Gaussian Mixture Model (GMM) on CIFAR-10 with PCA

Date: 29/01/2025

1. Tasks Assigned

Task 1: Study Gaussian Mixture Models (GMMs)

* Objective: Develop a thorough understanding of GMMs and their applications.
* Progress:
  + Studied fundamental concepts from Bishop's *Pattern Recognition and Machine Learning*, including:
    - Gaussian distributions and their significance in data modeling.
    - The Expectation-Maximization (EM) algorithm for GMM parameter estimation..

Task 2: Classification on CIFAR-10 using GMM

* Objective: Perform image classification on CIFAR-10 using GMM with dimensionality reduction through PCA.
* Progress:
  + Completed the workflow and evaluated the model's performance.

2. Workflow for CIFAR-10 Classification

Import Data

* Load the CIFAR-10 dataset batches (5 training batches and 1 test batch).
* Combine training batches into a single dataset.

Preprocessing

* Normalize the pixel values of the images to the range [0, 1].
* Flatten the images into 1D arrays.
* Apply Principal Component Analysis (PCA) to reduce dimensionality (e.g., 100 components).

Train-Test Split

* Use the provided training and testing split from the dataset (no manual split required).

Fit the GMM Model

* Define a Gaussian Mixture Model (GMM) with:
  + Number of components equal to the number of classes (10 for CIFAR-10).
  + Full covariance type.
* Train the GMM using the PCA-transformed training data.

Cluster-Label Mapping

* Assign GMM clusters to actual CIFAR-10 labels using majority voting on training data.

Evaluate on Test Data

* Transform the test data using the same PCA model.
* Predict clusters for test data using the trained GMM.
* Map predicted clusters to labels using the majority-voting mapping.
* Compute accuracy by comparing predictions with ground truth labels.

Single Image Prediction

* Take a single image from the test set.
* Normalize and flatten it.
* Transform the image using the fitted PCA model.
* Predict the cluster using the GMM and map it to the corresponding label.

3. Results and Observations

* Report the overall accuracy (e.g., 29% for PCA with 100 components).
* Note the incorrect classification for a single test image.

4. Code Reference

* Google Colab Link:  
  <https://colab.research.google.com/drive/1MI_UI6TcZj-orhuTgpy53XHdaZoG_C-R?usp=sharing>