Fahim Khan
CS6923 Machine Learning
5/1/20

Final Project: Multiclass Gaussian Kernel SVM

Gaussian Kernel SVM is an extension to SVM Dual Kernel.
Furthermore I implemented these with a multi-classification extension.

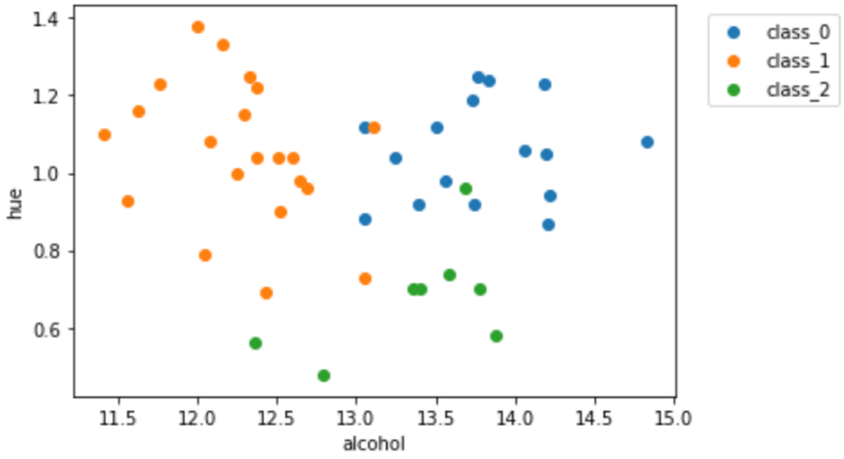
- I used the one-vs-rest strategy which is the default in sklearn's SVM algorithm.
    - Trains each class vs the rest of the data
    - Each test data is tested against each class to determine labels
    - Each test data point can have multiple labels detected, the majority label is chosen
- The original HW SVM Dual function is used within the multiclass function, same way as the numpy Gaussian SVM.

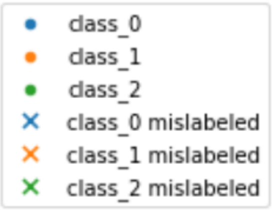
The original HW SVM Dual implementation had very poor accuracy with the multiclass strategy since the wine data isn't linearly separable with each class vs the rest. Therefore the accuracy is much lower than Sklearn's version, which does extra processing and goes through multiple iterations.

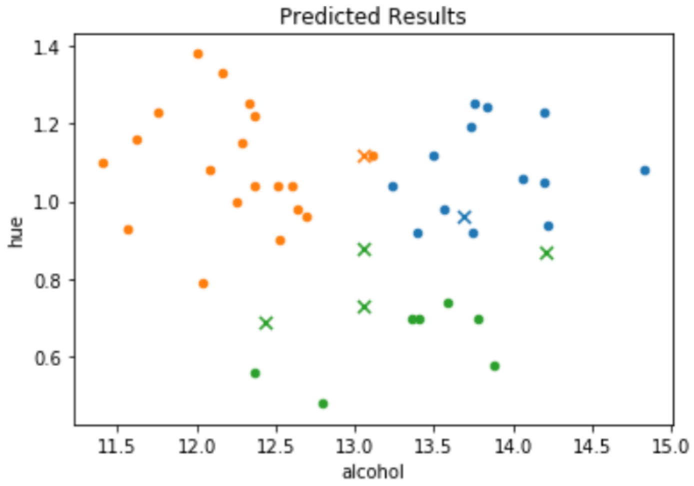
The rbf/Gaussian kernel worked better as expected, since it is best with non-linearly separable data. This is because rbf is a radial basis function which decays uniformly around the support vector, leading to spherical or bubble-like contours separating each class.

I used the wine dataset (alcohol vs hue) because it was linearly separable to some extent, so SVM Dual could be reasonably compared with SVM Gaussian. The points that mix in with other classes and the bubble-like shape of each class would show how SVM Gaussian does better.

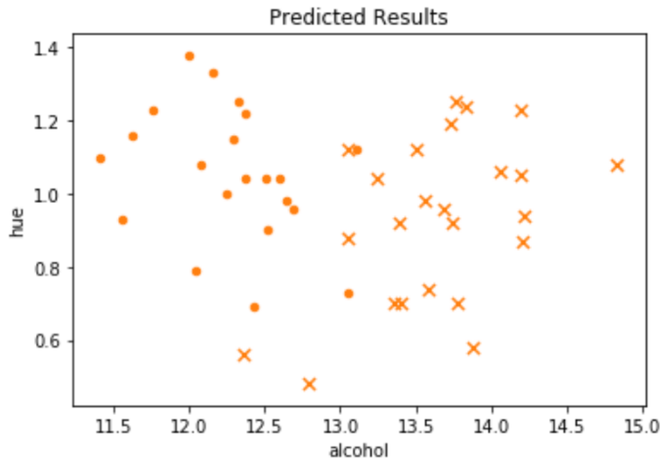
Wine Test Data





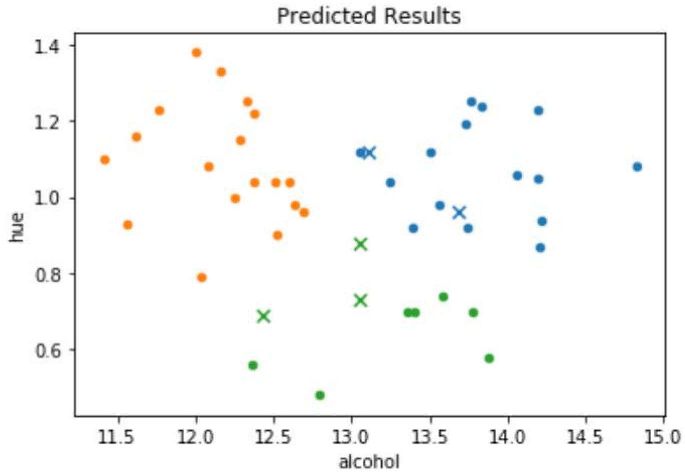


Sklearn SVM Dual: Accuracy = 87%

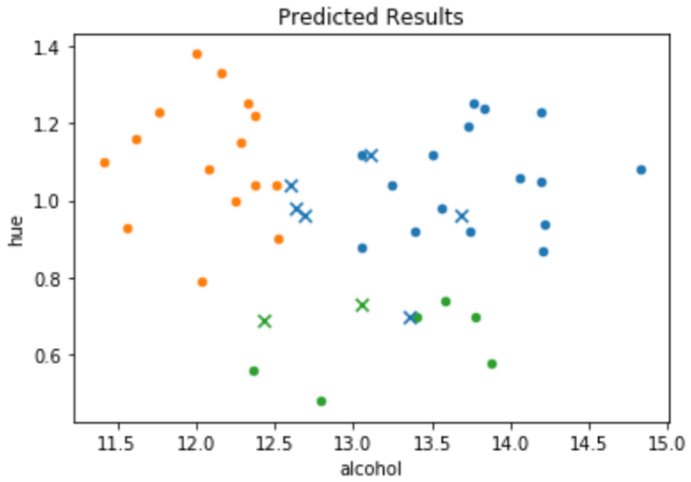


Numpy SVM Dual: Accuracy = 47%

*Note that most of the data was predicted as both class1 and class2 (class1 was the tie breaker). Class0 couldn't be predicted very well.



Sklearn RBF SVM: Accuracy = 89%



Numpy RBF SVM: Accuracy = 82%