Unsupervised Capstone:

Classify authors based on the style of text

By Karen McGee

Problem Statement

- Build an unsupervised model that will classify authors based on the style of writing using natural language processing.

Research Questions:

- Are authors consistently grouped into the same cluster?
- Does your clustering on those members perform as you'd expect?
- Have your clusters remained stable or changed dramatically?
- Does our model provide a consistent performance?

Solution statement

- Import data from ten different authors of various writing styles
- Clean, tokenize and lemmatize the data
- Generate features using TFIDF
- Generate clusters (i.e. K-mean, MeanShift…etc.)
- Evaluate the performance of the clusters
- Generate models (i.e. Random Forest, logistic…etc.)
- Evaluate the performance of the models

Evaluation metrics of clusters and models

- Utilize the Random index (RI) adjusted score to evaluate the performance of each clusters.
- Use confusion matrix, classification report and accuracy score to evaluate the performance of each model.

Analysis – input of raw text file

Example of raw text file – Macbeth

```
"[The Tragedie of Macbeth by William Shakespeare 1603]\n\n\nActus Primus. Scoena Prima.\n\n
Thunder and Lightning. Enter three Witches.\n\n  1. When shall we three meet againe?\nIn Th
under, Lightning, or in Raine?\n  2. When the Hurley-burley's done,\nWhen the Battaile's lo
st, and wonne\n\n   3. That will be ere the set of Sunne\n\n   1. Where the place?\n  2. Vp
on the Heath\n\n   3. There to meet with Macbeth\n\n   1. I come, Gray-Malkin\n\n   All. Pa
dock calls anon: faire is foule, and foule is faire,\nHouer through the fogge and filthie a
yre.\n\nExeunt.\n\n\nScena Secunda.\n\nAlarum within. Enter King Malcome, Donalbaine, Lenox
, with\nattendants,\nmeeting a bleeding Captaine.\n\n  King. What bloody man is that? he ca
n report,\nAs seemeth by his plight, of the Reuolt\nThe newest state\n\n   Mal. This is the
Serieant,\nWho like a good and hardie Souldier fought\n'Gainst my Captiuitie: Haile braue f
riend;\nSay to the King, the knowledge of the Broyle,\nAs thou didst leaue it\n\n   Cap. Do
ubtfull it stood,\nAs two spent Swimmers, that doe cling together,\nAnd choake their Art: T
he mercilesse Macdonwald\n(Worthie to be a Rebell, for to that\nThe multiplying Villanies o
f Nature\nDoe swarme vpon him) from the Westerne Isles\nOf Kernes and Gallowgrosses is supp
ly'd,\nAnd Fortune on his damned Quarry smiling,\nShew'd like a Rebells Whore: but all's to
o weake:\nFor braue Macbeth (well hee deserues that Name)\nDisdayning Fortune, with his bra
ndisht Steele,\nWhich smoak'd with bloody execution\n(Like Valours Minion) caru'd out his p
assage,\nTill hee fac'd the Slaue:\nWhich neu'r shooke hands, nor bad farwell to him,\nTill
he vnseam'd him from the Naue toth' Chops,\nAnd fix'd his Head vpon our Battlements\n\n   K
```

Analysis – Text file cleaned, lemmatized and tokenized

Example of a cleaned, lemmatized and tokenized file - Macbeth

['Actus Primus Scoena Prima Thunder and Lightning Enter three Witches 1 When shall we three
meet againe In Thunder Lightning or in Raine 2 When the Hurleyburleys done When the Battail
es lost and wonne 3 That will be ere the set of Sunne 1 Where the place 2 Vpon the Heath 3
There to meet with Macbeth 1 I come GrayMalkin All Padock call anon faire is foule and foul
e is faire Houer through the fogge and filthie ayre Exeunt Scena Secunda Alarum within Ente
r King Malcome Donalbaine Lenox with attendant meeting a bleeding Captaine King What bloody
man is that he can report As seemeth by his plight of the Reuolt The newest state Mal This
is the Serieant Who like a good and hardie Souldier fought Gainst my Captiuitie Haile braue
friend Say to the King the knowledge of the Broyle As thou didst leaue it Cap Doubtfull it
stood As two spent Swimmers that doe cling together And choake their Art The mercilesse Mac
donwald Worthie to be a Rebell for to that The multiplying Villanies of Nature Doe swarme v
pon him from the Westerne Isles Of Kernes and Gallowgrosses is supplyd And Fortune on his d
amned Quarry smiling Shewd like a Rebells Whore but alls too weake For braue Macbeth well h
ee deserues that Name Disdayning Fortune with his brandisht Steele Which smoakd with bloody
execution Like Valours Minion carud out his passage Till hee facd the Slaue Which neur shoo
ke hand nor bad farwell to him Till he vnseamd him from the Naue toth Chops And fixd his He
ad vpon our Battlements King O valiant Cousin worthy Gentleman Cap As whence the Sunne gin

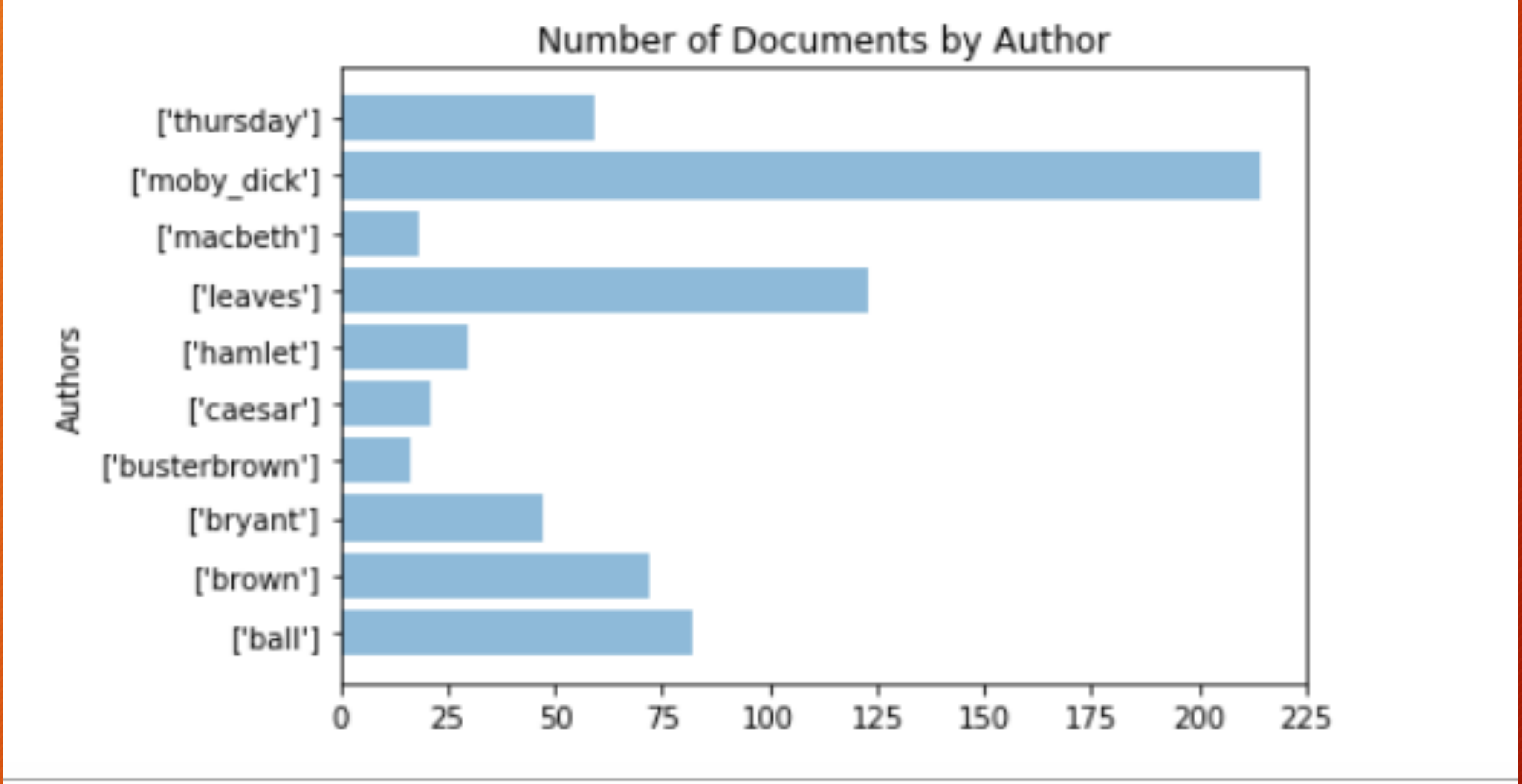
Analysis - combined document text, author and author code into a data frame.

	text	authors	author_codes
0	Actus Primus Scoena Prima Enter Flauius Murell...	caesar	4
1	Forgets the shewes of Loue to other men Cassi ...	caesar	4
2	would not so with loue I might intreat you Be ...	caesar	4
3	tell you that Ile nere looke you ith face agai...	caesar	4
4	is for Romans now Haue Thewes and Limbes like ...	caesar	4

Analysis – display number of documents group by author codes and authors.

```
author_codes  authors
0             ball              82
1             brown             72
2             bryant            47
3             busterbrown       16
4             caesar            21
5             hamlet            30
6             leaves           123
7             macbeth           18
8             moby_dick        214
9             thursday          59
Name: authors, dtype: int64
```

Analysis –visual bar chart displaying number of documents group by authors



Analysis – Feature generation of text file

Features produce by TFIDF

woods	woof	wool	woollen	word	wordless	words	wore	work	workd	worke	worked	worker	working	working
0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.031282	0.0	0.0	0.0	
0.0	0.0	0.0	0.0	0.013771	0.0	0.0	0.0	0.0	0.0	0.034234	0.0	0.0	0.0	
0.0	0.0	0.0	0.0	0.023609	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	
0.0	0.0	0.0	0.0	0.026765	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	
0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.033973	0.0	0.0	0.0	

5 rows × 12088 columns

Parameters used for TFIDF

```
#Generate features using TFIDF

from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(max_df=0.5, # drop words that occur in more than half the parag
                             min_df=3, # only use words that appear at least twice
                             stop_words=stopwords,
                             lowercase=True, #convert everything to lower case (since Alice
                             use_idf=True,#we definitely want to use inverse document freque
                             norm=u'l2', #Applies a correction factor so that longer paragra
                             smooth_idf=True #Adds 1 to all document frequencies, as if an e
                            )
```

Cluster analysis – Top terms identified for each KNN cluster and Author code

```
Top terms per cluster:
Cluster 0:
Authour code: 6
buster
joe
bear
browns
farmer
otter
blacky
pool
trout
boy
```

```
Cluster 1:
Authour code: 7
brown
father
flambeau
garden
margery
priest
door
prince
looked
boulnois
```

```
Cluster 2:
Authour code: 6
turnbull
macian
evan
quite
wall
sword
god
mean
garden
really
```

```
Cluster 3:
Authour code: 8
ye
queequeg
ahab
captain
ship
thou
starbuck
sea
whale
aye
```

```
Cluster 4:
Authour code: 8
ham
haue
lord
macb
king
enter
thou
hor
hamlet
vpon
```

```
Cluster 5:
Authour code: 6
syme
gregory
professor
bull
sunday
marquis
dr
secretary
anarchist
colonel
```

```
Cluster 6:
Authour code: 9
whale
boat
sperm
ship
ahab
stubb
sea
though
water
leviathan
```

```
Cluster 7:
Authour code: 6
king
came
michael
jackal
fir
story
tree
brown
mr
cross
```

```
Cluster 8:
Authour code: 2
caesar
brutus
bru
cassi
haue
cassius
cask
caes
antony
brut
```

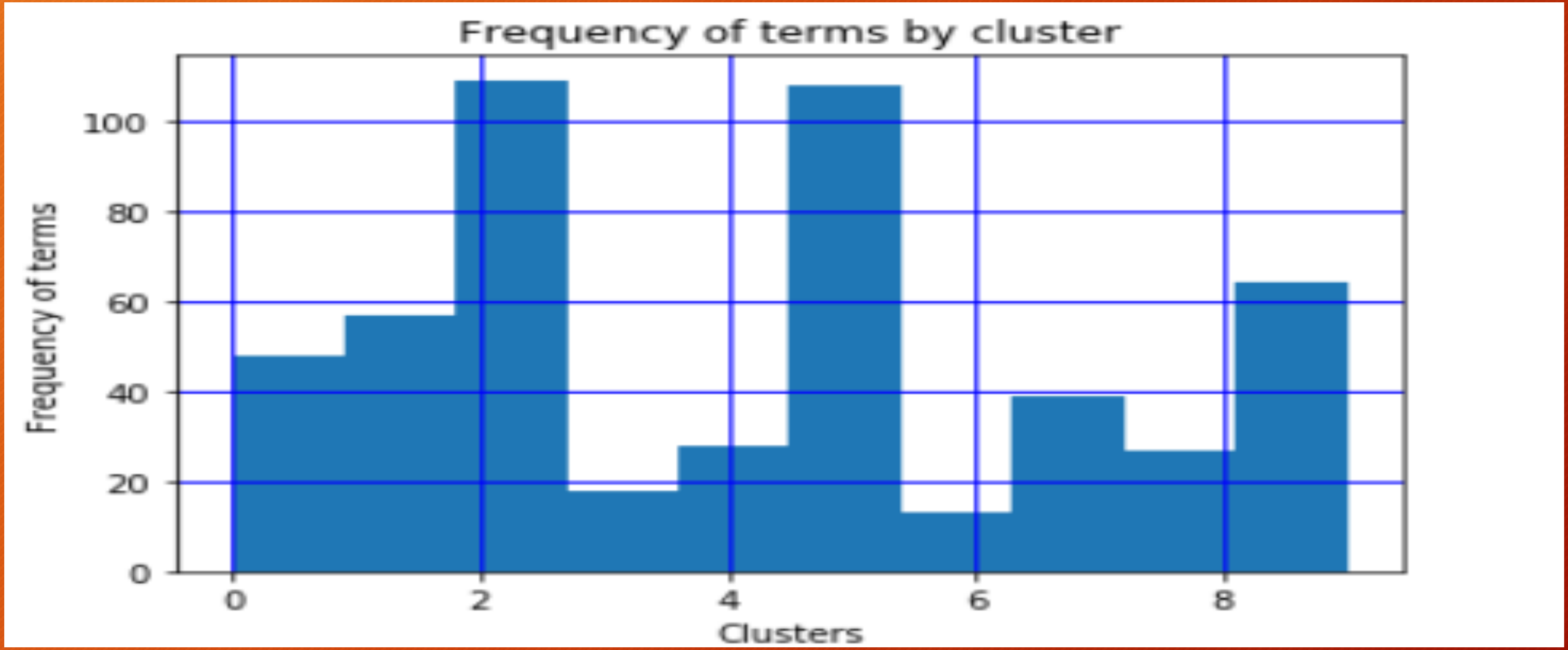
```
Cluster 9:
Authour code: 6
love
soul
thee
shall
song
earth
land
thy
woman
city
```

Top terms from KNN clusters:	Associated with author codes:
0,2,5,7 and 9	6 - Leaves
3 and 4	8 – Moby Dick
1	7 - Macbeth
6	9 - Thursday
8	6 - Bryant

Cluster analysis – Visual Distribution of top terms for each KNN cluster

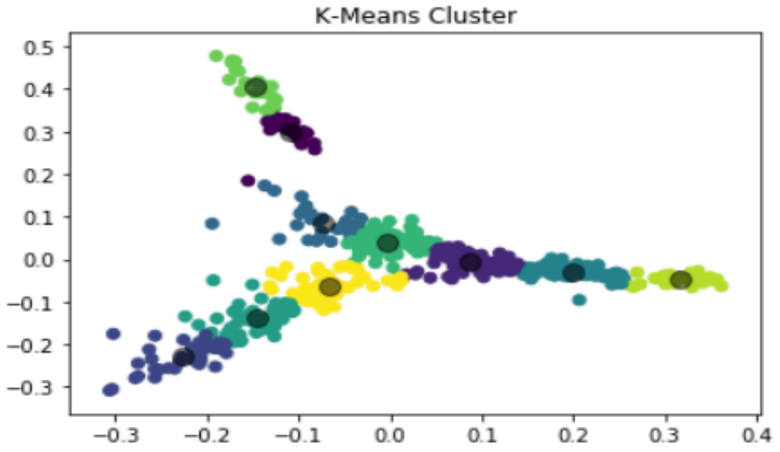
Parameters of Histogram

```
plt.hist(model.labels_, bins=n_clusters)
```



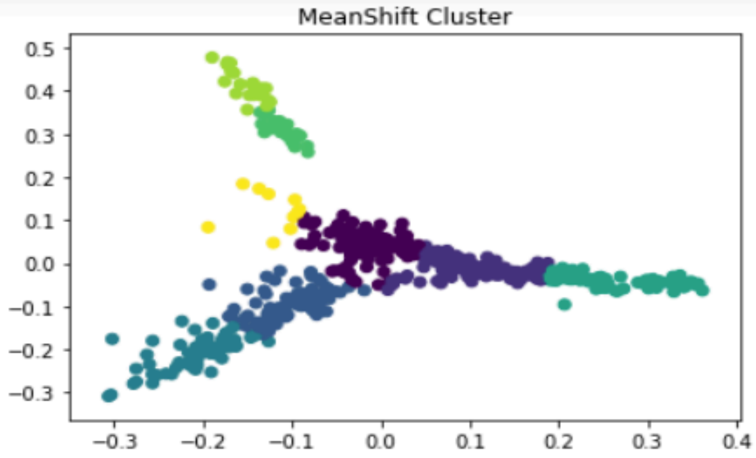
Cluster visualizations: Kmeans and MeanShift

Graphically visualization of each cluster



Comparing k-means clusters against author codes:

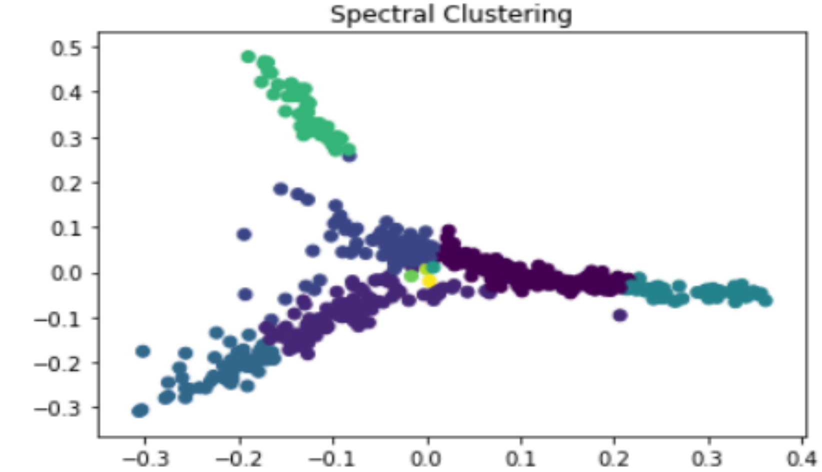
col_0 author_codes	0	1	2	3	4	5	6	7	8	9	Total
0	0	9	0	0	28	0	2	0	28	0	67
1	0	48	0	0	6	0	0	0	0	0	54
2	0	7	0	0	0	0	23	0	0	1	31
3	0	13	0	0	0	0	0	0	0	0	13
4	12	0	0	0	0	0	0	2	0	0	14
5	3	0	0	0	0	0	0	18	0	0	21
6	1	0	0	29	0	0	71	0	0	0	101
7	9	0	0	0	0	0	0	4	0	0	13
8	0	4	39	4	0	54	2	0	0	55	158
9	0	6	0	0	32	0	1	0	0	0	39
Total	25	87	39	33	66	54	99	24	28	56	511



Comparing meanshift clusters against authors:

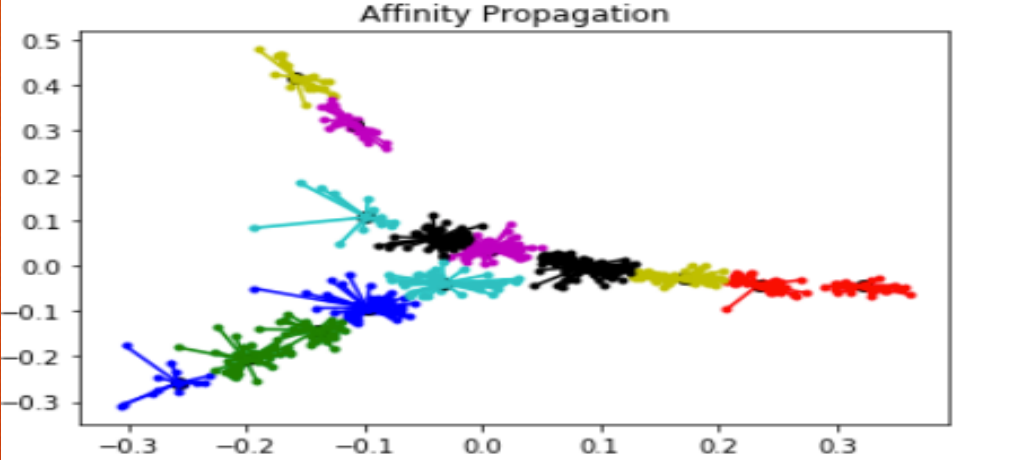
col_0 author_codes	0	1	2	3	4	5	6	7	Total
0	2	13	0	0	52	0	0	0	67
1	0	54	0	0	0	0	0	0	54
2	22	8	1	0	0	0	0	0	31
3	0	13	0	0	0	0	0	0	13
4	0	0	0	0	0	13	1	0	14
5	0	0	0	0	0	3	18	0	21
6	92	0	0	0	0	0	0	9	101
7	0	0	0	0	0	11	2	0	13
8	15	7	79	56	0	0	0	1	158
9	1	21	0	0	17	0	0	0	39
Total	132	116	80	56	69	27	21	10	511

Cluster visualizations Spectral and Affinity



Comparing spectral clusters against authors:

col_0	0	1	2	3	4	5	6	7	8	9	Total
author_codes											
0	49	15	0	0	0	2	0	0	0	1	67
1	0	52	0	0	0	2	0	0	0	0	54
2	0	28	0	0	0	1	0	0	0	2	31
3	0	13	0	0	0	0	0	0	0	0	13
4	0	0	0	0	14	0	0	0	0	0	14
5	0	0	0	0	21	0	0	0	0	0	21
6	0	8	0	1	0	1	1	0	1	89	101
7	0	0	0	0	13	0	0	0	0	0	13
8	0	0	50	0	0	92	0	1	0	15	158
9	5	34	0	0	0	0	0	0	0	0	39
Total	54	150	50	1	48	98	1	1	1	107	511



Comparing affinity clusters against authors:

col_0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	Total
author_codes															
0	0	0	24	0	2	0	0	0	0	26	0	0	7	8	67
1	0	0	0	0	0	0	0	0	0	0	0	0	9	45	54
2	0	0	0	0	21	0	2	1	0	0	0	0	0	7	31
3	0	0	0	0	0	0	0	0	0	0	0	0	0	13	13
4	0	0	0	0	0	0	0	0	0	0	0	14	0	0	14
5	0	0	0	0	0	18	0	0	0	0	0	3	0	0	21
6	0	0	0	13	41	0	45	0	0	0	2	0	0	0	101
7	0	0	0	0	0	2	0	0	0	0	0	11	0	0	13
8	14	31	0	1	0	0	3	44	35	0	29	0	0	1	158
9	0	0	0	0	1	0	0	0	0	5	0	0	30	3	39
Total	14	31	24	14	65	20	50	45	35	31	31	28	46	77	511

Estimated number of clusters: 14

Cluster evaluation results

Overall Spectral Clustering performed the best based on the Random Index (RI) Adjusted score.

	Cluster	Number of clusters	RI Score	RI adjusted score
0	K-Means	10	0.0171759	0.473769
1	MeanShift	8	0.0289061	0.465032
2	SpectralClustering	10	0.00935582	0.495523
3	AffinityPropagation	14	0.0138365	0.343228

Model Performance – Random Forest

```
RFC Training mean set score: 0.8046533422135997
RFC Testing mean set score: 0.6543994196433525

 Random Forest confusion matrix
 [[10  3  0  0  0  0  1  0  1  0]
 [ 4 12  0  0  0  0  0  0  1  1]
 [ 1  1 11  0  0  0  2  0  1  0]
 [ 1  0  1  1  0  0  0  0  0  0]
 [ 0  0  0  0  4  2  1  0  0  0]
 [ 0  0  0  0  0  9  0  0  0  0]
 [ 0  0  0  0  0  0 22  0  0  0]
 [ 0  0  0  0  0  1  0  4  0  0]
 [ 0  0  0  0  0  0  1  0 55  0]
 [ 1  2  0  0  0  0  3  0  4 10]]

 Random Forest classification report
              precision    recall  f1-score   support

           0       0.59      0.67      0.62        15
           1       0.67      0.67      0.67        18
           2       0.92      0.69      0.79        16
           3       1.00      0.33      0.50         3
           4       1.00      0.57      0.73         7
           5       0.75      1.00      0.86         9
           6       0.73      1.00      0.85        22
           7       1.00      0.80      0.89         5
           8       0.89      0.98      0.93        56
           9       0.91      0.50      0.65        20

   micro avg       0.81      0.81      0.81       171
   macro avg       0.85      0.72      0.75       171
weighted avg       0.83      0.81      0.80       171

Random Forest accuracy score: 0.8070175438596491
```

Model Performance – Logistic regression

```
LR Training mean set score: 0.9240892056625226
LR Testing mean set score: 0.5687751756050468

 Logistic regression confusion matrix
 [[15  0  0  0  0  0  0  0  0  0]
 [ 0 18  0  0  0  0  0  0  0  0]
 [ 0  0  5  0  0  0  4  0  7  0]
 [ 0  0  0  3  0  0  0  0  0  0]
 [ 0  0  0  0  7  0  0  0  0  0]
 [ 0  0  0  0  0  9  0  0  0  0]
 [ 0  0  0  0  0  0 22  0  0  0]
 [ 0  0  0  0  0  0  0  5  0  0]
 [ 0  0  0  0  0  0  0  0 56  0]
 [ 0  0  0  0  0  0  1  0  0 19]]

 Logistic classification report
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        15
           1       1.00      1.00      1.00        18
           2       1.00      0.31      0.48        16
           3       1.00      1.00      1.00         3
           4       1.00      1.00      1.00         7
           5       1.00      1.00      1.00         9
           6       0.81      1.00      0.90        22
           7       1.00      1.00      1.00         5
           8       0.89      1.00      0.94        56
           9       1.00      0.95      0.97        20

   micro avg       0.93      0.93      0.93       171
   macro avg       0.97      0.93      0.93       171
weighted avg       0.94      0.93      0.92       171

Logistic accuracy score:  0.9298245614035088
```

Model Performance – Gradient Boosting

```
Gradient Training mean set score: 0.9413435644920112
Gradient Testing mean set score: 0.8592366035265266

 Gradient Boosting confusion matrix
 [[15  0  0  0  0  0  0  0  0  0]
 [ 0 18  0  0  0  0  0  0  0  0]
 [ 0  0 13  0  0  0  1  0  2  0]
 [ 0  0  0  3  0  0  0  0  0  0]
 [ 0  0  0  0  7  0  0  0  0  0]
 [ 0  0  0  0  0  8  0  0  1  0]
 [ 0  1  0  0  0  0 21  0  0  0]
 [ 0  0  0  0  0  0  0  5  0  0]
 [ 0  0  0  0  0  1  0  0 55  0]
 [ 0  0  0  0  0  0  0  0  0 20]]

 Gradient Boosting classification report
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        15
           1       0.95      1.00      0.97        18
           2       1.00      0.81      0.90        16
           3       1.00      1.00      1.00         3
           4       1.00      1.00      1.00         7
           5       0.89      0.89      0.89         9
           6       0.95      0.95      0.95        22
           7       1.00      1.00      1.00         5
           8       0.95      0.98      0.96        56
           9       1.00      1.00      1.00        20

   micro avg       0.96      0.96      0.96       171
   macro avg       0.97      0.96      0.97       171
weighted avg       0.97      0.96      0.96       171

Gradient Boosting accuracy score: 0.9649122807017544
```

Model Performance - Knn

```
KNN Training mean set score: 0.9786117517268057
KNN Testing mean set score: 0.9363743799727022

 KNN Confustion Matrix
 [[15  0  0  0  0  0  0  0  0  0]
 [ 0 18  0  0  0  0  0  0  0  0]
 [ 0  1 14  0  0  0  1  0  0  0]
 [ 0  0  0  3  0  0  0  0  0  0]
 [ 0  0  0  0  7  0  0  0  0  0]
 [ 0  0  0  0  0  9  0  0  0  0]
 [ 0  0  0  0  0  0 22  0  0  0]
 [ 0  0  0  0  0  0  0  5  0  0]
 [ 0  0  0  0  0  0  0  0 56  0]
 [ 0  0  0  0  0  0  0  0  0 20]]

 KNN Classification Report
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        15
           1       0.95      1.00      0.97        18
           2       1.00      0.88      0.93        16
           3       1.00      1.00      1.00         3
           4       1.00      1.00      1.00         7
           5       1.00      1.00      1.00         9
           6       0.96      1.00      0.98        22
           7       1.00      1.00      1.00         5
           8       1.00      1.00      1.00        56
           9       1.00      1.00      1.00        20

   micro avg       0.99      0.99      0.99       171
   macro avg       0.99      0.99      0.99       171
weighted avg       0.99      0.99      0.99       171


 KNN accuracy score: 0.9883040935672515
```

Model Performance – Support Vector

```
SVC Training mean set score: 0.9942846872753414
SVC Testing mean set score: 0.9827296736464819

 Support vector cufusion matrix
 [[15  0  0  0  0  0  0  0  0  0]
 [ 0 18  0  0  0  0  0  0  0  0]
 [ 0  0 15  0  0  0  0  0  1  0]
 [ 0  0  0  3  0  0  0  0  0  0]
 [ 0  0  0  0  7  0  0  0  0  0]
 [ 0  0  0  0  0  9  0  0  0  0]
 [ 0  0  0  0  0  0 22  0  0  0]
 [ 0  0  0  0  0  0  0  5  0  0]
 [ 0  0  0  0  0  0  0  0 56  0]
 [ 0  0  0  0  0  0  0  0  0 20]]

 Support vector classification report
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        15
           1       1.00      1.00      1.00        18
           2       1.00      0.94      0.97        16
           3       1.00      1.00      1.00         3
           4       1.00      1.00      1.00         7
           5       1.00      1.00      1.00         9
           6       1.00      1.00      1.00        22
           7       1.00      1.00      1.00         5
           8       0.98      1.00      0.99        56
           9       1.00      1.00      1.00        20

   micro avg       0.99      0.99      0.99       171
   macro avg       1.00      0.99      1.00       171
weighted avg       0.99      0.99      0.99       171


 Support vector accuracy score: 0.9941520467836257
```

Summary results: Cluster and model performance

- Based on the crosstab results of the clusters, authors were not consistently grouped into the same cluster.
- I was expecting more clustering to occur on members who had more words per documents but the results of the clusters were not consistent with my expectation.
- Overall the clusters remained stable for every type of cluster (i.e. Kmeans, meanshift, spectural…etc.)
- Overall Model performance:
  - KNN and SVM where consistent with each other with as their accuracy scores range from 98 and 99%
  - Logistic regression, Random forest and Gradient boosting were not consistent as their accuracy scores range from 80, 92 and 96%