Sean Deery HW 5

# Introduction

Quote from the Library of Congress: https://guides.loc.gov/federalist-papers

The Federalist Papers were a series of eighty-five essays urging the citizens of New York to ratify the new United States Constitution. Written by Alexander Hamilton, James Madison, and John Jay, the essays originally appeared anonymously in New York newspapers in 1787 and 1788 under the pen name "Publius." A bound edition of the essays was first published in 1788, but it was not until the 1818 edition published by the printer Jacob Gideon that the authors of each essay were identified by name. The Federalist Papers are considered one of the most important sources for interpreting and understanding the original intent of the Constitution.

Eleven of the essays have disputed authorship between Hamilton and Madison. Hamilton wrote to claim the authorship before he was killed in a duel. Later Madison also claimed authorship. Historians were trying to find out which one was the real author. The following analysis examines the words in the documents to identify who wrote the disputed papers.

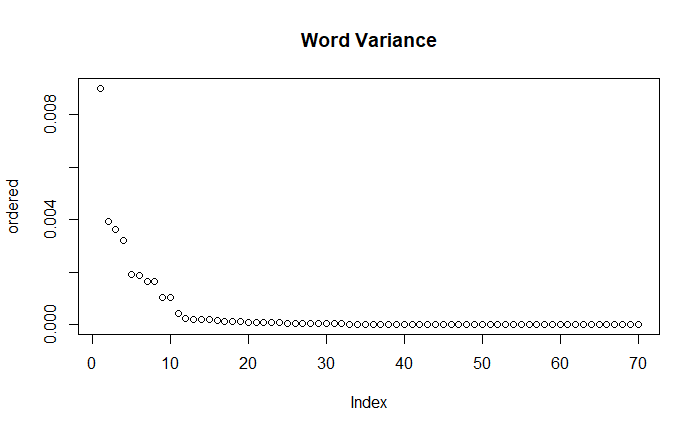
# Analysis and Models

## About the Data

The dataset contains a document term matrix of the Federalist Papers, including the author and the file name for each document. The author variable represents who wrote the paper. To prepare and clean the data for analysis, the author variable is reduced to Hamilton, Madison, and disputed. The author variable is then converted to a factor and the file name variable is converted into row names. The resulting data set contains 77 papers and 70 words.

The features are a set of “function words”, for example, “upon”. The feature value is the percentage of the word occurrence in an essay. For example, for the essay “Hamilton\_fed\_31.txt”, if the function word “upon” appeared 3 times, and the total number of words in this essay is 1000, the feature value is 3/1000=0.3%. This means the word counts have already been normalized for analysis.

Even though the words in the documents have been paired down to function words, there are still words in the dataset that are used similarly by all the authors. The plot below shows the variance of each word. Since the words with low variance will just add unwanted noise to the analysis, they are removed from the dataset. The words are identified by calculating the variance of the term frequency-inverse document frequency (tf-idf) of each word by the author. The words that have a variance in the bottom 40% are removed from the dataset.



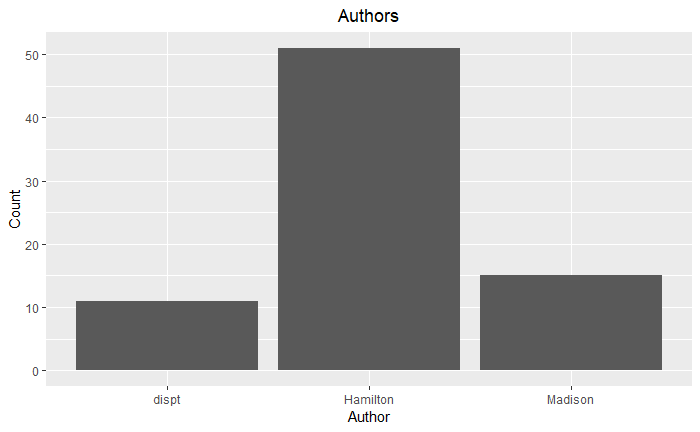
The author variables are Alexander Hamilton, James Madison, or labeled as disputed. 66.2% of the documents were authored by Hamilton. 19.5% of the documents were authored by Madison. 14.3% of the documents are disputed between Hamilton and Madison.

**Author Totals:**



**Author Percentages:**





The use of function words in each of the documents can be visualized with word clouds. The larger the word is in the cloud, the more it was used in the documents. In natural language processing, stopwords are a set of commonly used words, like “a” and “the”, that do not provide descriptive meaning. These stopwords are generally removed for analysis, but removing them resulted in a less helpful clustering analysis of who wrote the disputed documents. Therefore, all the words are kept in the analysis.

**Federalist Paper Wordcloud – All documents**

A close up of words

Description automatically generated with low confidence

**Hamilton’s Wordcloud**

A close up of words

Description automatically generated with low confidence

**Madison’s Wordcloud**

**A close up of words

Description automatically generated with low confidence**

**Disputed Papers Wordcloud**

A picture containing text, font, screenshot, graphics

Description automatically generated

## Cluster Analysis

Cluster analysis finds groups of objects such that the objects in a group will be similar to one another and different from the objects in other groups. This is an unsupervised learning model, which means that the group labels are not provided in the training data. The algorithm is tasked with creating groups based on the features.

### K-Means Clustering

K-Means Clustering is a partitional clustering technique. Partitional clustering creates a division of data objects into subsets such that each data object is in one subset. In K-means Clustering, the division of the objects is evaluated by minimizing the sum of square errors by recomputing the centroid of each cluster. The centroid of the cluster is calculated as the average of all the data examples in a cluster.

### Hierarchical Agglomerative Clustering (HAC)

Hierarchical clustering creates a set of nested clusters organized as a hierarchical tree. The clusters can be visualized as a dendrogram, which records the sequences of merges or splits. A clustering of the data objects is obtained by cutting the dendrogram at the desired level. In Hierarchical Agglomerative Clustering, each data point starts as a cluster. The clusters are then formed by evaluating the distance between clusters and merging the two closest. Different distance measures and linkage methods can be used to generate different clustering results. Distance measures include Euclidian, Manhattan, and Cosine among others. Linkage methods include Complete, Single, Average, and Median among others.

## Decision Tree

Decision Trees are a classification technique that classifies new data by asking a series of questions about the attributes of the new data. Each answer leads to a follow-up question until a conclusion is reached about the class label of the record. The questions can be organized into a decision tree containing a hierarchical structure of nodes and edges. A root node has no incoming edges and zero or more outgoing edges. Internal nodes have exactly one incoming edge and two or more outgoing edges. A leaf node has exactly one incoming edge and no outgoing edges. The leaf node is where the classification is completed.

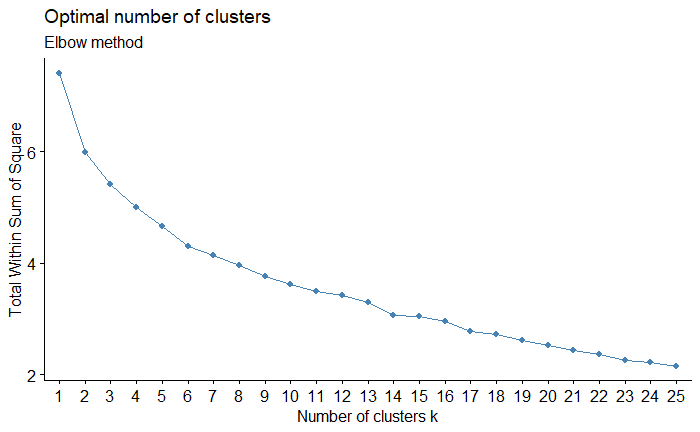
The order of the questions, or the order of attributes to split on, is decided based on the information gain and gain ratio. Information gain is a statistical measure that measures how well a given attribute separates the training examples according to their target classification. The gain ratio is the information gain divided by “split info”, which refers to a penalty for a large number of splits.

The strengths of the decision tree are that they are fast in prediction, they have interpretable patterns, and they are robust to noise in the data. The weaknesses of a decision tree are that they tend to overfit, they are error-prone when there are too many classes, and they are computationally expensive in training. The problem with overfitting arises because there may not be enough training data to fully represent all possible cases or the decision tree may be too detailed a fit to the training data. To account for overfitting, the decision tree model’s complexity can be controlled by pruning. Prepruning is setting a minimum information gain threshold, at which point the nodes do not split any further. Postpruning is removing branches from a decision tree that has already been trained.

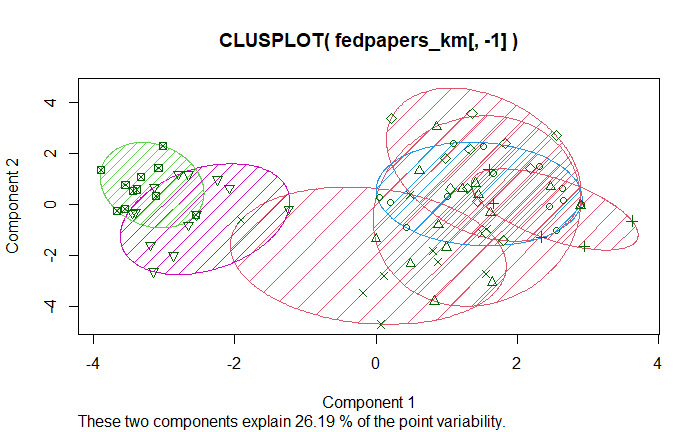
# Results

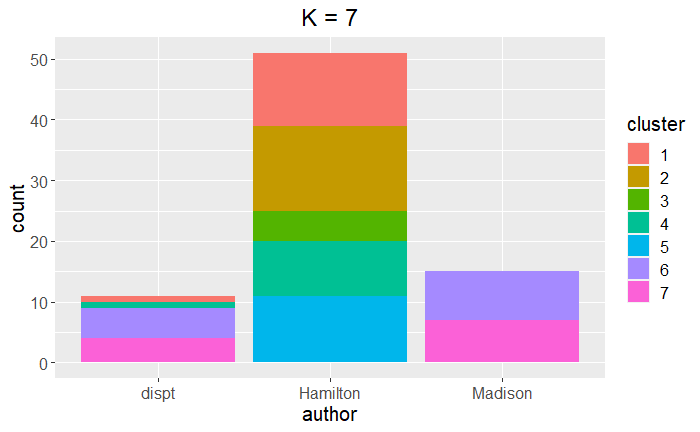
### K-Means Clustering

The K-Means clustering analysis was completed with a variety of total clusters to identify the optimal k value. To help identify the optimal number of clusters to use, the elbow method was applied to the dataset. The plot below shows the within-cluster sums of squared errors for 1 cluster up to 25 clusters. The method identified the optimal number of clusters to be 7.



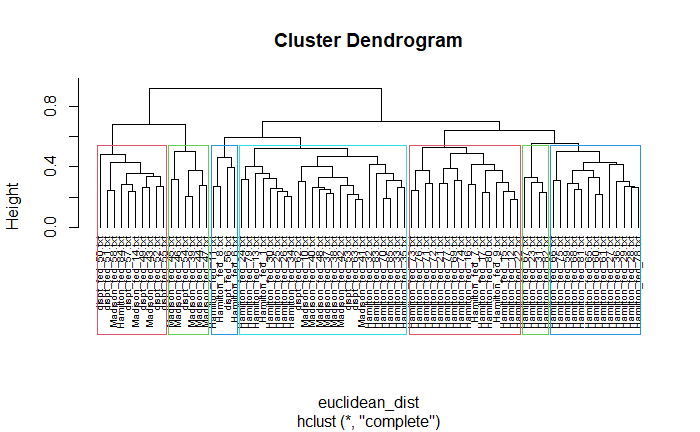
Running the model with five clusters resulted in the disputed papers being clustered equally between Hamilton and Madison as if both of them wrote the disputed papers. Running the model with nine clusters results in most of the disputed papers being clustered with Madison’s papers. Running the model with seven clusters resulted in the disputed papers being clustered reasonably between Hamilton and Madison. The results of using seven clusters are represented in the plots below. The algorithm clustered two papers (55 and 56) with Hamilton’s writing and nine papers (49, 50, 51, 52, 53, 54, 57, 62, and 63) with Madison’s writing. To add perspective to this result, the K-Means algorithm was run with different initial cluster centroids by changing the random seed. Changing the random seed made a large effect on the resulting clusters, and it mostly seemed to cluster more documents with Madison.

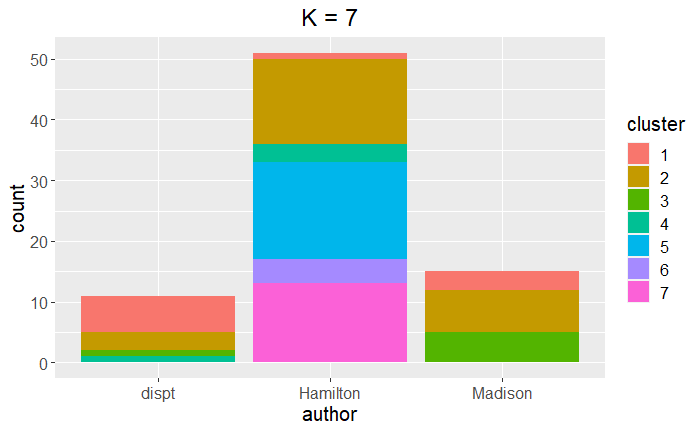




### Hierarchical Agglomerative Clustering (HAC)

The HAC analysis was completed with the same amount of clusters as the k-means analysis and a variety of different distance measures and linkage methods. Using the “single” method resulted in Hamilton and Madison being clustered in the same group, which is not very helpful for the analysis. Using the “complete” method, the algorithm was able to cluster the documents with some separation between Hamilton and Madison. The results of using the linkage method “complete” with a Euclidean distance measure are represented in the plots below. The separation between the two authors is not perfect, but the algorithm clustered one paper, 56, with Hamilton’s writing and ten papers (49, 50, 51, 52, 53, 54, 55, 57, 62, and 63) with Madison’s writing.





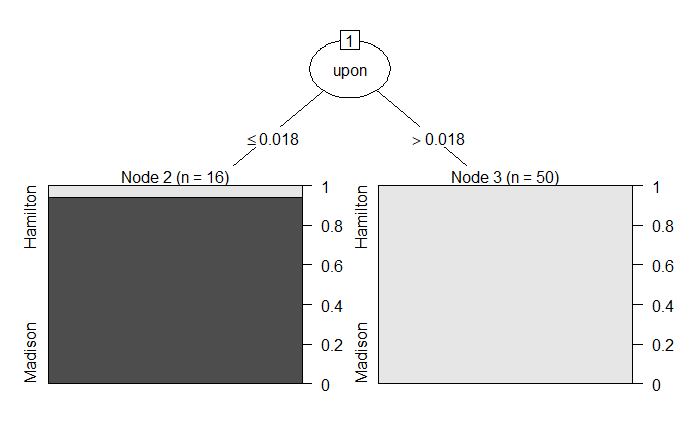
### Decision Tree

The decision tree was run with different pruning confidence and minimum instance thresholds and was then evaluated with 10-fold cross-validation. Running the algorithm with a minimum instance threshold of 1 resulted in 95.5% accuracy, whereas running it with a minimum instance threshold of 2 resulted in 97.0% accuracy. Setting the pruning confidence threshold of 0.1 to 0.5 did not change the result and had the same accuracy rate of 97%. The results shown below are from running the decision tree with a minimum instance threshold of 2 and a pruning confidence threshold of 0.3. The confusion matrix shows that it predicted 1 of Hamilton’s papers as Madison’s and 1 of Madison’s papers as Hamilton’s. The list of words below shows the information gain for each word that had an information gain above 0. The word ‘upon’ shows up as the word with the highest information gain, indicating the use of this word is what separated the two writers the most. The decision tree visualization below shows that the model split the documents by the word ‘upon’. If the word made up for over 0.018% of the words in the document, the model classified it as Hamilton. Otherwise, the model classified the document as Madison. Running the disputed papers through the prediction model resulted in all of the disputed papers being classified as Madison’s papers due to the lack of the word ‘upon’.

A picture containing text, font, screenshot, number

Description automatically generated





# Conclusions

In the attempt to understand who wrote the disputed Federalist Papers, the words in the papers were run through three different algorithms, k-means clustering, HAC clustering, and a decision tree. Clustering is an unsupervised learning technique that can group similar pieces of data and separate pieces of data that are different. Depending on the algorithm, the results can vary depending on the initial cluster centroid assignment. This is important to keep in mind to understand the uncertainty around the results. The decision tree on the other hand is a supervised learning technique where the answers are provided in the training data to assist in predicting the correct answer when presented with test data. The results can vary depending on the minimum instance threshold and the pruning confidence threshold.

The K-means algorithm clustered two papers (55 and 56) with Hamilton’s writing and nine papers (49, 50, 51, 52, 53, 54, 57, 62, and 63) with Madison’s writing. Changing the random seed for the K-means resulted in more documents being clustered with Madison’s writing. The HAC algorithm clustered one paper, 56, with Hamilton’s writing and ten papers (49, 50, 51, 52, 53, 54, 55, 57, 62, and 63) with Madison’s writing. The decision tree algorithm classified all the disputed papers as Madison’s writing. The differences in these findings indicate that the results are certainly not definitive, but they expose some understanding of who the most likely author was. The results show that the three different methods of analysis support each other and that the disputed documents were mostly, if not entirely, written by Madison.