ENSF 611 Project

Project proposal due date: March 24 at 11:59pm

Project due date: April 12 at 11:59pm

Total marks: 30 marks

Tasks:

1. Write a project proposal – submit in pdf format

2. Create a "README.md" file. It should include:

- Background on your code files

- How to run your code, guide to install any additional packages

- Results, interpretation, and reflection

3. Create your code file (.ipynb). You can include all your code in one file or you can create multiple files (modules)

Rubric:

1. Proposal (5 marks)
    a. Why: Question/Topic being investigated (1 mark)
    b. How: Plan of attack (2 marks)
    c. What: Dataset, models, framework, components (2 marks)
2. Final report (25 marks)
    a. Code (12 marks)
        i. Code runs (2 marks)
        ii. Code is explained well in the README.md file (4 marks)
        iii. Code is well organized and follows the format used in class (3 marks)
        iv. Code is well documented (comments, docustrings, etc.) (3 marks)
    b. Results (7 marks)
        i. Data is summarized and visualized effectively (4 marks)
        ii. Model selection is organized and well explained (3 marks)
    c. Interpretation (3 marks)
        i. As described in the proposal, was the question answered/topic investigated and how?
    d. Reflection (3 marks)
        i. Why did you select this problem to solve?
        ii. Were there any deviations from your proposal? Explain why (or why not).
        iii. What did you find difficult about this project? What did you find easy? What did you learn?

Example proposals:

Example 1

1. Why: Question being investigated
    a. I would like to understand better how the decision tree algorithm works.
2. How: Plan of attack
    a. In *insert reference here* a guide to implement decision trees from scratch is available. I will download the code, run with the tutorial dataset (very small) and then apply it to a real dataset (see below) and compare to scikit-learn. Furthermore, I will try and implement early stopping by introducting `max_depth` parameter. This is not part of the tutorial code and an extension.
3. What: Dataset, models, framework, components
    a. UCI dataset *insert url here* (13 features, 300 samples)
    b. Code from *insert code GitHub url*
    c. Scikit-learn DecisionTreeClassifier

Example 2

1. Why: Question being investigated
    a. I would like to see what it takes to participate in a Kaggle machine learning competition.
2. How: Plan of attack
    a. I have created an account on Kaggle and they recommend starting with the Titanic classification problem. I will follow these guidelines to prepare a notebook and submit it. If time permits, a second competition will be selected and prepared.
3. What: Dataset, models, framework, components
    a. Kaggle competition url *insert url*
    b. Scikit-learn classifiers: LogisticRegression, RandomForest, GradientBoosting, SVC
    c. Possibility to explore XGBoost library as an alternate classifier.

Example 3

1. Why: Question being investigated
    a. I am interested in learning more about putting a model into production.
2. How: Plan of attack
    a. Initial research showed, that mlflow *insert url* is a framework that allows for training and deploying machine learning models. On their website *insert url here* there are numerous tutorials. I am planning to follow the following:
        i. Tutorial 1 *add description*
        ii. Tutorial 2 *add description*
    b. Subsequently, I will adapt the tutorial code to create a website that allows entering Iris flower sepal and petal measurements, and the classifier displays the predicted type of Iris flower. To demonstrate this pipeline, only one classifier will be trained.

3. What: Dataset, models, framework, components
    a. mlflow *url here* with submodules *module 1* *module 2*
    b. mlflow to serve the model as a RESTapi
    c. Scikit-learn iris dataset
    d. Scikit-learn classifiers: LogisticRegression
    e. Flask framework to setup up webserver.
        i. The Flask server will provide the front-end for the website allowing the user to enter Iris measurements. Upon submission of the measurements, Flask will call the mlflow RESTapi to obtain the prediction results. Flask then displays the results as: predicted class, probability, and a sample image of the Iris class.

Project ideas:

Gaining more confidence

Use lab2 as a template and select a different dataset. Run through the steps that make sense for the data, add new steps if necessary, and show your solution.

Becoming competitive

Select a kaggle.com competition and try to put together a solution for the problem.

Implementing from scratch

Browse Data Sceince from Scratch for an algorithm of interest. Or use any other blog that guides through an implementation from scratch for your favourite algorithm. Demonstrate your working code on a different dataset than the guid/book used.

Machine learning history

Write about the history of machine learning and algorithms. Here it would be important to provide a new perspective. For example, can the sequence of algorithms reported be correlated with popularity of these algorithms?

Investigating a machine learning library or framework

If you find an interesting library or framework, you can follow the introduction tutorial and try to adapt it to new data.

Machine learning theory

Write about the mathematics used in one of your favourite algorithms. Here it would be important to connect any equation to code. The goal would be to make theory more accessible to others.