DAT5 SF: HOMEWORK 2 ASSIGNMENT

Assigned: Friday, March 21
Due: Sunday, March 30 by 11:59PM
Submission Method: Push your work to Github and create a new issue. Include @hallr and @ghego

The purpose of this homework is to gain deep, first-hand experience with cross-validation and selection of model parameters. Although the Scikit-learn package provides nicely packaged methods, cross-validation is such an important concept that we will implement it more directly. We will then use our implementation of cross-validation to select some model parameters – also called "hyperparameters" – for our KNN classifier on the Iris dataset.

DATA & CONTEXT

For this assignment, we will use the Iris dataset that we saw in the lab. This is a very well known dataset in the machine learning world. It is relatively simple, with only four features. Therefore, it is easy to develop an intuition about the data.

Unlike in class, for this assignment we will start with the Iris dataset in a .csv file, instead of using the "on a silver platter" version of the dataset included in the datasets library in Scikit-learn. This way, we will gain additional practice with the data wrangling skills we began building in HW1. In other words, do NOT use

```
iris = datasets.load_iris()
```

Note that we will actually be implementing a KNN classifier in this assignment. So, this is our first machine learning / predictive algorithm!

- The dataset is available from the UCI Irvine's Dataset Repository: https://archive.ics.uci.edu/ml/datasets/Iris
- Dataset: https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.names
- Detailed Description of Dataset (optional, in case you are interested): https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.names

No need to get the data from UCI; we have provided the HW2 dataset via Schoology.

SUBMITTING YOUR WORK

You may work in iPython notebook, via Python scripting, or both. Note that iPython notebook provides the ability to export Python scripts. Filenames for Python scripts should end in ".py". To execute a Python script from the command line, simply type:

```
python <filename.py>
```

Please submit your work by pushing it to your fork of the course Github repo and creating a new issue. Be sure to include @hallr and @ghego in the body or your issue.

HOMEWORK QUESTIONS

As we proceed through the course and increase our data science familiarity and problem-solving skills, the homework assignments will become less and less structured. For this assignment, some hints are provided with the questions. The questions are higher-level than in HW1.

1. Implement KNN classification, using the sklearn package. We learned how to do this in class.
   See also: http://scikit-learn.org/stable/modules/neighbors.html#nearest-neighbors-classification

2. Implement cross-validation for your KNN classifier. You may find it helpful to start with the cross-validation code from the lab. Note that you may need to re-write portions of that code to get it to work for you. **Use 5 folds for your cross-validation.** *Do NOT use the cross_val_score method from sklearn to do this "black box" for you.*
   See also: http://scikit-learn.org/stable/modules/cross_validation.html#

3. Use your KNN classifier and cross-validation code from (1) and (2) above to determine the optimal value of K (number of nearest neighbors to consult) for this Iris dataset. This hyperparameter will be a number between 1 and 150 ☺.

4. Using matplotlib, plot classifier accuracy versus the hyperparameter K for a range of K that you consider interesting. Explain in words what you are seeing.

5. OPTIONAL BONUS QUESTION: Using the value of K obtained in (3) above, vary the number of folds used for cross-validation across an interesting range, e.g. [2, 3, 5, 6, 10, 15]. How does classifier accuracy vary with the number of folds used? Do you think there exists an optimal number of folds to use for this particular problem? Why or why not?