CS 6316

Random Forest In-Class Activity

You may work with a partner on this activity. Both students must submit individually (but it's OK if you submit the same code shared between the two of you.)

1. Run `plot_ensemble_oob.py` file (see Resources on Collab). Understand what is going on here and what information is being conveyed.
2. Recall that Random Forest can be used to reduce the dimensionality of a given data set. Go to the following website, and follow this short tutorial by Andrew Cross that takes you through discovering which features (attributes) have the greatest impact on the prediction.
   URL: http://www.agcross.com/blog/2015/02/05/random-forests-in-python-with-scikit-learn/
   (Note: this code is written in Python 2.X, if you are using Python 3.X you may need to modify the "print" statement in the code to allow this code to run.)
   a. Write and run code on the ***iris data set*** (as described on this website)
   b. Find ***another data set*** to work with. You can look on the UCI machine learning repository (archive.ics.uci.edu/ml/index.php). Find a data set that has at least 15 features (attributes). Once you've chosen a data set, run this code on the data set and answer the following questions:
      i. What data set did you use (provide name and URL)?
      ii. How many total features does the original data set have?
      iii. After running the code, display the graph you obtained. How many features (out of the total) would you choose to keep based on the results? Which attributes are they (list the names)?
      iv. The ***cross-tab*** feature in this code is one way to look at how well the model performed. It is similar to a confusion matrix that describes how well the model performed on the test data set (basically anything off the diagonal is a misclassification.) Edit your data set to remove the columns corresponding to the features that you do NOT want.

v. Re run the code. Given you kept only those features that have the greatest impact on prediction, how well did your model perform now?
   Was it about the same (this is good; we reduced the dimensionality of the data set and still obtained comparable results)?
   Was it better (this is even better; reduced the dimensionality and obtained a higher prediction accuracy)?
   Or was it worse (oh dear… maybe too many features were removed…*try again?*). Discuss your findings.

Write up everything (PDF format please), include your code (.py is fine), and **submit your work by 8:00pm tonight on Collab under "Assignments."** Remember, if working with a partner, you can submit the same documents, but both of you need to submit (makes it easier to assign a grade for this in-class activity.)