CIS 419/519: Homework 3

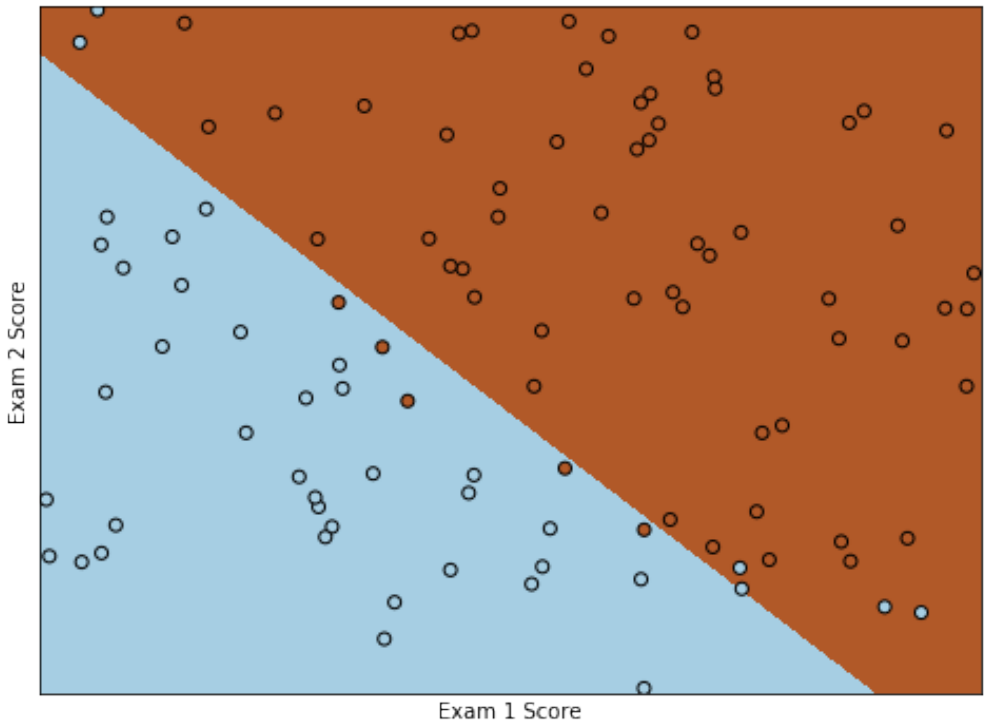
{Zhuheng Jiang}

Although the solutions are entirely my own, I consulted with the following people and sources while working on this homework: {Maian Zhang, Yihang Xu, Chang Liu}

I request to use 1 of the late days without penalty, please. Thank you.

1 1.2 Test Your Implementation

a. The logistic regression implementation's test result is shown in the following picture which is similar to the document's:



So my implementation of the Logistic Regression has been verified.

2 1.3 Analysis

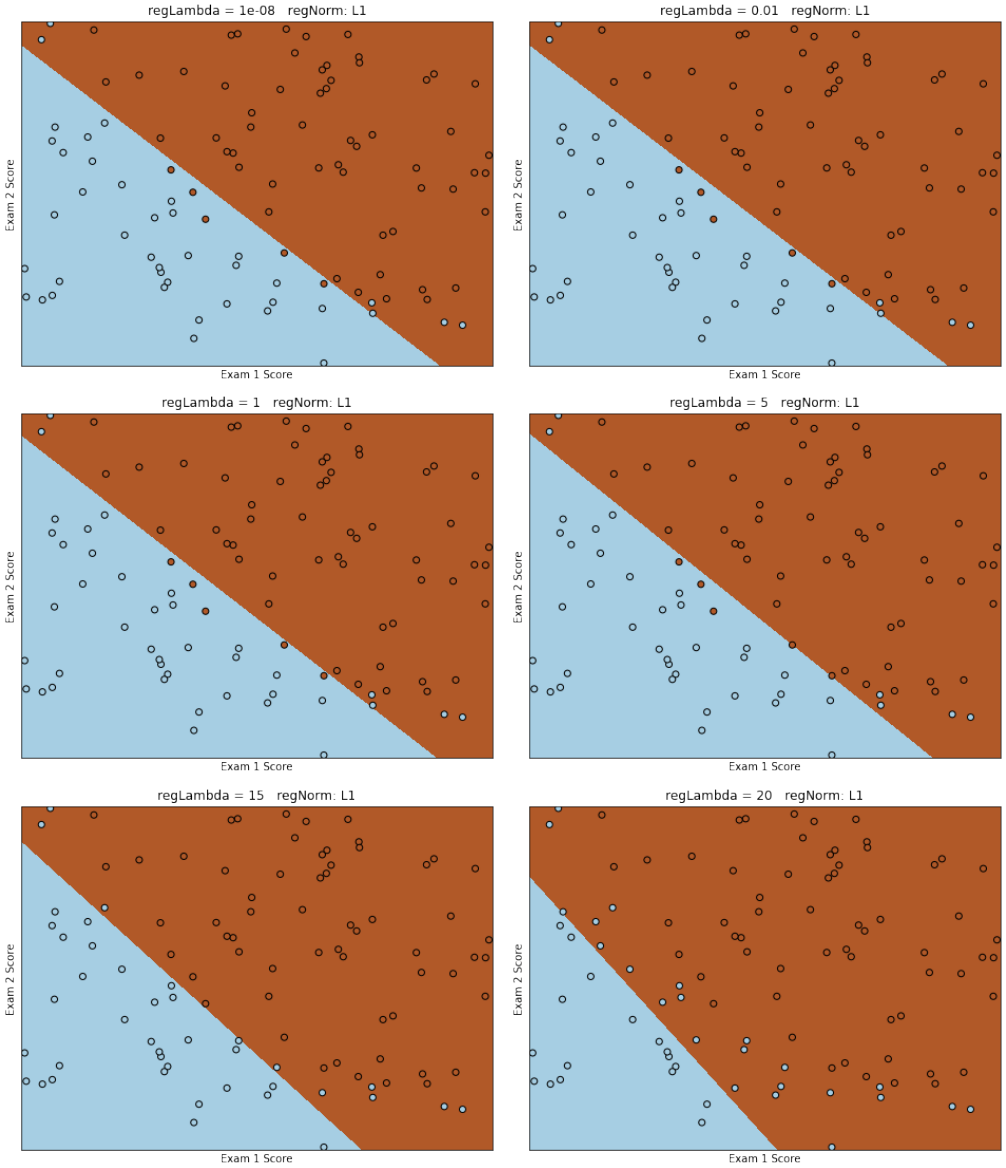
a. I attempt seven different values within a large range for regLambda and compute the corresponding converge iteration and final cost for estimation under both regNorm L1 and L2. The numerical results are shown in the following table:

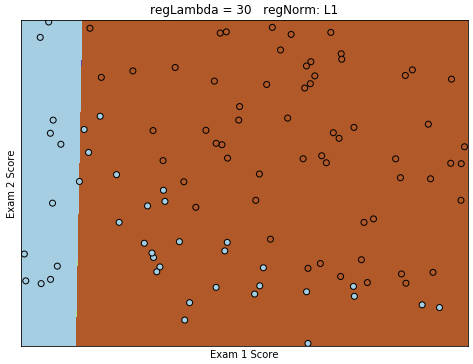
:

Under L1 Norm

regLambda	Converge Iteration	Cost
1E-8	804	0.3498
1E-2	795	20.4268
1	406	26.8861
5	116	42.7612
15	39	61.1883
20	22	65.3180
30	Not converged	/

And the corresponding plots under L1 norm are attached



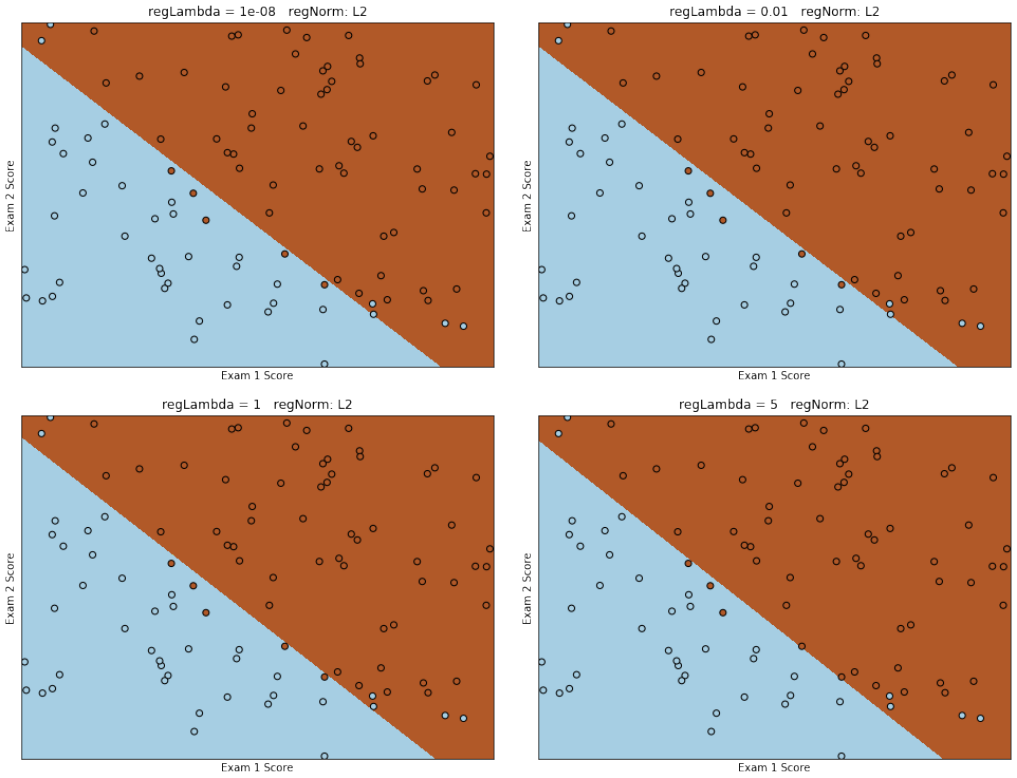


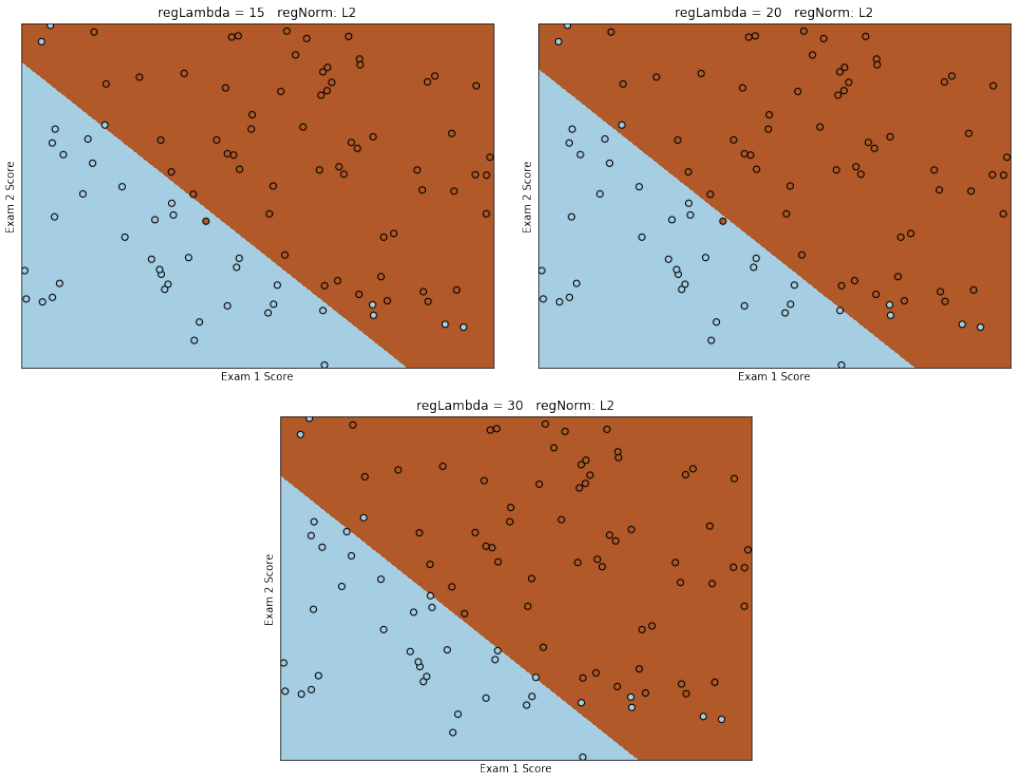
Then we test the qualities of the model under L2 norm. The table and plots are shown below:

Under L2 Norm

regLambda	Converge Iteration	Cost
1E-8	804	20.3499
1E-2	770	20.6405
1	189	32.7687
5	63	46.4143
15	35	56.6918
20	34	59.0129
30	28	61.7876

And the corresponding plots under L2 norm are attached:

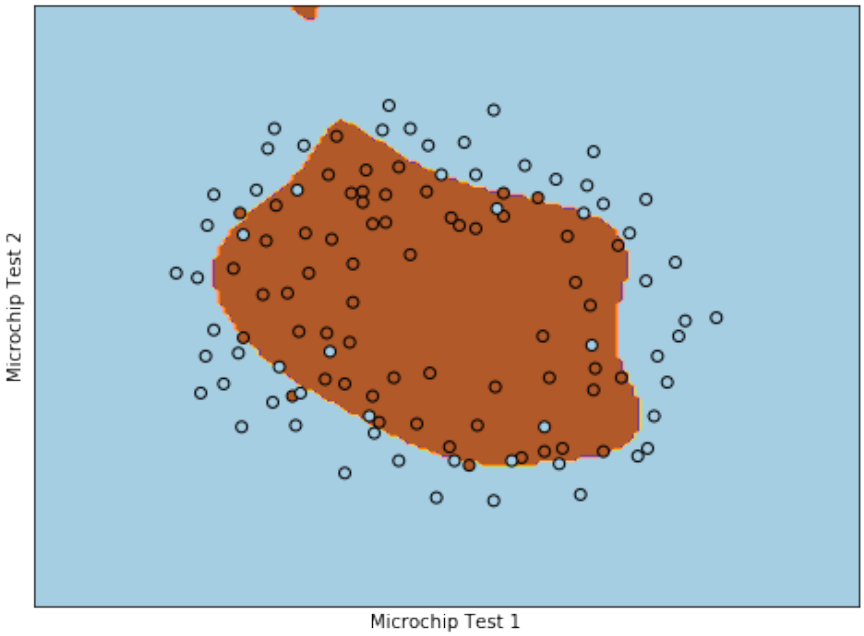




From the plots we can derive that:
When applying the L1 norm and increase the regLambda, the model's boundary is gradually rotate. When the regLambda is extremely large, the penalty is intense which will cause the boundary to be almost vertical or horizontal. That is because according to the trajectory of the norm, the L1 norm is gradually dragging one of the theta to zero and only the other one exerts action on the model. Only one of the intercepts is decreasing.
And when applying the L2 norm, this trend changes that the boundary is gradually translating to one corner. That is because two theta is changing in the same time. If the line's both intercepts changes, the line will translate.
Then, from the comparison table we can conclude that:
1. Increasing the regLambda can remarkably reduce the iterations to convergence, but correspondingly the cost will increase significantly.
2. The L2 norm turns out to be less strict than L1 norm when the regLambda is relatively large due to less iterations and higher cost. But when we are employing typical and small regLambda, these two criterions' performances are close.

3 1.4 Analysis

a. The plot I get is shown in the following picture:



Non linear logistic regression(L2 Norm - $regLambda = 1E - 8$)

4 2.2 Comparing Algorithm

a. Description of datasets:
1. Breast Cancer Wisconsin (Diagnostic):
Instances: 569
Features: 10 (computed for each cell nucleus):
a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter$^2/area$ - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

The ID has been removed from the dataset so the dataset should be much purer for training a more precise model. And the rest features are all about the cancer's feature which should be much more convincing. But I notice there exists many 0s under some features which are not supposed to be none. So I think we might have to preprocess these values.
Then, this dataset contains both "perimeter", "area" and "compactness". These three features have functional relations between each other. Thus, this dataset contains features which are not independent, and this may negatively affect the quality.
2. Retinopathy Diagnosis:
Instances: 1151
Features: 5 0) The binary result of quality assessment. 0 = bad quality 1 = sufficient quality.
1) The binary result of pre-screening, where 1 indicates severe retinal abnormality and 0 its lack.
2-7) The results of MA detection. Each feature value stand for the number of MAs found at the confidence levels alpha = 0.5, . . . , 1, respectively. 8-15) contain the same information as 2-7) for exudates.

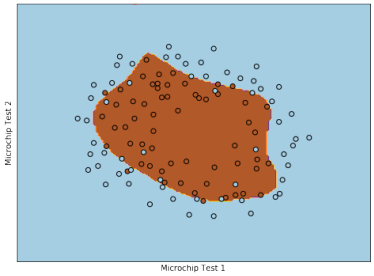
However, as exudates are represented by a set of points rather than the number of pixels constructing the lesions, these features are normalized by dividing the number of lesions with the diameter of the ROI to compensate different image sizes.
16) The euclidean distance of the center of the macula and the center of the optic disc to provide important information regarding the patientâ^{TM}s condition. This feature is also normalized with the diameter of the ROI.
17) The diameter of the optic disc.
18) The binary result of the AM/FM-based classification.
19) Class label. 1 = contains signs of DR (Accumulative label for the Messidor classes 1, 2, 3), 0 = no signs of DR.

This dataset's quality was assessed, there exists a few bad quality data instances which should be remove. And the second feature is the binary result of pre-screening, where 1 indicates severe retinal abnormality and 0 its lack. Some instances lacks the pre-screening which might influce the quality of the model. But there is no omissions in the rest features such as the distances and diameters which is perfect for training the model.

3. Diabetes:
Instances: 768
Features: 9
1) Pregnancies Number of times pregnant
2) Glucose Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3) BloodPressure Diastolic blood pressure (mm Hg)
4) SkinThickness Triceps skin fold thickness (mm)
5) Insulin 2-Hour serum insulin (mu U/ml)
6) BMI Body mass index (weight in kg/(height in m)^2)
7) DiabetesPedigreeFunction Diabetes pedigree function
8) Age (years)
9) Outcome Class variable (0 or 1) 268 of 768 are 1, the others are 0

From the data quality, this dataset is kind of not comprehensive for studying the diabetes for all races. It is only focusing on the specific Pima Indians women.
Besides, this data is relatively less complete since there are many omissions displayed as 0 in the features of blood pressure and skin thickness. This can be found since these two features' histogram do not conform the Gaussian's distribution and contains distinct outliers. Thus, this dataset may be collected under relatively severe condition which means its confidence may not be strong.

Then, firstly we check the implementation of the Adagrad model by ploting fitting the same dataset in testlogreg2 function. The graph is shown below:



Adagrad logistic regression implementation

From this plot we can see the Adagrad can get even better performance than merely mapping the features and applying batch gradient descent. What is more, the computational cost has been remarkably reduced

in this way.
Then we start to analyze and compare the model. The first step is to use cross validation to tune the regLambda for each model and L1 and L2 criterion. The significance of doing this procedure is to improve every model into its best condition in case we are comparing two models under different performances. I employed the regLambdaValues used in previous homework as the regLambda set.

$$regLambdaValues = [1E-8, 0.001, 0.003, 0.006, 0.01, 0.03, 0.06, 0.1, 0.3, 0.6, 1, 3, 10]$$

Then for each data set, the optimal regLambda is tested and the results are shown in the following table. The accuracy for all the value can be accessed by implementing the program. Changing the function of "tuneRegLambda()" to use different model objects('logRegression' or 'logRegressionAdagrad') and change the input of 'regNorm' to switch between L1 and L2 norm.
1. For Breast Cancer Wisconsin (Diagnostic):

"WDBC" Optimal regLambda and accuracy:

Models	L1 Norm	L2 Norm
Log Reg	$\lambda = 0.3$ $Accuracy = 0.9666$	$\lambda = 0.1$ $Accuracy = 0.9672$
Adagrad	$\lambda = 1E-8$ $Accuracy = 0.9766$	$\lambda = 0.01$ $Accuracy = 0.9783$

2. For Retinopathy Diagnosis:

"Retino" Optimal regLambda and accuracy:

Models	L1 Norm	L2 Norm
Log Reg	$\lambda = 0.3$ $Accuracy = 0.6913$	$\lambda = 1E-8$ $Accuracy = 0.6823$
Adagrad	$\lambda = 1E-8$ $Accuracy = 0.7017$	$\lambda = 1E-8$ $Accuracy = 0.7008$

3. For Diabetes:

"Diabetes" Optimal regLambda and accuracy:

Models	L1 Norm	L2 Norm
Log Reg	$\lambda = 1$ $Accuracy = 0.7157$	$\lambda = 0.03$ $Accuracy = 0.7166$
Adagrad	$\lambda = 0.001$ $Accuracy = 0.7773$	$\lambda = 1E-8$ $Accuracy = 0.7764$

Accuracy Comparison:
From the about tables, we can derive some conclusion around the prediction accuracy on several aspects:

1. For the data sets, the accuracy of the model trained and tested by Breast Cancer data set is conspicuously higher than the other two. This phenomenon can prove that the quality of this data set is relatively high so the model is easier to use the features to do the model training.

2. For the L1 and L2 norm 's prediction performances, the table shows the models' best accuracy under corresponding best regLambda. After comparing the accuracy of L1 and L2 norm under the same training model we can find these two methods do not have so much discrepancy on the prediction accuracy. Generally, the accuracy is almost the same when using L1 and L2 norm. And in some data sets the L1's accuracy is higher while in the others the L2 is slightly higher.

3. For the Logistic Regression and Adagrad, the Accuracy of this two model is different. Though the Logistic Regression uses Batch Gradient Descent which follows the global optimal path, its best accuracy is not as high as the Adagrad. The accuracy of the Adagrad is more accurate more or less. In the Diabetes, the Adagrad's accuracy is almost 6% higher than the logistic regression.
This is because in the Adagrad method the learning rate is changing separately for each features. This will cause the more frequent features' learning rate to accumulate faster and becoming smaller so that it will

decay slower while the less frequent features will decay faster. It can be interpreted as: if we have seen a feature for multiple times, then we are supposed to know more about this specific feature. Then in the future, we should reduce the intensity of learning this feature.

Model Speed Comparison:

Then we only focus on the Breast Cancer data set due to its high accuracy to study the iteration or the models' speed. I used the best regLambda tuned in the previous section, and record the iterations before convergence. The results are shown in the following table. The reason for using iteration instead of inner time

"Diabetes" Iterations:

Models	L1 Norm	L2 Norm
Log Reg	$Iterations$: 9001	$Iterations = 2992$
Adagrad	$Iterations = 5931$	$Iterations = 1799$

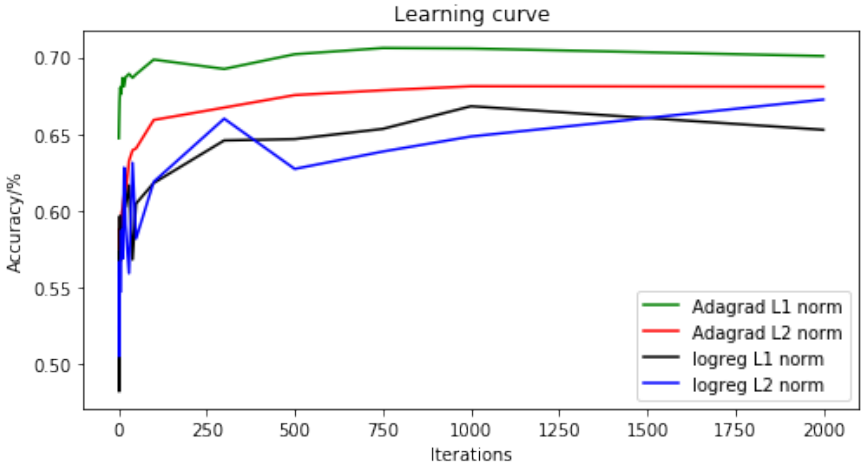
From the results we can conclude that:

1) For the Logistic regression and Adagrad, the Adagrad requires significantly less iterations to fit the model than the Logistic regression. Technically, this is the advantage of using Adagrad instead of logistic regression. It comes from that the adaptive learning rate can speed up the fitting speed and control the speed by accumulating the learning rate. And this method also acts on each element of the theta which makes the theta updates more quickly.

2) For the L1 and L2 norm, the iterations turn out to be much fewer when using L2 norm. The required iterations under L1 norm is over 3 times of the L2's. This phenomenon is due to the larger penalty L2 norm acting on the gradient. In computing gradient, L2 norm penalizes with the theta while L1 norm penalizes less.

5 2.3 Learning Curves

a. For this part I still employ the "Retinopathy" data set for estimation since this data set contains less data which is relatively convenient for computation, and its average accuracy is low so we can easily find the variation as iteration increases. The maxIterNum will vary within a large discrete range of [1, 2, 3, 4, 5, 7, 9, 12, 16, 20,30,40,50,100,300, 500, 750, 1000, 2000]. And the convergence check is removed to go through all the iterations. The plot I get is shown in the following picture. Alpha = 0.01, epsilon = 0.001 and regLambda = 1E-4. And in cross validation I picked 3 trials and 4 folds.



Learning Curve

From the learning curve we can find that:

1. Changing the learning rate can effectively increase the model's accuracy. From both the curves' comparison and the data during the computation, we can find the logistic regression model's accuracy is lower

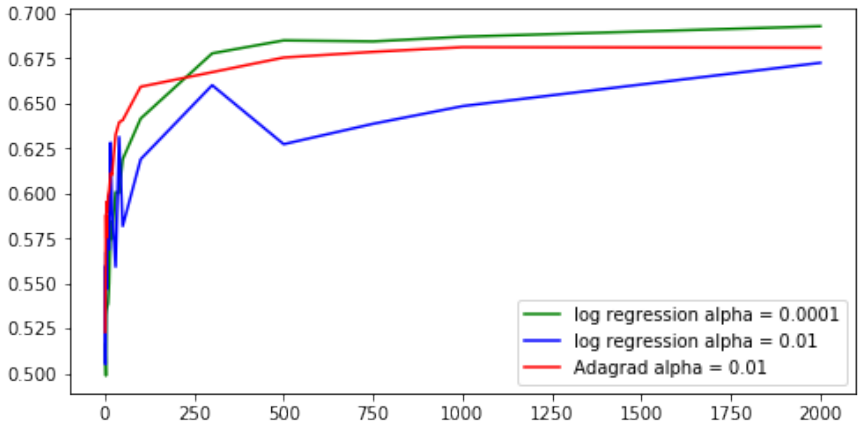
than adagrad even the max iteration is beyond the convergence point. This phenomenon also verifies the conclusions in the last section.

2. L1 generally performs better than L2 due to the higher accuracy. But in all the curves will have subtle oscillations when the iteration increases.

3. The accuracy increases quickly at the beginning, then it converge to a stable accuracy level.

4. This result also verifies the conclusion that the changing learning rate (Adagrad) can improve the model's performance compared with the fixed learning rate(logistic regression).

Then we can dive deeper into the variable learning rate. According to the suggestions in the write-up, the logistic regression model can be adjusted to have almost the same accuracy with the Adagrad model. Since the Adagrad model employs the adaptive learning rate which will be gradually diminished as the iteration or time goes, the learning rate of Adagrad ought to be pretty small when approaching the convergence point. Thus, we can set the logistic regression's fixed learning rate to be relatively tiny, sacrificing the computational cost and times, to get a better accuracy. In the previous section, the alphas were all 0.01. Now I set the logistic regression's learning rate to be 0.0001 and plot the new curve with Adagrad under 0.01 alpha and logistic regression under 0.01 alpha. The curve is attached below:



From this graph we can find:

1. The Adagrad converges to the stable condition faster than the logistic regression at the beginning segment.

2. After diminishing the fixed learning rate for logistic regression, its learning curve becomes much closer to the Adagrad model's learning curve compared with the original one which means the accuracy and the performances were improved in this way.