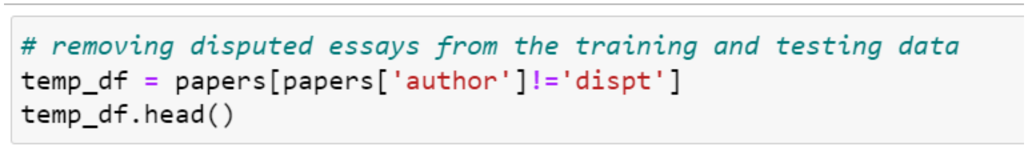
Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu

HW 05 – DECISION TREES

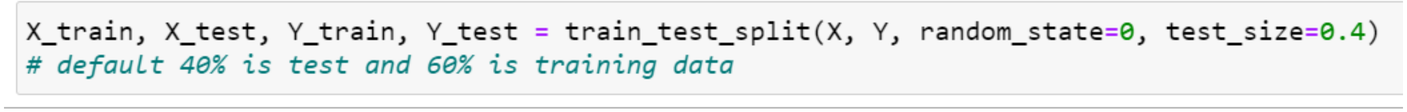
DATA PREPARATION

1. Removed all the disputed essays from the data set

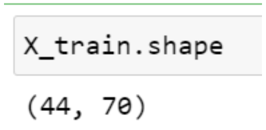


```
# removing disputed essays from the training and testing data
temp_df = papers[papers['author']!='dispt']
temp_df.head()
```

2. Essays (without disputed authorship) are divided as **training and testing dataset with 60% and 40%** proportion respectively

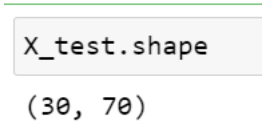


```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, random_state=0, test_size=0.4)
# default 40% is test and 60% is training data
```



```
X_train.shape
```

```
(44, 70)
```

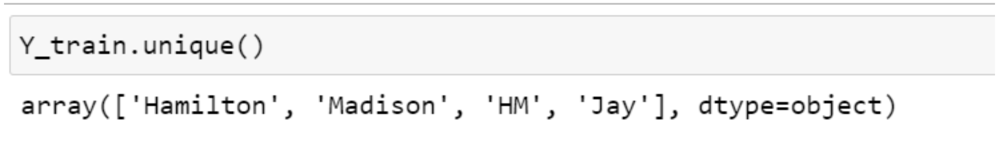


```
X_test.shape
```

```
(30, 70)
```

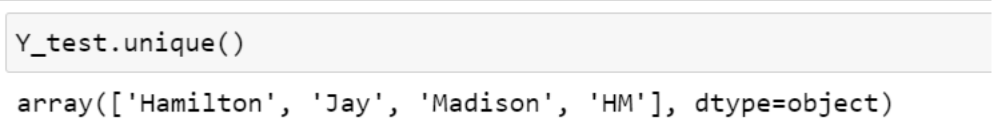
So, 44 essays are used for **training** the decision tree classifier and **30 essays** for **testing** the predictions.

3. Both training and testing essays have the essays of all the authors – Hamilton, Madison, Jay, and Hamilton & Madison



```
Y_train.unique()

array(['Hamilton', 'Madison', 'HM', 'Jay'], dtype=object)
```

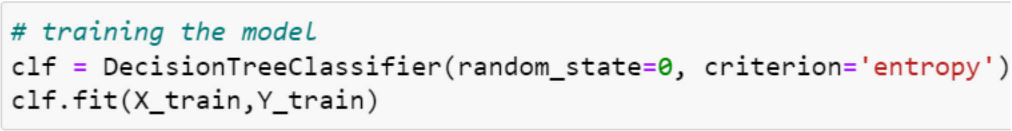


```
Y_test.unique()

array(['Hamilton', 'Jay', 'Madison', 'HM'], dtype=object)
```

BUILDING MODEL AND POST PRUNING

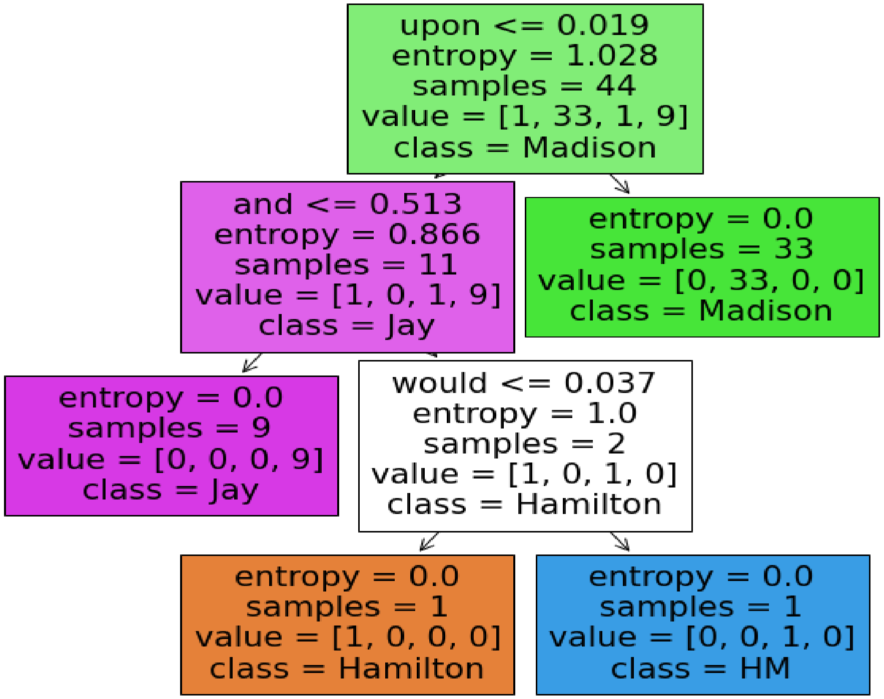
1. Created decision tree classifier with 'entropy' criterion for determining the best split at every level



```
# training the model
clf = DecisionTreeClassifier(random_state=0, criterion='entropy')
clf.fit(X_train,Y_train)
```

Decision tree created by the model from the training data looks like:

Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu

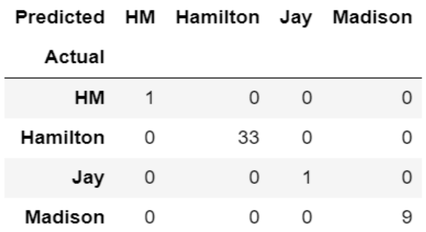


Textual Representation of the Decision Tree

```
|--- upon <= 0.02
|   |--- and <= 0.51
|   |   |--- class: Madison
|   |--- and > 0.51
|   |   |--- would <= 0.04
|   |   |   |--- class: HM
|   |   |--- would > 0.04
|   |   |   |--- class: Jay
|--- upon > 0.02
|   |--- class: Hamilton
```

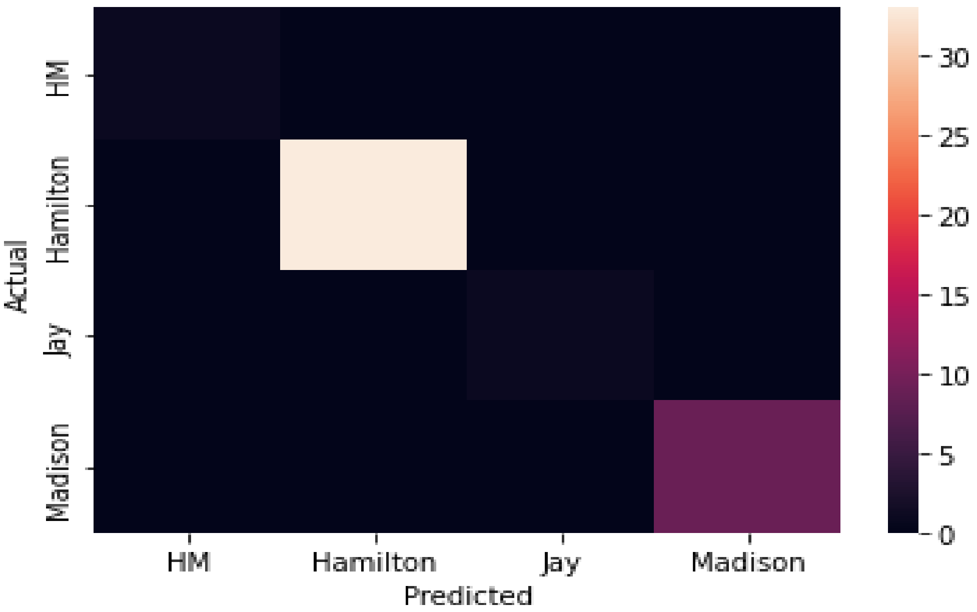
Feature **upon** has the highest information gain and is considered to be the first feature to classify the authors.

2. Validating the model to check if there is any overfitting problem using confusion matrix
   a. Training data is having 100% accuracy

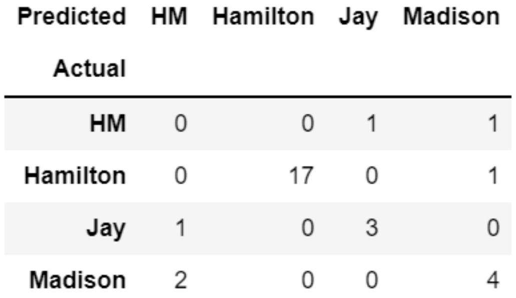


Predicted	HM	Hamilton	Jay	Madison
Actual				
HM	1	0	0	0
Hamilton	0	33	0	0
Jay	0	0	1	0
Madison	0	0	0	9

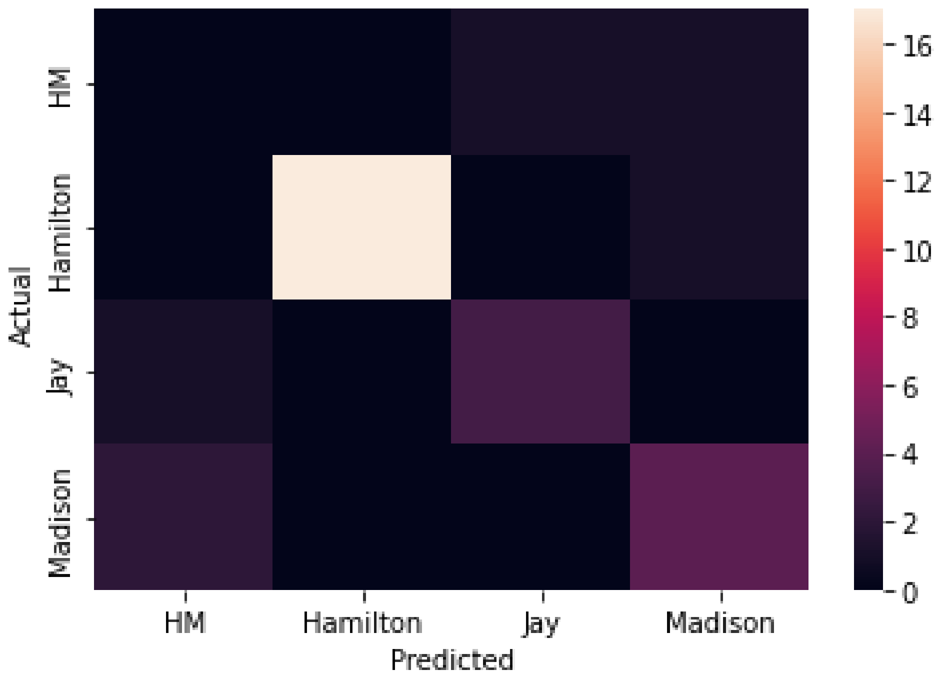
Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu



b. Model is not performing well on the testing data. Essays authored by Jay and Madison are being classified as HM. It can happen that the model is overfitted which is common problem with decision trees



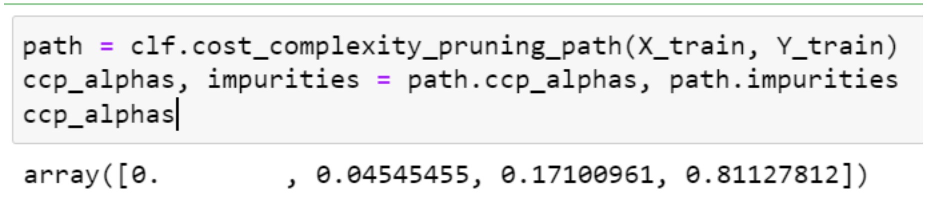
Predicted	HM	Hamilton	Jay	Madison
Actual				
HM	0	0	1	1
Hamilton	0	17	0	1
Jay	1	0	3	0
Madison	2	0	0	4



Performing Post Pruning to avoid overfitting

Pruning technique is parameterized by the cost complexity parameter, **ccp_alpha**. Greater values of **ccp_alpha** increase the number of nodes pruned. It is necessary to choose right **ccp_alpha** to cut down the branches of the decision tree.

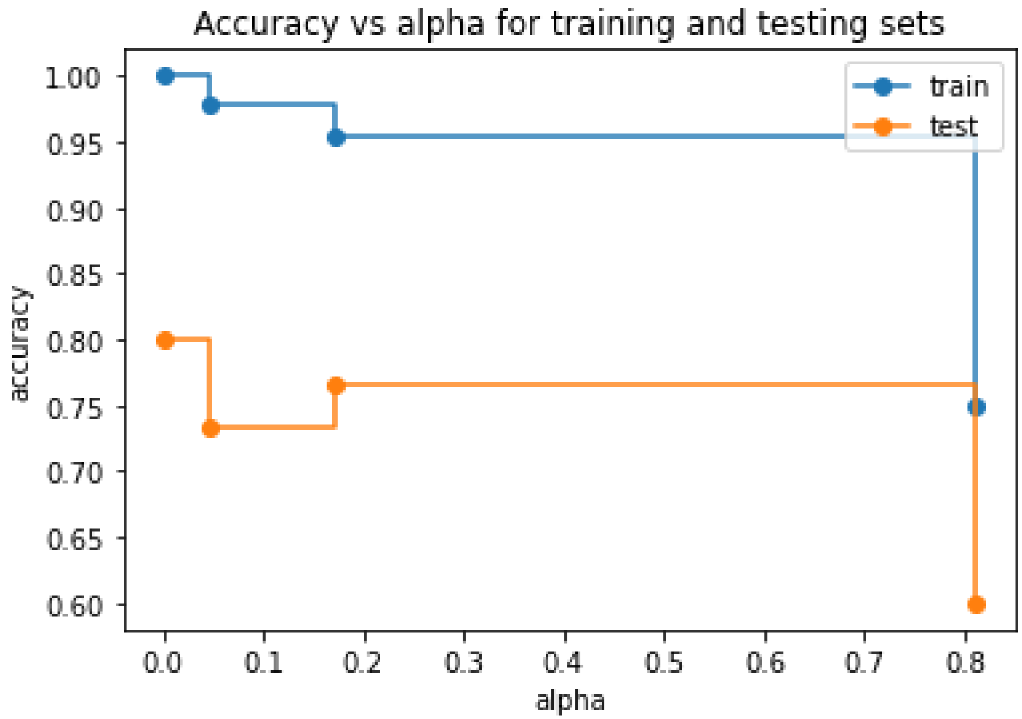
Based on different ccp_alpha values found from the training data, accuracy was plotted for training and testing data sets



```
path = clf.cost_complexity_pruning_path(X_train, Y_train)
ccp_alphas, impurities = path.ccp_alphas, path.impurities
ccp_alphas
```

```
array([0.        , 0.04545455, 0.17100961, 0.81127812])
```

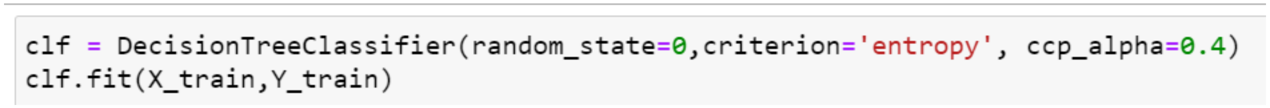
Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu



From the plot, cost complexity value between 0.2 to 0.8 seems to be stable with accuracy on training and testing data sets.

We can also choose alpha value above 0.0 but less than ~0.05 because the accuracy remained constant for training and testing data around that value.

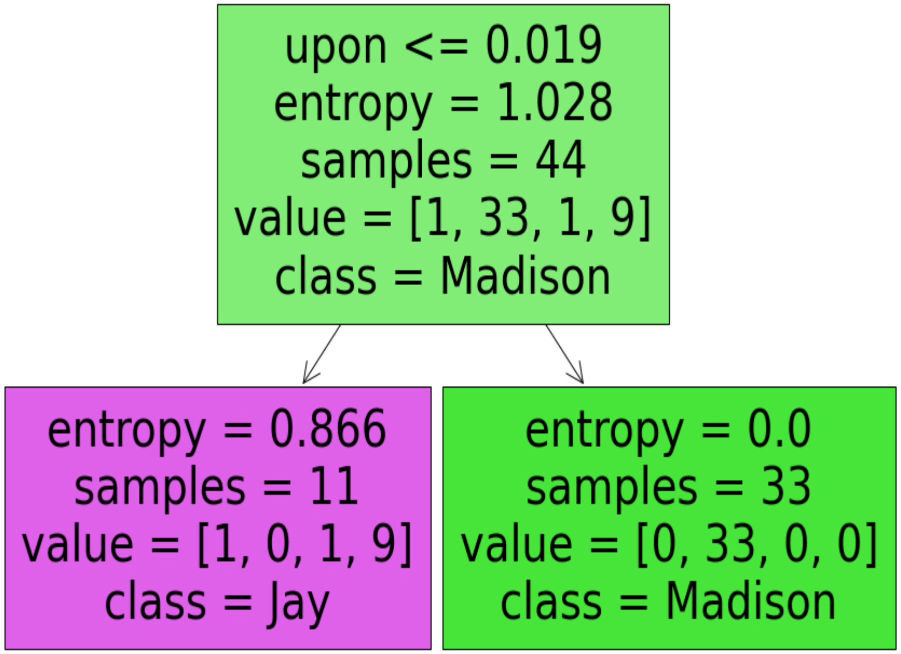
Training the model with minimum cost complexity parameter = 0.4



```
clf = DecisionTreeClassifier(random_state=0,criterion='entropy', ccp_alpha=0.4)
clf.fit(X_train,Y_train)
```

Decision Tree resulted from the above classifier looks like:

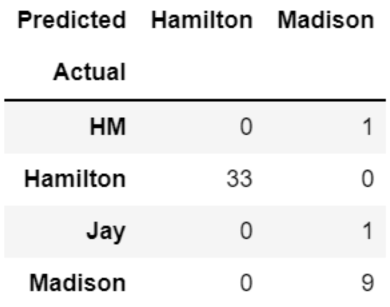
Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu



```
|--- upon <= 0.02
|   |--- class: Madison
|--- upon > 0.02
|   |--- class: Hamilton
```

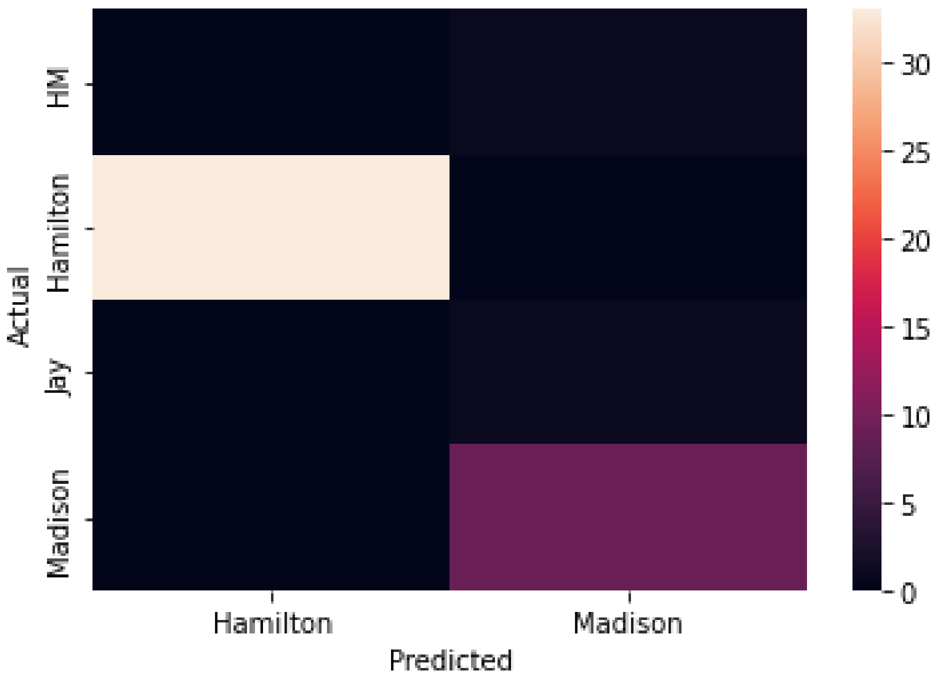
This model is performing well in classifying Hamilton and Madison essays but not w.r.t Jay and Hamilton & Madison essays.

Confusion matrix of the training data

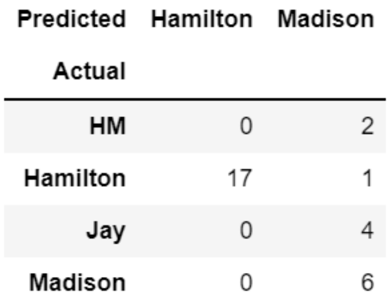


Predicted	Hamilton	Madison
Actual		
HM	0	1
Hamilton	33	0
Jay	0	1
Madison	0	9

Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu

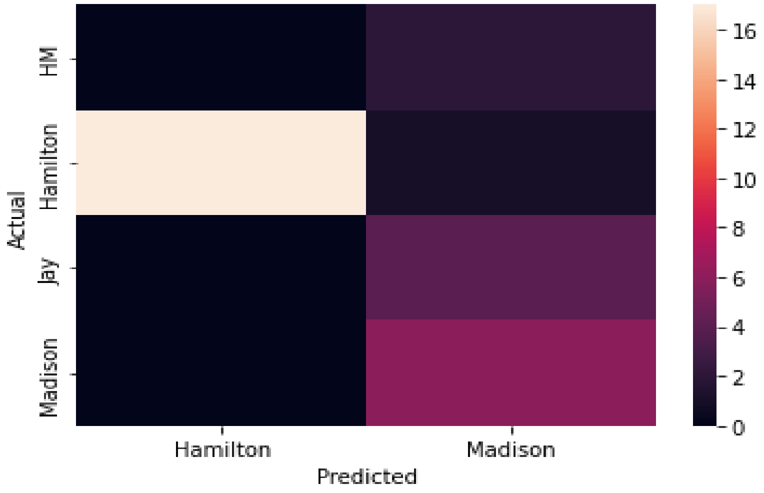


Confusion matrix of the testing data



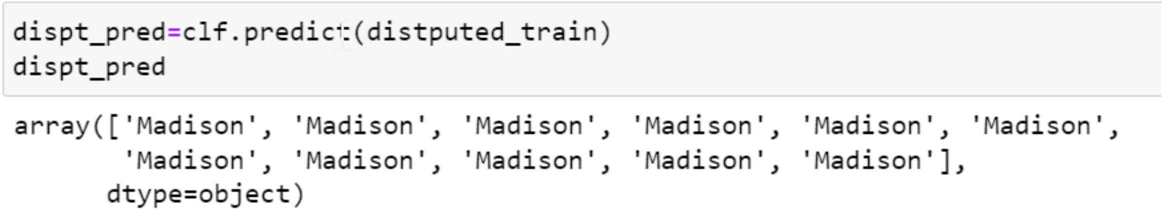
Predicted	Hamilton	Madison
Actual		
HM	0	2
Hamilton	17	1
Jay	0	4
Madison	0	6

Chaithra Kopparam Cheluvaiah
SUID 326926205
ckoppara@syr.edu



PREDICTING AUTHOR OF DISPUTED ESSAYS

According to the prediction by the decision tree classifier, all the **disputed essays are written by Madison**.



```
dispt_pred=clf.predict(distputed_train)
dispt_pred
```

```
array(['Madison', 'Madison', 'Madison', 'Madison', 'Madison', 'Madison',
       'Madison', 'Madison', 'Madison', 'Madison', 'Madison'],
      dtype=object)
```