# CS60050_Machine Learning: Programming Assignment 3

> ### Part B: K-Means Clustering - Anuran Calls Dataset (MFCCs)

## Project Code: PartB_ChandranshSingh_22CS30017

- Data Preprocessing and Exploration:
  - The dataset was loaded, and essential libraries were imported.
  - Initial exploration included examining data distributions and scaling features using techniques like MinMaxScaler, helping prepare the dataset for clustering.
- Optimal Number of Clusters:
  - The Elbow Method and silhouette analysis were employed to determine the ideal cluster count.
  - This process aimed to balance cluster compactness and separation quality.
- Clustering Evaluation:
  - Metrics such as silhouette score, Davies-Bouldin Index, and Calinski-Harabasz Index were applied to evaluate clustering quality.
  - Silhouette Score: This metric helped in assessing how well-separated and compact the clusters were.
  - Davies-Bouldin Index and Calinski-Harabasz Index: Provided additional insights into inter-cluster distances and the ratio of within-cluster to between-cluster dispersion.

## Insights from Visualizations

Dimensionality reduction techniques, PCA, were likely used for visualizing clusters in 2D space, providing clarity on how well-separated the clusters appeared based on the chosen features.

## Limitations of K-Means and Other Clustering Algorithms

- K-Means:
  - Strengths: Efficient and effective when clusters are spherical and evenly distributed. It’s also computationally scalable.
  - Limitations: Sensitive to outliers and assumes clusters are spherical, which might not suit the real structure of this dataset. K-Means also requires specifying the number of clusters beforehand, which might not always align with natural groupings.
- Agglomerative Hierarchical Clustering:

    - Strengths: Does not require a predefined cluster count and can reveal hierarchical relationships.
    - Limitations: High computational demand makes it less suitable for large datasets, and the outcome can vary based on the chosen linkage method.
- DBSCAN:
    - Strengths: Can identify clusters of arbitrary shapes and is robust against outliers, labeling sparse data points as noise.
    - Limitations: Parameter sensitivity (eps and min_samples) and difficulty handling clusters of varying density can limit DBSCAN's performance, especially in higher dimensions