

README

Name : Tushar Kumar

NetId : tusharku

Collaborators : None

December 13, 2018

1 Introduction

This project, which was implemented by myself, and **did not have any collaboration from anybody else**, implements two learning algorithms , Decision Trees and Neural Networks. This project was undertaken as part of the graduate course(CSC 442) at University of Rochester.

2 Technology Used

- Python as programming language(3.7.0)
- PyCharm used as IDE
- [Draw.io](https://draw.io) for creating design artifacts

3 Project dependencies

- Python 3.7.0
- numpy 1.15.1

4 Running the project

- Download the **Learning-tusharku.zip** file(which you would have if you are reading this file.
- Unzip the file to get the Learning-tusharku folder
- In case you dont have numpy please use the below mentioned command to download the requirement

```
cd Learning-tusharku
pip3 install -r requirements.txt
```

- Run the below mentioned command . Please mind the line break created because the command being of greater length than width of the page.

```
cd Learning-tusharku

python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ aima-restaurant --train_test_split 0 --model dtree --seed 11
↪ --splitting_criteria entropy --print_tree
```

This, would run the decision tree algorithm on the AIMA restaurant example with entropy as the splitting criteria and print the learned tree.

5 Executing Models

I would now provide a detailed description of all the command line arguments and how to use them to evaluate the different project requirements and run the different models These are the commandline arguments with their description and possible values that can be provided in order to run the models.

- **dataset** : This represents dataset on which you want to perform the learning on.
Possible Values : aima-restaurant(for aima-restaurant example), iris(for iris dataset), connect-4(for learning on connect-4 dataset)
Default Value : iris
- **train_test_split** : What is the split you want to use to train the model and test.
Possible Values : float (between 0 and 1)
Default Value : 0.2(so a 80/20 split)
- **model** : Which model you want to use for learning
Possible Values : dtree(for decision tree) or nnet(for neural net)
Default Value : dtree
- **log_interval** : After how many epochs do you wish to log the training loss and testing accuracy for neural network
Possible Value : int
Default Value : 5
- **-batch_size** : What batch size you want to use
Possible Value : int
Default Value : 1

- **epochs** : Number of epochs you want to run neural net for
Possible Values : Integer
Default Value : 50
- **lr** : Learning rate you want to use for Neural network
Possible Values : float
Default Value : 0.05
- **seed** : Seed value you wish to use
Possible Values : int
Default Value : 11
- **max_depth** : Maximum depth the decision tree should be allowed to grow to
Possible Values : int
Default Value : None
- **splitting_criteria** : What criteria to use for splitting the branches in decision trees
Possible Values : entropy or gini(for gini-impurity)
Default Value : entropy
- **print_tree** : If present in the command the program will print the tree learned
Default Value : Program will not print the tree

5.1 Performing Learning on Aima-Restaurant

Here are some sample scenarios with learning on the restaurant example.

- **Learning decision trees with entropy as splitting criteria and printing tree**

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ aima-restaurant --train_test_split 0 --model dtree --seed 11
↪ --splitting_criteria entropy --print_tree
```

- **Learning decision trees with gini impurity as splitting criteria and printing tree**

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ aima-restaurant --train_test_split 0 --model dtree --seed 11
↪ --splitting_criteria gini --print_tree
```

5.2 Performing Learning on Iris

- Learning decision trees with train test split of 80/20 , entropy as splitting criteria, no depth restrictions and printing tree

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ iris --train_test_split 0.2 --model dtree --seed 11
↪ --splitting_criteria entropy --print_tree
```

- Learning decision trees with train test split of 80/20 , entropy as splitting criteria, max depth allowed 3 and printing tree

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ iris --train_test_split 0.2 --model dtree --seed 11 --max_depth 3
↪ --splitting_criteria entropy --print_tree
```

- Learning decision trees with train test split of 80/20 , gini impurity as splitting criteria, max depth allowed 3 and printing tree

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ iris --train_test_split 0.2 --model dtree --seed 11 --max_depth 3
↪ --splitting_criteria gini --print_tree
```

- Learning neural network with train test split of 80/20 , learning rate of 0.1, batch size of 1 and logging accuracy and loss at every 5 epochs for 100 epochs with seed of 11

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ iris --train_test_split 0.2 --model nnet --log_interval 5
↪ --batch_size 1 --epochs 100 --lr 0.1 --seed 11
```

5.3 Performing Learning on Connect-4 dataset

- Learning decision trees with train test split of 80/20 , entropy as splitting criteria, no depth restrictions

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ connect-4 --train_test_split 0.2 --model dtree --seed 11
↪ --splitting_criteria entropy
```

- Learning decision trees with train test split of 80/20 , entropy as splitting criteria, max depth allowed 11

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ connect-4 --train_test_split 0.2 --model dtree --seed 11
↪ --max_depth 11 --splitting_criteria entropy
```

- Learning decision trees with train test split of 80/20 , gini impurity as splitting criteria, max depth allowed 11

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ connect-4 --train_test_split 0.2 --model dtree --seed 11
↪ --max_depth 11 --splitting_criteria gini
```

- Learning neural network with train test split of 80/20 , learning rate of 0.05, batch size of 1 and logging accuracy and loss at every 5 epochs for 50 epochs with seed of 11

```
python3 -m com.uofr.course.csc442.hw.hw4.models.runModels --dataset
↪ connect-4 --train_test_split 0.2 --model nnet --log_interval 5
↪ --batch_size 1 --epochs 50 --lr 0.05 --seed 11
```