CSE6250: Homework 2 Answer

Name: Yichao Zhang
GT User ID: yzhang3414

Deadline: Feb 3, 2019, 11:55 PM AoE

1 Logistic Regression [25 points]

1.1 Batch Gradient Descent

The negative log-likelihood can be calculated according to

$$NLL\left(D, \mathbf{w}\right) = -\sum_{i=1}^{N} \left[(1-y_i)\log(1-\sigma(\mathbf{w}^T\mathbf{x}_i)) + y_i \log \sigma(\mathbf{w}^T\mathbf{x}_i)\right] \tag{1}$$

where $\sigma(t) = \frac{1}{1+e^{-t}}$ is the sigmoid function.

a. Derive the gradient of the negative log-likelihood in terms of w for this setting. [5 points]

$\because$

$$\frac{\partial}{\partial t}\sigma(t) = \frac{\partial}{\partial t}\frac{1}{1+e^{-t}} = \frac{-1}{(1+e^{-t})^2}\frac{\partial}{\partial t}e^{-t} = \frac{e^{-t}}{(1+e^{-t})^2} = \sigma(1-\sigma) \tag{2}$$

$\therefore$

$$\nabla_{\mathbf{w}}\sigma(\mathbf{w}^T\mathbf{x}_i) = \sigma(1-\sigma)\nabla_{\mathbf{w}}(\mathbf{w}^T\mathbf{x}_i) = \sigma(1-\sigma)\mathbf{x}_i \tag{3}$$

$\therefore$

$$\begin{aligned}
\nabla_{\mathbf{w}} NLL\left(D, \mathbf{w}\right) &= -\sum_{i=1}^{N}\left[\left(1-y_i\right)\nabla_{\mathbf{w}}\log(1-\sigma(\mathbf{w}^T\mathbf{x}_i)) + y_i\nabla_{\mathbf{w}}\log\sigma(\mathbf{w}^T\mathbf{x}_i)\right] \\
&= -\sum_{i=1}^{N}\left[-\left(1-y_i\right)\frac{1}{1-\sigma} + y_i\frac{1}{\sigma}\right]\nabla_{\mathbf{w}}\sigma(\mathbf{w}^T\mathbf{x}_i) \\
&= -\sum_{i=1}^{N}\left[-\left(1-y_i\right)\frac{1}{1-\sigma} + y_i\frac{1}{\sigma}\right]\sigma(1-\sigma)\mathbf{x}_i \\
&= -\sum_{i=1}^{N}\left[-\left(1-y_i\right)\sigma + y_i(1-\sigma)\right]\mathbf{x}_i \\
&= -\sum_{i=1}^{N}\left[y_i - \sigma(\mathbf{w}^T\mathbf{x}_i)\right]\mathbf{x}_i
\end{aligned} \tag{4}$$

1.2 Stochastic Gradient Descent

If N and d are very large, it may be prohibitively expensive to consider every patient in D before applying an update to $\mathbf{w}$. One alternative is to consider stochastic gradient descent, in which an update is applied after only considering a single patient.

a. Show the log likelihood, l, of a single $(\mathbf{x}_t, y_t)$ pair. [5 points]

From eq(1), by simply removing the summation, we get the log likelihood for a single patient :

$$nll\left(D_t, \mathbf{w}\right) = -\left[\left(1-y_t\right)\log(1-\sigma(\mathbf{w}^T\mathbf{x}_t)) + y_t\log\sigma(\mathbf{w}^T\mathbf{x}_t)\right] \tag{5}$$

b. Show how to update the coefficient vector $\mathbf{w}_t$ when you get a patient feature vector $\mathbf{x}_t$ and physician feedback label y_t at time t using $\mathbf{w}_{t-1}$ (assume learning rate η is given). [5 points]

From eq(4), by simply removing the summation, we get the gradient of a single patient:

$$\nabla_{\mathbf{w}} nll\left(D_t, \mathbf{w}\right) = -\left[y_t - \sigma(\mathbf{w}^T\mathbf{x}_t)\right]\mathbf{x}_t \tag{6}$$

$\therefore$ to minimize the negative log likelihood of a single patient, we update the $\mathbf{w}$ by

$$\begin{aligned}
\mathbf{w}_t &= \mathbf{w}_{t-1} - \eta\nabla_{\mathbf{w}} nll\left(D_t, \mathbf{w}\right)\Big|_{\mathbf{w}=\mathbf{w}_{t-1}} \\
&= \mathbf{w}_{t-1} + \eta\left[y_t - \sigma(\mathbf{w}_{t-1}^T\mathbf{x}_t)\right]\mathbf{x}_t
\end{aligned} \tag{7}$$

c. What is the time complexity of the update rule from b if $\mathbf{x}_t$ is very sparse?

[2 points]

- Update for a single patient in 1 iteration:

  We need to update each component of $\mathbf{w}_t$ where the respective component of $\mathbf{x}_t$ is non-zero. Suppose $\mathbf{x}_t$ has m non-zero components in average, then the time complexity is $O(m)$.

  If $\mathbf{x}_t$ is very sparse, then $m \to 1$, it becomes $O(1)$.

- Update for N patient in 1 iteration:

  The time complexity is $O(mN)$.

  Similarly, if $\mathbf{x}_t$ is very sparse, the time complexity becomes $O(N)$

- Update for I iterations:

  The time complexity is $O(mNI)$.

  Similarly, if $\mathbf{x}_t$ is very sparse, the time complexity becomes $O(NI)$

d. Briefly explain the consequence of using a very large η and very small η. [3 points]

A very large learning rate η means we update $\mathbf{w}$ by very large steps, so SGD may not converge to the minimum. And SGD may finish searching in a few iterations, or never stop.

A very small learning rate η means we update $\mathbf{w}$ by very small steps, so it may take much more iterations for SGD to converge to the minimum.

e. Show how to update $\mathbf{w}_t$ under the penalty of L2 norm regularization. In other words, update $\mathbf{w}_t$ according to $l - \mu\|\mathbf{w}\|_2^2$, where μ is a constant. What's the time complexity? [5 points]

When we add a penalty term: $-\mu\|\mathbf{w}\|_2^2$ to the log likelihood for single patient, the negative log likelihood for single patient becomes: $nll\left(D_t, \mathbf{w}\right) + \mu\|\mathbf{w}\|_2^2$.

$\because$

$$\nabla_{\mathbf{w}}\|\mathbf{w}\|_2^2 = \nabla_{\mathbf{w}}(\mathbf{w}^T\mathbf{w}) = 2\mathbf{w} \tag{8}$$

$\therefore$ we update $\mathbf{w}$ by:

$$\begin{aligned}
\mathbf{w}_t &= \mathbf{w}_{t-1} - \eta\nabla_{\mathbf{w}}\left[nll\left(D_t, \mathbf{w}\right) + \mu\|\mathbf{w}\|_2^2\right]\Big|_{\mathbf{w}=\mathbf{w}_{t-1}} \\
&= \mathbf{w}_{t-1} - \eta\nabla_{\mathbf{w}} nll\left(D_t, \mathbf{w}\right) - \eta\mu\nabla_{\mathbf{w}}\|\mathbf{w}\|_2^2\Big|_{\mathbf{w}=\mathbf{w}_{t-1}} \\
&= \mathbf{w}_{t-1} + \eta\left[y_t - \sigma(\mathbf{w}_{t-1}^T\mathbf{x}_t)\right]\mathbf{x}_t - 2\eta\mu\mathbf{w}_{t-1}
\end{aligned} \tag{9}$$

Time complexity:

- For a single patient in 1 iteration: we need to update each component of $\mathbf{w}_t$, no matter $\mathbf{x}_t$ is sparse or not. Here $\mathbf{w} \in \mathbf{R}^d$, so the time complexity is $O(d)$.
- For N patient in 1 iteration: the time complexity is $O(dN)$.
- For I iterations: the time complexity is $O(dNI)$.

2 Programming [75 points]

2.1 Descriptive Statistics [15 points]

Metric	Deceased patients	Alive patients
Event Count		
1. Average Event Count	1027.739	683.155
2. Max Event Count	16829	12627
3. Min Event Count	2	1
Encounter Count		
1. Average Encounter Count	24.839	18.695
2. Max Encounter Count	375	391
3. Min Encounter Count	1	1
Record Length		
1. Average Record Length	157.042	194.703
2. Median Record Length	25	16
3. Max Record Length	5364	3103
4. Min Record Length	0	0
Common Diagnosis	DIAG320128	DIAG320128
	DIAG319835	DIAG319835
	DIAG313217	DIAG317576
	DIAG197320	DIAG42872402
	DIAG132797	DIAG313217
Common Laboratory Test	LAB3009542	LAB3009542
	LAB3023103	LAB3000963
	LAB3000963	LAB3023103
	LAB3018572	LAB3018572
	LAB3016723	LAB3007461
Common Medication	DRUG19095164	DRUG19095164
	DRUG43012825	DRUG43012825
	DRUG19049105	DRUG19049105
	DRUG956874	DRUG19122121
	DRUG19122121	DRUG956874

Table 1: Descriptive statistics for alive and dead patients

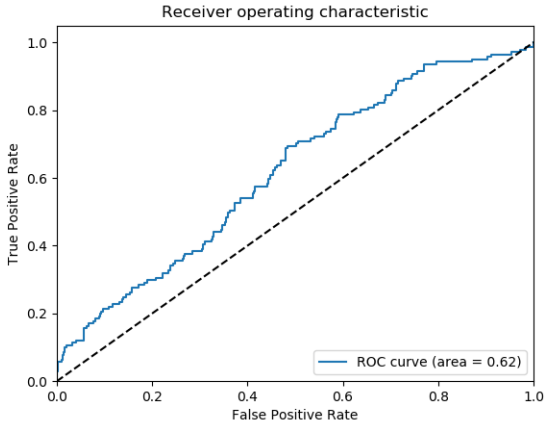
2.2 Transform data [20 points]

2.3 SGD Logistic Regression [20 points]

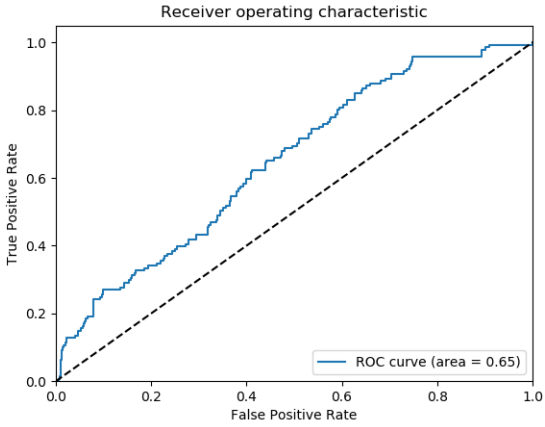
Train and test a classifier by running

1. cat ../pig/training/part-r-00000 | python train.py -f 3618
2. cat ../pig/testing/part-r-00000 | python test.py

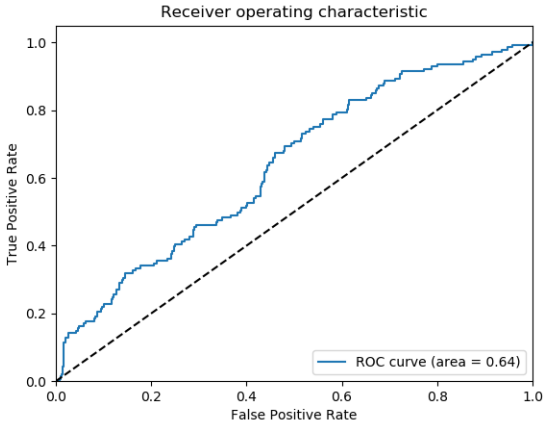
b. Show the ROC curve generated by test.py in this writing report for different learning rates η and regularization parameters μ combination and briefly explain the result. [5 points]



(a) $\eta = 0.01, \mu = 0, ROC = 0.62$



(b) $\eta = 0.1, \mu = 0, ROC = 0.65$

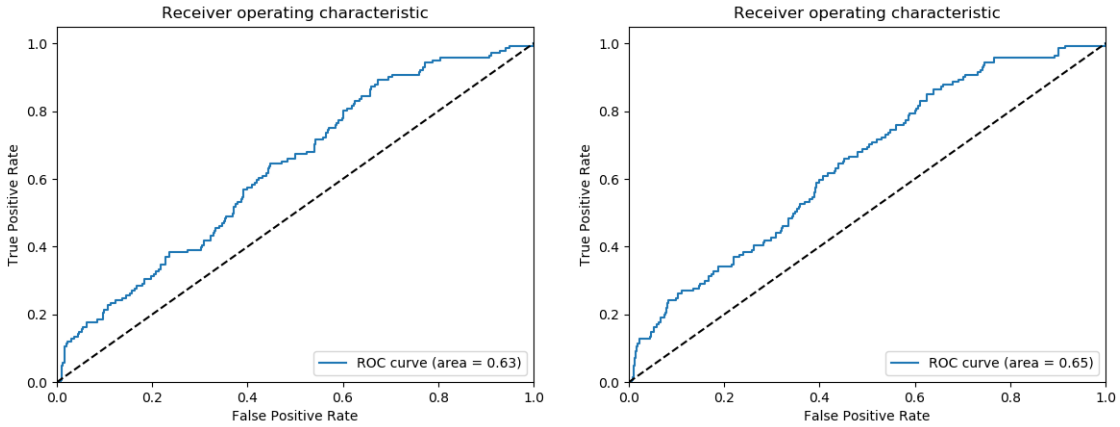


(c) $\eta = 0.2, \mu = 0, ROC = 0.64$

Figure 1: Tune η, with fixed $\mu = 0$

Firstly, I fixed $\mu = 0$, and tune the learning rate η from 0.01 to 0.2. I find that the highest ROC score (0.65) is on $\eta = 0.1$. If the learning rate is higher than 0.1, it may not

converge to the minimum. And if it is much lower than 0.1, then it cannot finish searching minimum in short times.



(a) $\eta = 0.1, \mu = 0.01, ROC = 0.63$ (b) $\eta = 0.1, \mu = 0.001, ROC = 0.65$

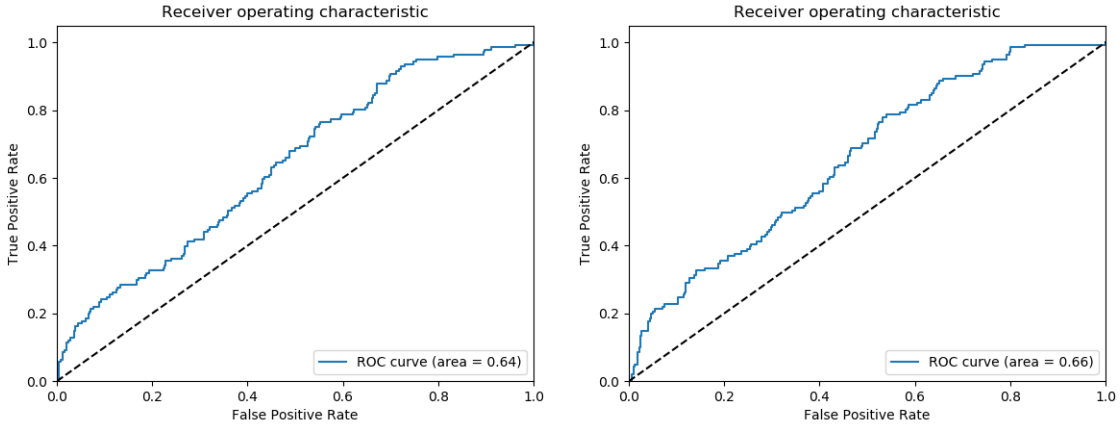
Figure 2: Tune μ, with fixed $\eta = 0.1$

Secondly, I fixed $\eta = 0.01$, and tried some different L2 penalty constant μ. When μ is a little large, like 0.1, the ROC performance decrease due to the trade-off between optimizing the parameters and simplifying the parameters.

When μ is very small, like 0.001, the performance is as good as $\mu = 0$. However, the penalty term makes the parameters smaller and simpler. Considering the Occam's Razor principle, this model is likely to have a better generalizability.

2.4 Hadoop [15 points]

c. Compare the performance with that of previous problem and briefly analyze why the difference. [5 points]



(a) $\eta = 0.1, \mu = 0.001, ROC = 0.64$ (b) $\eta = 0.4, \mu = 0.001, ROC = 0.66$

Figure 3: ROC performance of ensemble models, model number $n = 5$, sample ratio $r = 0.4$

By using the best parameter in the single model: $\eta = 0.1, \mu = 0.001$, the ROC score of ensemble model is 0.64, lower than the single model, which has 0.65.

However, when I increased the learning rate η to 0.4, the ROC of ensemble model raised to 0.66, better than single model. That means the single model with small value of $\eta = 0.1$ may overfits the training data. In addition, the larger value of learning rate increase the speed of ensemble model.