# CS434 - Assignment 2 Logistic Regression

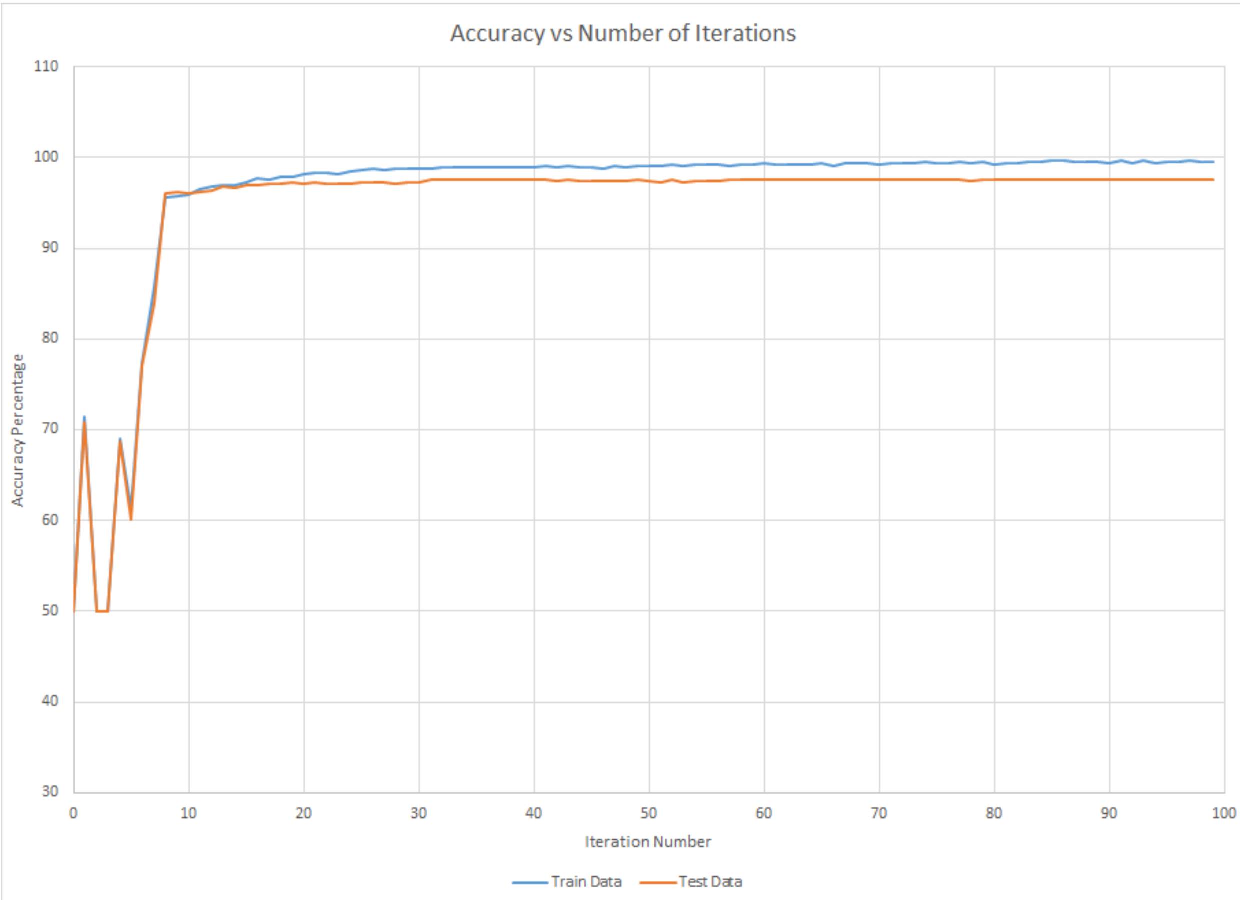
*Nathan Woodworth*
*Sean Hammond*
*Stephanie Ison*

## Part 1: Finding a Good Learning Rate

| Learning Rate | Iterations | Accuracy |
|---|---|---|
| 0.1 | 170 | 100% |
| | 165 | 99.9% |
| | 160 | 99.9% |
| 0.01 | 170 | 100% |
| | 165 | 99.9% |
| | 160 | 99.9% |
| 0.001 | 170 | 100% |
| | 165 | 100% |
| | 160 | 99.6% |
| 0.0001 | 170 | 100% |
| | 165 | 99.9% |
| | 160 | 99.9% |

From our own observations we found 0.001 to be the most suitable learning rate for our training algorithm, because it achieved 100% accuracy with less iterations than the other learning rates. The above data is a small sample of the range of values that we tested when searching for the best learning rate to use for our algorithm.

# Part 2 Accuracy as a Function of the Number of Iterations




The training data and test data follow the same path in accuracy until about 20 iterations. The test data then begins to plateau as the training data continues to increase. Once the training data reaches 100% accuracy, no more changes occur. It can be observed that in both sets of data, a 4 and a 9 have a great enough variation in pixels that the trends of accuracy are very similar from the 0 iterations to the last iteration.

Part 3 Pseudo Code for Logistic Regression with Regularization Algorithm

**TrainDataLambda(X, Y, W, ɳ, λ):**

$W \leftarrow (0, 0, 0, ..., 0)$

For i = 0 to number of iterations do

$d \leftarrow (0, 0, 0, ..., 0)$

For j = 0 to N do

$y_{sigma} \leftarrow \frac{1}{1 + e^{-W^T X^j}}$

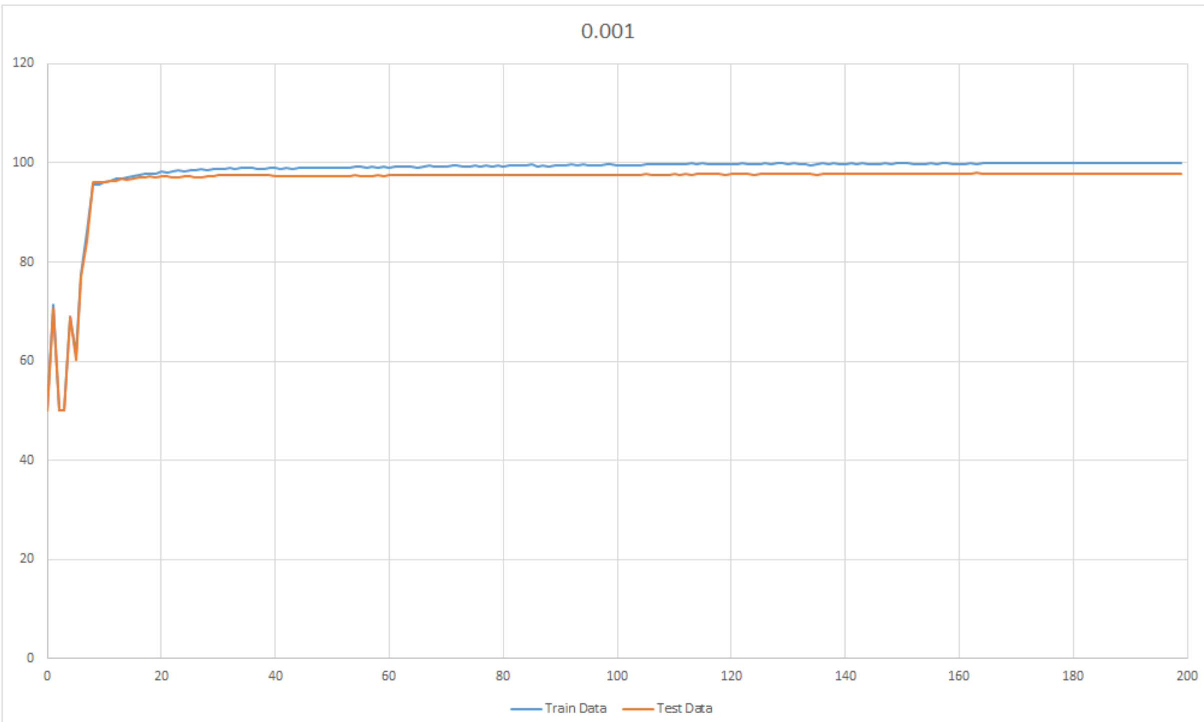
$error \leftarrow Y^j - y_{sigma}$

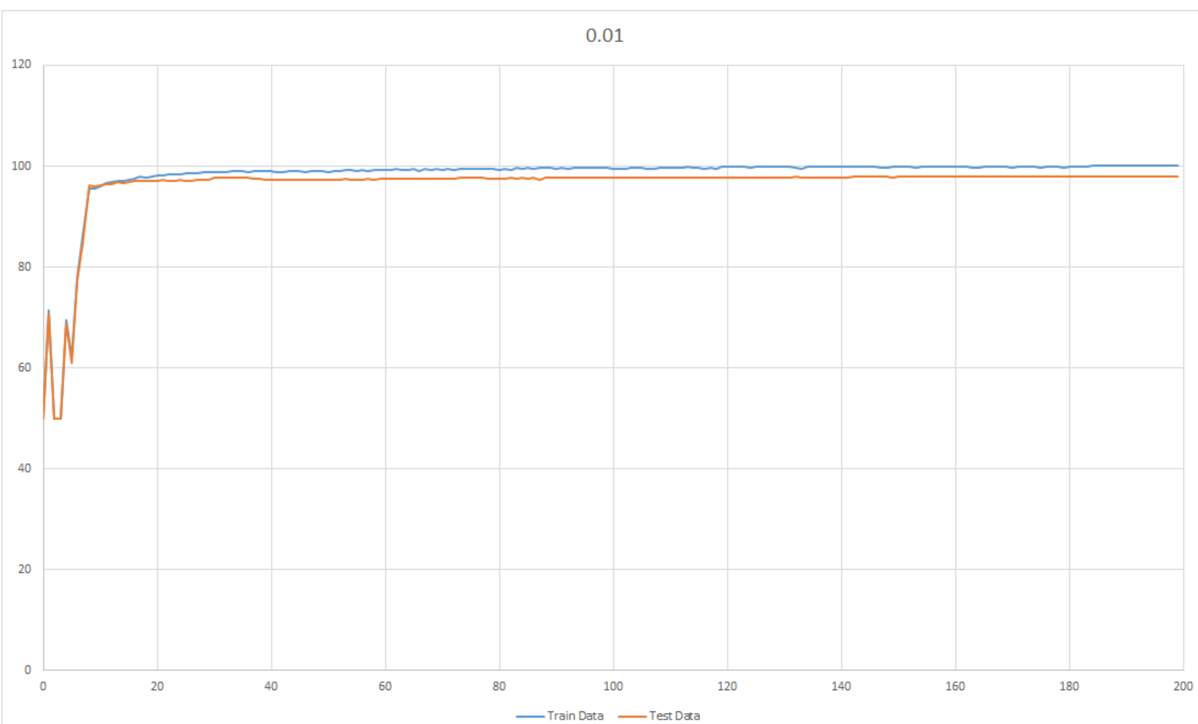
$d \leftarrow d + error * X^j + \lambda W$

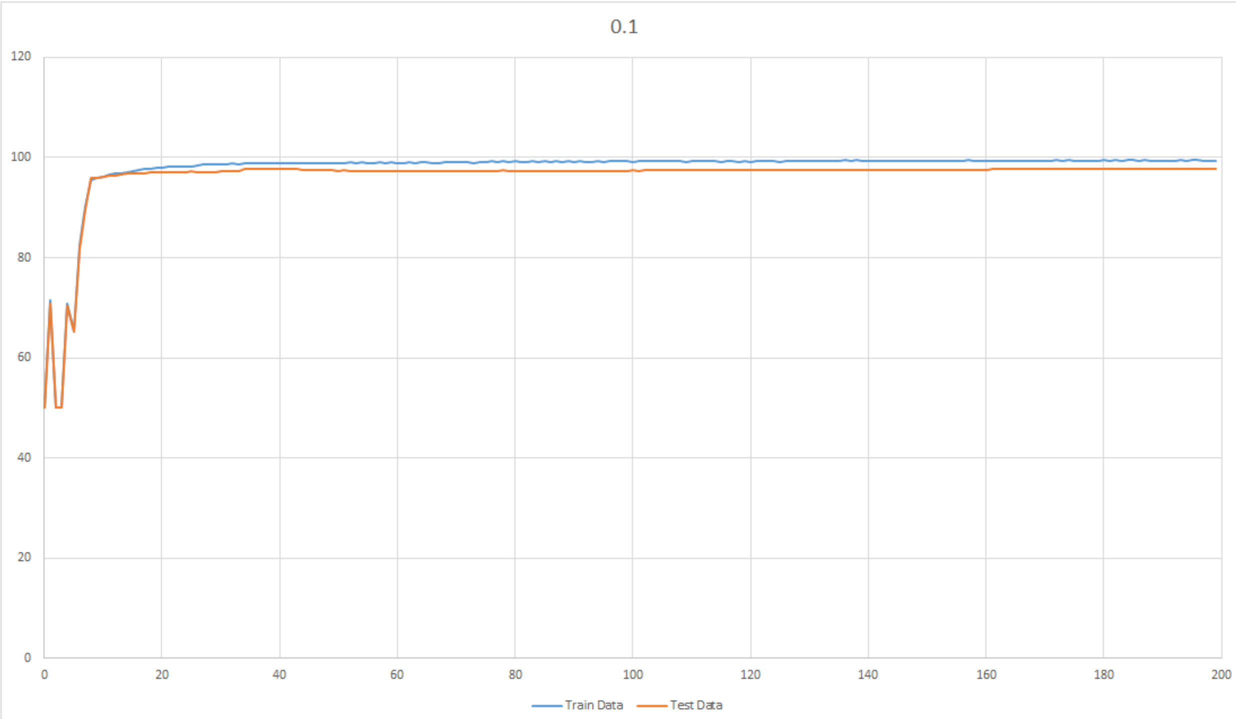
$W \leftarrow W + ɳd$

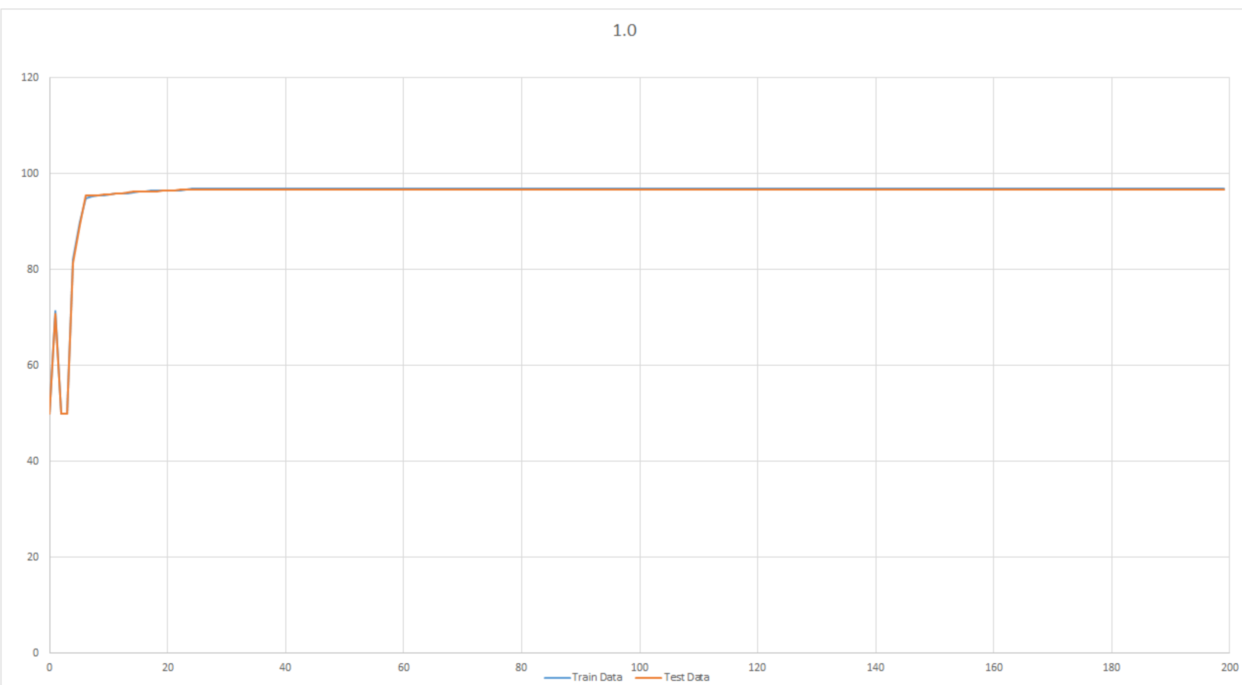
# Part 4 Logistic Regression with Regularization Algorithm Results

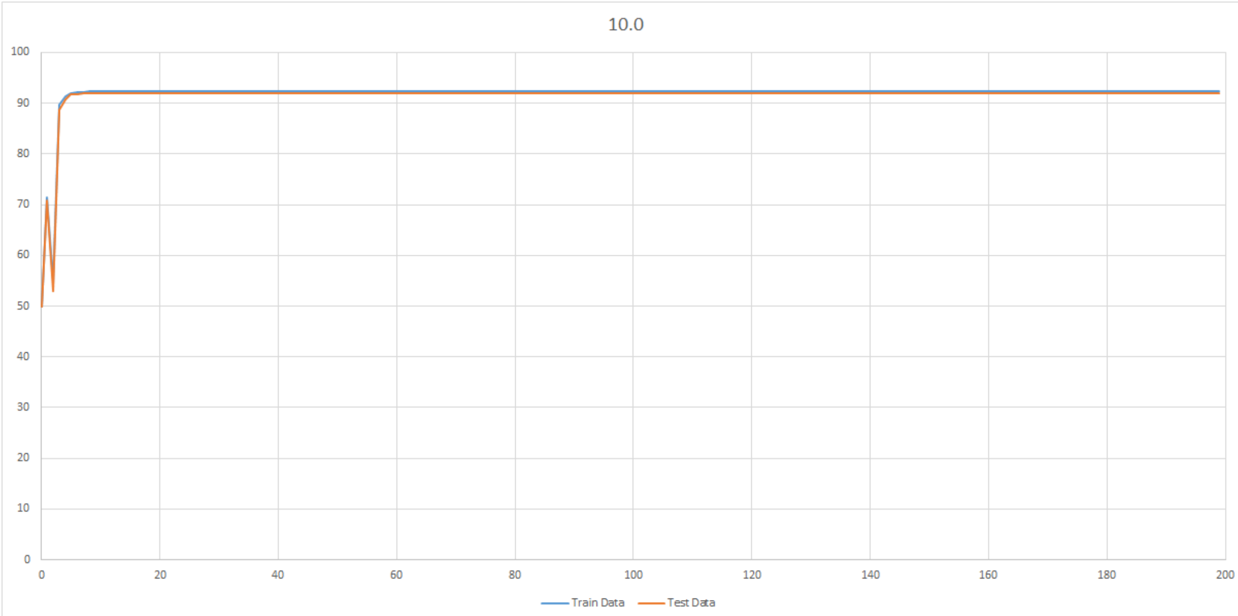
Below are charts of different lambda values from 0.001 to 100.0. The x-axis is the number of iterations and the y-axis is the accuracy in percentage.

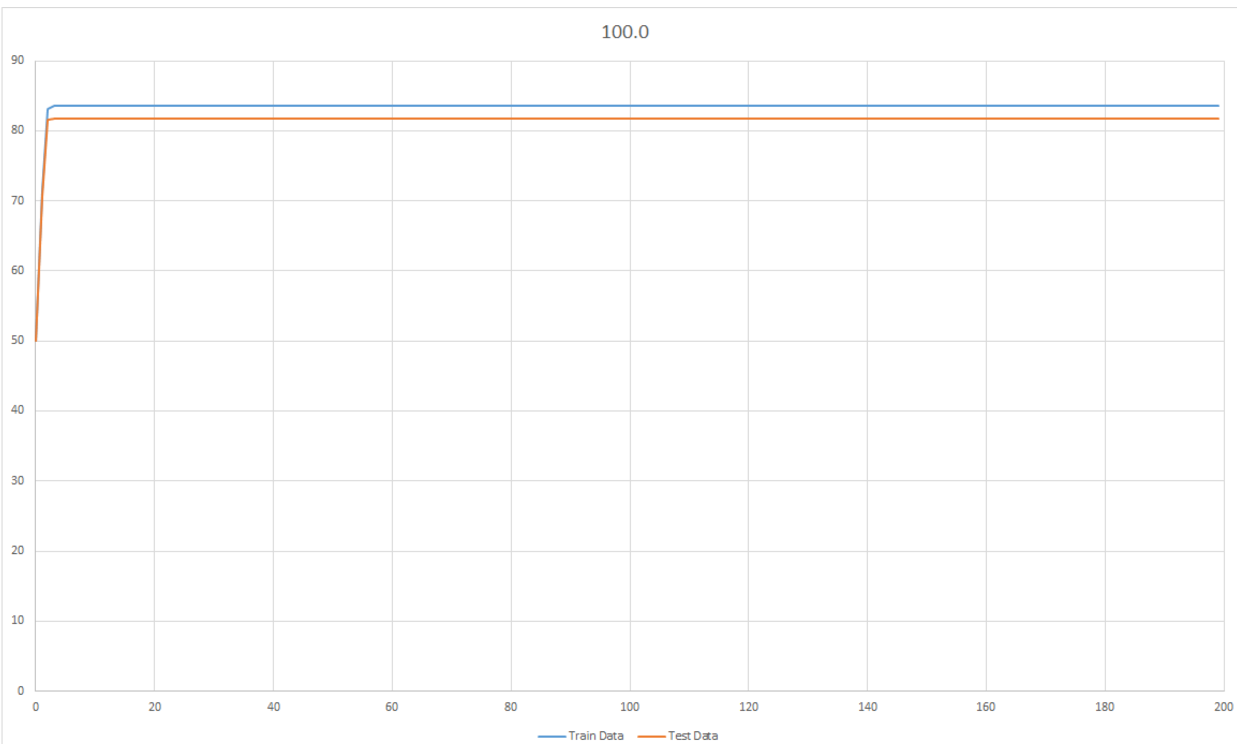



0.1
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
180
200
Train Data
Test Data


1.0
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
180
200
Train Data
Test Data




As the lambda value increases, the accuracy of both the training and test data decreases over 200 iterations. By the rate of change of the charts, the accuracy will not improve with more iterations. What is observable is that as the lambda value does increase, the spikes of accuracy seen in the beginning and the jagged behavior of the training accuracy smooths out. The L2 regularization decreases the overall regularization, but makes it quicker for the learning algorithm to find its optimal w vector with the given lambda faster.