Leticia Lima Huang CIS9660- Data Mining for Business Analytics Technical report

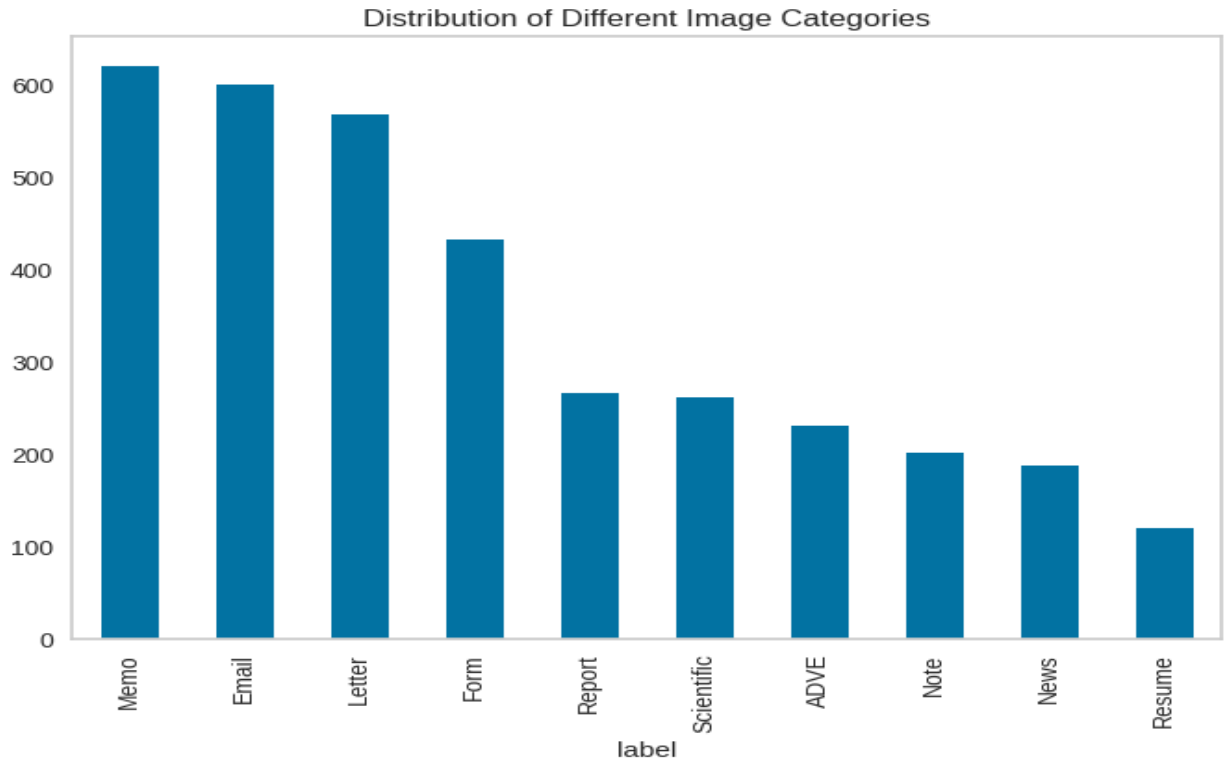
My classification project involved creating an app that sorts financial documents into categories, providing a useful tool for procurement professionals managing large volumes of paperwork. I selected this topic based on my experience conducting procurement through CUNY Buy, which frequently involves handling numerous financial documents. Such an application could streamline the process of reviewing and accurately sorting these documents by department. The dataset used comes from a Kaggle dataset (link available in my Jupiter notebook), but due to its size (about 9GB), I sampled the data to extract image features efficiently.

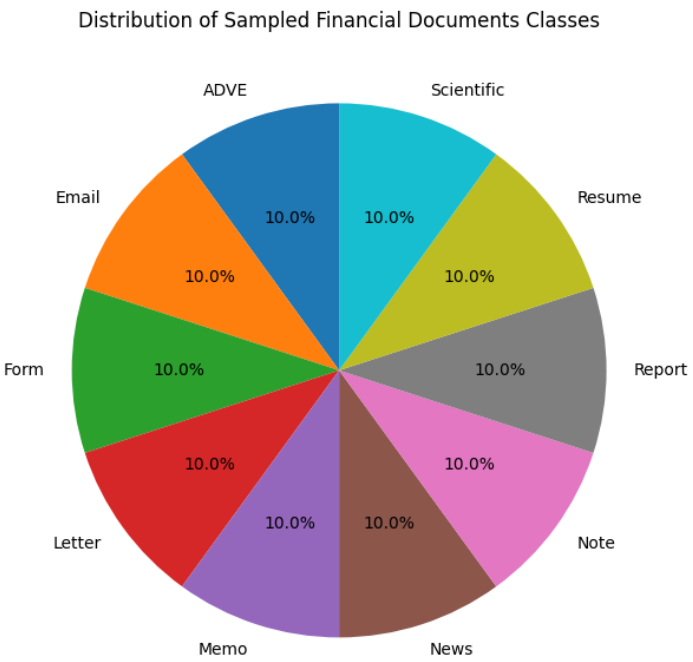
I started the project by loading my original dataset and revising its structure, which initial contained 3,482 images. Various dataset structures were tested to determine which would be most effective given system constraints and to achieve a reliable cross-validation score. A reduced dataset was created with 120 images per class to ensure fair distribution, as some classes had approximately 600 images while others, such as the resume category, only had 120. This adjustment aimed to prevent bias toward overrepresented classes. After creating my data frame, I reviewed sample documents, cleaned the data, and extracted image features for classification. One of the main considerations was whether to use TensorFlow, which offered faster and more accurate results, or Scikit-learn, a lightweight library compatible with Streamlit cloud deployment but slower in performance. Ultimately, TensorFlow was selected, and compatibility with Streamlit was achieved by using Python version 3.10, which supports the TensorFlow library. Just keep in mind that when working with the same repository on GitHub, it is advisable to create a file specifying the Python library used by each application. Some applications experienced failures due to dependencies on Python version 3.13. The use of config.toml to define the required libraries for each application successfully resolved these issues.

During preprocessing and model training, I first used K-Nearest Neighbors (KNN), finding an optimal K of 7 and tested various distance metrics, with both Euclidean and Minkowski yielding comparable results. A total of eleven models were assessed and evaluated using cross-validation; Logistic Regression achieved the highest accuracy at 63.33%. Unsupervised K-Means performed best with k=8 (score: 738093.81) and cluster accuracy of 16.11%, less accurate than supervised methods, that outperformed k-means by 47.22%. Overall, supervised models consistently achieved superior performance compared to unsupervised approaches. The confusion matrix revealed a satisfactory rate of correct predictions, with minor confusion observed between classifying 'letter' and 'report' due to similarities in image characteristics.

This model presents a valuable solution for environments that require daily sorting, offering the advantage of automating document classification and thereby reducing time spent on manual sorting. It enhances document organization by ensuring materials are correctly categorized and improves workflow efficiency when integrated with databases that route documents to the appropriate departments. By minimizing the need for manual intervention, it also contributes to cost savings through reduced staffing requirements.

Appendix


Distribution of Different Image Categories
600
500
400
300
200
100
0
Memo
Email
Letter
Form
Report
Scientific
ADVE
Note
News
Resume
label


Distribution of Sampled Financial Documents Classes
ADVE
Scientific
Email
Resume
Form
Report
Letter
Note
Memo
News
10.0%

Sample of Images included in the data frame

ADVE

ADVE

Email

Email

Form

Form

Letter

Letter

Memo

Memo

News

News

Note

Note

Report

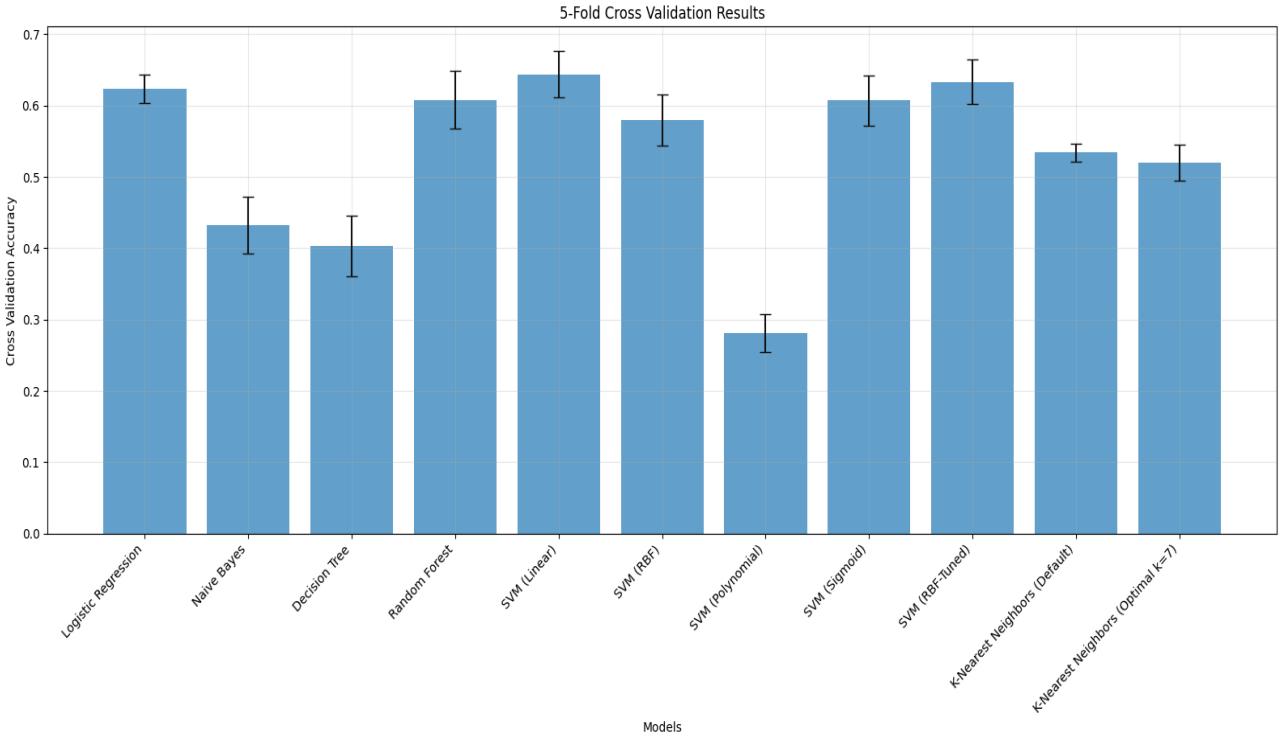
Report

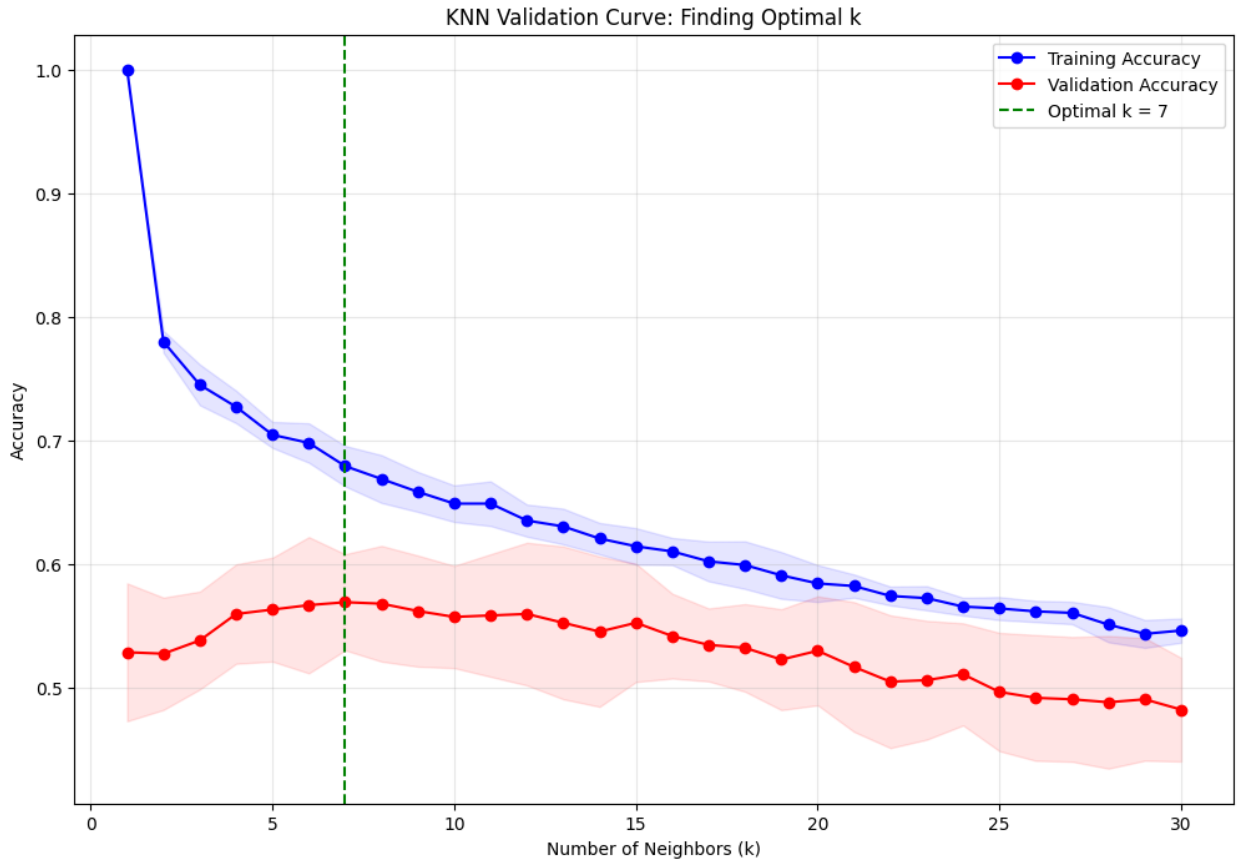
Resume

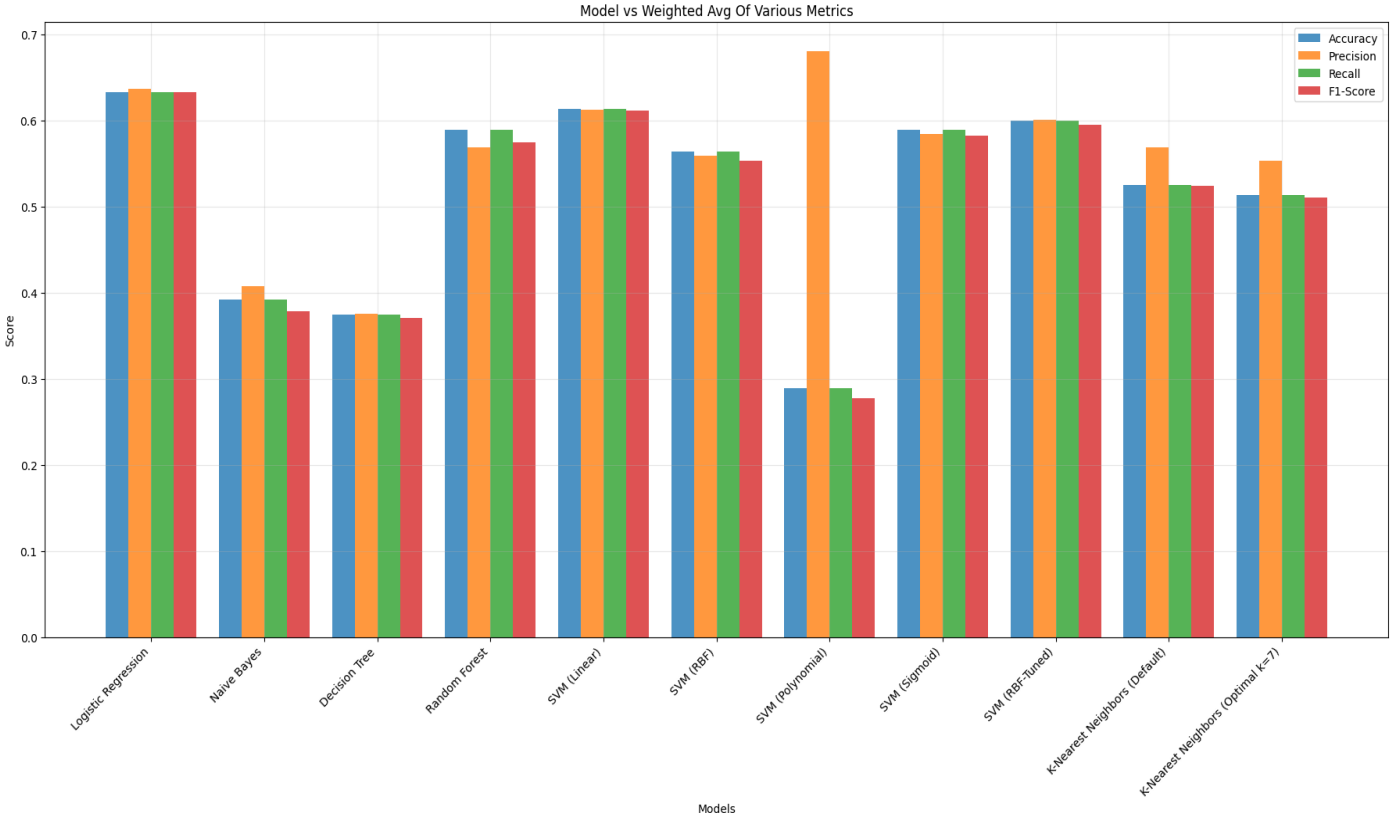
Resume

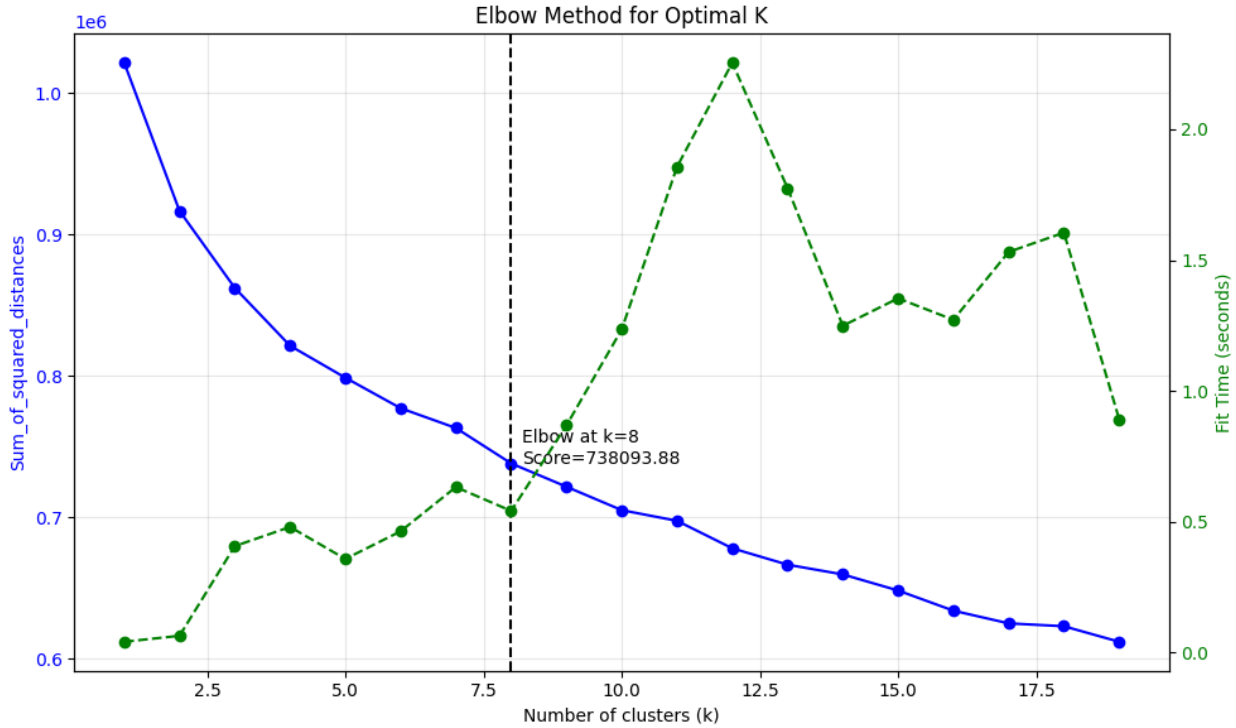
Scientific

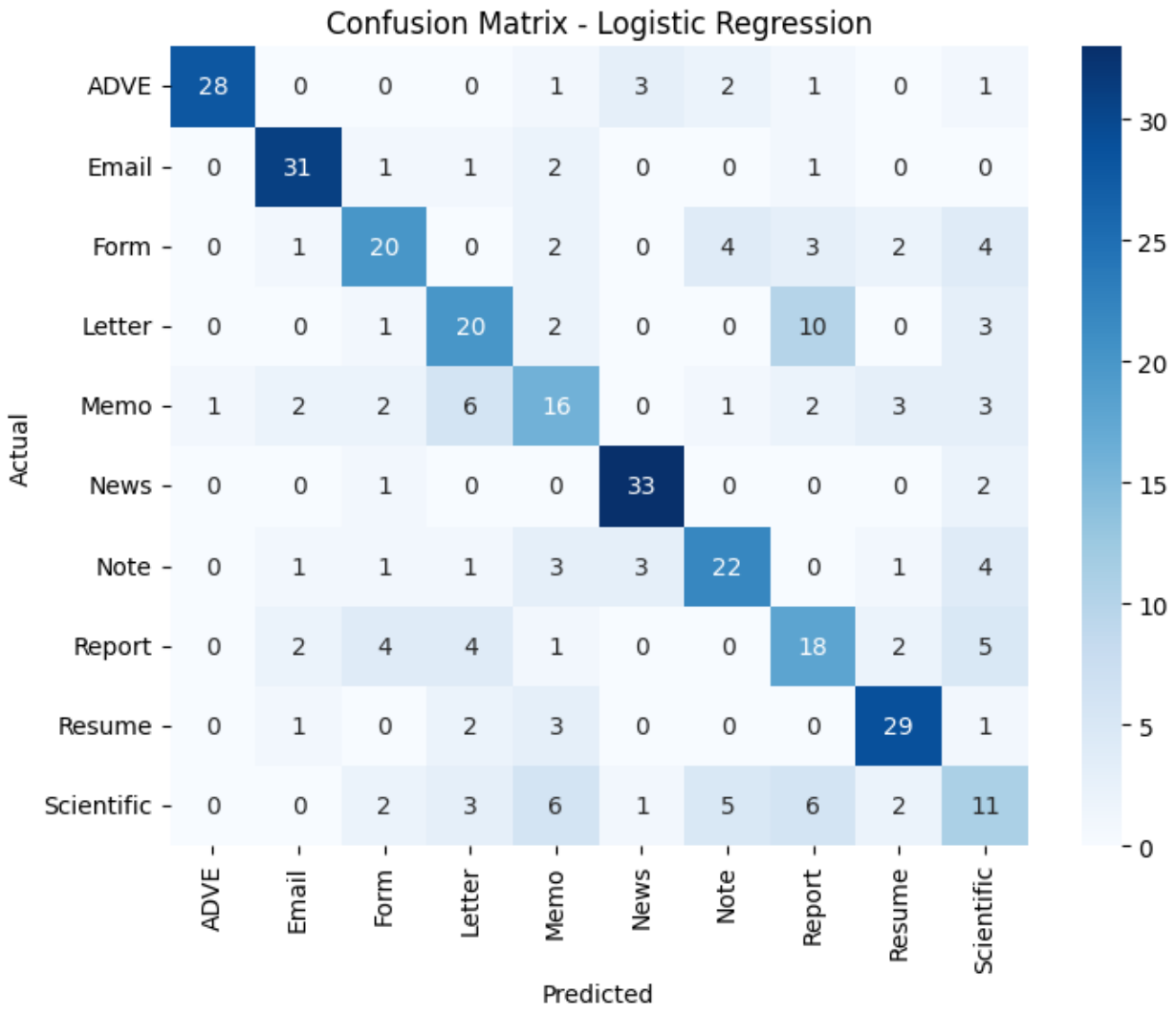
Scientific


5-Fold Cross Validation Results
Cross Validation Accuracy
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Logistic Regression
Naive Bayes
Decision Tree
Random Forest
SVM (Linear)
SVM (RBF)
SVM (Polynomial)
SVM (Sigmoid)
SVM (RBF-Tuned)
K-Nearest Neighbors (Default)
K-Nearest Neighbors (Optimal k=7)
Models


KNN Validation Curve: Finding Optimal k
Training Accuracy
Validation Accuracy
Optimal k = 7
Accuracy
1.0
0.9
0.8
0.7
0.6
0.5
0
5
10
15
20
25
30
Number of Neighbors (k)


Model vs Weighted Avg Of Various Metrics
Accuracy
Precision
Recall
F1-Score
Score
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Logistic Regression
Naive Bayes
Decision Tree
Random Forest
SVM (Linear)
SVM (RBF)
SVM (Polynomial)
SVM (Sigmoid)
SVM (RBF-Tuned)
K-Nearest Neighbors (Default)
K-Nearest Neighbors (Optimal k=7)
Models


Elbow Method for Optimal K
1e6
Sum_of_squared_distances
Fit Time (seconds)
Elbow at k=8
Score=738093.88
0.6
0.7
0.8
0.9
1.0
0.0
0.5
1.0
1.5
2.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Number of clusters (k)


Confusion Matrix - Logistic Regression
ADVE 28 0 0 0 1 3 2 1 0 1
Email 0 31 1 1 2 0 0 1 0 0
Form 0 1 20 0 2 0 4 3 2 4
Letter 0 0 1 20 2 0 0 10 0 3
Memo 1 2 2 6 16 0 1 2 3 3
News 0 0 1 0 0 33 0 0 0 2
Note 0 1 1 1 3 3 22 0 1 4
Report 0 2 4 4 1 0 0 18 2 5
Resume 0 1 0 2 3 0 0 0 29 1
Scientific 0 0 2 3 6 1 5 6 2 11
Actual
ADVE
Email
Form
Letter
Memo
News
Note
Report
Resume
Scientific
Predicted
30
25
20
15
10
5
0