Machine Learning Decal Fall 2018 Homework 3

Hyperparameter Tuning with SVMs

Daniel Geng
UC Berkeley
dangengdg@berkeley.edu

March 1, 2018

1 Introduction

In this homework you will be using SVMs on the Wisconsin Breast Cancer dataset which can be found at the following URL:

https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29

This URL contains all relevant information on how the data is formatted. The data was collected by Dr. William H. Wolberg from the University of Wisconsin Hospital between 1989 and 1991. The dataset was never actually intended for any machine learning to be done on it, but you will be seeing how well you can get an SVM to classify the data. **For this assignment please submit an ipython notebook with all relevant code, write-up, and images.**

2 Tasks

2.1 The Data

Use the above link to familiarize yourself with the data. For your convenience we've removed a few bad data points from the dataset.

2.2 The Tools

You will be using sklearn's implementation of an SVM for this assignment. In particular, you might find

```
sklearn.svm.SVC
sklearn.model_selection.train_test_split
matplotlib.pyplot
pandas
numpy
```

useful although you are by no means restricted to just these libraries.

2.3 Grid Search

In class we covered the notion of hyperparameters and the importance of tuning them correctly. The SVM is a good model to see this process at work. Recall that the soft-margin SVM has a hyperparameter called C which determines the amount with which we penalize misclassified points.

Also recall that the SVM model can make non-linear decision boundaries by making projecting points on to a higher dimension space. This is called often called the "kernel method" and involves picking a "kernel" to use, which determines how exactly data points are projected. For example, the polynomial kernel of degree 2 projects data points on to a paraboloid as shown in this video:

https://www.youtube.com/watch?v=3liCbRZPrZA

Thus when using a SVM with a polynomial kernel there are two hyperparameters to tune. The value of C and the degree of the polynomial, d. Your first task will be to find the optimal pair of (C, d) that gives the best results on a test set.

2.3.1 Specifications

- Right now your data are all scales from 1 to 10. Normalize your data to have 0 mean and 1 standard deviation
- For your labels 1 should indicate malignant and 0 should indicate benign
- Please use $C \in \{.0001, .001, .01, .1, 1, 10, 100\}$ and $d \in \{1, 2, 3, 4, 5\}$ for a total of 35 different pairs to test.
- Use a 70-30 split on the data (you might find "sklearn.model_selection.train_test_split" useful)
- You are free to use "sklearn.model_selection.GridSearchCV" but honestly just writing a nested for loop will probably be easier
- Graphically represent the results of the grid search. Let the x-axis be C and the y-axis be d and plot the accuracies of each pair on the test set.
  - Hint: You can just make a 2d numpy array, and then call "plt.imshow([array])", "plt.colorbar()", "plt.show()" to create a nice looking heatmap
  - Look at the documentation for how to give the plot a title and correct axis labels
- Indicate the best pair of parameters for the dataset and the best accuracy.
- Describe any general trends that you can see. What can you say about overfitting?
- As a bonus problem (i.e. optional but pretty instructive and not that hard to do) in medicine people often want to distinguish a false positive rate and a false negative rate instead of just an accuracy. This is because there are different consequences for the two categories of false results. False positives will result in unnecessary treatment while false negatives will result in lack of treatment. Google what a confusion matrix is and then create one using the SVM classifier that gave you the best results from the above grid search. Report the false negative and false positive rate.

2.4 Cross Validated Grid Search

As you might remember from lecture in addition to a polynomial kernel there is also a Radial Basis Function (RBF) kernel. This kernel has a hyperparameter called γ that determines the "fineness" of the boundaries. A higher gamma will result in a more smooth boundary and a lower gamma will result in a more jagged boundary. Use cross validated grid search to find the best combination of γ and C on the breast cancer dataset.

2.4.1 Specifications

- Using "sklearn.model_selection.GridSearchCV" will make your life considerably easier. Check out the documentation (especially the example) for reference:

  http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

- Use $C \in \{.0001, .001, .01, .1, 1, 10, 100\}$ and $\gamma \in \{.001, .01, .1, 1, 10, 100\}$ for your gridsearch
- Report the best accuracy and which parameters gave the best accuracy

3 Conclusion

As you have seen SVMs easily get above 90% on this dataset. With just a bit of hyperparameter tuning we can push this number up to a remarkable 97%. Moreover, SVMs are a very simple algorithm and should have taken no more than a few seconds to train on your computer. The potential to create models so easily that are this accurate on a task as complicated as breast cancer classification is perhaps one of the reasons that machine learning has become such a hot topic as of late.

3.1 Bonus Question

Why are SVMs a better model choice for this dataset as opposed to state-of-the-art models such as neural networks?