Hw 2

Due Sunday by 11:59pm **Points** 100 **Submitting** a file upload

Use the listed technique in SKLearn to make the appropriate predictions for each of these data sets.

Classify mice using k-nearest-neighbors (KNN)
http://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression
(http://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression)

Predict cases of heart disease using **all 3** of the following techniques:
Naive Bayes, decision tree, random forest
http://archive.ics.uci.edu/ml/datasets/Heart+Disease (http://archive.ics.uci.edu/ml/datasets/Heart+Disease)

Complete instructions for submission:

Use a header comment in your with at least your name and which thing you're working on.

Please reasonably use comments throughout your code to indicate or explain things which aren't generally obvious.

Submit files in the format

<UID>_<n>.py

where <UID> is your directory ID (your email, NOT your number) and <n> is the number of the data set you're working on as listed above.
Your codes will be run standalone in bash via the command
python3 <filename>.py <original datafile name>

Our method here of running the code with the datafile as an argument requires the use of *sys.argv*, which is easy to use and easy to look up.

Your code will need to print the accuracy of the results and nothing else.

You also shouldn't make any raw edits to the data files for labelling columns etc.
If you're worried about compliance and compatibility, you can refer to the class github for the source of the grading software (feel free to submit PRs or email me if you find any issues with it).
For full credit, for each data set, you will need to:
-Write code that runs in Python3
-Use SKLearn

-Use the right technique(s)
-Use a reasonable division of the data for training and testing (should be randomly selected (i.e. not just the first 90%))
-Not overtrain the model on your data (i.e. don't use an unreasonably large number of epochs)
-Use a data import and cleaning method that is clean and easily human readable (which should also be true for the entire code!)
-Correctly use one hot encoding
-Make a reasonable attempt to play with model parameters.
-Not crash the grading system!!
-Not hack or attempt to hack the grading system!!
-Reasonably handle missing data, if present

Training should be limited to 2 minutes.
Do not attempt to edit the local files
The following libraries can be used (in the most recent versions):
numpy
sklearn
pandas
If you want to use others, please let us know explicitly by email (you'll likely want to, especially for the image processing).
If you email with questions and desire meaningful answers, don't wait until right before the deadline!
Grading:
30% code runs in a reasonably functional manner
50% code does what it's supposed to do (as outline above)
20% code performs well (in terms of accuracy, run time, etc)

File Upload | Google Doc | BOX

Upload a file, or choose a file you've already uploaded.

File: Choose File No file chosen

+ Add Another File

Click here to find a file you've already uploaded

Comments...

Cancel | Submit Assignment