STAT 570 Probabilistic Machine Learning
Assignment 3

For this assignment, you can use either R or Python. There is no automatic testing component. Submit your code and your write-up in PDF format. Submit all code for Problem 1 in `problem1.py` (or `problem1.R`) and for Problem 2 in `problem2.py` (or `problem2.R`).

Problem 1: Gibbs sampling for Dirichlet process mixture models

In this problem, we will implement Gibbs algorithms to sample from Dirichlet process mixture models described in "R. M. Neal, Markov Chain Sampling Methods for Dirichlet Process Mixture Models. J. Comput. Graph. Stat. 9, 249–265 (2000)".

The data is given in `Y.csv`. This data was generated from a mixture of bivariate Normal distributions with K components:

$$\begin{aligned}
\pi &\sim \text{Dirichlet}(1/K, ..., 1/K) && (1)\\
z_i &\sim \text{Categorical}(\pi) && (2)\\
y_i|z_i, \theta &\sim \text{BivariateNormal}(\mu_{z_i}, \Gamma_{z_i}^{-1}). && (3)
\end{aligned}$$

The model parameters of interest are the data likelihood parameters $\theta_k = (\mu_k, \Gamma_k)$, where Γ_k denotes the 2×2 precision matrix.

Question 1 (30 points)

To start, we will make a simplifying assumption that Γ_k is a diagonal matrix:

$$\Gamma_k = \begin{pmatrix} \gamma_{k,1} & 0 \\ 0 & \gamma_{k,2} \end{pmatrix}.$$

For prior, assume $\mu_k \sim \text{BivariateNormal}(0, I)$ and $\gamma_{k,p} \sim \text{Gamma}(\alpha, \theta)$, where α, θ denote the shape and scale parameters. We will fix $\alpha = 1$ and $\theta = 0.5$.

Your tasks are to implement MH-within-Gibbs, Algorithm 2 of Neal (2000) and analyze the data.

- Specify the proposal distributions for μ_k and $\gamma_{k,1}, \gamma_{k,2}$.
- Analyze the data using Algorithm 2. Describe how you infer the clustering structure from your MCMC samples. For example, explain how you determined cluster assignments or the number of clusters from your posterior. Generate a figure using the posterior samples to support your conclusion.
- Provide your estimates for μ_k and Γ_k for $k = 1, ..., K$. Generate a contour plot using your final estimates of the mean and covariance parameters similar to Figure 1.

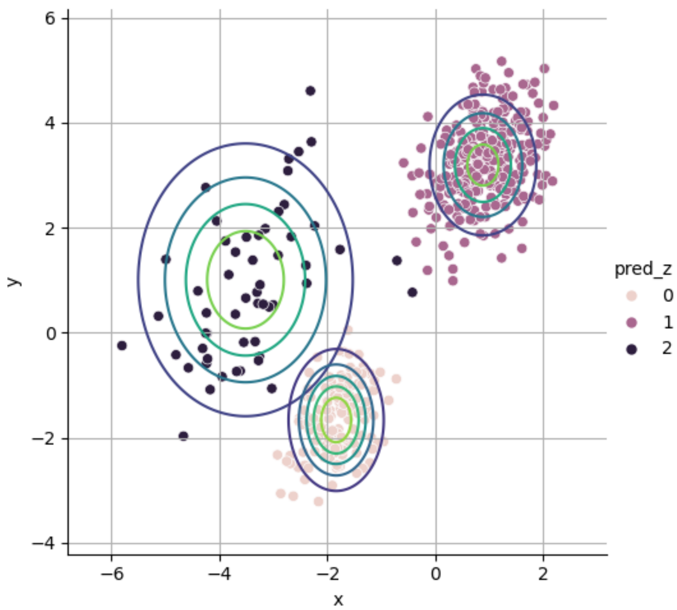

Figure 1: Contour plot of Bivariate normal densities with diagonal precision matrices.

Question 2 (40 points)

In Question 1, we assumed that the precision matrix is diagonal. In this question, we will implement Algorithm 3 of Neal (2000) but with full precision matrix. To that end, we will use Wishart distribution as the base measure, which is specified by two parameters, the degree of freedom $\nu > 0$ and a positive definite matrix V (referred to as scale matrix).

- Clearly specify your chosen Wishart hyperparameters and briefly justify these choices by considering interpretability, numerical stability, and whether the prior is informative or weakly informative. You can illustrate your justification by sampling and visualizing random precision matrices (using `scipy.stats.wishart` in Python or `rWishart` in R).
- Analyze the data using Algorithm 3. Describe how you infer the clustering structure from your MCMC samples. For example, explain how you determined cluster assignments or the number of clusters from your posterior. Generate a figure using the posterior samples to support your conclusion.
- Provide your estimates for μ_k and Γ_k for $k = 1, ..., K$. Generate a contour plot using your final estimates of the mean and covariance parameters similar to Figure 2.

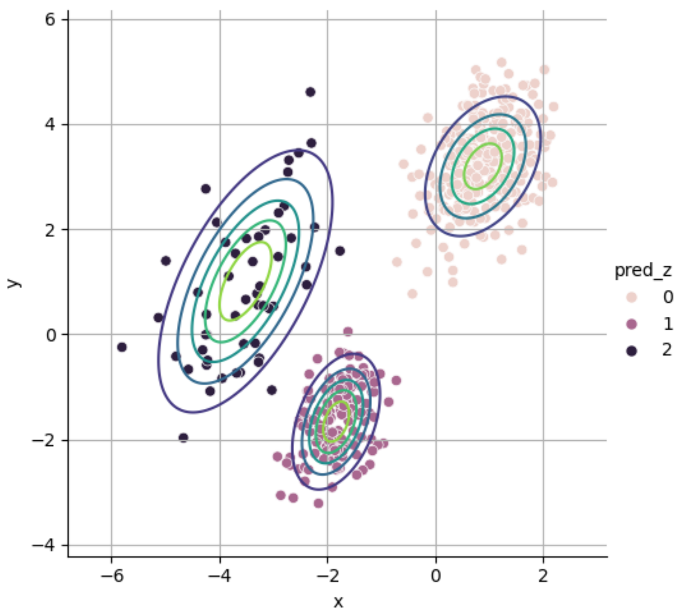

Figure 2: Contour plot of Bivariate normal densities with full precision matrices.

Problem 2: Particle filter (30 points)

We will reproduce the results from "P. Fearnhead, P. Clifford, On-line inference for hidden Markov models via particle filters. J. R. Stat. Soc. Series B Stat. Methodol. 65, 887–899 (2003)."

The well-log data is provided in `welldata.csv`. Your task is to reproduce Figure 2 in Fearnhead and Cliffod (2003).

- An implementation of the optimal sampling algorithm described in Section 4 of the paper is provided in `problem2.py`.

- Implement a particle filter for online inference in a hidden Markov model (HMM). Use the model specification described in Section 3 of the paper.
- Apply your particle filter to the `welldata.csv` data.
- Reproduce Figure 2, which shows the posterior mean of the hidden states at each time point, overlaid on the observed data.