EE 559 **Course Project Assignment** Posted: 3/30/2019
Jenkins v1.3 Due: Tues., 4/30/2019, 2:00 PM

Introduction

The goal of this project is to develop a pattern recognition system that operates on a given real-world dataset that is nontrivial to solve. In doing so, you will apply tools and techniques that we have covered in class. You will also confront and solve issues that occur in practice and that have not been covered in class. Discussion sessions and past homework problems will provide you with some tips and pointers for this; also internet searches and piazza discussions will likely be helpful.

Projects can be individual (one student working on the project) **or team** (2 students working together on one project).

You will have significant freedom in designing what you will do for your project. You must cover various topics that are listed below (as "Required Elements"); the methods you use, and degree of depth you go into each topic, are up to you. And, you are encouraged to do more than just the required elements.

Everyone will choose their dataset(s) from the three datasets listed below. Collaboration and comparing notes on piazza may be helpful, and may make it more fun and engaging.

Datasets:

Choose from the following three datasets.

1. Adult Dataset (U.S. Census Data)

Predict whether a person's income is greater than (or less than) $50K per year. Features have been taken from U.S. Census data, and include age, education, occupation category, etc. This is a 2-class problem.

Training and testing sets are posted on D2L. The files labeled "SMALLER" have been down-sampled by factor of 3 (stratified), from the complete versions. The complete versions are also posted on D2L in case you choose to work with them also.*

For more information on the dataset:

https://archive.ics.uci.edu/ml/datasets/Adult

Difficulty level: 3.

2. Dota2 Games Results Dataset

Predict which team will win a match of the Dota2 game, given the 5 heroes that each team selects to play. The input feature information is which heroes are selected by each team (out of 113 possible heroes in total), plus 3 other features. This is a 2-class problem.

Training and testing are posted on D2L; the training set labeled "SMALLER" has been down-sampled by factor of 3 (stratified), from the complete training set. The complete training set is also posted on D2L in case you choose to work with it also.* Please use the complete test set, as posted on D2L, for your test set. (There is no down-sampled version of the test set.)

For more information on the dataset:

https://archive.ics.uci.edu/ml/datasets/Dota2+Games+Results

Difficulty level: 2.5.

3. APS Failure at Scania Trucks Dataset

The dataset consists of data collected from heavy Scania trucks in everyday usage. The system in focus on the "Air Pressure system (APS) which generates pressurized air that are utilized in various functions in a truck, such as braking and gear changes." [1]. This is a 2-class problem, and the goal is to predict the failure of components in the APS system, given various inputs.

Training and testing are posted on D2L; the training set labeled "SMALLER" has been down-sampled by factor of 3 (stratified), from the complete training set. The complete training set is also posted on D2L in case you choose to work with it also.* Please use the complete test set, as posted on D2L, for your test set. (There is no down-sampled version of the test set.)

For more information on the dataset:

https://archive.ics.uci.edu/ml/datasets/APS+Failure+at+Scania+Trucks

Difficulty level: 5.

***Note on all 3 datasets (Adult, Dota2, and APS):** the down-sampled sets (labeled "SMALLER") are provided to save you some computation time. Everyone is required to use the SMALLER versions of the training sets (and also for the testing set in the case of the Adult dataset), and to report your results. For additional work or to improve your results further, you may optionally use the complete set(s) also and report your results on them as well.

Which datasets can you use?

Individual projects may use any one dataset (Adult, Dota2, or APS), or may optionally use both the Adult and Dota2 datasets.

Team projects must use the APS dataset, and may optionally use one other dataset also (Adult or Dota2).

Computer languages and available code

You may use Matlab, Python, or C/C++. (To use another language, please ask the TA or instructor first.)

Matlab and Python are recommended and supported.

You may use any toolbox or libraries you prefer.

For Matlab users, some common choices include PRTools and LIBSVM.

For Python users, scikit-learn and LIBSVM are common choices.

Be sure to state in your project report what languages and toolboxes/libraries you used; what you coded up yourself specifically for this class project; and any code from other sources.

Required elements

- **The items below give the minimal set of items you are required to include in your project, for each dataset you report on.** Note that you are welcome and encouraged to do more than the minimal required elements (for example, where you are required to use one method, you are welcome to try more than one method and compare the results). Doing more work will increase your workload score and might increase your interpretation score, and might improve your final system's performance.
- **Preprocessing**
    - Tip: you might find it easier to let Matlab (using the "import" button) or Python (using pandas) handle csv parsing.
    - Consider the feature types: numerical, ordered categorical, or unordered categorical. If your data has any non-numerical feature data, re-cast its representation as appropriate. [Tips given in Discussion 11.]
    - Missing data. If there is any missing data, decide how you will deal with it and implement your approach. (Applies to APS and Adult datasets.) [Tips given in Discussion 11.]
    - Normalization. Decide whether, and how, you will normalize the data, and which features will be normalized. [See Discussion 7.]
- **Compensation for unbalanced data**
    - If your dataset has unbalanced data (percent representation of data points from each class varies significantly from class to class), choose and implement a method to accommodate the imbalance. Describe the method you used in your final report. This applies to APS dataset, and can optionally be applied to Adult dataset.
- **Feature-space dimensionality adjustment.**
    - Use a method to try reducing and/or expanding the dimensionality, and to choose a good dimensionality.
- **Cross validation**
    - Use for choosing parameter values, comparing different models or classifiers, and/or for dimensionality adjustment.

- **Training and classification.**
  - Try at least 3 different classification techniques that we have covered in class; include both distribution-free and statistical classification. Beyond this, feel free to optionally try other methods (either from those we covered in class or other pattern recognition/machine learning methods).
- **Proper dataset (and subset) usage.**
  - Final test set, training set, validation sets, cross validation.
- **Interpretation.**
  - Interpret intermediate results and final results. Can you explain (or hypothesize reasons for) what you observe?
- **Performance evaluation and baseline comparison**
  - Use the measures given below.
  - Compare with a baseline system as given below.
- **Written final report**
  - Submit by the deadline.
  - More detail given below.

Evaluation of performance.

- For Adult dataset, provide the accuracy (percent correct) measure, the confusion matrix, and the F1 score. For purposes of the F1 score, define the minority class as the positive class.
- For Dota2 dataset, provide the accuracy (percent correct) measure and the confusion matrix.
- For APS dataset, provide the unnormalized weighted-error measure that is defined in the aps_failure_description.txt file on the UCI website; this provides a score for your system (lower scores are better). Also report the confusion matrix, and the F1 score.
  - Note that the "positive" class is the APS failure class; "type 1" error means a false positive, and "type 2" error means a false negative.

General Tips

1. Be careful to keep your final test set uncorrupted, by using it only to evaluate performance of your final system(s).
2. If you find the computation time too long using the provided down-sampled dataset(s), you could try the following: check that you are using the routines and code efficiently, consider using other classifiers, or down-sample the training dataset further to use a smaller *N* for your most repetitive work. In the latter case, once you have narrowed your options down, you can do some final choices or just your final training using the required training dataset(s) on D2L.

3. If possible, it can be helpful to consider number of learning variables, and number of constraints, as discussed in class. However, this is easier to evaluate for some classifiers than for others; and yet for others, such as SVM, it doesn't directly apply.
4. It's good to start out with a baseline system and its result to compare with. Baseline systems to use for each dataset are given below.
5. If using Python, consider using a util.py file for your import statements, then import util in each of your code files.

Dataset Tips and Baselines

1. For Adult dataset, other works have found only modest gains over a simple baseline. Don't be discouraged if you can only gain a few percent over baseline accuracy; that's very good for this dataset.

   If you want to add some additional work on this dataset, you could consider adding a confidence measure, and using a reject class for data that has low confidence; what percentage of data points need to be rejected to get a significantly higher accuracy rate on the unrejected data? Can you draw any conclusions or make any conjectures from this? Which of your systems does best when a modest amount of rejects are allowed? Why? Be sure to define your confidence measure in your report.
2. For Dota2 dataset, you may find some surprises in accuracy for different approaches.

   If you want to add to the amount of work on this dataset, one option is to explore further how to explain the performance for different approaches: conjecture an explanation, then think of some experiment that will verify or refute your conjecture, and implement the experiment.
3. Baseline for Adult dataset: a classifier that always decides the majority class.
4. Baseline for Dota2 dataset: A classifier that randomly chooses its output (tosses a fair coin to get each output).
5. Baseline for APS dataset: A classifier that always decides majority class. Note that this will give a very high accuracy, but a sub-par weighted-error score and a high number of false negatives (Type 2 errors).

Grading criteria

Please note that your project will not be graded like a homework assignment. There is no set of problems with pre-defined completion points, and for many aspects of your project there is no one correct answer. The project is open-ended, and the work you do is largely up to you. You will be graded on the following aspects:

- workload (effort in comparison with required elements, additional work beyond minimal required);
- difficulty of the problem (defined by the dataset(s) used – see below);

- approach (soundness and quality of work);
- performance of final system (as measured by stated performance metric for each dataset);
- analysis (understanding and interpretation);
- final report write-up (clarity, completeness, conciseness).

In each category above, you will get a score, and the weighted sum of scores will be your total project score.

For team projects, both members of a team will typically get the same score, but in some cases the score can vary among team members (depending on quantity and quality of each team member's effort).

Final report

In your final report you will describe the work that you have done, including the problem statement, your approach, your results, and your interpretation and understanding.

Note that where you describe the work of others, or include information taken from elsewhere, it must be cited and referenced as such; similarly, code that is taken from elsewhere must be cited in comments. Instructions for citing and referencing will be included with the final report instructions. Plagiarism (copying information from elsewhere without crediting the source) of text, figures, or code will cause substantial penalty.

For team projects, each team will submit one final report; the final report must also describe which tasks were done by which team member.

You will submit one pdf of your (or your team's) report, one pdf of all code, and a zip file with all your code as you ran it. We will provide instructions for where a team can turn in their report and code.

More detailed guidelines for the written report will be posted later.

Appendix: Difficulty-of-problem scores

Dataset(s)	**Individual project**	**Team project**
Dota2 only	6	N/A
Adult only	7	N/A
APS only	10	7
Dota2 and Adult	10	N/A
Dota2 and APS		8.5
Adult and APS		10

Note that the *workload* score is separate from the *difficulty-of-problem* score. The workload score is based on how much work you do for each dataset, relative to the minimum required elements for that dataset. The difficulty-of-problem score is based on the minimum required elements for each dataset.

For example, doing fewer datasets, with more work per dataset, can give you the same workload+difficulty total, as doing more datasets, with less work per dataset. You can use the above table as a guideline for your workload+difficulty score if you do only the required elements and nothing more, on each dataset.