Grouping a Large Dataset of News Documents and being able to predict its Classes

1. Frame problem at hand : Predict Classes of Large Dataset New Documents

- Unsupervised Learning -> Large Dataset News Documents -> features -> predict Classes
- Potential Stakeholders
  - Politicians ???
  - Companies??
- Potential Added Value via Project Implementation
  - **Time saved on manual labor**
  - Metric/s
    - **Time**
  - What happens if No Project Implementation
- Current Base Model to measure Potential Added Value??

2. Collect the raw data needed for the problem

Regarding the chosen dataset:

- Dataset contains collection 18828 News Documents
- Each News Document represents a Single Newsgroup
- Message in each News Document is text of some Newsgroup
- Kaggle Dataset can be found:
    - https://www.kaggle.com/crawford/20-newsgroups/home
- Much regards to its previous contributors of analysis

3. Process & Explore the Data before In-Depth Analysis

Original Data

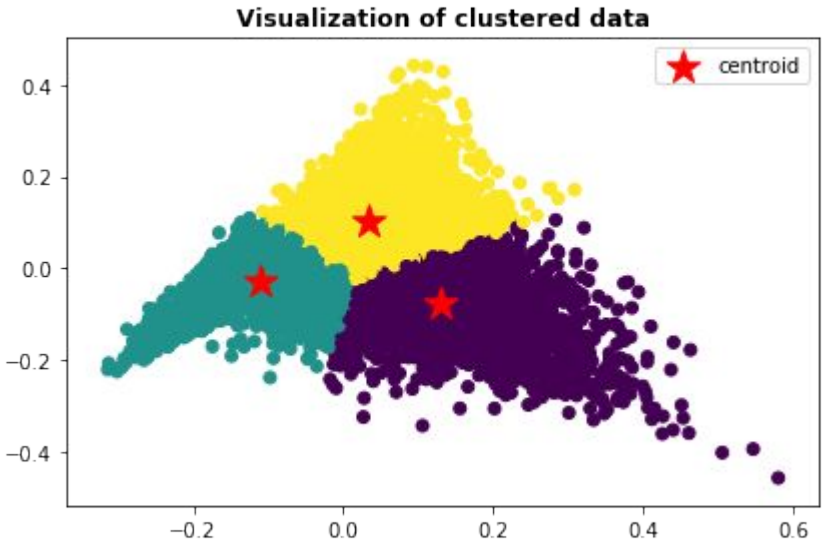
- List of 18828 News Documents
- List of 18828 pathnames of News Documents
- List of 18828 Newgroups each New Document belongs to

Clustering Dataset

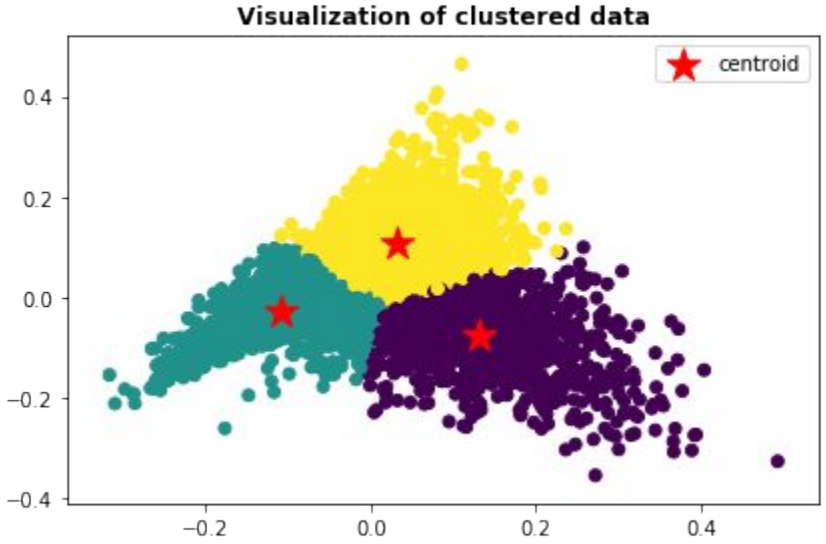
- 5000 features
- 18828 rows

K-Means Clustering

K-Means Training dataset



K-Means Test dataset



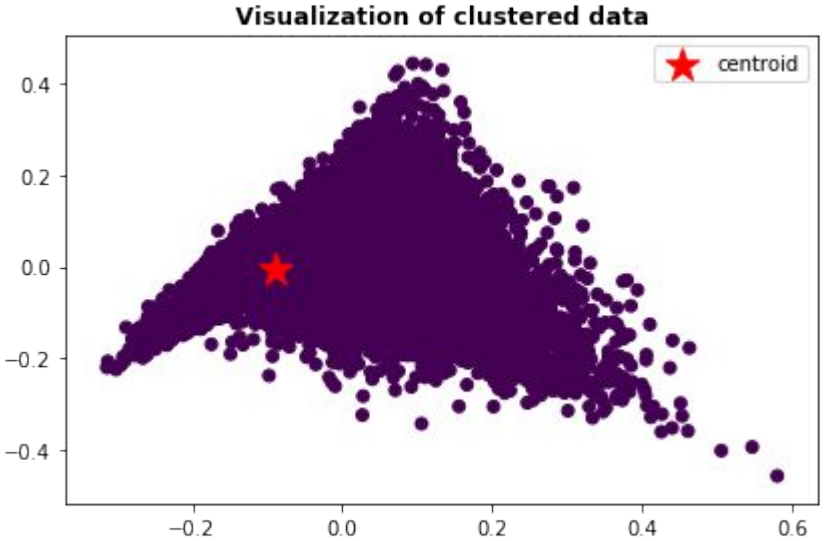
- **Somewhat radially symmetrical isotropic true clusters**
  - **-> somewhat captures underlying patterns**

K-Means Clustering Evaluation

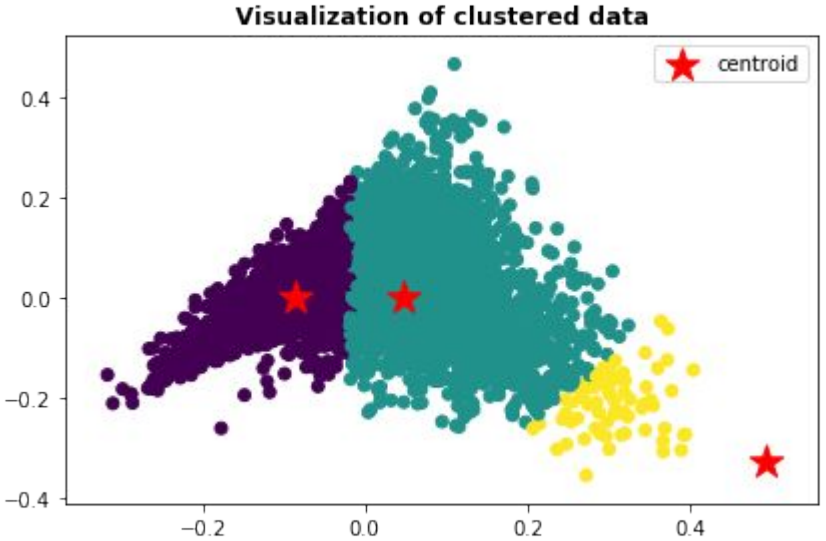
- ARI
  - **0.04 -**
    - **-> relation datapoint pairs ground truth & new solution -> close perfect randomness**
- Similarity Silhoutte Coefficient
  - **.007**
  - **.007**
  - **.006**
  - **.007**
    - **-> consistency coefficients of subsets**
    - **-> samples very close to neighboring clusters**

Mean Shift Clustering

Mean Shift Training dataset



Mean Shift Test dataset



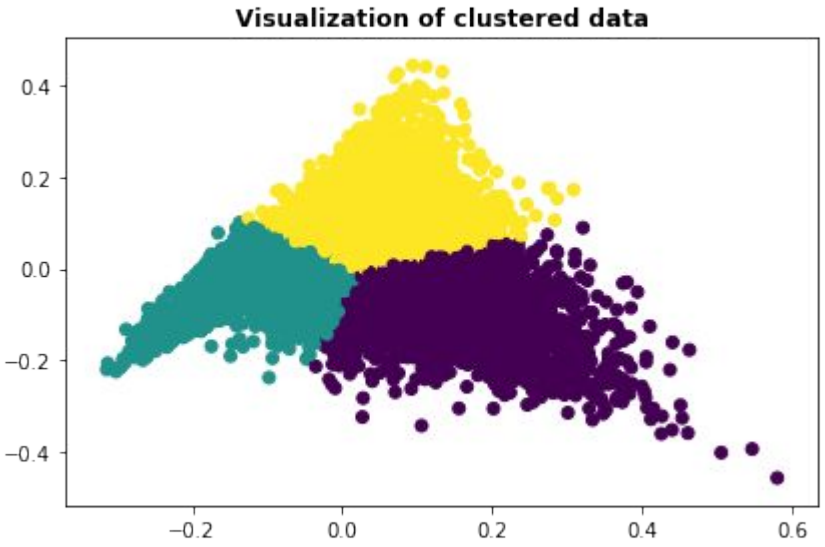
- Somewhat radially symmetric isotropic shape
  - **-> Somewhat captures underlying data patterns**

Mean Shift Clustering Evaluation

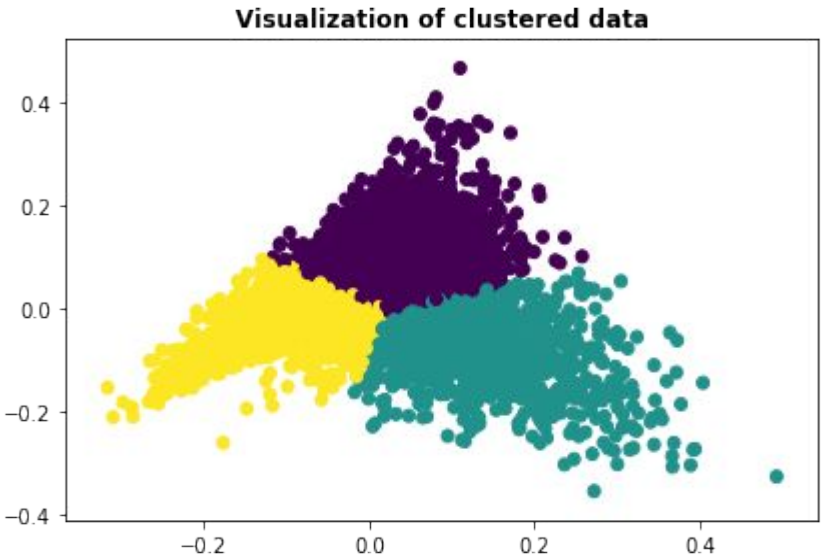
- ARI
  - **.0004**
    - **-> relation datapoint pairs ground truth & new solution -> close perfect randomness**
- Similarity Silhoutte Coefficient
  - **-.06**
  - **-.05**
  - **-.06**
  - **-.06**
    - **Consistency of coefficients between subsets**
    - **Samples assigned to WRONG clusters**

Spectral Clustering

Spectral Training dataset



Spectral Test dataset



- **Somewhat radially symmetric isotropic shape**
  - **-> Somewhat captures underlying data patterns**

Spectral Clustering Evaluation

- ARI
  - **.03**
    - **-> relation datapoint pairs ground truth & new solution -> close perfect randomness**
- Similarity Silhoutte Coefficient
  - **.007**
  - **.007**
  - **.006**
  - **.007**
    - **-> consistency coefficients of subsets**
    - **-> samples very close to neighboring clusters**

Clustering Algorithms Evaluation

WORST in Capturing data patterns:

- MeanShift
  - Least true cluster shape
  - ARI
    - **Ground truth vs new solution most close to perfect randomness**
  - Similarity Silhoutte Coefficient
    - **Negative**
      - **-> samples assigned to wrong clusters**

BEST in Capturing data patterns:

- K-Means vs Spectral
  - **K-Means**
    - **Slightly better ARI evaluation score**

4. In-Depth Analysis

Training vs Test Accuracy Score on ML models:

	Training data set Acc Score	Test data set Acc Score
Random Forest Classifier	**.994**	**.684**
Logistic Regression	**.991**	**.812**
Multinomial Naive Bayes	**.858**	**.802**

- **RFC ml model overfitting immensely -> captures alot of noise**
- **LR ml model overfitting -> still captures noise**
- **MNB ml model not overfitting + not underfitting**

5. Communicate Results of analysis (Potential Data Product)

Uncovered Insights for Proposal Implementation

- Multiclass Multinomial Naive Bayes best + solid ml model performer
  - Closest training and test mean accuracy scores with training being tad bit higher-
    - -> not overfitting
  - Training mean accuracy score decent
    - .858
      - -> not underfitting
  - For even closer + higher training and test mean accuracy scores;
    - dimension reduction on features?
    - experimenting with NLP & Neural Network features
    - tuning parameters

Link to Corresponding Folder & Jupyter Notebook:

Folder:

https://github.com/pman117/Data_Science_Portfolio/tree/master/unsupervised_learning_capstone

Jupyter Notebook:

https://github.com/pman117/Data_Science_Portfolio/blob/master/unsupervised_learning_capstone/Unsupervised_Learning_Capstone.ipynb