Classification with Gaussian Mixture Models: Theory and Application on the Old Faithful Dataset

Tudor Gulin

Abstract

This project explores the classification of data using Gaussian Mixture Models (GMMs). The theoretical framework relies on the Expectation-Maximization (EM) algorithm to estimate the parameters of the mixture—means, covariances, and mixing weights. The implementation uses the `sklearn.mixture.GaussianMixture` library and is applied to the classic Old Faithful Geyser dataset. The workflow follows a standard semi-supervised approach: training the model on 70% of the data to estimate parameters Θ and classifying the remaining 30% by maximizing the posterior probability.

1 Introduction

Gaussian Mixture Models (GMMs) are probabilistic models that assume all data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. Unlike K-Means, which assigns hard clusters, GMMs provide a soft classification based on probability density functions. This project implements a GMM classifier to distinguish between the two eruption modes of the Old Faithful geyser.

2 Theoretical Framework

We consider a dataset $Y = \{y^{(1)}, \dots, y^{(N)}\}$. The probability density function of a mixture with K components is defined as:

$$p(y) = \sum_{k=1}^{K} \alpha_k \cdot \mathcal{N}(y \mid \mu_k, \Sigma_k) \tag{1}$$

where:

- α_k is the mixing weight of component k, with $\sum \alpha_k = 1$,
- μ_k is the mean vector of component k,
- Σ_k is the covariance matrix of component k,
- $\mathcal{N}(y \mid \mu_k, \Sigma_k)$ is the multivariate Gaussian density.

The set of parameters to be estimated is $\Theta = \{\alpha_1, \dots, \alpha_K, \mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K\}$.

2.1 Parameter Estimation: The EM Algorithm

Since the latent variables (cluster assignments) are unobserved, we cannot compute the parameters directly. We use the **Expectation-Maximization (EM)** algorithm to maximize the log-likelihood:

$$\ln p(Y \mid \Theta) = \sum_{i=1}^{N} \ln \left(\sum_{k=1}^{K} \alpha_k \mathcal{N}(y^{(i)} \mid \mu_k, \Sigma_k) \right)$$

The algorithm iterates between two steps until convergence:

2.1.1 E-Step (Expectation)

We compute the posterior probability (responsibility) that data point $y^{(i)}$ belongs to component k:

$$\gamma_{ik} = \frac{\alpha_k \mathcal{N}(y^{(i)} \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \alpha_j \mathcal{N}(y^{(i)} \mid \mu_j, \Sigma_j)} \tag{2}$$

2.1.2 M-Step (Maximization)

We update the parameters Θ using the responsibilities computed in the E-step:

$$\mu_k^{new} = \frac{1}{N_k} \sum_{i=1}^{N} \gamma_{ik} y^{(i)} \tag{3}$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{i=1}^{N} \gamma_{ik} (y^{(i)} - \mu_k^{new})(y^{(i)} - \mu_k^{new})^T \tag{4}$$

$$\alpha_k^{new} = \frac{N_k}{N} \tag{5}$$

where $N_k = \sum_{i=1}^{N} \gamma_{ik}$ is the effective number of points in cluster k.

3 Implementation

3.1 Dataset: Old Faithful

The Old Faithful dataset contains 272 observations with two features:

- **Eruptions:** Duration of the eruption (minutes).
- **Waiting:** Time until the next eruption (minutes).

The data exhibits two natural clusters: short eruptions followed by short waits, and long eruptions followed by long waits.

3.2 Methodology

The classification pipeline consists of four steps:

1. **Preprocessing:** The features are standardized using Z-score scaling ($z = \frac{x-\mu}{\sigma}$) to ensure that Euclidean distances are not dominated by the larger magnitude of the "Waiting" variable.
2. **Splitting:** The data is split into a training set (70%) and a test set (30%).
3. **Training:** A GMM with $K = 2$ components and full covariance is fitted on the training set using `sklearn.mixture.GaussianMixture`. This step runs the EM algorithm to estimate Θ.
4. **Classification:** For each point in the test set, we calculate the posterior probability for both clusters and assign the label corresponding to the maximum probability:

$$\text{Class}(y_{new}) = \arg \max_{k \in \{1,2\}} P(C_k \mid y_{new})$$

4 Results and Visualization

4.1 Classification Map

The classifier successfully separates the dataset into two distinct groups. Since the components are well-separated in the feature space, the decision boundary (where $P(C_1|y) \approx P(C_2|y)$) lies in the gap between the two clusters. The accuracy on the test set is near 100%, confirming that the bimodal Gaussian assumption fits the physical reality of the geyser.

4.2 Density Estimation

The 1D density plots (projections of the 2D Gaussians) reveal two distinct "bell curves."

- **Cluster 1:** Centered around shorter eruption times (≈ 2 min).
- **Cluster 2:** Centered around longer eruption times (≈ 4.5 min).

The minimal overlap between the curves indicates high confidence in the classification model.

5 Conclusion

This project demonstrates that GMM is a powerful tool for unsupervised pattern recognition. By estimating the parameters Θ via the EM algorithm, we successfully modeled the underlying generative process of the Old Faithful geyser and built a robust classifier for predicting eruption types.