**MACHINE LEARNING**

A machine learning project involves several steps.

**Steps:**

1. Importing Data
2. Clean the Data
3. Split the data into training & test sets
4. Create a Model
5. Train the Model
6. Make Predictions
7. Evaluate & Improve

**Jupiter Shortcuts:**

* python -m notebook / jupyter notebook ( Command to open Jupiter notebook)
* Press ‘Esc’ (Escape Key) for Command mode (blue)/Edit mode (green)
* Press ‘H’ in command mode – For Keyboard shortcuts
* Press ‘B’ in command mode – It opens new cell below
* Press ‘A’ in command mode – It opens new cell Above
* Ctrl + Enter – Shortcut for Run
* Press ‘Tab’ key – For Intellisence
* Press ‘Shift+Tab’ on methos – It shows the details of that method
* ‘Ctrl + /’ – Shortcut comment/Uncomment both

The CSV file used in this tutorial:

Music dataset - [https://bit.ly/3muqqta](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbHFyWngxWUJXUFNmUTkxVmNvYzJ0SURWQmpkUXxBQ3Jtc0tudDc5T0ZRLTF5V2h0RFlfTXN1VHQ4YkdSVUxEZGlSRlBxRkUzQUVRcWR1U01kMk0tTGJOalQ2aVpSR1BBRjlhYmNGSElQRW5WQXlUbG1CbDlyREdMVXF0SlgyT0Y1aGZWdmVtRzdrV0xnYmdUOUdmWQ&q=https%3A%2F%2Fbit.ly%2F3muqqta)

Video games dataset - <https://www.kaggle.com/gregorut/videogamesales?select=vgsales.csv>

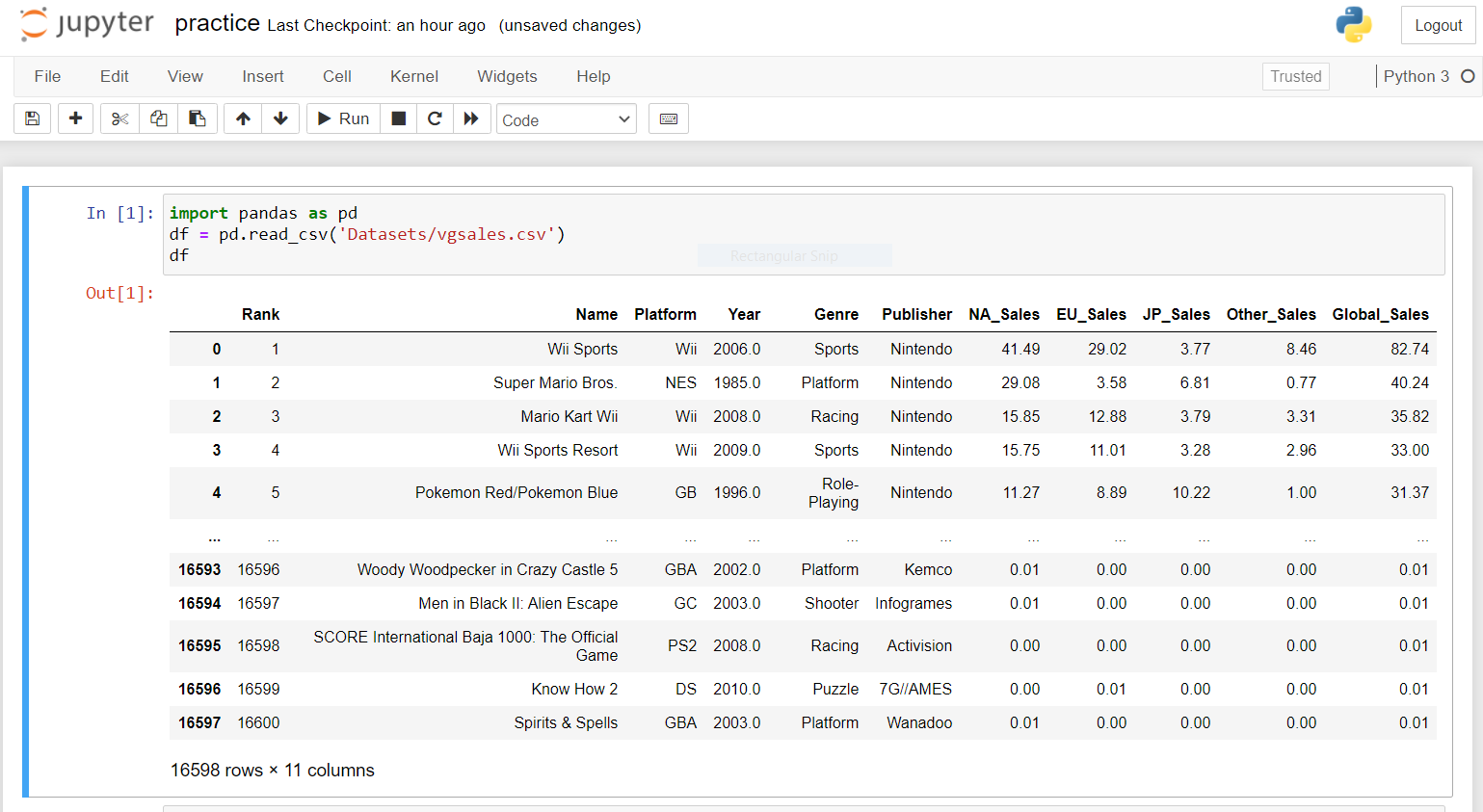
**1.Importing Data:** Imports data from CSV file using ‘pandas’

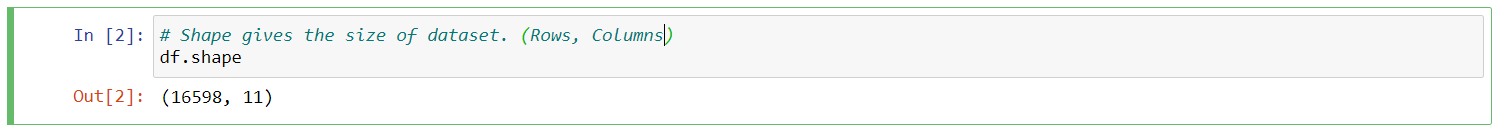
**Snippet:**

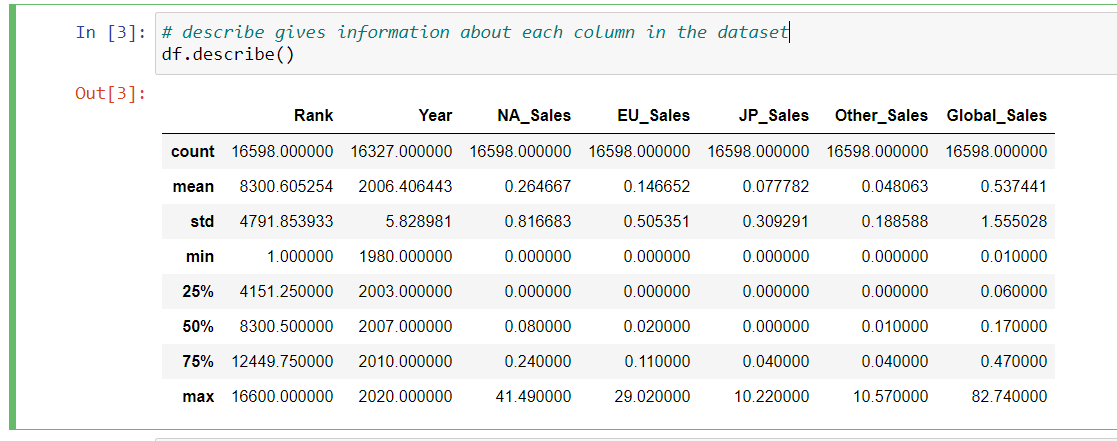
*import pandas as pd*

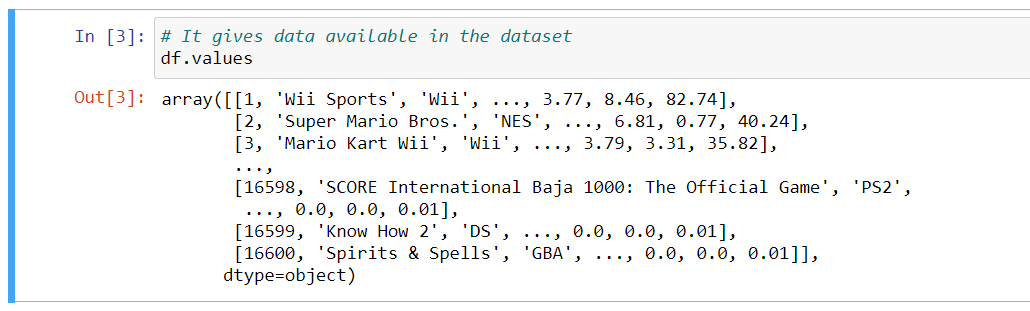
*df = pd.read\_csv('Datasets/vgsales.csv')*

*df*





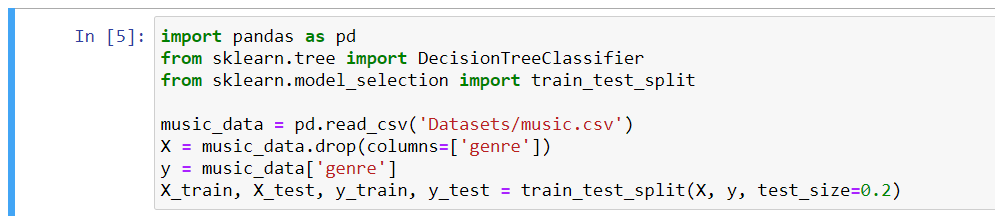




**2. Clean the Data:** If there are any junk/empty cells clean the data.

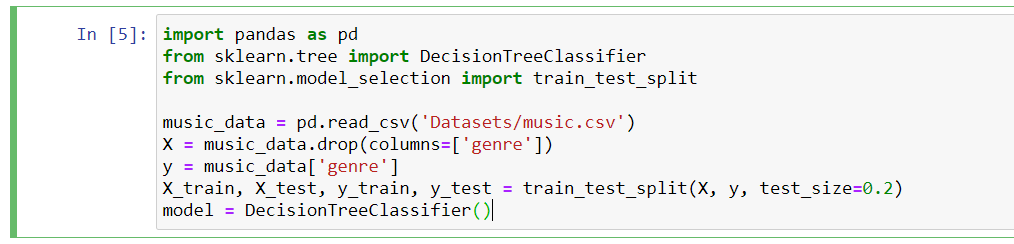
**3. Split the data into training & test sets:**

Split the data into both training & test. For getting accurate/best results we need to split the dataset into 80% as training and 20% test data.



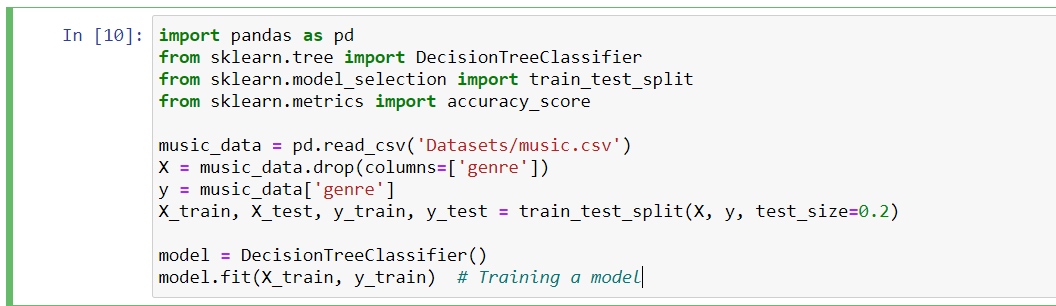
**4. Create a Model:**

There are many models available. Here we are using ‘***DecisionTreeClassifier’***



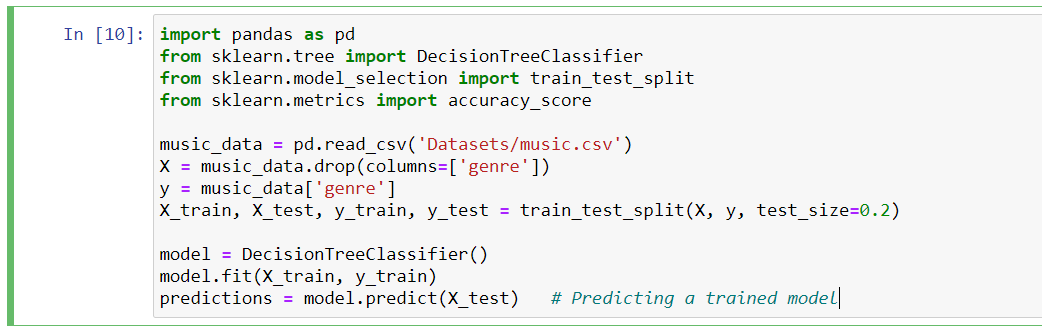
**5. Train the Model:**

Here we are training the model using ***X\_train*** (input training data, which is 80%)**, *y\_train*** (Output/target data)



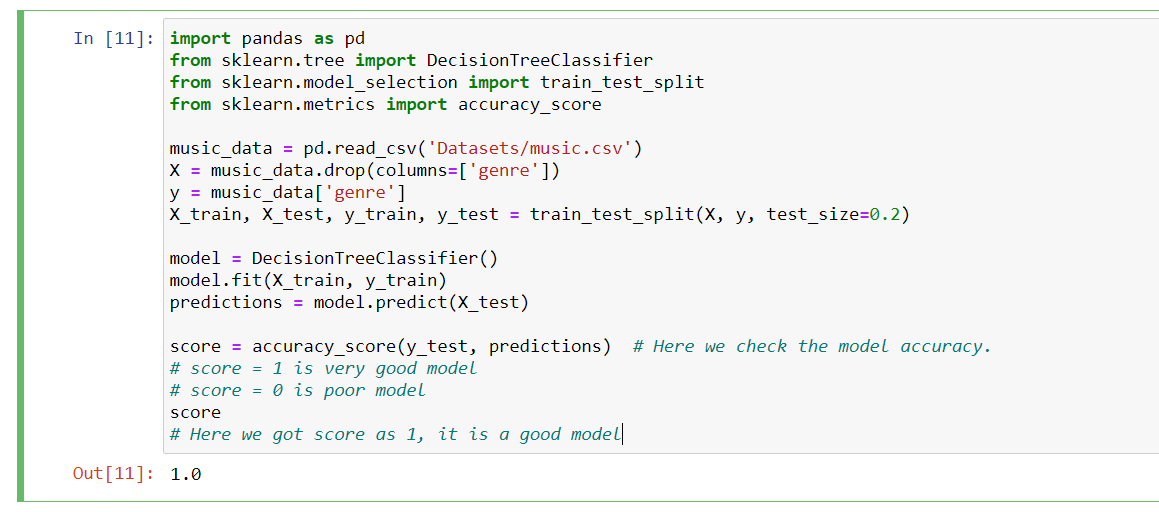
**6. Make Predictions:**

Here we are predicting using ‘***X\_test’***, which is testing data (20%). It is different from trained data.

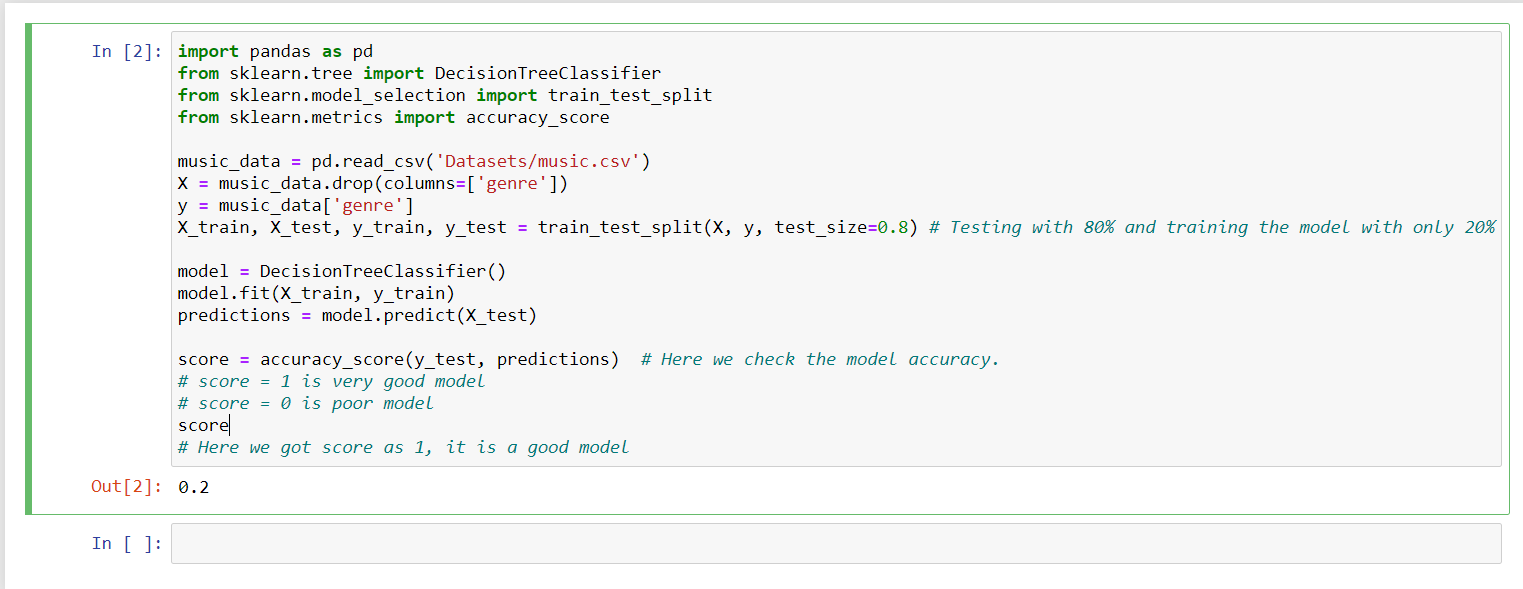


**7. Evaluate & Improve:**

Below we can see the predicted accuracy, which is 1. So, it is 100% accurate.



* If you change the test size to 80% ***(‘test\_size=0.8’)*** , see the prediction accuracy is very low (Here it is 0.2). So, divide the data as 80% for training and 20% for testing.
* If we train the model with more data, the accuracy will be more.



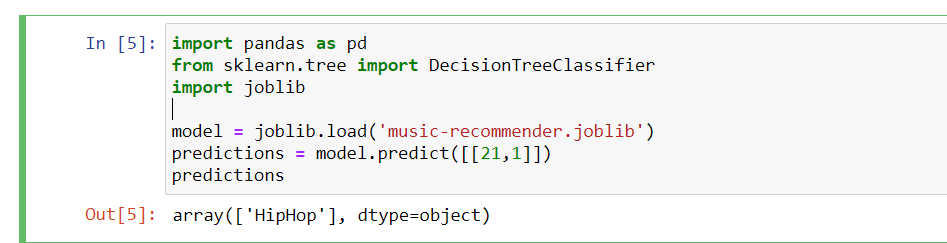
Note: If you run the model multiple times, accuracy will not be same. It may change every time.

**Persisting/Reusing Models:**

* Joblib object has methods for saving & loading models.
* Instead of creating & training a model every time, we can save the trained model using ***joblib*** and use it whenever required using ***load*** method.
* See below screen, model was saved as **‘music-recommender.joblib’** using ***dump*** method.



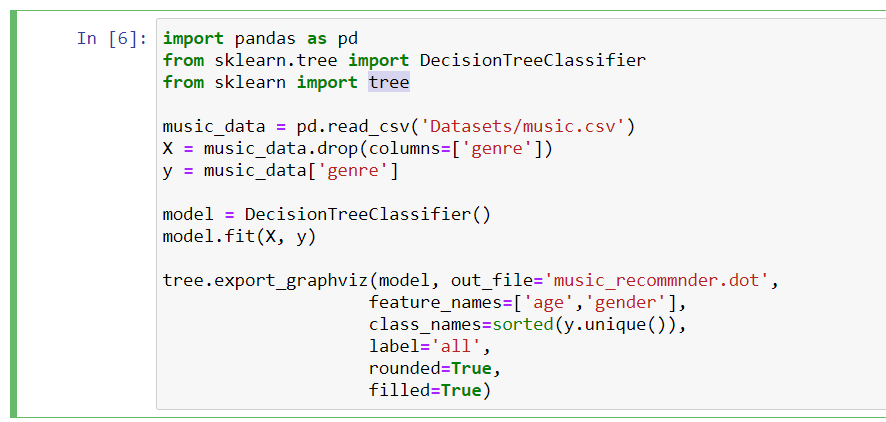
* In below snippet, we are loading the saved model using ***load*** method and predicted with input.



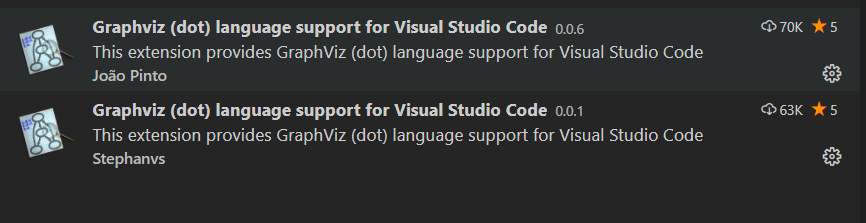
**Visualizing a Decision tree:**

***Tree*** object has a method for exporting our decision tree in graphical format using below code.

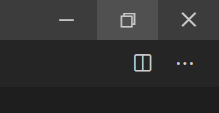
* Below code will be saved as ‘***music\_recommnder.dot***’



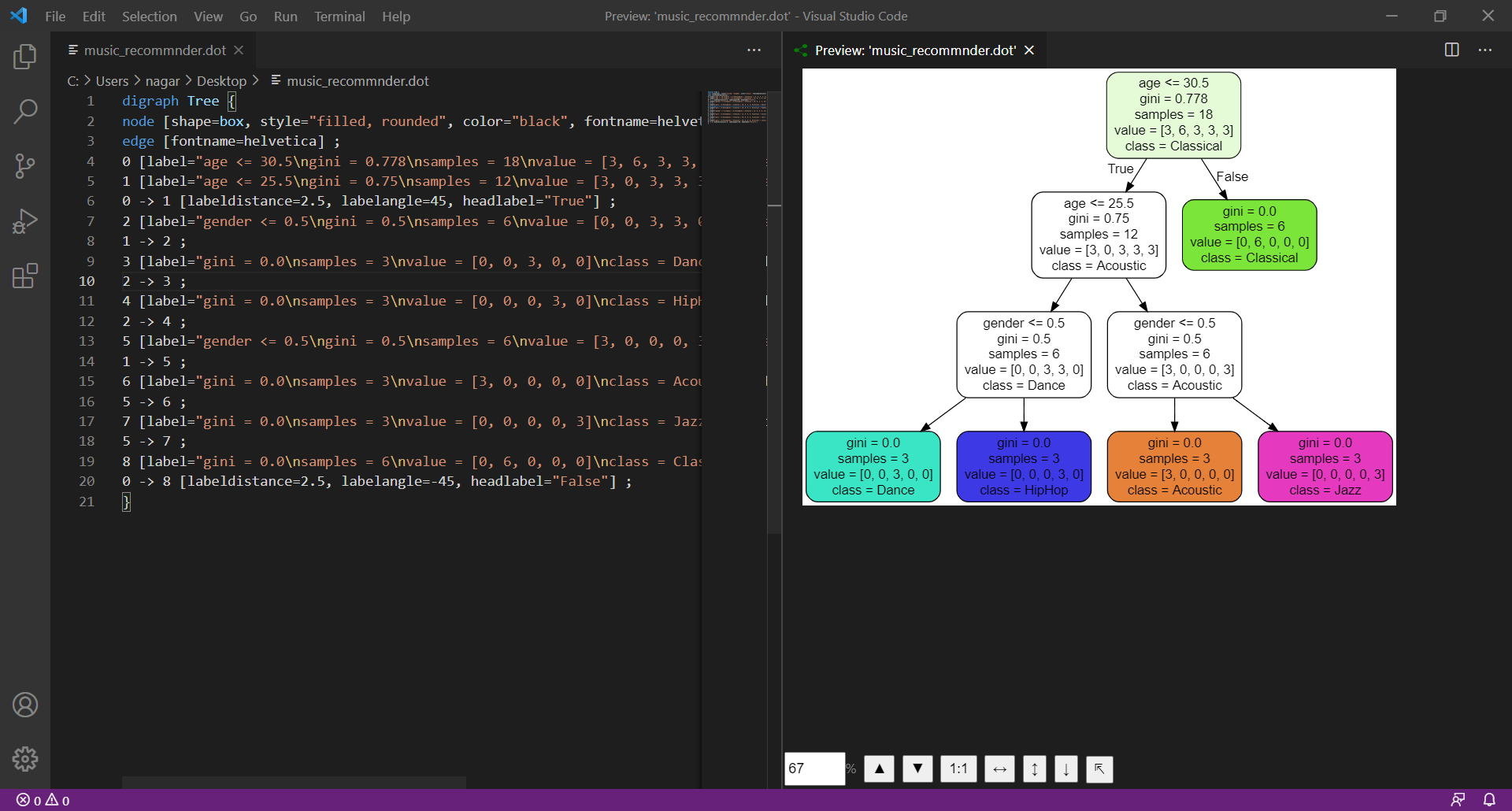
* Open that file (***music\_recommnder.dot***) in vscode. Before that install any one of the below extensions (For me stephanvs didn’t worked)



* Next open ‘Open Preview to the Side’, it is available on the top right corner under 3 dots (…).



* Now, you can see the model visualization like below



**Reference:** <https://www.youtube.com/watch?v=7eh4d6sabA0> ( By Mosh Hamedani)