1 Mixture Model EM Derivations

The Q function is given by

$$Q(\mathcal{D}, \theta, \pi, \phi) = \sum_{n=1}^{N} \sum_{z=1}^{K} \phi_{zn} \left(\log P(\mathbf{X} = \mathbf{x} | \mathbf{Z} = \mathbf{z}, \theta_z) + \log P(\mathbf{Z} = \mathbf{z}, \pi) - \log \phi_{zn} \right)$$

The objective function can be formulated as
$Q^* = \arg\max_{\theta, \phi, \pi} Q(\mathcal{D}, \theta, \pi, \phi)$
Subject to the constraints,

1. $\sum_{z=1}^{K} \phi_{zn} = 1 \quad \forall n$
2. $\phi_{zn} \geq 0 \quad \forall n$ and $z \in [1, K]$
3. $\sum_{z=1}^{K} \pi_z = 1$
4. $\pi_z \geq 0 \quad$ For $z \in [1, K]$

The Lagrangian form of the function is given by

$$\mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta) = \sum_{n=1}^{N} \sum_{z=1}^{K} \phi_{zn} \left(\log P(\mathbf{X} = \mathbf{x} | \mathbf{Z} = \mathbf{z}, \theta_z) + \log P(\mathbf{Z} = \mathbf{z}, \pi) - \log \phi_{zn} \right)$$
$$- \alpha \left(\sum_{z=1}^{K} \phi_{zn} - 1 \right) - \beta \left(\sum_{z=1}^{K} \pi_z - 1 \right) - \sum_{z=1}^{K} \gamma_{zn} \phi_{zn} - \sum_{z=1}^{K} \delta_z \pi_z$$

Where the Lagrangian multipliers are given by $\alpha, \beta, \gamma, \delta$

1.1 Lagrange multipliers ϕ

To maximize $Q(\mathcal{D}, \theta, \pi, \phi)$ with respect to ϕ, we differentiate with respect to ϕ_{zn}.

$\nabla_{\phi_{zn}} \mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta)$

$= \frac{\partial}{\partial \phi_{zn}} \mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta)$

$= \log P(\mathbf{X} = \mathbf{x} | \mathbf{Z} = \mathbf{z}, \theta) + \log P(Z = z | \pi) - \phi_{zn} \frac{1}{\phi_{zn}} - \log \phi_{zn} - \alpha - \gamma_{zn} \equiv 0$

$\Rightarrow \log P(\mathbf{X} = \mathbf{x} | \mathbf{Z} = \mathbf{z}, \theta) + \log P(Z = z | \pi) = 1 + \log \phi_{zn} + \alpha + \gamma_{zn}$

Substituting this equation in the Lagrangian,

$$\mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta) = \sum_{n=1}^{N} \sum_{z=1}^{K} \phi_{zn} \left(1 + \alpha + \gamma_{zn}\right) - \alpha \left(\sum_{n=1}^{N} \sum_{z=1}^{K} \phi_{zn} - 1 \right) - \beta \left(\sum_{z=1}^{K} \pi_z - 1 \right) - \sum_{n=1}^{N} \sum_{z=1}^{K} \gamma_{zn} \phi_{zn} - \sum_{z=1}^{K} \delta_z \pi_z$$

1.2 Lagrange multipliers π

To maximize $Q(\mathcal{D}, \theta, \pi, \phi)$ with respect to π, we differentiate with respect to π_z.

$\nabla_{\pi_z} \mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta)$

$= \frac{\partial}{\partial \pi_z} \mathcal{L}(\phi, \pi, \alpha, \beta, \gamma, \delta)$
$= \phi_{zn} \left(\frac{\partial}{\partial \pi_z} \log P(Z = z | \pi) \right) - \beta - \delta_z$
$= \phi_{zn} \left(\frac{\partial}{\partial \pi_z} \log \pi_z \right) - \beta - \delta_z$
$= \frac{\phi_{zn}}{\pi_z} - \beta - \delta_z \equiv 0$

2 Laplacian Mixture Models

2.1 Log marginal likelihood

The Marginal Likelihood is given by

$$P(\mathbf{X} = \mathbf{x}, \theta) = \sum_{z=1}^{K} P(\mathbf{X} = \mathbf{x}|\mathbf{Z} = \mathbf{z}, \theta_z) P(\mathbf{Z} = \mathbf{z}|\pi_z)$$

The Log marginal likelihood over the dataset $\mathcal{D}$ is given by

$$\mathcal{L}(\mathcal{D}, \theta) = \sum_{n=1}^{N} \log \left(\sum_{z=1}^{K} P(\mathbf{X} = \mathbf{x}|\mathbf{Z} = \mathbf{z}, \theta_z) P(\mathbf{Z} = \mathbf{z}|\pi_z) \right)$$

Substituting the values of the Laplacian and Multinoulli distributions for the mixure component and distribution respectively, we obtain

$$\mathcal{L}(\mathcal{D}, \theta) = \sum_{n=1}^{N} \log \left(\sum_{z=1}^{K} \prod_{d=1}^{D} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \ \pi_z \right)$$

2.2 Posterior distribution

The posterior distribution is given by

$$P(Z = z|\mathbf{X} = \mathbf{x}, \theta) = \frac{P(\mathbf{X}=\mathbf{x}|\mathbf{Z}=\mathbf{z}, \theta_z) P(\mathbf{Z}=\mathbf{z}|\pi_z)}{\sum_{m=1}^{K} P(\mathbf{X}=\mathbf{x}|\mathbf{Z}=\mathbf{m}, \theta_z) P(\mathbf{Z}=\mathbf{m}|\pi_z)}$$

$$= \frac{\prod_{d=1}^{D} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \ \pi_z}{\sum_{m=1}^{K} \prod_{d=1}^{D} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \ \pi_m}$$

2.3 Marginal likelihood maximization method

The function $P(Z = z|X = x)$ includes a product of exponents term $\prod_{d=1}^{D} \exp\left(-\frac{|x_d - \mu_{dz}|}{b_{dz}}\right)$. If the exponent is a large value, this may cause numerical overflow. To perform the computation safetly, we can take the log on both sides and by using Jensen's inequality, we can . *PyTorch* and *scipy* provide the **logsumexp** method, which is neumerically stabilized.

2.4 Unconstrained parameters

The Log marginal likelihood using the unconstrained parameters is given by,

$$\mathcal{L}(\mathcal{D}, \theta) = \sum_{n=1}^{N} \log \left(\sum_{z=1}^{K} \prod_{d=1}^{D} \frac{1}{2\exp(b'_{dz})} \exp(-\frac{|x_d - \mu_{dz}|}{\exp(b'_{dz})}) \cdot \frac{\exp(\pi'_z)}{\sum_{m=1}^{K} \pi'_m} \right)$$

2.5 Implementation

1. **Library** For the implementation of the **fit** method, I used **PyTorch** and its *autograd* mechanics. The *ndarray* were converted to PyTorch tensors. *DataLoader* to sample the data. Experimenting with various batch sizes, **64** was chosen, since it provided the best performance compared to the reference model.

2. **Optimizer** I tried various optimizers such as Adam, SGD and RMSProp (with and without momentum). SGD with momentum had higher error as compared to without it. **Adam**, with the default values for β_1 (0.9) and β_2 (0.99), provided the best results.

3. **Parameters**: Parameters were sampled from a standard normal distribution.

4. **Convergence**: I ran the training for 20 epochs, while monitoring the log likelihood.

2.6 Plot of K

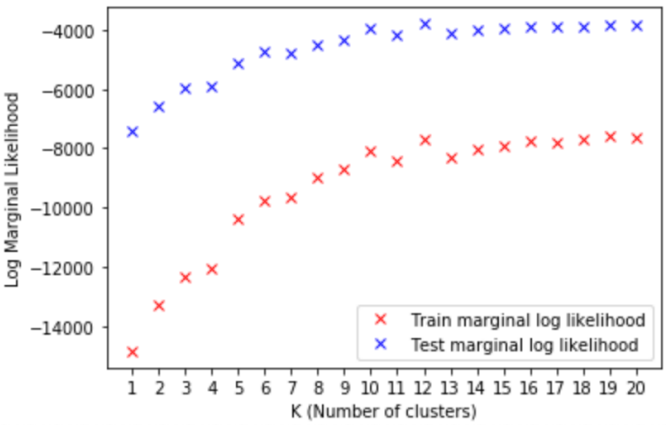

Figure 1: Log marginal likelihood for Train and test data. Trained on the entire dataset instead of batches

As we see, the marginal log likelihood improves with the number of clusters.

3 Mixture models and missing data

3.1 Closed form expression

Let $\mathbf{D_o}$ represent the observed dimensions and $\mathbf{D_m}$ represent the missing dimensions.
The distribution over observed is obtained by marginalizing the missing dimensions.

$$P(\mathbf{X_o} = \mathbf{x_o}|Z = z, \theta_z) = \int \left(\prod_{d=1}^{D} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \right) d\mathbf{x_m}$$

$$= \int \left(\prod_{d=1}^{D_o} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \prod_{d=1}^{D_m} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \right) d\mathbf{x_m}$$

The observed dimensions $\mathbf{D_o}$ are constant with respect to $\mathbf{x_m}$. So we can take them outside the integral. Moreover, the integral over the missing dimensions $\mathbf{D_m}$ is $\mathbf{1}$ (By Law of Total Probability). We can simplify as,

$$= \prod_{d=1}^{D_o} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}})$$

3.2 Posterior distribution

The posterior distribution is given by

$$P(Z = z|\mathbf{X_o} = \mathbf{x_o}, \theta) = \frac{P(\mathbf{X}=\mathbf{x_o}|\mathbf{Z}=\mathbf{z},\theta_z)P(\mathbf{Z}=\mathbf{z}|\pi_z)}{\sum_{m=1}^{K} P(\mathbf{X_o}=\mathbf{x_o}|\mathbf{Z}=\mathbf{m},\theta_z)P(\mathbf{Z}=\mathbf{m}|\pi_z)}$$

$$= \frac{\prod_{d=1}^{D_o} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \cdot \pi_z}{\sum_{z'=1}^{K} \prod_{d=1}^{D_o} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \cdot \pi_{z'}}$$

3.3 Imputation

$P(Z = z|\mathbf{X_o} = \mathbf{x_o})$ gives a distribution over the predictions of the clusters for a given datapoint, conditioned on the observed dimensions.

We impute the missing value by the mean of the cluster for a given dimension. Since we only have the distribution of cluster predcitions, we obtain a distribution over the missing values $P(X_d = x_d|\mathbf{X_o} = \mathbf{x_o})$. Therefore, the missing value is given by $\mathbb{E}[P(X_d = x_d|\mathbf{X_o} = \mathbf{x_o})]$, which is equivalent to the dot product of the dimension means ($\mathbb{R}^K$) and the cluster probabilities ($\mathbb{R}^K$).

$$x_d = < \mu_d \cdot P(Z = z|\mathbf{X_o} = \mathbf{x_o}) > \quad d \in \{D\}$$

This can be further generalized to N data points, by replacing dot product with matrix multiplication.

3.4 Marginal likelihood

The marginal likelihood is given by

$$\sum_{z=1}^{K} P(\mathbf{X_o} = \mathbf{x_o}|\mathbf{Z} = \mathbf{z}, \theta_z)P(\mathbf{Z} = \mathbf{z}|\pi_z)$$

Substituting the value of $P(\mathbf{X_o} = \mathbf{x_o}|\mathbf{Z} = \mathbf{z}, \theta_z)$ from above, we get

$$= \sum_{z=1}^{K} \prod_{d=1}^{D_o} \frac{1}{2b_{dz}} \exp(-\frac{|x_d - \mu_{dz}|}{b_{dz}}) \cdot \pi_z$$

3.5 Plot of K

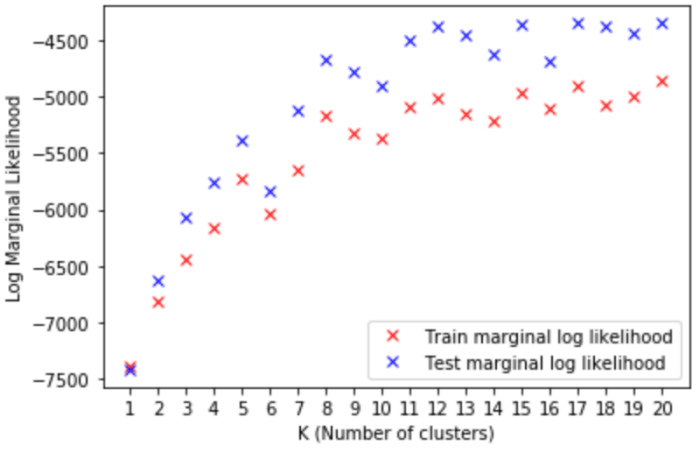

Figure 2: Log marginal likelihood for Train and test data. Trained on the entire dataset instead of batches

As we see, the marginal log likelihood improves with the number of clusters. However, the model trained on the complete data provides better marginal log likelihood as compared to the model trained on incomplete data.

3.6 Test

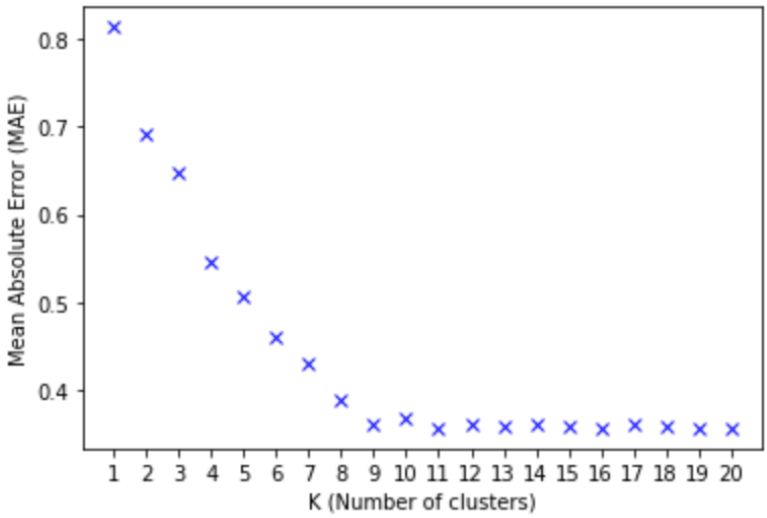

Figure 3: Mean Absolute error vs K (cluster) size

As we can see from the plot, as the size of the clusters increases, the log marginal likelihood increases. Similarly, the Mean absolute imputation error decreases. Therefore, for $K = 20$, we get the highest test log marginal likelihood and consequently the least imputation error.