Name: Kushal Sharma
Email: kushals@usc.edu
Course: INF 552 Summer 2020
Assignment: HomeWork1 (Decision Tree)

Data Structure:

Following data structures are used for different purposes:

1. Pandas Data frame –
   i. to load the data set from the location.
   ii. to delimit the data.
   iii. to split the data.
   iv. to set appropriate column name.
2. Numpy Functions –
   i. to find out unique values and their corresponding count by column wise(numpy.unique function).
   ii. to find the index of max value (numpy.argmax).
   iii. to take log base 2 (numpy.$\log_2$ functions).
   iv. to sum up all the elements of array (numpy.sum function).
3. Dictionary –
   i. to store the decision tree.
   ii. to pass the query for which prediction needs to be derived.
4. Tuple -
   i. to store the unique value and count from numpy.unique() function.

Code Level Optimization:

1. Use of pandas dataframe and numpy functions help in faster execution.
2. When calculating information gain for specific attribute, only that attribute and target variable data is passed to 'infogain' method which help in consuming lesser memory.
3. In 'decisiontree' method, if two or more than two attributes have same information gain, it picks the one having minimum no. of branches. In given dataset, it didn't make any difference, but in other it can.
4. In 'decisiontree' method, in termination condition if there are no attributes left to be expanded further, the highest class of previous dataset is returned.

Challenges:

1. What could be termination condition.
2. Which data structure tree can be stored
3. Which data structure query can be passed

Decision Tree:

Below decision tree is produced in pprint of dictionary format. I have used following color coding to describe the nodes.

Root node – red
Intermediate node – blue
Leaf node - green

```
{'Occupied': {'High': {'Location': {'City-Center': array(['Yes'], dtype=object),
                                    'German-Colony': array(['No'], dtype=object),
                                    'Mahane-Yehuda': array(['Yes'], dtype=object),
                                    'Talpiot': array(['No'], dtype=object)}},
              'Low': {'Location': {'City-Center': {'Price': {'Cheap': array(['No'], dtype=object),
                                                             'Normal': {'Music': {'Quiet': {'VIP': {'No': {'Favorite Beer':
{'No': 'No'}}}}}}}},
                                   'Ein-Karem': {'Price': {'Cheap': array(['Yes'], dtype=object),
                                                           'Normal': array(['No'], dtype=object)}},
                                   'Mahane-Yehuda': array(['No'], dtype=object),
                                   'Talpiot': array(['No'], dtype=object)}},
              'Moderate': {'Location': {'City-Center': array(['Yes'], dtype=object),
                                        'Ein-Karem': array(['Yes'], dtype=object),
                                        'German-Colony': {'VIP': {'No': array(['No'], dtype=object),
                                                                  'Yes': array(['Yes'], dtype=object)}},
                                        'Mahane-Yehuda': array(['Yes'], dtype=object),
                                        'Talpiot': {'Price': {'Cheap': array(['No'], dtype=object),
                                                              'Normal': array(['Yes'], dtype=object)}}}}}}
```

<u>Query in dictionary form :</u>

{'Occupied': 'Moderate', 'Price': 'Cheap', 'Music': 'Loud', 'Location': 'City-Center', 'VIP': 'No', 'Favorite Beer': 'No'}

<u>Prediction:</u>

Yes

Part 2:

DecisionTreeClassifier from sklearn.tree is widely used library function for decision tree classification problems.

```
from sklearn.tree import DecisionTreeClassifier
```

It requires the feature data and target data in separate array variables. Data can be split into training and testing data by using train_test_split library function.

```
from sklearn.model_selection import train_test_split
```

Target values are require to be in int or float before they can be passed to fit(). String can’t be passed to model fit(). So if there are strings then they are required to convert into encoding such as label encoding, one hot encoding. If data is in dataframe, then pandas.get_dummies() can also be used to convert string in the features into encoded form.

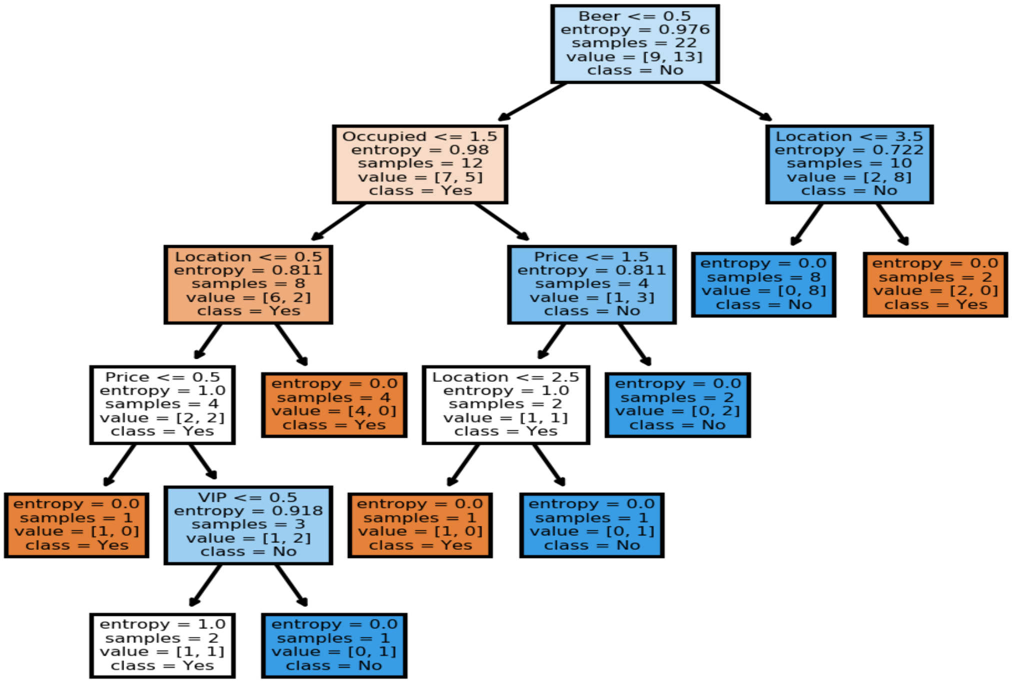
It is also easy to use techniques to print decision tree such as matplotlb.pyplot, graphviz, sklearn.tree.plot_tree.

Below is the code to use sklearn library functions on the given data. Corresponding decision tree is also displayed.

```
import matplotlib.pyplot as plt
from sklearn import tree
from sklearn import preprocessing

## SKLEARN Training and Testing
    le = preprocessing.LabelEncoder()
    dfData = dataFrame.values
    for i in range(len(dataFrame.columns)):
        dfData[:, i] = le.fit_transform(dfData[:, i])
    X = dfData[:,:-1]
    y = dfData[:,-1]
    y = y.astype('int')
    # X_train, X_test, y_train, y_test =
train_test_split(X, y, random_state=0)
    clf = tree.DecisionTreeClassifier(criterion="entropy")
    # print(clf, type(clf))
    model = clf.fit(X, y)
    # print(model, type(model))
    feature_name = dataFrame.columns.values[:-1]
    target_name = ['Yes', 'No']
    fig, axes = plt.subplots(nrows=1, ncols=1, figsize=(4,
4), dpi=300)
```

```
tree.plot_tree(model, feature_names=feature_name,
               class_names=target_name,
               filled = True);
fig.savefig('dtree.png')
```



Part 3:

Decision trees are widely applied in multiple applications successfully. Below are some application areas:

1. Business Management - Decision trees are used to extract useful information from databases in domain of business and management.
2. Customer Relationship Management – Decision trees are used to find the customer’s needs and preferences based on customer’s record of sales or item access data specially online service.
3. Fraudulent Statement Detection – Decision tree could correctly classify all non-fraud cases and 92% of fraud cases in one of the case study on financial statements of 76 firms which deals with large number of variables in addition to hidden information pertaining to relationship among those variables.
4. Fault Diagnosis – Engineers measure the vibration and acoustic emission(AE) signals emanated from the rotary machine to detect the existence of faulty bearing. However, measurement involves a number of variables, some of which may be less relevant to the investigation. Decision tree help to remove irrelevant variables for purpose of feature selection.