CICS 397A – Fall 2020

Homework 3

Due Wednesday, November 18th at 11:59pm ET

You are encouraged to discuss the assignment in general with your classmates, and may optionally collaborate with one other student. If you choose to do so, you must indicate with whom you worked. Multiple teams (or non-partnered students) submitting the same code or output will be considered plagiarism.

Code must be written in a reasonably current version of Python (>3.0). You are free to use Python's standard modules for data structures and utilities, as well as the modules mentioned in class (pandas, seaborn, etc.).

Working with scikit-learn

The goal of this assignment is to get you familiar with the `scikit-learn` toolkit. As with the last assignment, the requirements are open-ended — we want to give you a chance to work with your data on something you find interesting or insightful. In addition to submitting code, you'll also be writing up your findings.

The intention of this assignment is to allow you to continue to work with your data set from previous assignments, but you are not required to do so and may use new data if you prefer.

1. Unsupervised Learning

The first task is to perform some cluster analysis on your data. In lecture, we talked about **agglomerative** and **k-means** clustering, which are both implemented by scikit-learn, along with many, many others; see: https://scikit-learn.org/stable/modules/clustering.html).

Choose a clustering algorithm from the ones listed at the link above, and write some code to run it on your data. The documentation provides code samples and examples that you can use as a guide.

As to which algorithm you use, it's up to you, and you're encouraged to experiment. Think carefully about which attributes you use for clustering and how the distance metric you use captures a meaningful notion of similarity.

Once you've been able to segment your data into clusters, write up a short explanation describing your efforts. You should include:

- A short description of your data set in terms of where it came from and what it represents
- Examples of some of the clusters that were created

- A description of the intuitive meaning of the clustering you've produced and how that's reflected in the algorithm and data choices you've made

Your goal here is to explain the type of clusters you were looking for and whether you were able to achieve that goal. For instance, you could cluster movie data into genres, time periods, groups of similar actors, etc., and each of those options would entail different choices in terms of data and/or approach. If your efforts to produce meaningful clusters were unsuccessful, that's okay — just make sure you provide a description of the things you tried and a hypothesis on why the algorithms failed to produce good results.

2. Supervised

Next, you'll use `scikit-learn` to do some classification. In addition to the **Naive Bayes**, **Decision Tree**, and **k-NN** approaches we talked about in class, you can use one of the other model types that are listed here: https://scikit-learn.org/stable/supervised_learning.html

As with Part 1, it's up to you to choose an algorithm and read the docs to find the meaning of the different parameters. You should use cross validation to get an idea of how your model(s) perform, and experiment with different approaches and settings.

You should provide a short write-up your findings, including but not limited to:

- A short description of the prediction task, including the target values and the attribute(s) you are incorporating into your model
- A summary of how your model performs at the prediction task, based on accuracy, precision/recall, or some other measure. What does this tell you about your data?
- A description of the models tried, different parameter settings, and the effects on performance

Tips for Success

Data scientists often joke that they spend 90% of their time preparing their data, and you might find that you need to do a bit of wrangling to get your data in a form that will play nicely with `scikit-learn`. In particular, you'll need to transform numerical attributes (see https://pbpython.com/categorical-encoding.html for some good suggestion). You are advised to get started early, and seek help from the course staff if you get stuck or would like some input.

Requirements and Grading

As with the last assignment, think of this as a mini-project. Maximum points will be awarded for results and presentation that are thoughtful and insightful. The final project will be building on these efforts, so time spent now will help you down the road.

What to Submit

- A pair of files called `cluster.py` and `classify.py` which contain the code for reading in your data from csv, running the clustering and classification algorithms, and outputting results and/or performance metrics (you may also store your code in a Jupyter notebook if you'd like).
- The data file that you used called `data.csv`
- A `summary.txt` (or `summary.pdf`) containing your writeups for parts 1 and 2 described above