Assignment 3: Unsupervised Learning and Dimensionality Reduction

Due: April 1st 11:55pm
Please submit via T-Square

Why?

So far this term we have explored supervised learning algorithms. Now it's time to explore unsupervised learning algorithms. This assignment asks you to use some of the clustering and dimensionality reduction algorithms we've looked at in class and to revisit earlier assignments. The goal is for you to think about how these algorithms are the same as, different than, and interact with your earlier work.

The same ground rules apply for programming languages as with assignments #1 and #2.

Read everything below carefully!

The Problems Given to You

You are to implement (or find the code for) six algorithms. The first two are clustering algorithms:

- *k*-means clustering
- Expectation Maximization

You will need to choose your own, potentially problem-dependent, measures of distance/similarity. You need to justify your choices, but you're practiced at that sort of thing by now.

The last four algorithms are dimensionality reduction algorithms:

- PCA
- ICA
- Randomized Projections
- Any other feature selection algorithm you desire

You are to run a number of experiments. Come up with at least two datasets. You are encouraged to use the datasets from previous assignments.

You will then need to follow the procedure below:

1. Run the clustering algorithms on the data sets and describe what you see.
2. Apply the dimensionality reduction algorithms to the two datasets and describe what you see.
3. Reproduce your clustering experiments, but on the data after you've run dimensionality reduction on it.
4. Apply the dimensionality reduction algorithms to one of your datasets from assignment #1 (if you've reused the datasets from assignment #1 to do experiments 1-3 above then you've already done this) and rerun your neural network learner on the newly projected data.
5. Apply the clustering algorithms to the same dataset to which you just applied the dimensionality reduction algorithms (you've probably already done this), treating the clusters as if they were new features. In other words, treat the clustering algorithms as if they were dimensionality reduction algorithms. Again, rerun your neural network learner on the newly projected data.

What to Turn In

1. A file named *README.txt* that contains instructions for running your code
2. Your code
3. A file named *analysis.pdf* that contains your write-up.
4. Any supporting files you need (for example, your datasets).

The file *analysis*.pdf should contain:

- A discussion of your datasets, and why they're interesting: If you're using the same datasets as before at least briefly remind us of what they are so we don't have to revisit your old assignment write-up.
- Explanations of your methods: How did you choose *k*?
- A description of the kind of clusters that you got.
- Analyses of your results. Why did you get the clusters you did? Do they make "sense"? If you used data that already had labels (for example data from a classification problem from assignment #1) did the clusters line up with the labels? Do they otherwise line up naturally? Why or why not? Compare and contrast the different algorithms. What sort of changes might you make to each of those algorithms to improve performance? How much performance was due to the problems you chose? Be creative and think of as many questions you can, and as many

answers as you can. Take care to justify your analysis with data explicitly.

- Can you describe how the data look in the new spaces you created with the various algorithms? For PCA, what is the distribution of eigenvalues? For ICA, how kurtotic are the distributions? Do the projection axes for ICA seem to capture anything "meaningful"? Assuming you only generate *k* projections (*i.e.*, you do dimensionality reduction), how well is the data reconstructed by the randomized projections? PCA? How much variation did you get when you re-ran your RP several times?
- When you reproduced your clustering experiments on the datasets projected onto the new spaces created by ICA, PCA and RP, did you get the same clusters as before? Different clusters? Why? Why not?
- When you re-ran your neural network algorithms were there any differences in performance? Speed? Anything at all?

It might be difficult to generate the same kinds of graphs for this assignment as you did before; however, you should come up with some way to describe the kinds of clusters you get. If you can do that visually all the better. **Note: Analysis write-up is limited to 10 pages.**

One example of analysis is available in the resource section of T-Square.

Grading Criteria

This assignment is out of **10 points**, each representing a percentage point of your final grade. The breakdown for grading is as follows:
1 point: A working implementation of all approaches listed above for your two classification problems.
9 points: Your analysis. You are being graded on your analysis more than anything else. However, analysis without proof of working code will harm your analysis grade.

Follow the directions carefully. Failure to turn in files without the proper naming scheme, or anything else that makes the grader's life unduly hard will lead to an ignored assignment. There will be no late assignments accepted.