EECS545 Machine Learning
Homework #2 Solutions

2024/2/15

1 [25 points] Logistic Regression

(a) **Answer [5 points]:** Recall that we have $g'(z) = g(z)(1 - g(z))$ where $g(z) = \sigma(z)$, and thus for $h(\mathbf{x}) = g(\mathbf{w}^\top \mathbf{x})$, we have $\frac{\partial h(\mathbf{x})}{\partial \mathbf{w}_k} \partial h(\mathbf{x}) = h(\mathbf{x})(1 - h(\mathbf{x}))x_k$.
Remember we have shown in class:

$$\frac{\partial l(\mathbf{w})}{\partial \mathbf{w}_k} = \sum_{i=1}^{N} (y^{(i)} - h(\mathbf{x}^{(i)}))x_k^{(i)}$$

By taking second derivative, we get

$$\begin{aligned} H_{kl} &= \frac{\partial^2 l(\mathbf{w})}{\partial \mathbf{w}_k \partial \mathbf{w}_l} \\ &= \sum_{i=1}^{N} -\frac{\partial h(\mathbf{x}^{(i)})}{\partial \mathbf{w}_l} x_k^{(i)} \\ &= \sum_{i=1}^{N} -h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)}))x_l^{(i)} x_k^{(i)} \end{aligned}$$

In a matrix form,

$$H = -\sum_{i=1}^{N} h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)}))\mathbf{x}^{(i)}\mathbf{x}^{(i)\top}$$

(b) **Answer [6 points]:** To prove H is negative semidefinite, we show $\mathbf{z}^\top H \mathbf{z} \leq 0$ for all $\mathbf{z}$.

$$\begin{aligned} \mathbf{z}^\top H \mathbf{z} &= \textbf{-}\mathbf{z}^\top \left(\sum_{i=1}^{N} h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)}))\mathbf{x}^{(i)}\mathbf{x}^{(i)\top} \right) \mathbf{z} \\ &= -\sum_{i=1}^{N} h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)}))\mathbf{z}^\top \mathbf{x}^{(i)}\mathbf{x}^{(i)\top}\mathbf{z} \\ &= -\sum_{i=1}^{N} h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)})) \left(\mathbf{z}^\top \mathbf{x}^{(i)}\right)^2 \\ &\leq 0 \end{aligned}$$

with the last inequality holding, since $0 \leq h(\mathbf{x}^{(i)}) \leq 1$, which implies $h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)})) \geq 0$, and $(\mathbf{z}^\top \mathbf{x}^{(i)})^2) \geq 0$.

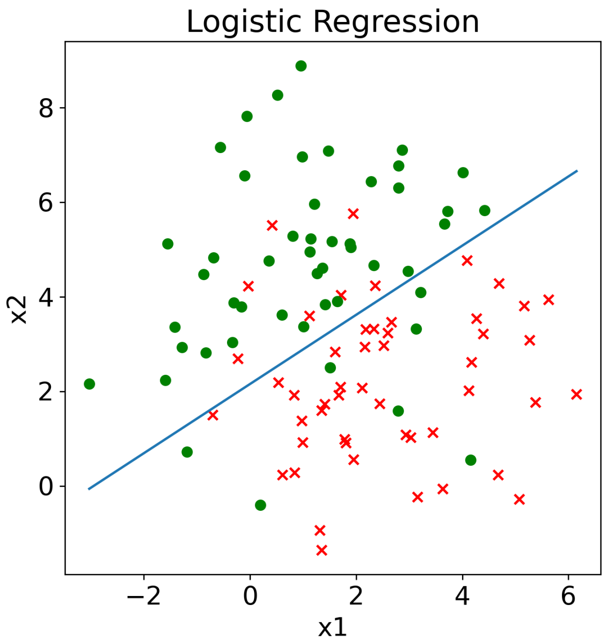
(c) **Answer [2 points]:** (note: the answer is in the lecture note)

$\mathbf{w} = \mathbf{w} - H^{-1} \frac{\partial l(\mathbf{w})}{\partial \mathbf{w}_k}$

(d) **Answer [8 points]:** See attached `logistic_regression.py`.

(e) **Answer [2 points]:** $\mathbf{w} = (-1.8492, -0.6281, 0.8585)$ with the first entry corresponding to the intercept term.

(f) **Answer [2 points]:** As shown in the figure, the data sample $x^{(i)}$ with label $y^{(i)} = 0$ is plotted as red cross, and the data sample $x^{(i)}$ with label $y^{(i)} = 1$ is plotted as green dot.



2 [27 points] Softmax Regression via Gradient Ascent

(a) **Answer [11 points]** Derive the gradient ascent update rule for the log-likelihood of the training data. We have:

$$l(\mathbf{w}) = \sum_{i=1}^{N}\sum_{k=1}^{K} \log \left[p(y^{(i)} = k|\mathbf{x}^{(i)}, \mathbf{w})\right]^{\mathbb{I}(y^{(i)}=k)}$$

Taking gradient with respect to $\mathbf{w}_m$ ($p \leq m \leq K-1$):

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

As the log term on the right does not contain k, it can be taken out of the summation and since $\sum_{k=1}^{K}\mathbb{I}(y^{(i)} = k) = 1$, we obtain the following:

$$\begin{aligned}
&= \sum_{i=1}^{N} \nabla_{\mathbf{w}_m} \Big(\Big[\sum_{k=1}^{K} \mathbb{I}(y^{(i)} = k)\mathbf{w}_k^\top \phi(\mathbf{x}^{(i)})\Big] - \log\Big(1+\sum_{j=1}^{K-1}\exp(\mathbf{w}_j^\top \phi(\mathbf{x}^{(i)}))\Big)\Big) \\
&= \sum_{i=1}^{N} \nabla_{\mathbf{w}_m} \Big[\sum_{k=1}^{K} \mathbb{I}(y^{(i)} = k)\mathbf{w}_k^\top \phi(\mathbf{x}^{(i)})\Big] - \nabla_{\mathbf{w}_m}\log\Big(1+\sum_{j=1}^{K-1}\exp(\mathbf{w}_j^\top \phi(\mathbf{x}^{(i)}))\Big)
\end{aligned}$$

The left term contains $\mathbf{w}_m$ iff $m = y^{(i)}$. Second term contains $\mathbf{w}_m$ (produced by summation)

$$\begin{aligned}
&= \sum_{i=1}^{N} \mathbb{I}(y^{(i)} = m)\phi(\mathbf{x}^{(i)}) - \frac{\exp(\mathbf{w}_m^\top \phi(\mathbf{x}^{(i)}))\phi(\mathbf{x}^{(i)})}{1+\sum_{j=1}^{K-1}\exp(\mathbf{w}_j^\top \phi(\mathbf{x}^{(i)}))} \\
&= \sum_{i=1}^{N} \phi(\mathbf{x}^{(i)})[\mathbb{I}(y^{(i)} = m) - p(y^{(i)} = m|\mathbf{x}^{(i)})]
\end{aligned}$$

(b) **Answer [14 points]** See `gda.py`.

(c) **Answer [2 points]** Instructor solution achieves 92.0% accuracy; students should be able to get an accuracy above 90%. SciKit-Learn gets an accuracy of $2-94\%$, depending on the version.

3 [25 points] Gaussian Discriminate Analysis

(a) **Answer [6 points]**

Note, parameters can be omitted.

$p(y = 1 \mid \mathbf{x}; \phi, \Sigma, \mu_0, \mu_1) = p(y = 1 \mid \mathbf{x})$
$p(\mathbf{x} \mid y = 1; \phi, \Sigma, \mu_0, \mu_1) = p(\mathbf{x} \mid y = 1)$
$p(\mathbf{x} = 1; \phi, \Sigma, \mu_0, \mu_1) = p(\mathbf{x} = 1)$
$p(y = 1; \phi, \Sigma, \mu_0, \mu_1) = p(y = 1)$

Now,

$$
\begin{aligned}
p(y = 1 \mid \mathbf{x}; \phi, \Sigma, \mu_0, \mu_1) &= p(y = 1 \mid \mathbf{x}) \\
&= \frac{p(\mathbf{x} \mid y = 1)p(y = 1)}{p(\mathbf{x})} \\
&= \frac{p(\mathbf{x} \mid y = 1)p(y = 1)}{p(\mathbf{x} \mid y = 1)p(y = 1) + p(\mathbf{x} \mid y = 0)p(y = 0)} \\
&= \frac{\exp\left(-\frac{1}{2}(\mathbf{x} - \mu_1)^\top \Sigma^{-1} (\mathbf{x} - \mu_1)\right)\phi}{\exp\left(-\frac{1}{2}(\mathbf{x} - \mu_1)^\top \Sigma^{-1} (\mathbf{x} - \mu_1)\right)\phi + \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_0)^\top \Sigma^{-1} (\mathbf{x} - \mu_0)\right)(1 - \phi)} \\
&= \frac{1}{1 + \exp\left(\log\left(\frac{(1-\phi)}{\phi}\right) - \frac{1}{2}(\mathbf{x} - \mu_0)^\top \Sigma^{-1} (\mathbf{x} - \mu_0) + \frac{1}{2}(\mathbf{x} - \mu_1)^\top \Sigma^{-1} (\mathbf{x} - \mu_1)\right)} \\
&= \frac{1}{1 + \exp\left(\log\left(\frac{1-\phi}{\phi}\right) + \mathbf{x}^\top \Sigma^{-1} \mu_0 - \mathbf{x}^\top \Sigma^{-1} \mu_1 - \frac{1}{2}\mu_0^\top \Sigma^{-1} \mu_0 + \frac{1}{2}\mu_1^\top \Sigma^{-1} \mu_1\right)} \\
&= \frac{1}{1 + \exp\left(\log\left(\frac{1-\phi}{\phi}\right) + \mathbf{x}^\top \Sigma^{-1} (\mu_0 - \mu_1) - \frac{1}{2}\mu_0^\top \Sigma^{-1} \mu_0 + \mu_1^\top \Sigma^{-1} \mu_1\right)}
\end{aligned}
$$

By setting

$\mathbf{w}_0 = \frac{1}{2}\left(\mu_0^\top \Sigma^{-1} \mu_0 - \mu_1^\top \Sigma^{-1} \mu_1\right) - \log \frac{1-\phi}{\phi}$,
$\mathbf{w} = -\Sigma^{-1}(\mu_0 - \mu_1)$,
A constant intercept term $\mathbf{x}_0 = 1$,

We get: $p(y = 1 \mid \mathbf{x}) = \frac{1}{1+\exp(-(\mathbf{w}^\top \mathbf{x} + \mathbf{w}_0 \mathbf{x}_0))} = \frac{1}{1+\exp(-\mathbf{w}^\top \mathbf{x})}$.

(b) **Answer [14 points]** Let us derive the general case directly:

$$
\begin{aligned}
\ell\left(\phi, \mu_0, \mu_1, \Sigma\right) &= \log \prod_{i=1}^{N} p(\mathbf{x}^{(i)} \mid y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{N} \log p(\mathbf{x}^{(i)} \mid y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) + \sum_{i=1}^{N} \log p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{N} \left[\log \frac{1}{(2\pi)^{\frac{M}{2}} \left|\Sigma\right|^{\frac{1}{2}}} - \frac{1}{2} \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^{\top} \Sigma^{-1} \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right) + \log \phi^{y^{(i)}} + \log\left(1-\phi\right)^{\left(1-y^{(i)}\right)} \right] \\
&\simeq \sum_{i=1}^{N} \left[-\frac{1}{2} \log|\Sigma| - \frac{1}{2} \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^{\top} \Sigma^{-1} \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right) + y^{(i)} \log \phi + \left(1 - y^{(i)}\right) \log\left(1-\phi\right) \right]
\end{aligned}
$$

(the constant term is independent of the parameters, thus removed.)

Then, the likelihood is maximized by setting the derivative with respect to each parameter to zero:

(1) with respect to ϕ:

$$
\begin{aligned}
\frac{\partial \ell}{\partial \phi} &= \sum_{i=1}^{N} \left(\frac{y^{(i)}}{\phi} - \frac{1-y^{(i)}}{1-\phi}\right) \\
&= \sum_{i=1}^{N} \frac{1(y^{(i)}=1)}{\phi} + \frac{N - \sum_{i=1}^{N} \mathbb{I}(y^{(i)}=1)}{1-\phi}
\end{aligned}
$$

Therefore, $\phi = \frac{1}{N}\sum_{i=1}^{N} 1(y^{(i)}=1)$, i.e. the percentage of the training examples such that $y^{(i)} = 1$.

(2) with respect to μ_0:

$$
\begin{aligned}
\nabla_{\mu_0} \ell &= -\frac{1}{2} \sum_{i:y^{(i)}=0} \nabla_{\mu_0} \left(\mathbf{x}^{(i)} - \mu_0\right)^{\top} \Sigma^{-1} \left(\mathbf{x}^{(i)} - \mu_0\right) \\
&= -\frac{1}{2} \sum_{i:y^{(i)}=0} \nabla_{\mu_0} \left[-2\mu_0^{\top} \Sigma^{-1} \mathbf{x}^{(i)} + \mu_0^{\top} \Sigma^{-1} \mu_0\right] \\
&= -\frac{1}{2} \sum_{i:y^{(i)}=0} \left[-2\Sigma^{-1} \mathbf{x}^{(i)} + 2\Sigma^{-1} \mu_0\right]
\end{aligned}
$$

By setting the gradient to zero,

$$
\sum_{i:y^{(i)}=0} \left[\Sigma^{-1} \mathbf{x}^{(i)} - \Sigma^{-1} \mu_0\right] = 0
$$

$$
\sum_{i=1}^{N} \mathbb{I}\left\{y^{(i)} = 0\right\} \Sigma^{-1} \mathbf{x}^{(i)} - \sum_{i=1}^{N} \mathbb{I}\left\{y^{(i)} = 0\right\} \Sigma^{-1} \mu_0 = 0
$$

Thus we obtain $\mu_0 = \frac{\sum_{i=1}^N \mathbb{I}\left\{y^{(i)}=0\right\}\mathbf{x}^{(i)}}{\sum_{i=1}^N \mathbb{I}\left\{y^{(i)}=0\right\}}$

(3) with respect to μ_1:

The calculations are similar for μ_1. The resulting maximum likelihood estimate is: $\mu_1 = \frac{\sum_{i=1}^N \mathbb{I}\left\{y^{(i)}=1\right\}\mathbf{x}^{(i)}}{\sum_{i=1}^N \mathbb{I}\left\{y^{(i)}=1\right\}}$

(c) **Answer [5 points]** With respect to Σ:

The last step is to calculate the gradient with respect to Σ. Here, we assume $M = 1$, i.e., $|\Sigma| = \sigma^2$. The log-likelihood of the data then can be written:

$$\begin{aligned}
\ell(\phi, \mu_0, \mu_1, \Sigma) &\simeq \sum_{i=1}^N \left[-\frac{1}{2}\log|\Sigma| - \frac{1}{2}\left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^\top \Sigma^{-1}\left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right) + y^{(i)}\log\phi + \left(1 - y^{(i)}\right)\log(1-\phi)\right] \\
&= \sum_{i=1}^N \left[-\log\sigma - \frac{1}{2\sigma^2}\left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^\top \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right) + y^{(i)}\log\phi + \left(1 - y^{(i)}\right)\log(1-\phi)\right]
\end{aligned}$$

By taking derivative with respect to σ and set it to zero:

$$\nabla_\sigma \ell = \sum_{i=1}^N \left[-\frac{1}{\sigma} + \frac{1}{\sigma^3}\left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^\top \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)\right] = 0$$

You obtain: $\Sigma = \frac{1}{N}\sum_{i=1}^N \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)^\top \left(\mathbf{x}^{(i)} - \mu_{y^{(i)}}\right)$

Elaborated above.

4 [23 points] Naive Bayes for Classifying SPAM

(a) **Answer [6 points]** Naive Bayes with Bayesian Smoothing. Recall that the likelihood from lecture is:

$$\begin{aligned}
P(\mathbf{X}, \mathbf{y} \mid \boldsymbol{\mu}, \phi) = \prod_i P(\mathbf{x}^{(i)}, y^{(i)} \mid \boldsymbol{\mu}, \phi) &= \prod_{s=1}^K \left[\left(\prod_{i:y^{(i)}=s}^N \prod_{k=1}^{\mathrm{len}\left(x^{(i)}\right)} \prod_{j=1}^M \left(\mu_j^s\right)^{\mathbb{I}\left(x_k^{(i)} = \text{“}j\text{”th word}\right)}\right)\left(\prod_{i:y^{(i)}=1}^N \phi_s\right)\right] \\
&= \prod_{s=1}^K \left[\left(\prod_{j=1}^M \left(\mu_j^s\right)^{N_j^s}\right)\phi_s^{N^s}\right]
\end{aligned}$$

Then, the MAP objective is:

$$\begin{aligned}
P(\mathbf{X}, \mathbf{y} \mid \boldsymbol{\mu}, \phi)P(\boldsymbol{\mu}) &= \prod_{s=1}^K \left[\left(\prod_{j=1}^M \left(\mu_j^s\right)^{N_j^s}\right)\phi_s^{N^s}\right]\left(\frac{1}{Z}\prod_{s=1}^K\prod_{j=1}^M \left(\mu_j^s\right)^\alpha\right) \\
&= \frac{1}{Z}\prod_{s=1}^K \left[\left(\prod_{j=1}^M \left(\mu_j^s\right)^{N_j^s}\right)\phi_s^{N^s}\left(\prod_{j=1}^M \left(\mu_j^s\right)^\alpha\right)\right] \\
&= \frac{1}{Z}\prod_{s=1}^K \left[\left(\prod_{j=1}^M \left(\mu_j^s\right)^{N_j^s+\alpha}\right)\phi_s^{N^s}\right]
\end{aligned}$$

Note this MAP objective is the same as the likelihood, but we added α to each N_j^s count! Using the MLE derivation from lecture (which can still be applied for $K > 2$. See $\diamondsuit$), we maximize the likelihood w.r.t. μ's and we get the final objective:

$$P(x_j \mid C_i) = \mu_j^i = \frac{N_j^{C_i} + \alpha}{\sum_{j'} N_{j'}^{C_i} + \alpha M}.$$

$\diamondsuit$ Why does the derivation from lecture still hold for $K > 2$? We use the fact that μ_j^a and μ_j^b are *independent* if $a \neq b$ (they come from two independent multinomial distributions from the Naive Bayes assumption). This is also (implicitly) used in the lecture derivation. Notice then that this implies:

$$\frac{\partial}{\partial \mu_j^a} \log \mu_j^b = \begin{cases} \frac{1}{\mu_j^a} & a = b \\ 0 & a \neq b \end{cases}$$

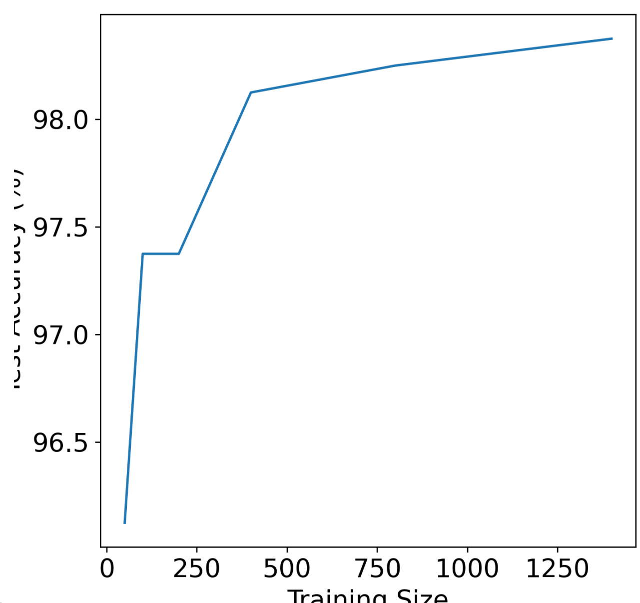
After removing all independent variables from the gradient calculations results in an equation similar to the equation derived in the lecture (Lecture 6 slide 26-28). Note that this does **not** apply for μ_a^i and μ_b^i for $a \neq b$. These are *dependent*, as they are parameters of the same multinomial distribution, from the constraint $\sum_j^M \mu_j^i = 1$. These can substituted using the trick shown in lecture (Lecture 6 slide 27).

(b) i. **Answer [9 points]** See `naive_bayes_spam.py` for instructor solution.

ii. **Answer [2 points]** Top 5 spam tokens are `['httpaddr' 'spam' 'unsubscrib' 'ebai' 'valet']`.

iii. **Answer [2 points]** These should be the accuracy breakdowns.

- Training set size 50: Test set accuracy = 96.1250%
- Training set size 100: Test set accuracy = 97.3750%
- Training set size 200: Test set accuracy = 97.3750%
- Training set size 400: Test set accuracy = 98.1250%
- Training set size 800: Test set accuracy = 98.2500%
- Training set size 1400: Test set accuracy = 98.3750%



iv. **Answer [2 points]**

v. **Answer [2 points]** The training set of size 1400 gives the best accuracy. A model trained with a smaller size of data could be more subject to noise.