Homework #5

Objective:

This homework's objective is to implement a decision tree model to predict mushroom type (edible vs. poisonous).

Details:

For this assignment you will be using the data obtained from The Audubon Society Field Guide to North American Mushrooms (1981) publication. This dataset is publicly available from University of California Irvine (UCI) Machine learning repository as well as kaggle dataset repository. This dataset contains information about various mushroom characteristics from 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom. You can find more information about this dataset here:
https://www.kaggle.com/uciml/mushroom-classification

Use input file homework5_input_data.csv for this assignment. The rows of this csv-formatted file are different mushroom samples and the columns are various input variables for this dataset. The first column titled "class" is the output variable you will be predicting based on all the other input variables.

Implement a decision tree model to predict the mushroom class using all the other variables. This will be a binary classifier because there are only 2 classes available in this dataset.

Because the input variables are categorical and do not contain numeric values scikitlearn libraries will give an error if you try to use raw data for the classification task. You have to convert these categorical variables to numeric values. There are several options for doing this:

1. You can map each category to a numeric value and replace each categorical value with its mapped numeric value.
2. You can create new binary variable column for each category available in the original column. In this scenario you will end up with as many new columns as the number of categories in the original column and the values will be 0 or 1 to indicate if the observation contains the particular category value in the original column.

Here is a useful post to help you understand options for categorical variable encoding:
https://pbpython.com/categorical-encoding.html

Homework # 5

Here is an example for how you might choose to do the conversion. Let's say your raw data is contained in the pandas dataframe called X. You can use the following code to convert categorical data to numeric data:

```
X_numeric = pd.get_dummies(X, columns=X.columns, prefix=X.columns)
X_numeric.head()
```

	cap-shape_b	cap-shape_c	cap-shape_f	cap-shape_k	cap-shape_s	cap-shape_x	cap-surface_f	cap-surface_g	cap-surface_s	cap-surface_y	...	population_s	po
0	0	0	0	0	0	1	0	0	1	0	...	1	
1	0	0	0	0	0	1	0	0	1	0	...	0	
2	1	0	0	0	0	0	0	0	1	0	...	0	
3	0	0	0	0	0	1	0	0	0	1	...	1	
4	0	0	0	0	0	1	0	0	1	0	...	0	

5 rows × 117 columns

You will have to use X_numeric to perform training-test set split.

You can use the code from my notebook examples as a reference to help you get started:

- DecisionTrees.Breast.ipynb

<u>Your submission should include the following:</u>

1. Load the dataset.
2. Convert categorical variable to numeric either using the method described above or your own implementation.
3. Break the data into the training and test datasets.
4. Train a decision tree model (DecisionTreeClassifier) to predict the class variable. Report (print out) 5-fold cross-validation accuracies (for all 5 folds as well as the mean accuracy).
5. Train a decision tree model on all the training data and report prediction accuracy on the test data.
6. Plot two confusion matrices for test set predictions (one non-normalized and one normalized). You can choose to use the same implementation of plotting a confusion matrices as I showed in my examples or include a different implementation. If you use code examples from the internet then make sure to site your sources in your notebook.