**Eth. AI, HW1, Explainability**

**Due Thursday, Week 3, 11:59PM**

**Goal:** Examine different machine learning models’ level of explainability.

**Resources:** Tutorials on python and sklearn library can be found both in [W3Schools](https://www.w3schools.com/python/default.asp) as well as [here (Python course Slides](https://ind657-my.sharepoint.com/:f:/g/personal/jrusert_pfw_edu/Ei_EBaJU-IJBgpwM2A7mmvUB-GrO9jUSvSq0CgMgwZgwYw?e=DJcAgX)). Also, sklearn has demonstrations of models and functions on the specific model pages as well (see below links).

**Models:** You will be working with 3 types of models for this homework, all from the sklearn python library:

[Decision Tree](https://scikit-learn.org/1.5/modules/tree.html#classification)

[Logistic Regression](https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html)

[Neural Networks (MLP)](https://scikit-learn.org/1.5/modules/neural_networks_supervised.html#classification)

This homework consists of 2 parts, training/testing each model and explaining the decisions of each model. Each model can be trained, tested, and explained withing its own function.

For each model:

1. Train a default model (default parameters) on the given train.csv file. This file consists of 5 columns: [“salary”, “age”, “credit score”, “debt”, “approved”]. The model should use the first 4 columns as features and the last column as the target (or label).
2. Test the model on the train data and the test data. Both scores should be recorded in the report.
3. Find which features each model relies on the most to make its decision. (See hints below)

Each model should have its own function named “MODEL\_Explain(train\_file, test\_file)” where MODEL is substituted by DT, LR or MLP for decision tree, logistic regression, and neural networks respectively.

**Training Hints:**

* Training sklearn classifiers are simple, see the included sklearn resources.
* For the neural network and logistic regression you’ll want to normalize the features (see resources or ["sklearn normalize").](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.normalize.html)

**Explainability Hints:**

* Trained decision trees can be printed using [tree.plot\_tree(clf)](https://scikit-learn.org/1.5/modules/tree.html#classification)
* Logistic regression’s weights for each feature can be printed using .coef\_
* As noted in class, Neural Networks have too many hidden layers/weights to easily help with explainability. You’ll have to leverage another strategy discussed in class (see slides or instructor if unable to think of any).

**To Turn In:**

* Python file (**USERNAME\_HW1.py**.) which contains at least 3 functions (one for each noted ML method) which train/test/explain each model.
* Report which contains:
  + A table which contains the train accuracy and test accuracy for each model.
  + Discussion on how you found which features helped each model (for every model).
  + Which features helped each model the most.
  + An image of your trained decision tree.
  + Discussion of issues or difficulties encountered.
  + A note of any AI tools used in coding or writing.

**Additional Rules (MUST BE FOLLOWED):**

1. All Homeworks should follow the overall [homework guidelines](https://ind657-my.sharepoint.com/:p:/g/personal/jrusert_pfw_edu/EZxf1ZsRXjBEkQLNhEhdTgUBt6U64KiT1DJ1YHtkARgKwA?e=bwHrbf)
2. The code should be written in python 3.
3. The functions must follow the naming and number of arguments as demonstrated.
4. You should make your code modular to the different steps. (You may have more functions to help your main functions)
5. You should be adding comments to document your code. **If I can’t understand why you perform an action, then I can’t credit you for performing that action.**
6. The report should be readable and reference your code, **without explicitly including code.**
7. You should include your name and homework number in the comments at the beginning of the python file.

**Report**

The reports for the homeworks are necessary to communicate your learning and thinking through of the material. Examples of good reports can be found on brightspace under Additional Resources/Guides. Note that your report style may differ, but it is a good reference to start with.

**Grading**

Assignment will be graded as follows:

|  |  |
| --- | --- |
| **Description** | **Points** |
| **(.py file)** Train/Test Functionality of all 3 Models | 10 |
| **(.py file)** Explanation Functionality of all 3 Models | 10 |
| **(Report)** Train/Test scores in report | 10 |
| **(Report)** Discussion of Explainability in Report | 10 |
| **(Report)** Other Report Requirements | 5 |
| **(.py file)** Documentation/Code (Comments, functions, etc) | 5 |
| **Total:** | **50** |

* **If the code does not run, it cannot be graded well.** (Many points can be lost if the code cannot be run, as I will not be able to fully test the implementation of the functions).
* **Breaking of the additional rules can result in applied penalties.** (Always make sure you are checking against the rules)

**Suggestions**

* **Documentation is key for showing your effort in this homework.** Make sure you are noting why you make certain decisions all throughout your code.
* The slides for previous classes are posted, so please refer to these and the book for ideas during implementation.
* Start simple, build up complexity. You should always make sure your new ideas being added do not cause your program to crash. So starting simple is the best way to a) maintain the ability to keep your code running, b) add in comments for documentation and thought process as you add more code.
* Work through the homework yourself, rather than sharing ideas (especially not code) with other students. **As a reminder, plagiarism (or sharing) of code is strictly prohibited.** This assignment is complex enough that significant overlap between students will be suspicious.
* If you have not worked with python before, w3schools can help you translate your previous coding experience to python (<https://www.w3schools.com/python/default.asp>)
* Stop by office hours to discuss ideas. I am always happy to help you think through your process!