# Assignment 2

**Important Note: While you are free to utilize any Python libraries for your models in Assignment 2, developing your classifier from scratch will earn you an additional 20% bonus points.**

# Question 1: Binary Classification with Custom Naive Bayes

Objective:

The objective of this assignment is to develop a binary classification model using the Naive Bayes algorithm. Students will gain hands-on experience in data loading, preprocessing, visualization, and model evaluation, applying statistical fundamentals to create an effective classifier.

Components:

* Data Loading and Preprocessing(15 points):
  + Load the dataset from the provided URL into a suitable data structure (like a pandas Data Frame).
  + Clean the data by handling missing values, and normalizing numerical features as needed.
  + Split the dataset into training and testing subsets to facilitate model evaluation.
* Data Visualization:
  + Generate visual plots (e.g., histograms, bar charts, box plots) to understand the distribution and characteristics of the dataset.
  + Visualize the class distribution to check for class imbalance and consider strategies like resampling if needed.
* Binary Classification Model Development:
  + Implement the Naive Bayes classifier. Document the mathematical formulation and programming logic used if you develop from scratch.
  + Optimize the model by experimenting with different techniques like feature selection and hyperparameter tuning.
* Code Implementation and Testing:
  + Write clean, modular, and well-documented Python code for the entire classification process.
  + Test the classifier on the test set and ensure your code produces reliable outputs.
* Performance Analysis:
  + Evaluate the classifier using metrics such as accuracy, precision, recall, F1 score, and ROC curve.
  + Draw a confusion matrix to understand true positives, true negatives, false positives, and false negatives.
* Reflection and Insights:
  + Reflect on the performance of the Naive Bayes classifier, providing an analysis of the results.
  + Discuss any observed limitations and propose potential improvements or future work that could enhance the classifier's performance.

## Submission Format:

* A Jupyter notebook with fully executed code and comments.
* A comprehensive report summarizing your methodology, results, and insights.

Notes:

* Ensure your code is well-commented to explain your logic and steps.
* Your report should be concise, focusing on significant findings and possible applications.
* Adhere to ethical guidelines and academic integrity in your analysis and reporting.

Dataset:

* The dataset to be used for this assignment is available at Support Descriptions Dataset.

**Evaluation:**

Data Loading and Preprocessing (15 points):

* Proper data loading and handling: 5 points
* Correct handling of missing values and data cleaning: 5 points
* Correct dataset splitting into training and testing sets: 5 points

Data Visualization (15 points):

* Clear and insightful visual plots: 10 points
* Accurate representation and analysis of class distribution: 5 points

Model Development (30 points):

* Correct Naive Bayes algorithm: 20 points
* Optimization through feature selection or hyperparameter tuning: 10 points

Code Implementation and Testing (15 points):

* Quality, modularity, and documentation of the code: 10 points
* Effective handling of edge cases and reliability of outputs: 5 points

Performance Analysis (15 points):

* Accurate calculation of performance metrics (accuracy, precision, recall, F1 score): 10 points
* Correct and insightful interpretation of the ROC curve and confusion matrix: 5 points

Reflection and Insights (10 points):

* Depth of reflection on model performance: 5 points
* Quality of insights and suggestions for improvement: 5 points Total: 100 points

**Additional Notes:**

* Each section must be completed for full credit in subsequent sections; partial credit may be awarded as appropriate.
* The quality of writing, clarity of explanations, and organization of the report will be considered in each section's grading.
* Late submissions may incur penalties as per course policy.
* Academic integrity is paramount; any evidence of plagiarism or cheating will result in disciplinary action.

# Question 2: Emotion Analysis using SVM and K-Means Clustering

Objective:

Develop an understanding of emotion recognition in text using Support Vector Machines (SVM) for classification and K-Means clustering for pattern discovery. This assignment will help you grasp the nuances of supervised and unsupervised learning techniques in Natural Language Processing (NLP).

Components:

* Data Preparation:
  + Load and familiarize yourself with the EmotionLines dataset.
  + Conduct text preprocessing: tokenize, stem/lemmatize, and remove stop words.
  + Transform text into numerical representations using TF-IDF.
* SVM for Emotion Classification:
  + Construct an SVM classifier to categorize emotions in text data.
  + Optimize the classifier by experimenting with different kernels and hyperparameters.
  + Validate the model using cross-validation and compute classification metrics.
* K-Means Clustering:
  + Implement K-Means clustering on the preprocessed text data.
  + Identify the optimal number of clusters with methods like the elbow technique.
  + Interpret the clusters to find patterns corresponding to different emotions.
* Model Insights:
  + Analyze the performance of the SVM classifier and the clusters formed by K-Means.
  + Compare and contrast the results obtained from both SVM and K-Means.
  + Offer insights into the emotional trends captured by the models.
* Report and Documentation:
  + Document your process, code, and results in a clear and structured report.
  + Reflect on the classifier’s performance and the clustering outcomes.
  + Suggest improvements and real-world applications of your findings.

Submission Format:

* A Jupyter notebook with fully executed code and comments.
* A comprehensive report summarizing your methodology, results, and insights.

Notes:

* Ensure your code is well-commented to explain your logic and steps.
* Your report should be concise, focusing on significant findings and possible applications.
* Adhere to ethical guidelines and academic integrity in your analysis and reporting.

Dataset:

The dataset to be used is the EmotionLines dataset, which can be found here: EmotionLines Dataset

Data Preparation (20 points):

* Data Loading (5 points): Correctly loading the dataset into a workable format.
* Preprocessing (10 points): Effective execution of tokenization, stemming/lemmatization, and stop words removal.
* Feature Transformation (5 points): Appropriate use of TF-IDF to convert text into numerical data.

SVM for Emotion Classification (30 points):

* Model Implementation (20 points): Proper implementation of the SVM algorithm.
  + Validation (10 points): Application of cross-validation and computation of classification metrics.

K-Means Clustering (20 points):

* + Algorithm Implementation (10 points): Correct application of the K-Means algorithm to the text data.
  + Optimal Clusters (10 points): Accurate determination of the number of clusters using appropriate methods.

Model Insights (20 points):

* + Performance Analysis (10 points): Detailed evaluation of the SVM classifier's performance.
  + Comparative Analysis (10 points): Insightful comparison between SVM classification and K-Means clustering results.

Report and Documentation (10 points):

* + Clarity and Structure (5 points): The report is well-organized, with clear explanations of methods and findings.
  + Insights and Applications (5 points): Thoughtful discussion on the implications of findings and potential real-world applications.

Total: 100 points Notes:

* + Partial credit may be awarded in each category for attempts that demonstrate

a good understanding but are not fully correct.

* + Points may be deducted for lack of clarity, incorrect execution, or incomplete analysis.
  + Students are expected to follow academic integrity guidelines; any form of plagiarism will result in a score of zero for the entire assignment.

For your assignment on Emotion Analysis using SVM (Support Vector Machines) and K-Means Clustering, you'll be delving into both supervised and unsupervised learning techniques within the field of Natural Language Processing (NLP). Here's a structured approach to start your assignment and the key concepts to focus on:

### 1. \*\*Data Preparation\*\*

- \*\*Key Concepts\*\*:

- \*\*Text Preprocessing\*\*: Learn about tokenization, stemming/lemmatization, and removing stop words. These are foundational NLP techniques.

- \*\*TF-IDF (Term Frequency-Inverse Document Frequency)\*\*: Understand how TF-IDF converts text into numerical representations, which is crucial for machine learning models.

- \*\*Resources\*\*:

- NLP books or online tutorials covering text preprocessing and TF-IDF.

### 2. \*\*SVM for Emotion Classification\*\*

- \*\*Key Concepts\*\*:

- \*\*Support Vector Machines\*\*: Understand the theory behind SVMs, how they work for classification tasks, and their effectiveness in high-dimensional spaces.

- \*\*Kernel Trick and Hyperparameter Tuning\*\*: Learn about different kernels (like linear, polynomial, and RBF) and how to tune hyperparameters for optimal performance.

- \*\*Model Validation\*\*: Cross-validation techniques and classification metrics (like accuracy, precision, recall).

- \*\*Resources\*\*:

- Textbooks on machine learning or online courses focusing on SVM.

### 3. \*\*K-Means Clustering\*\*

- \*\*Key Concepts\*\*:

- \*\*K-Means Algorithm\*\*: Grasp the basics of K-Means clustering, an unsupervised learning method.

- \*\*Optimal Number of Clusters\*\*: Techniques like the elbow method to determine the best number of clusters.

- \*\*Cluster Interpretation\*\*: Analyzing and understanding the clusters formed.

- \*\*Resources\*\*:

- Online tutorials or courses on clustering algorithms, particularly K-Means.

### 4. \*\*Model Insights\*\*

- \*\*Key Concepts\*\*:

- \*\*Performance Analysis\*\*: Evaluate and compare the performance of SVM and K-Means.

- \*\*Result Interpretation\*\*: Draw insights from the outcomes of both models.

- \*\*Resources\*\*:

- Case studies or research papers on emotion analysis using machine learning.

### 5. \*\*Report and Documentation\*\*

- \*\*Key Concepts\*\*:

- \*\*Effective Reporting\*\*: Skills in documenting your process, code, and results comprehensively.

- \*\*Critical Analysis\*\*: Ability to critique and suggest improvements.

- \*\*Resources\*\*:

- Guides on technical writing and report preparation.

### Getting Started

1. \*\*Familiarize with the Dataset\*\*: Start by exploring the EmotionLines dataset to understand its structure and content.

2. \*\*Sequential Approach\*\*: Begin with data preparation, followed by model building (SVM and then K-Means), and finally analysis and reporting.

3. \*\*Practical Implementation\*\*: Alongside studying the concepts, implement them in a step-by-step manner to gain practical insights.

### Online Resources

- \*\*Coursera\*\*, \*\*Udemy\*\*, and \*\*edX\*\* offer courses in machine learning and NLP.

- \*\*Kaggle\*\*: Find similar projects or kernels for practical insights.

- \*\*Scikit-Learn Documentation\*\*: For practical implementation of SVM, K-Means, and TF-IDF in Python.

- \*\*Stack Overflow\*\*: For coding-related queries.

This assignment is an excellent opportunity to apply theoretical concepts in a practical setting, enhancing your understanding of both supervised and unsupervised learning in the realm of NLP.