 - y^{(i)}) \frac{1}{1 - \sigma(w^T x)} - \sigma(w^T x)(1 - \sigma(w^T x))(w^T x) \tag{1-5}$$

$$= \sum_{i=1}^{N} y^{(i)} (1 - \sigma(w^T x))(w^T x)' - (1 - y^{(i)}) \sigma(w^T x)(w^T x)' \tag{1-6}$$

$$= \sum_{i=1}^{N} y^{(i)} x^{(i)} - \sigma(w^T x^{(i)}) x^{(i)} \tag{1-7}$$

$$= \sum_{i=1}^{N} (y^{(i)} - \sigma(w^T x^{(i)})) x^{(i)} \tag{1-8}$$

Calculate second order gradient:

$$\begin{align}
H = \nabla_w^2 l(w) &= \frac{\partial l(w)}{\partial w_j \partial w_k} \tag{1-9}\\
&= -\sum_{i=1}^{N} \nabla_{w_k} \sigma(w^T x^{(i)}) x^{(i)} \tag{1-10}\\
&= -\sum_{i=i}^{N} \sigma(w^T x^{(i)})(1-\sigma(w^T x^{(i)}))(w^T x^{(i)})' x^{(i)} \tag{1-11}\\
&= -\sum_{i=i}^{N} h(x^{(i)})(1-h(x^{(i)})) x^{(i)} {x^{(i)}}^T \tag{1-12}\\
&= X^T \begin{bmatrix} h(x^{(1)})(1-h(x^{(1)})) & & 0 \\ & \cdots & \\ 0 & & h(x^{(N)})(1-h(x^{(N)})) \end{bmatrix} X \tag{1-13}
\end{align}$$

where, X has shape of (N, M), N is the number of data, M is the size of feature.

Hence, each entries of Hessian matrix is given by:

$$H_{jk} = -\sum_{i=i}^{N} h(x^{(N)})(1-h(x^{(N)})) x_j^{(i)} x_k^{(i)} \tag{1-14}$$

where, $x_j^{(i)}$ is the jth element of ith sample.

$$Q.E.D.\blacksquare$$

1.2 Show H is NSD

From the definition:

$$\begin{align}
z^T H z &= \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{k=1}^{M} -z^{(i)} h(x^{(i)})(1-h(x^{(N)})) x_j^{(i)} x_k^{(i)} z^{(i)} \tag{1-15}\\
&= \sum_{i=1}^{N} h(x^{(i)})(1-h(x^{(N)})) \sum_{j=1}^{M} \sum_{k=1}^{M} -z^{(i)} x_j^{(i)} x_k^{(i)} z^{(i)} \tag{1-16}\\
&= -\sum_{i=1}^{N} h(x^{(i)})(1-h(x^{(N)})) ||{x^{(i)}}^T z||_2^2 \tag{1-17}\\
& \tag{1-18}
\end{align}$$

We also have:

$$\begin{align}
||{x^{(i)}}^T z||_2^2 \geq 0 \tag{1-19}\\
0 \leq h(x) \leq 1 \tag{1-20}\\
0 \leq 1 - h(x) \leq 1 \tag{1-21}
\end{align}$$

Therefore:

$$z^T H z = -\sum_{i=1}^{N} h(x^{(i)})(1 - h(x^{(N)}))||{x^{(i)}}^T z||_2^2 \geq 0 \tag{1-22}$$

$$\Leftrightarrow H\ is\ NSD \tag{1-23}$$

$$Q.E.D.\blacksquare$$

1.3 Newton's method

$$w := w - H^{-1}\nabla_w l(w) \tag{1-24}$$

1.4 Code Implementation

Submitted in autograder

1.5 Result of w

```
  >> w = [-1.84922892 -0.62814188  0.85846843]
```

1.6 Result of plot

See Figure 1.

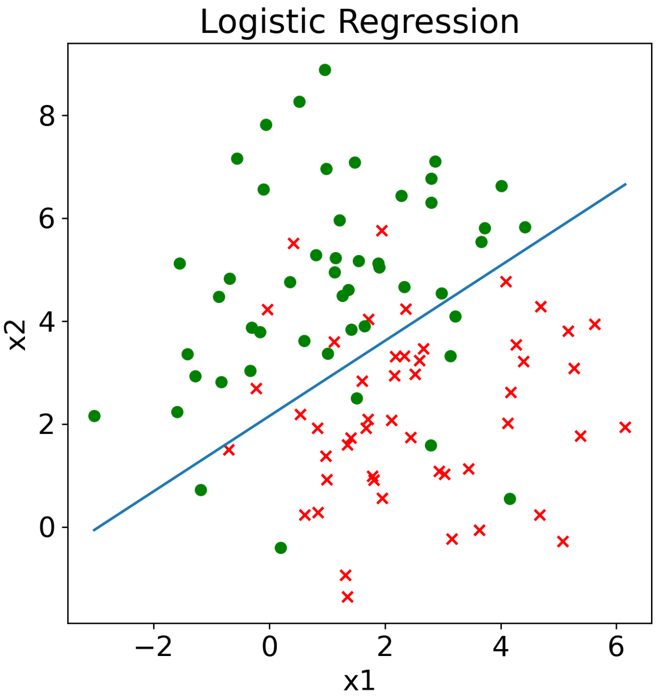

Figure 1: Logistic Regression Visualization

2 Softmax Regression via Gradient Ascent

2.1 Gradient of Softmax

$$l(w) = \log L(w) \tag{1-25}$$

$$= \sum_{i=1}^{N} \sum_{k=1}^{K} \log([p(y^{(i)} = k|x^{(i)}, w)]^{\mathbb{1}\{y^{(i)}=k\}}) \tag{1-26}$$

$$= \sum_{i=1}^{N} \log p(y^{(i)}|x^{(i)}, w) \tag{1-27}$$

Take the first order derivative:

$$\nabla_{w_m} l(w) = \sum_{i=1}^{N} \frac{\partial}{\partial w_m} \log p(y^{(i)} = m|x^{(i)}, w) \tag{1-28}$$

$$= \sum_{i=1}^{N} \frac{\partial}{\partial w_m} \log \frac{exp(w_m^T \phi(x^{(i)}))}{1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))} \tag{1-29}$$

$$= \sum_{i=1}^{N} \frac{\partial}{\partial w_m} [w_m^T \phi(x^i) - \log(1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)})))] \tag{1-30}$$

$$= \sum_{i=1}^{N} \phi(x^{(i)}) - \frac{1}{1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))} \frac{\partial}{\partial w_m} (1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))) \tag{1-31}$$

$$= \sum_{i=1}^{N} \phi(x^{(i)}) - \frac{1}{1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))} \frac{\partial}{\partial w_m} exp(w_m^T \phi(x^{(i)})) \tag{1-32}$$

$$= \sum_{i=1}^{N} \phi(x^{(i)}) - \frac{exp(w_m^T \phi(x^{(i)}))}{1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))} \phi(x^{(i)}) \tag{1-33}$$

$$= \sum_{i=1}^{N} \phi(x^{(i)}) \left(\mathbb{1}\{y^{(i)} = m\} - \frac{exp(w_m^T \phi(x^{(i)}))}{1 - \sum_{k=1}^{K-1} exp(w_k^T \phi(x^{(i)}))} \right) \tag{1-34}$$

$$= \sum_{i=1}^{N} \phi(x^{(i)}) (\mathbb{1}\{y^{(i)} = m\} - p(y^{(i)} = m|x^{(i)}, w)) \tag{1-35}$$

$$Q.E.D. \blacksquare$$

2.2 Code Implementation

Submitted to autograder

2.3 Accuracy on test data

```
    >> The accuracy of Softmax Regression - our implementation: 94.00 %
```

3 Gaussian Discriminate Analysis

3.1 Prove posterior distribution takes logistic function form

The logit is defined as:

$$\begin{aligned}
a &= \log \frac{p(y=1|x,\phi,\Sigma,\mu_0,\mu_1)}{p(y=0|x,\phi,\Sigma,\mu_0,\mu_1)} && \text{(1-36)}\\
&= \log \frac{p(y=1,x|\phi,\Sigma,\mu_0,\mu_1)}{p(y=0,x|\phi,\Sigma,\mu_0,\mu_1)} && \text{(1-37)}\\
&= \log \frac{p(x|y=1,\phi,\Sigma,\mu_0,\mu_1)p(y=1|\phi,\Sigma,\mu_0,\mu_1)}{p(x|y=0,\phi,\Sigma,\mu_0,\mu_1)p(y=0|\phi,\Sigma,\mu_0,\mu_1)} && \text{(1-38)}
\end{aligned}$$

where, $p(x|y=0,\phi,\Sigma,\mu_0,\mu_1) \sim \mathcal{N}(\mu_0,\Sigma)$ and $p(x|y=1,\phi,\Sigma,\mu_0,\mu_1) \sim \mathcal{N}(\mu_1,\Sigma)$.

Thus:

$$\begin{aligned}
a &= \log\left(\frac{\exp\left\{-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right\}}{\exp\left\{-\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\right\}}\right) + \log\left(\frac{p(y=1)}{p(y=0)}\right) && \text{(1-39)}\\
&= \left\{-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right\} - \left\{-\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\right\} + \log\left(\frac{p(y=1)}{p(y=0)}\right) && \text{(1-40)}\\
&= (\mu_1-\mu_2)^T\Sigma^{-1}x - \frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1 + \frac{1}{2}\mu_2^T\Sigma^{-1}\mu_2 + \log\left(\frac{p(y=1)}{p(y=0)}\right) && \text{(1-41)}\\
&= w^T\hat{x} && \text{(1-42)}
\end{aligned}$$

where, $w = \begin{bmatrix} (\mu_1-\mu_2)^T\Sigma^{-1} \\ -\frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1 + \frac{1}{2}\mu_2^T\Sigma^{-1}\mu_2 + \log\left(\frac{p(y=1)}{p(y=0)}\right) \end{bmatrix}$, and $\hat{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}$

From posterior distribution, we have:

$$\begin{aligned}
p(y=1|x;\phi,\Sigma,\mu_0,\mu_1) &= \frac{p(y=1|x;\phi,\Sigma,\mu_0,\mu_1)}{p(y=0|x;\phi,\Sigma,\mu_0,\mu_1) + p(y=1|x;\phi,\Sigma,\mu_0,\mu_1)} && \text{(1-43)}\\
&= \frac{\frac{p(y=1|x;\phi,\Sigma,\mu_0,\mu_1)}{p(y=0|x;\phi,\Sigma,\mu_0,\mu_1)}}{1 + \frac{p(y=1|x;\phi,\Sigma,\mu_0,\mu_1)}{p(y=0|x;\phi,\Sigma,\mu_0,\mu_1)}} && \text{(1-44)}\\
&= \frac{exp(a)}{1+exp(a)} && \text{(1-45)}\\
&= \frac{1}{1+exp(-a)} && \text{(1-46)}\\
&= \frac{1}{1+exp(w^T\hat{x})} && \text{(1-47)}
\end{aligned}$$

$$Q.E.D.\blacksquare$$

3.2 Prove ϕ, μ_0, μ_1 with MLE

The log-likelihood for generative model is:

$$\begin{aligned} l(w) = \log P(D|w) &= \log \prod_{i=1}^{N} p(x^{(i)}, y^{(i)}|w) && \text{(1-48)} \\ &= \log \prod_{i=1}^{N} p(x^{(i)}|y^{(i)}, w)p(y^{(i)}|w) && \text{(1-49)} \\ &= \sum_{i=1}^{N} \log p(x^{(i)}|y^{(i)}, w) + \log p(y^{(i)}|w) && \text{(1-50)} \end{aligned}$$

where, D is data and w are the parameters which is $\phi, \mu_0, \mu_1, \Sigma$ in this problem

3.2.1 Proof of ϕ

For ϕ, it is not depends on $x^{(i)}$, so we are only maximizing the sencond term $\log p(y^{(i)}|w)$.

$$\begin{aligned} p(y^{(i)}|w) &= \phi^{y^{(i)}}(1 - \phi^{(1-y^{(i)})}) && \text{(1-51)} \\ \log p(y^{(i)}|w) &= \log \phi^{y^{(i)}} + \log(1 - \phi^{(1-y^{(i)})}) && \text{(1-52)} \\ &= y^{(i)} \log \phi + (1 - y^{(i)}) \log(1 - \phi) && \text{(1-53)} \end{aligned}$$

Take derivative;

$$\begin{aligned} \frac{\partial l(w)}{\partial \phi} &= \frac{\partial \sum_{i=1}^{N} \log p(y^{(i)}|w)}{\partial \phi} && \text{(1-54)} \\ &= \sum_{i=1}^{N} \frac{y^{(i)}}{\phi} - \frac{1 - y^{(i)}}{1 - \phi} && \text{(1-55)} \end{aligned}$$

Set the derivative to zero to maximize the likelihood:

$$\begin{aligned} \sum_{i=1}^{N} \frac{y^{(i)}}{\phi} - \frac{1 - y^{(i)}}{1 - \phi} &= 0 && \text{(1-56)} \\ \phi &= \frac{\sum_{i=1}^{N} y^{(i)}}{N} && \text{(1-57)} \end{aligned}$$

Since this is the binary classification problem, so we have:

$$\phi = \frac{\sum_{i=1}^{N} \mathbb{1}\{y^{(i)} = 1\}}{N} \tag{1-58}$$

Q.E.D.■

3.2.2 Proof of μ_1 and μ_0

First take derivative with respect to mu_1

$$\frac{\partial l(w)}{\partial \mu_1} = \frac{\partial}{\partial \mu_1} \sum_{i=1}^{N} \log p(x^{(i)}|y^{(i)}, w) + \log p(y^{(i)}|w) \tag{1-59}$$

$$= \frac{\partial}{\partial \mu_1} \sum_{i=1}^{N} \log p(x^{(i)}|y^{(i)}, w) \tag{1-60}$$

$$= \frac{\partial}{\partial \mu_1} \sum_{i=1}^{N} \log(\mathcal{N}(x^{(i)}; \mu_1, \Sigma)^{y^{(i)}}) + \log(\mathcal{N}(x^{(i)}; \mu_0, \Sigma)^{1-y^{(i)}}) \tag{1-61}$$

$$= \frac{\partial}{\partial \mu_1} \sum_{i=1}^{N} y^{(i)} \log(\mathcal{N}(x^{(i)}; \mu_1, \Sigma)) \tag{1-62}$$

$$= \frac{\partial}{\partial \mu_1} \sum_{i=1}^{N} y^{(i)} \log\left(\frac{1}{\sqrt{(2\pi)^k |\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right)\right) \tag{1-63}$$

$$= \sum_{i=1}^{N} y^{(i)} \frac{\partial}{\partial \mu_1} \left(-\frac{1}{2} \log|\Sigma| - \frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right) \tag{1-64}$$

$$= \sum_{i=1}^{N} y^{(i)} \Sigma^{-1} (x^{(i)} - \mu_1) \tag{1-65}$$

Set the derivative to zero to maximize the likelihood:

$$\sum_{i=1}^{N} y^{(i)} \Sigma^{-1} (x^{(i)} - \mu_1) = 0 \tag{1-66}$$

$$\Sigma^{-1} \sum_{i=1}^{N} y^{(i)} (x^{(i)} - \mu_1) = 0 \tag{1-67}$$

Since Σ^{-1} is non-zero:

$$\sum_{i=1}^{N} y^{(i)} x^{(i)} - y^{(i)} \mu_1 = 0 \tag{1-68}$$

$$\mu_1 = \frac{\sum_{i=1}^{N} y^{(i)} x^{(i)}}{\sum_{i=1}^{N} y^{(i)}} \tag{1-69}$$

$$= \frac{\sum_{i=1}^{N} \mathbb{1}\{y^{(i)} = 1\} x^{(i)}}{\sum_{i=1}^{N} \mathbb{1}\{y^{(i)} = 1\}} \tag{1-70}$$

Similar to previous, we take derivative with respect to μ_0, and can get similar result:

$$\mu_1 = \frac{\sum_{i=1}^{N} \mathbb{1}\{y^{(i)} = 0\} x^{(i)}}{\sum_{i=1}^{N} \mathbb{1}\{y^{(i)} = 0\}} \tag{1-71}$$

$Q.E.D.$■

3.3 Prove Σ with MLE

The log-likelihood function for can be rewritten as:

$$l(\Sigma) = \sum_{i=1}^{N} \left[-\frac{1}{2} \log |\Sigma| - \frac{1}{2} (\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} (\mathbf{x}^{(i)} - \mu_{y^{(i)}}) \right]$$

Take the derivative with respect to Σ:

$$\frac{\partial}{\partial \Sigma} l(\Sigma) = -\frac{1}{2} \sum_{i=1}^{N} \Sigma^{-1} + \Sigma^{-1} (\mathbf{x}^{(i)} - \mu_{y^{(i)}})(\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1}$$

Setting the derivative equal to zero to find the maximum likelihood estimate::

$$-\frac{1}{2} N \Sigma^{-1} + \sum_{i=1}^{N} \Sigma^{-1} (\mathbf{x}^{(i)} - \mu_{y^{(i)}})(\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} = 0$$

Multiplying through by -2Σ and dividing by N:

$$\Sigma = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \mu_{y^{(i)}})(\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T$$

$$Q.E.D. \blacksquare$$

4 Naive Bayes for Classifying SPAM

4.1 Naive Bayes with Bayesian Smoothing

Given the Dirichlet prior:

$$P(\mu) = \frac{1}{Z} \prod_{i=1}^{K} \prod_{j=1}^{M} (\mu_{ij})^{\alpha}$$

The likelihood of observing the data D given parameters μ is:

$$P(D|\mu) = \prod_{i=1}^{K} \prod_{j=1}^{M} (\mu_{ij})^{N_{C_i}^j}$$

The MAP estimate maximizes the posterior:

$$P(\mu|D) \propto P(D|\mu) P(\mu)$$

$$P(\mu|D) \propto \prod_{i=1}^{K} \prod_{j=1}^{M} (\mu_{ij})^{N_{C_i}^j} \prod_{i=1}^{K} \prod_{j=1}^{M} (\mu_{ij})^{\alpha}$$

Combining exponents of μ_{ij}:

$$P(\mu|D) \propto \prod_{i=1}^{K}\prod_{j=1}^{M}(\mu_{ij})^{N_{C_i}^j+\alpha}$$

Taking the logarithm of the posterior for differentiation:

$$\log P(\mu|D) \propto \sum_{i=1}^{K}\sum_{j=1}^{M}(N_{C_i}^j+\alpha)\log\mu_{ij}$$

Introducing Lagrange multipliers λ_i for the constraint $\sum_{j=1}^{M}\mu_{ij}=1$:

$$\frac{\partial}{\partial\mu_{ij}}\left(\sum_{j=1}^{M}(N_{C_i}^j+\alpha)\log\mu_{ij}-\lambda_i\left(\sum_{j=1}^{M}\mu_{ij}-1\right)\right)=0$$

$$(N_{C_i}^j+\alpha)/\mu_{ij}-\lambda_i=0$$

$$\mu_{ij}=(N_{C_i}^j+\alpha)/\lambda_i$$

Summing over j gives us:

$$\sum_{j=1}^{M}(N_{C_i}^j+\alpha)/\lambda_i=1$$

$$\lambda_i=\sum_{j=1}^{M}N_{C_i}^j+\alpha M$$

Plugging λ_i back into μ_{ij}:

$$\mu_{ij}=\frac{N_{C_i}^j+\alpha}{\sum_{j=1}^{M}N_{C_i}^j+\alpha M}$$

$$Q.E.D.\blacksquare$$

4.2 SPAM classifier

4.2.1 Code implementation

Submitted to autograder

4.2.2 Top 5 spam tokens

```
 >> Top 5 most indicative tokens are: ['httpaddr' 'spam' 'unsubscrib' 'ebai' 'valet'].
```

4.2.3 Classification accuracy for different classifier

```
    >> (50, 1448)
    >> Accuracy for 50 mails (data/q4_data/MATRIX.TRAIN.50): 96.1250%
    >> (100, 1448)
    >> Accuracy for 100 mails (data/q4_data/MATRIX.TRAIN.100): 97.3750%
    >> (200, 1448)
    >> Accuracy for 200 mails (data/q4_data/MATRIX.TRAIN.200): 97.3750%
    >> (400, 1448)
    >> Accuracy for 400 mails (data/q4_data/MATRIX.TRAIN.400): 98.1250%
    >> (800, 1448)
    >> Accuracy for 800 mails (data/q4_data/MATRIX.TRAIN.800): 98.2500%
    >> (1400, 1448)
    >> Accuracy for 1400 mails (data/q4_data/MATRIX.TRAIN.1400): 98.3750%
```

4.2.4 Size-accuracy plot

See Figure 2

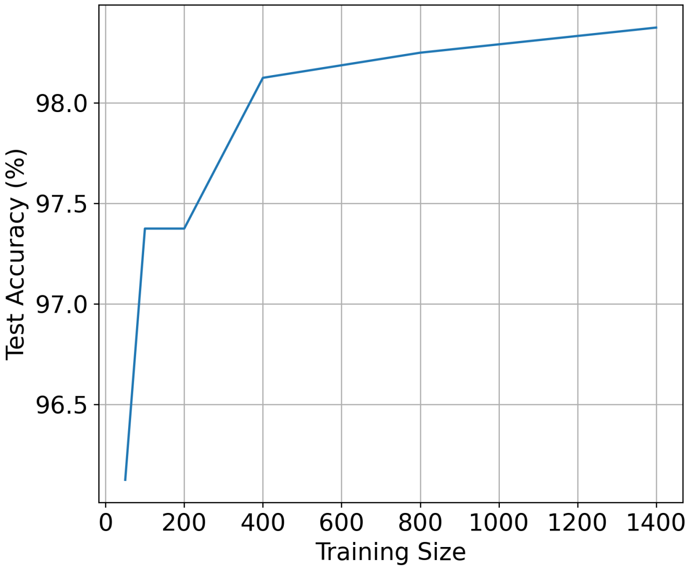

Figure 2: Size-accuracy

4.2.5 Discussion

The size of 1400 gives the best test classification accuracy. The figure shows that bigger training size yields better test accuracy, this is saying that generally we want more data to train a better model. But the slope is becoming less steep when the training size is even larger, this is also indicating that the accuracy will not increase dramatically after the size of training set it too big.