NTUST, CSIE

Machine Learning (CS5087701), Fall 2025

Final Project

Guidelines

A. Each project should be done by a group of 1 to 3 students. Let me know if you have any requests to form a group of more than 3 students.

B. A project should include an oral presentation (15%), scheduled soon, and a written report (15%) due near the end of this semester (a confirmed deadline to be determined). A team needs to have each member on stage for the oral presentation. The written report should have fewer than six pages, with references included.

C. Project title due as soon as possible.

D. You can either use Python, Matlab, R, Julia or other languages to write your own code or use software/packages to help you do experiments and evaluations. That means writing code is not a must, but you get some credits by writing code by yourself.

E. Any further questions can be discussed with me.

Data-oriented or method-oriented projects

The candidate topics may be categorized as data-oriented or method-oriented ones. The data-oriented projects mean that you use one or more methods to analyze a given data set. The candidate datasets shall be given soon. In this case, the goal is to achieve as accurate a result as possible for the data set. Moreover, you may need to consider the following items in your discussion and analysis.

- Compare to the state-of-the-art approach: compare your result to others, or the state-of-the-art approach in particular. (Where can you find the state-of-the-art approach?) What makes your approach better (or worse) than others?
- Feature engineering: finding any possible set of features so that you may have an improved result. You can either extend the feature set or shrink the feature set (dimensionality reduction). What if you use deep learning? Do you need to find your own feature set?
- (Optional) Compare to end-to-end models, such as deep learners, to see if the deep learners really work better than those we have learned in class.
- Try your best to deliver a final statement about the dataset. The statement should provide some general guidelines for novices if they want to analyze the dataset.

After all, the larger the dataset, the more complicated the dataset you work on, the more credits you can receive for your project.

On the other hand, the method-oriented projects means that you focus one method or one group of methods to analyze several datasets (could be datasets of similar kind). In this case, the conclusion should be like: "Method A is better than Method B", and provide the reason why it is so. What could be your choice of method-oriented project is given in another document.

The general issues to study and discuss

You should emphasize the following items in your oral and written presentations.

A. Model Effectiveness

The prediction accuracy is not the only one to judge how effective a model is; however, it is the most common one. A typical approach to estimate the prediction accuracy is based on a cross-validation procedure, e.g., 10-fold, with several repeats, e.g., 5 different partitions of the original training set, for a total of 50 trials. Is there any other way to measure the model effectiveness? Clearly state the evaluation measure in your experiments and explain why this measure should be used in your case.

B. Model Complexity

How many attributes do you use in your model? You may try to use as few attributes as possible, given similar performance from your model. On the other hand, you should care about the model size. State clearly your own definition of model complexity. Generally speaking, more complex models may have a better chance of overfitting the data or may introduce more local optima. Make sure that you do not run into the above problems in your modeling procedure.

C. The Data Size

In general, the more data, the better for the model's effectiveness. Can you see this from your experiments? At the same time, can you use a smaller subset of the whole set to make your model perform as good as using the whole dataset?

D. Comparison to the baselines

Some candidates of baseline benchmark learning methods include *k*NN and naïve Bayes (if applicable). It is good that your model is at least more effective than those models. Moreover, you can compare your model to the state-of-the-art models. At the same time, you can also compare your model to other models in terms of model complexity. We can say a simple model is preferable to a complex model based on the above discussion.

E. Creative ideas

How is your model designed? Any creative preprocessing, design of experiment process is used? You may also share the ideas on why you chose particular modeling techniques.

Below are considered some advanced features with bonus:

F. How much domain knowledge is involved?
Usually domain knowledge helps us to improve the learning model. You can show how you use some domain knowledge to help you achieve better performance in your evaluation.

G. Can you operate an approach called *small-data* research, such as using a base model than fine-tuning the model based on a small dataset you have on the side?

H. Is the problem you face considered as a *highly nonlinear problem*? Any evidences such as removing some activation functions significantly hurts the model performance.

I. Can GPT or any *generative AI* useful in this project? Propose your ideas and integrate them into your system.

J. Similar to the previous item, can we consider *data augmentation* in the project? State details are necessary.

K. Can we play with different versions or different parts of the original data and study whether or not *transfer learning* is indeed helpful? In this case, we can rely on the knowledge learned from one part of data and contribute it to the learning of another part.

Datasets & Software

- WEKA: http://www.cs.waikato.ac.nz/ml/weka/
- Julia: https://julialang.org
- OpenCV: Open Source Computer Vision, http://opencv.org/
- MALLET: MAchine Learning for LanguagE Toolkit, http://mallet.cs.umass.edu/
- MLC++: Machine learning library in C++, http://www.sgi.com/tech/mlc/
- Stalib
  Data, software and news from the statistics community, http://lib.stat.cmu.edu
- GALIB: MIT GALib in C++ (http://lancet.mit.edu/ga)
- Kaggle: (https://www.kaggle.com/)
- UCI
  Machine Learning Data Repository UC Irvine,
  http://www.ics.uci.edu/~mlearn/MLRepository.html
- UCI KDD Archive: http://kdd.ics.uci.edu/summary.data.application.html
- Delve
  Data for Evaluating Learning in Valid Experiments,
  http://www.cs.utoronto.ca/~delve

Major journals & conferences

- PAMI (IEEE T. on Pattern Analysis and Machine Intelligence (PAMI))
- JMLR (Journal of Machine Learning Research)
- NIPS (Neural Information Processing Systems)
- ICML (International Conference on Machine Learning)
- ECML (European Conference on Machine Learning)
- UAI (Uncertainty in Artificial Intelligence)
- COLT (Computational Learning Theory)
- IJCAI (International Joint Conference on Artificial Intelligence)