**NNU-Machine-Learning-2020**

Developed by Aleesha Chavez, Garrisen Cizmich, and Enoch Levandovsky

**Folder Contents**

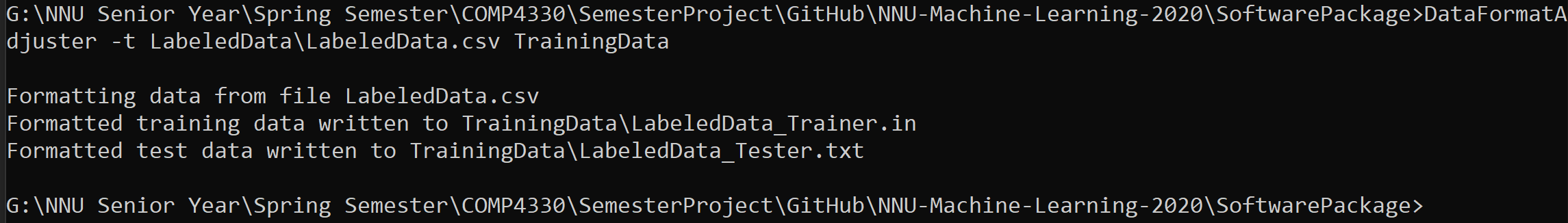
In this folder, you will find a DataFormatAdjuster program and a DecisionTree program, both meant to be run from the command line. You will also find sample labeled data in the LabeledData folder. All source code for both programs and Visual Studio solution files are in the SourceCode folder.

**DataFormatAdjuster Program: -t**

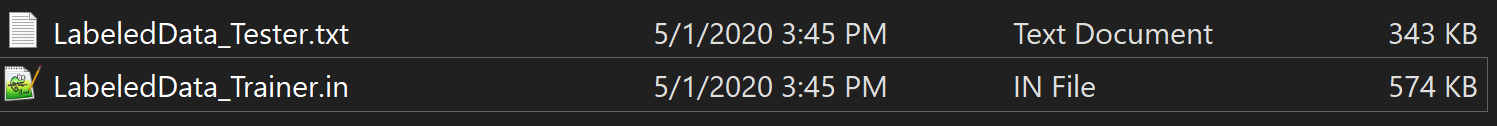
To generate training data and associated test data from labeled data, run the DataFormatAdjuster program with the -t option. Here is an example using the LabeledData.csv file.

Statement: "DataFormatAdjuster -t LabeledData\LabeledData.csv TrainingData"

Successful results:



TrainingData Folder contents:



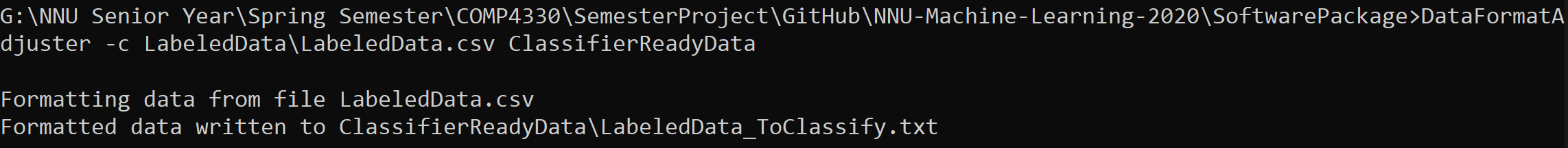
This statement will generate two files in the TrainingData folder: LabeledData\_Trainer.in and LabeledData\_Tester.txt. The .in file contains training data in the format expected by the DecisionTree program, while the .txt file contains mutually exclusive test data that is ready to classify.

**DataFormatAdjuster Program: -c**

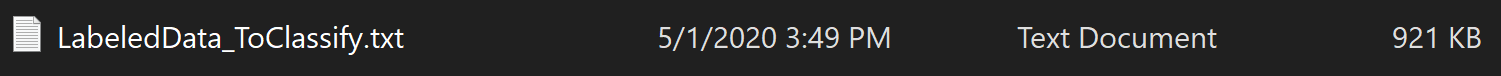
To generate formatted data to classify without training data, run DataFormatAdjuster with the -c option. Here is an example using the LabeledData.csv file.

Statement: "DataFormatAdjuster -c LabeledData\LabeledData.csv ClassifierReadyData"

Successful results:



ClassifierReadyData contents:



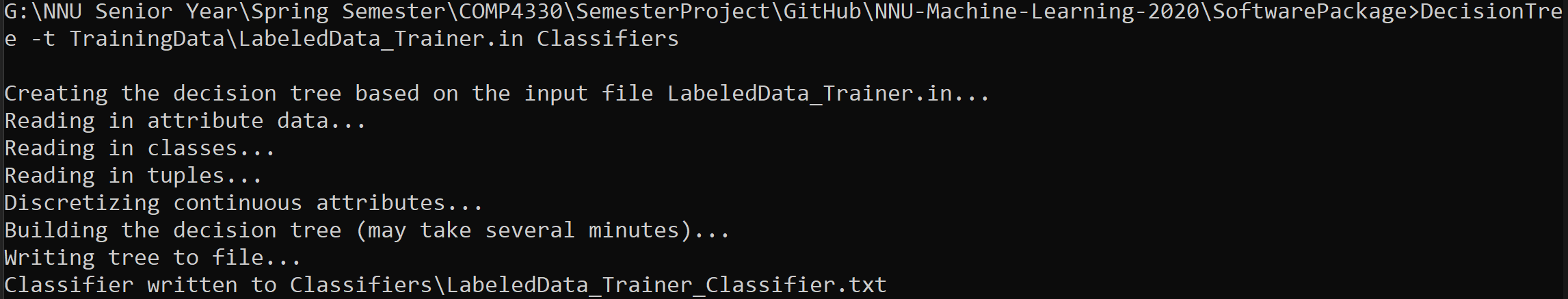
This statement will generate one file in the ClassifierReadyData folder: LabeledData\_ToClassify.txt. This file contains every tuple from the input file in the format needed for the DecisionTree program to classify it.

**DecisionTree Program: -t**

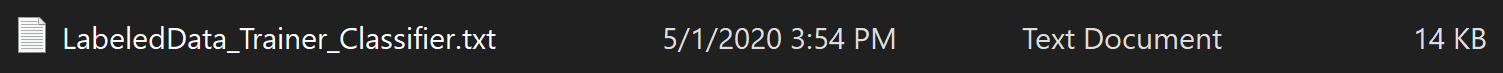
To train an ID3 decision tree model, run DecisionTree with the -t option. Here is an example using the training data from DataFormatAdjuster -t.

Statement: "DecisionTree -t TrainingData\LabeledData\_Trainer.in Classifiers"

Successful results:



Classifiers contents:



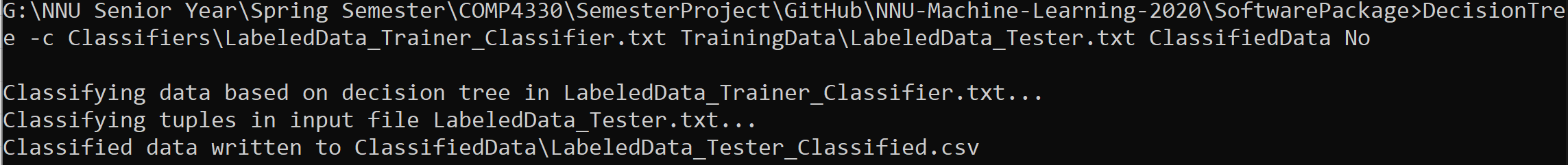
This statement will generate a classifier file in the Classifiers folder named "LabeledData\_Trainer\_Classifier.txt." This classifier file can be used with the DecisionTree program to classify tuples in files generated by the DataFormatAdjuster program.

**DecisionTree Program: -c**

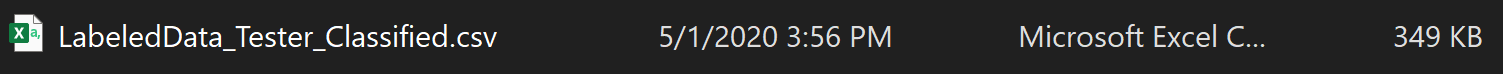
To classify tuples with an existing classifier, run DecisionTree with the -c option. Here is an example using the testing data from DataFormatAdjuster -t and the classifier from DecisionTree -t.

Statement: "DecisionTree -c Classifiers\LabeledData\_Trainer\_Classifier.txt TrainingData\LabeledData\_Tester.txt ClassifiedData No"

Successful results:



ClassifiedData contents:



This statement will generate a file containing each tuple and its associated class in the ClassifiedData folder named "LabeledData\_Tester\_Classified.csv." The Ans column in this file is the classifier’s decision on each tuple. The final argument in the statement above is the required default class if the classifier cannot otherwise classify a tuple.

**Further Information**

For more information on each of the programs or on a specific option, run either program without any arguments from the command line or run the program with only the option specified (e.g. "DecisionTree" or "DataFormatAdjuster -c").