James Alford-Golojuch

CS460G

November 5, 2016

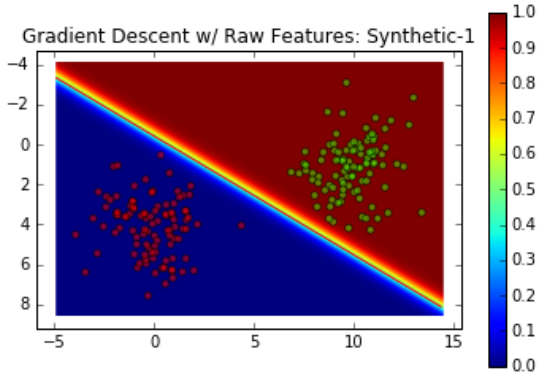
Assignment 3 Logistic Regression Report

I have four separate files. There are two files for each implementation of logistic regression where each implementation has two files where one file has the program using the raw data while the other has quadratic data. Since the main difference between the two files for each revolves around whether the function is quadratic or linear covering one file's methods will pretty much cover the entirety of the other program besides the simple method of converting the data into a quadratic function.

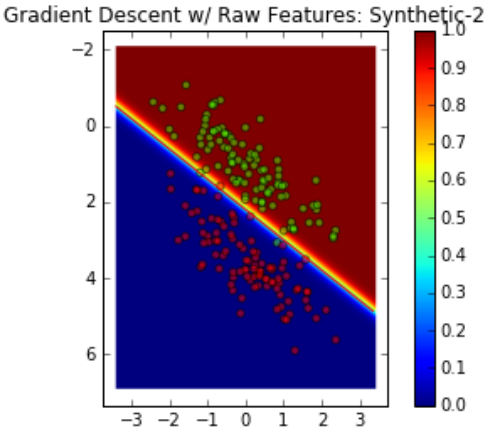
For the low level gradient descent implementation I had four main functions which are named sigmoid, costfunc, gradDec and predict. The sigmoid function performs the actions for the sigmoid function for the logistic regression function. It takes as input the theta values transposed and dotted with the data values. This is called in the costfunc which is the cost function. Using the sigmoid function is finds the probability of class 1 with the given data. Costfunc then finds the log likelihood vector and returns the mean value of that vector. I also have the gradDec function which returns the gradient descent vector which represents the final theta values for the logistic function. I also have a predict function which finds the correct classification rate of the logistic regression function which I then use to find the misclassification rate of the function.

For the high level implementation I used the logistic regression model located in python's numpy. I didn't need to use any of my own functions in this since the logistic regression model has its own functions which perform the same actions as the functions that I used above. The have the exact same functional uses as above so I won't go through and describe each one again. Also I used a bias term for both implementations of the logistic function. I used the high level implementation of the logistic regression model to make sure that my low level implementation was done correct. Comparing the misclassification rates I believe that my implementation is good. I did notice I forgot to convert the misclassification into an actual percentage so for example when the misclassification is .03 it really is 3% . Otherwise I don't believe there is anything wrong with my implementations or my results.

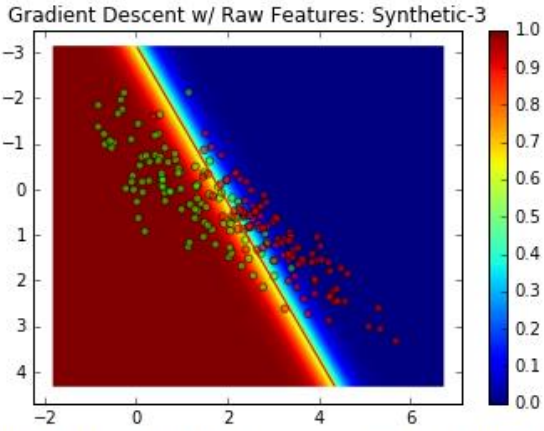
Gradient Descent Implementation of Logistic Regression with Raw Features:



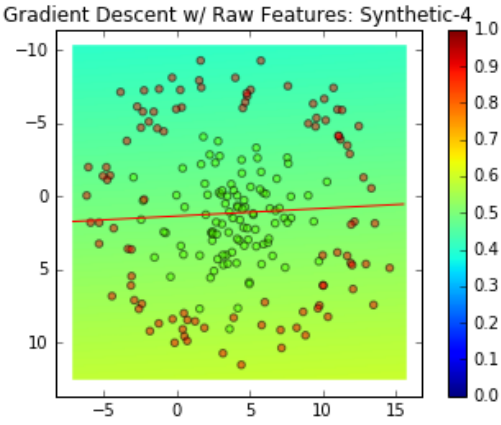
The misclassification rate is 0.0 %



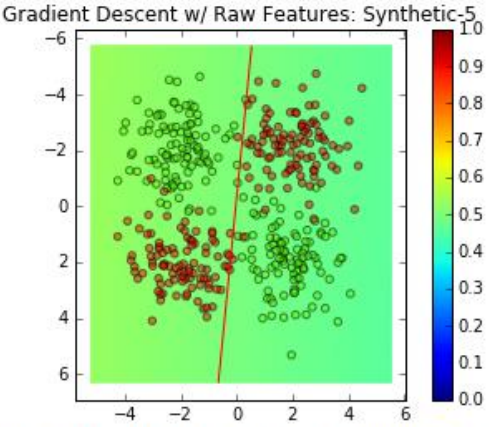
The misclassification rate is 0.010000000000000009



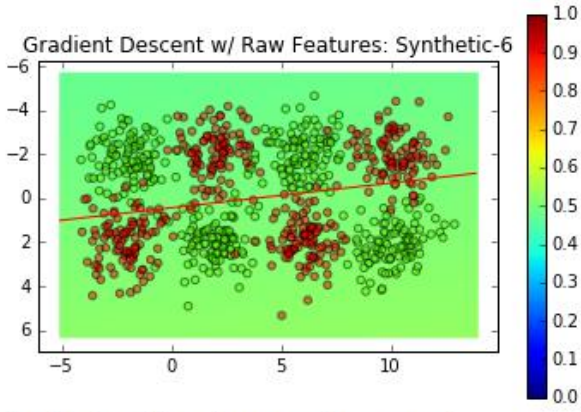
The misclassification rate is 0.10999999999999999



The misclassification rate is 0.475

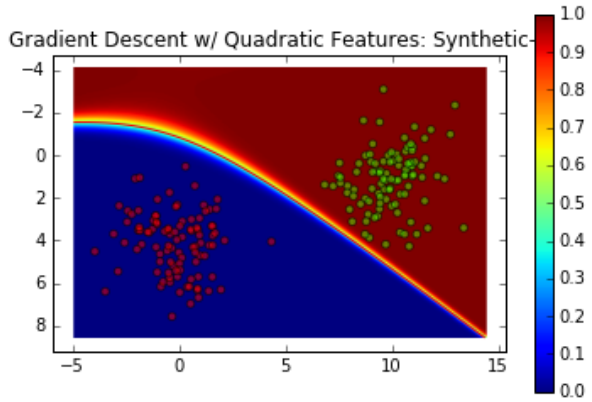


The misclassification rate is 0.49250000000000005

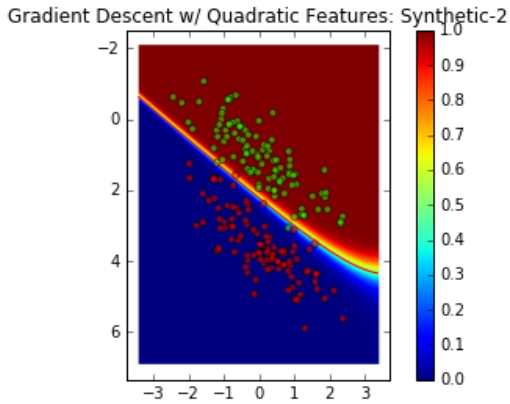


The misclassification rate is 0.495

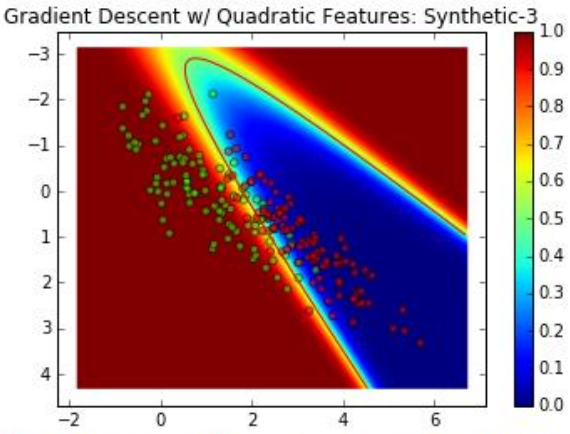
Gradient Descent Implementation of Logistic Regression with Quadratic Features:



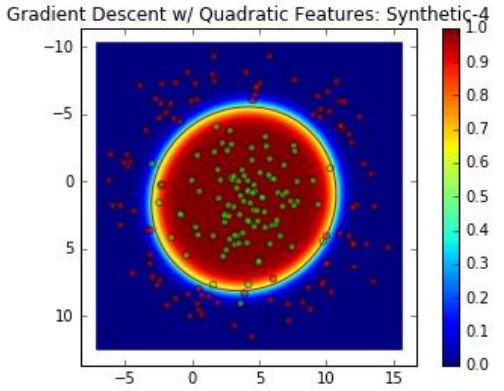
The misclassification rate is 0.0 %



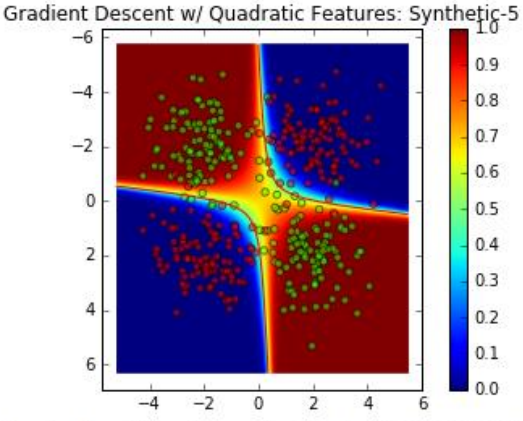
The misclassification rate is 0.010000000000000009



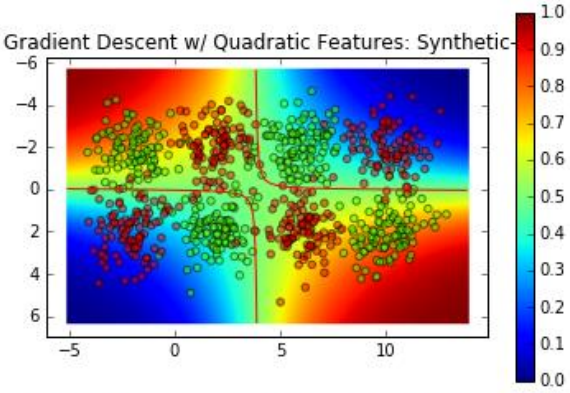
The misclassification rate is 0.11499999999999999



The misclassification rate is 0.020000000000000018

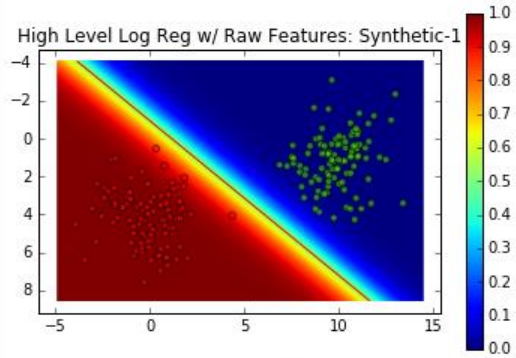


The misclassification rate is 0.03249999999999997

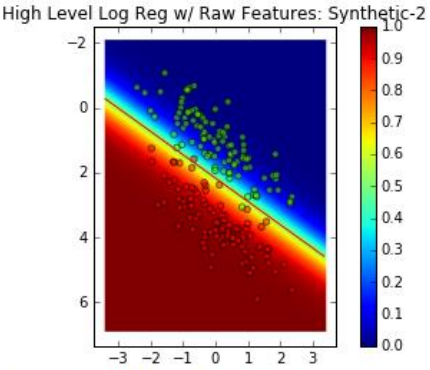


The misclassification rate is 0.49250000000000005

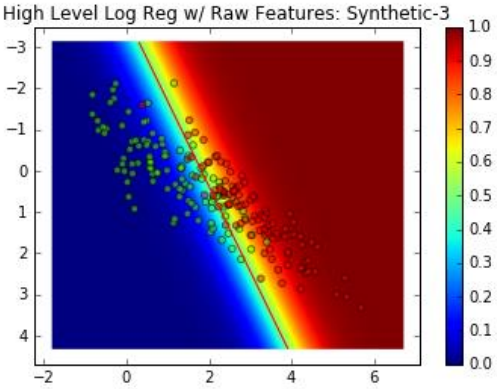
High Level Implementation of Logistic Regression with Raw Features:



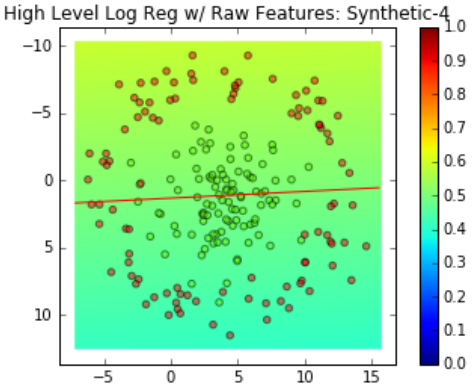
The misclassification rate is 0.0



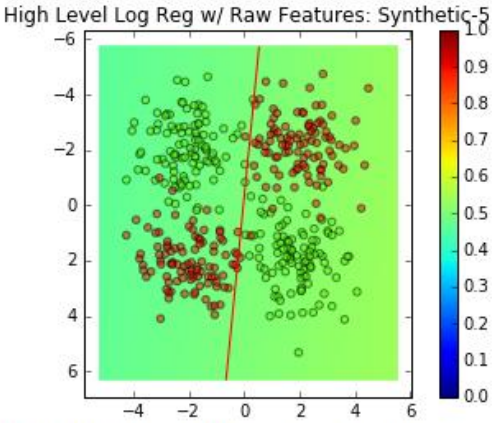
The misclassification rate is 0.01



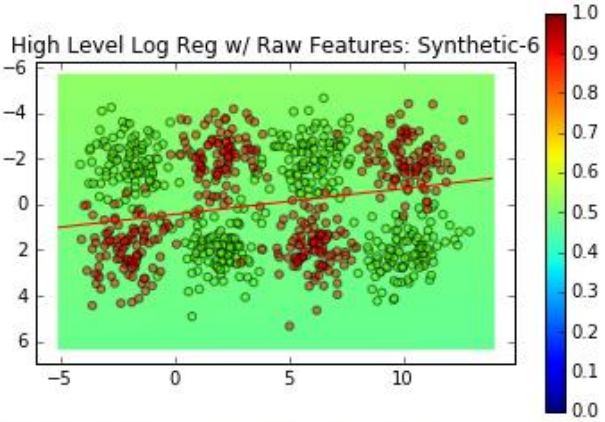
The misclassification rate is 0.12



The misclassification rate is 0.475

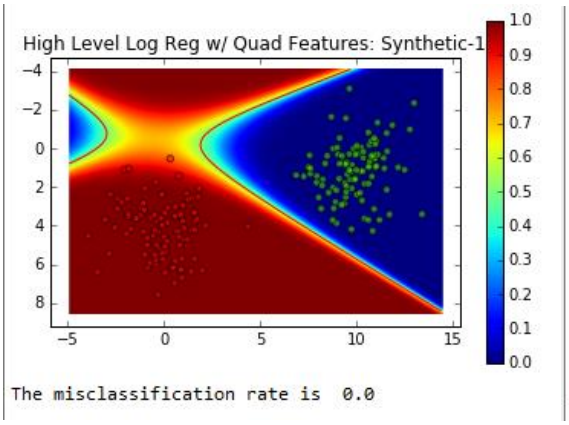


The misclassification rate is 0.4925

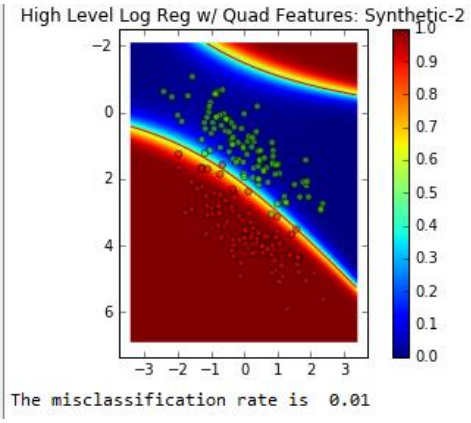


The misclassification rate is 0.495

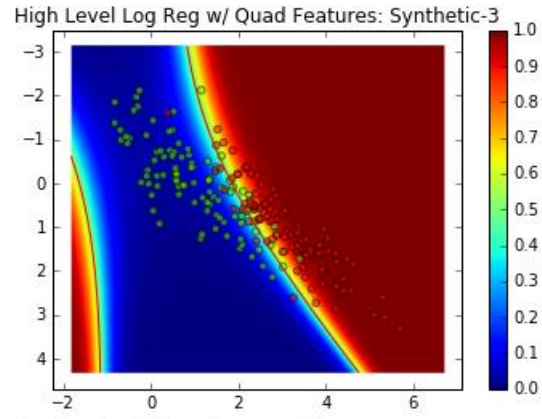
High Level Implementation of Logistic Regression with Quad Features:



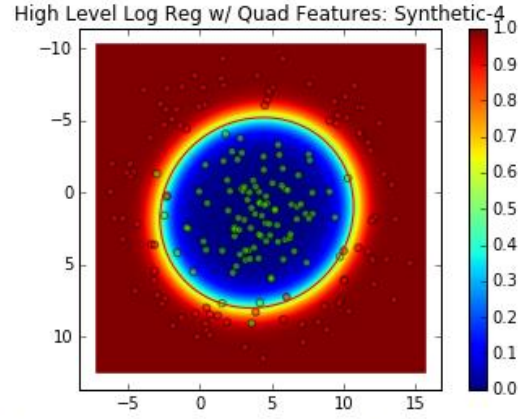
The misclassification rate is 0.0



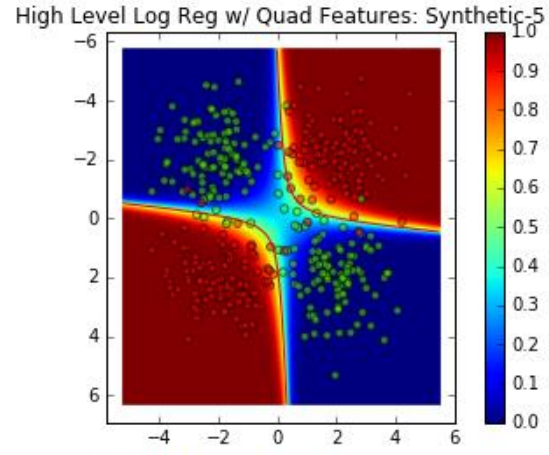
The misclassification rate is 0.01



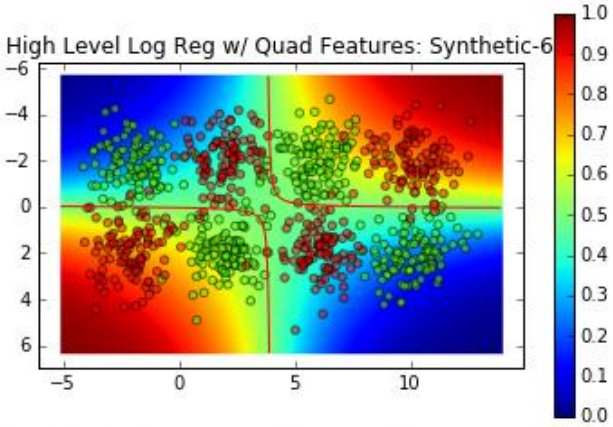
The misclassification rate is 0.105



The misclassification rate is 0.025



The misclassification rate is 0.03



The misclassification rate is 0.4925