Homework 5

Kyle Walter

2/14/2021

#Hamilton verse Madison Round 2

In our previous model we looked at using clustering algorithms to look at the the Federalist Papers. These were 85 essays written by Hamiliton, Madison, and John Jay, with some co-authored, and some disputed authorship (between Hamilton and Madison). Cluster works by looking at the data set as a whole and trying to group together similar data points while creating a most distance from other based on their similar nature. We then inferred from the results which clusters belong to each of the three authors but comparing the known authorship.

For this test we’ll actually use a supervised machine learning technique known as a Decision tree. This will build a model based on samples given to the algorithm with the known results. The algorithm then builds a model that will classify the documents as Hamilton, Madison, John Jay, or Hamilton and Madison. We will then utilze that model on a test and see if it can correctly classify a series of articles correctly.

Lastly we’ll test the model on the disputed and see if we obtain similar results to the clustering model for the unknown authorship. ##Data Preprocessing

First step in any data mining process is to load and prepare the data for use. We were previously provided a converted dataset of word occurrences in each of the 85 documents, along with two leading columns. The first with the authorship details for the known and 11 unknown marked as disputed. The 2nd Column refers to the particular paper and author. Lastly 70 columns of commonly seen words and their percent of occurence within the articles.

We’ll first bring them into R. Next we’ll remove the 2nd column as it refers to the each of the separate documents like an ID of sorts. Since Decision tree models have a liking of entropy this column has the potential to throw off the model. Next we’ll convert the first column of Author names to factors, since the J48 decision tree algorithm does not work with text strings.

#Calls the packages used int the code  
require(readr)

## Loading required package: readr

require(RWeka)

## Loading required package: RWeka

require(gdata)

## Loading required package: gdata

## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.

##

## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.

##   
## Attaching package: 'gdata'

## The following object is masked from 'package:stats':  
##   
## nobs

## The following object is masked from 'package:utils':  
##   
## object.size

## The following object is masked from 'package:base':  
##   
## startsWith

FedPapers <- read\_csv("fedPapers85.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## .default = col\_double(),  
## author = col\_character(),  
## filename = col\_character()  
## )  
## i Use `spec()` for the full column specifications.

FedPapers <- FedPapers[,-2]  
FedPapers$author <- as.factor(FedPapers$author)

Now that the data is a form that algorithm can process it, the training set, test set, and validation set will need to be created. The validation set is the easiest of the 3, since it will include the disputed papers, which are represented by the first 11 observations in the data set provided. next the for the remaining, we’ll need to use a 67-33 split on the rest of the data. However; because the authorship isn’t balanced, to get a good test set we’ll first need to separate the sets by each author and then apply selection to generate a training set for each author and test set for each and combine them back into consolidated sets.

#creates a list of by Authors  
SplitSet <- split.data.frame(FedPapers, FedPapers$author)  
ValidationSet <- SplitSet[[1]] #Extracts the Validation Set  
SplitSet <- SplitSet[-1] #removes the validation set from the list  
#creates random index for each author and calculates a cut point in the data  
set.seed(47)  
RandomIndex <- sapply(SplitSet, function(x){sample(1:dim(x)[1])})  
CutPoint <- sapply(SplitSet, function(x){floor(2\*dim(x)[1]/3)})  
  
#calls each of the data sets out from the list  
Hamilton <- SplitSet[[1]]  
HM <- SplitSet[[2]]  
Jay <- SplitSet[[3]]  
Madison <- SplitSet[[4]]  
  
#Creates the training set using the random index and 2/3rd cutpoint  
HamiltonTrain <- Hamilton[RandomIndex[[1]][1:CutPoint[1]],]  
HMTrain <- HM[RandomIndex[[2]][1:CutPoint[2]],]  
JayTrain <- Jay[RandomIndex[[3]][1:CutPoint[3]],]  
MadisonTrain <- Madison[RandomIndex[[4]][1:CutPoint[4]],]  
  
#Creates the test data sets from remaining 3rd of records  
HamiltonTest <- Hamilton[RandomIndex[[1]][(CutPoint[1]+1):dim(Hamilton)[1]],]  
HMTest <- HM[RandomIndex[[2]][(CutPoint[2]+1):dim(HM)[1]],]  
JayTest <- Jay[RandomIndex[[3]][(CutPoint[3]+1):dim(Jay)[1]],]  
MadisonTest <- Madison[RandomIndex[[4]][(CutPoint[4]+1):dim(Madison)[1]],]  
  
#Combines the data sets back into Train and Test Data sets and removes no longer needed objects  
TrainSet <- rbind.data.frame(HamiltonTrain, HMTrain, JayTrain,MadisonTrain)  
TestSet <- rbind.data.frame(HamiltonTest, HMTest, JayTest,MadisonTest)  
keep(FedPapers, ValidationSet, TrainSet, TestSet, sure = T)

##Building the Models

Now that the data is prepared, let’s apply the Decision Tree Model to the training dataset.

BaseModel <- J48(author~.,TrainSet)  
BaseModel

## J48 pruned tree  
## ------------------  
##   
## upon <= 0.018  
## | up <= 0.009  
## | | of <= 0.638: Jay (3.0)  
## | | of > 0.638: Madison (10.0)  
## | up > 0.009: HM (3.0/1.0)  
## upon > 0.018: Hamilton (33.0)  
##   
## Number of Leaves : 4  
##   
## Size of the tree : 7

As we can see from the output certain words have been identified for each set of Authors. Upon for Hamilton, Up has an appearance with the with the work of both Hamilton and Madison. Madison and Jay have been identified by their use of “up” and and the word “of”

Here’s the graphical representation of the tree.

require(partykit)

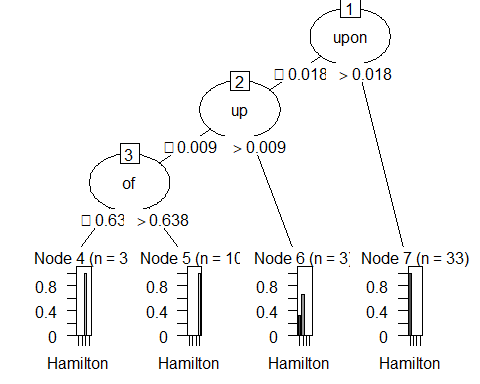
## Loading required package: partykit

## Loading required package: grid

## Loading required package: libcoin

## Loading required package: mvtnorm

plot(BaseModel)



Now that a graphical representation of the model can been seen. It is time to test the model to check it’s accuracy. We’ll use the test data set created in the prior section in order to see how the model is preforming.

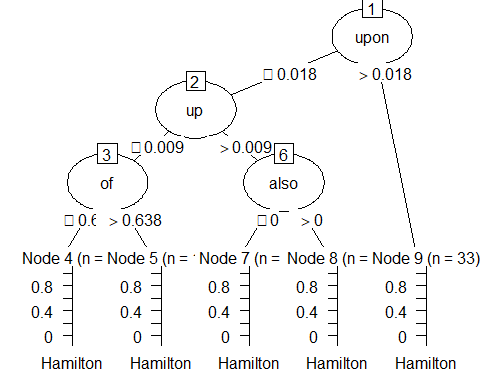
Pred1 <- predict(BaseModel, TestSet)  
table(TestSet$author, Pred1)[-1,-1] #Cross validation without the "Disputed" Column and Row.

## Pred1  
## Hamilton HM Jay Madison  
## Hamilton 17 0 0 0  
## HM 0 0 0 1  
## Jay 0 0 0 2  
## Madison 0 0 0 5

As we can see from the output, the model correctly classifies the Articles belonging to Hamilton and Madison, which is important for our cause. However; it incorrectly classifies John Jay’s articles as belong to Madison. Similar to what was seen previously in the Kmean clustering review of these same article, the papers published by Hamilton and Madison are getting classified under Madison. This likely indicates that these articles have features that are more like Madison’s than Hamilton’s. Thus the Alg thinks it belongs to Madison.

The model appears to highly preform with the base settings, but let’s see what happens if we change some of the inputs to the model to see if it can correctly predict the ariticles by John Jay.

TunedModel <- J48(author~., TrainSet, control = Weka\_control(C=.5, M=1))  
plot(TunedModel)



We adjusted two items in our formula. First the Confidence, which tells the formula to review the nodes of the tree, and those with minimal impact or adding to error of the model’s prediction of the training set to remove them. The Default is .25 which means to preform this process quite aggressively. As the number gets closer to 1, it preforms this process less aggressively. In addition we also decreased the M value of the formula. The M value tells the algorithm to look for fewer instances of a feature before adding it to the tree. Since John Jay only wrote 5 of the federalist papers this gives us the best chance at find him in the noise. Unfortunately as we see from the output model above similar to our first model, the deliniation between John Jay and Madison is driven by the word “of”.

In just to show the output more simply

Pred2 <- predict(TunedModel, TestSet)  
table(TestSet$author, Pred2)[-1,-1]

## Pred2  
## Hamilton HM Jay Madison  
## Hamilton 17 0 0 0  
## HM 0 0 0 1  
## Jay 0 0 0 2  
## Madison 0 0 0 5

We get the same table output as before. That being said though. The model can determine between Hamilton and Madison which we believe from their infighting that one or the other authored them.

##Prediction

Now that we have a working model for predicting between Hamilton and Madison using the Decision tree, let’s run the disputed data through the model and see which it predicts for each of them

PredV <- predict(TunedModel, ValidationSet)  
table(ValidationSet$author,PredV)[1,]

## dispt Hamilton HM Jay Madison   
## 0 1 0 0 10

The Model predict that Hamilton wrote one of the disputed papers, while Madison wrote 10 of them. This actually quite similar to what the Kmeans clustering predicted, where Hamilton wrote 2 of them and Madison wrote 9 of them. The output from both models seem to lend varsity to Madison’s statement that he was the writer of the disputed papers over Hamilton who claimed shortly before his death to have written them.