# Advanced Probabilistic Machine Learning and Applications

Pablo Sánchez and Isabel Valera

## 1 Tutorial 4: Infinite Mixture Model (iMM)+ Gibbs sampling

In this tutorial we will continue working with the CMM and Twitter data-set presented in Tutorial 2. We will use Bayesian nonparametric (BNP) to assume (a priori) infinite number of mixture components $K \to \infty$. In practice, we will only be able to "see" a finite number, $K^+$, of components $K^+ < N-1 << \infty$ which is constrained by the number of observations in our data-set. Again, we will rely on the Gibbs sampling algorithm to infer the posterior distribution of the unknown variables and parameters in our model.

### Introduction

**Notation:** Through this document we will use the following notation:

- $K^+$: number of "seen" mixture components, i.e., we interpret them as topics/clusters.
- $N$: number of documents, i.e., tweets.
- $I$: dictionary
- $|I|$: number of words in $I$.
- $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_k\}_{k=1}^{K^+}$: set of likelihood parameters.
- $\mathbf{x}_n \in \mathbb{R}^{W_n}$: $n$-th document with length (i.e., number of words) $W_n$.
- $\boldsymbol{X} = \{\mathbf{x}_n\}_{n=1}^{N}$: set of all documents.
- $\boldsymbol{X}_{-n} = \{\mathbf{x}_i | i \neq n\}_{i=1}^{N}$: set of all documents except for $\mathbf{x}_n$.
- $z_n$:component assignment variable of document $\mathbf{x}_n$.
- $\boldsymbol{Z} = \{z_n\}_{n=1}^{N}$: set of all component assignment variables.
- $\boldsymbol{Z}_{-n} = \{\mathbf{z}_i | i \neq n\}_{i=1}^{N}$: set of all component assignment variables except for $z_n$.

**Summary of Generative Model:** We will work with the following Infinite Mixture Model

$$p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{\Theta}) = p(\boldsymbol{\pi}|\boldsymbol{\alpha}) p(\boldsymbol{\Theta}|\boldsymbol{\gamma}) \prod_{n=1}^{N} [p(z_n|\boldsymbol{\pi}) p(\mathbf{x}_n|z_n, \boldsymbol{\Theta})]$$

The conjugate prior for the categorical distribution is the Dirichlet distribution. Therefore, we define the prior distribution for $\boldsymbol{\theta}_k$ for all $k$ as Dirichlet distributions with parameters $\boldsymbol{\gamma}$. Notice the prior distributions for each $\boldsymbol{\theta}_k$ share the same set of parameters.

$$p(\boldsymbol{\Theta}|\boldsymbol{\gamma}) = \prod_{k=1}^{K^+} \text{Dir}(\boldsymbol{\theta}_k|\boldsymbol{\gamma}) \quad p(z_n|\boldsymbol{\pi}) = \text{Cat}(z_n|\boldsymbol{\pi}) \quad p(\mathbf{x}_n|z_n, \boldsymbol{\Theta}) = \prod_{j=1}^{W_n} \text{Cat}(x_{nj}|\boldsymbol{\theta}_{z_n})$$

For convenience, we define the prior distribution for $\pi$ as a symmetric Dirichlet (also known as multivariate beta) with concentration parameter $\alpha_k = \alpha/K$. Recall that $K \to \infty$. Since we can not sample from an infinity distribution we have to collapse $\pi$. This means we cannot obtain a standard Gibbs sampling algorithm.

$$p(\pi|\alpha) = \text{Dir}(\pi|\{\alpha/K\}_{k=1}^{\infty})$$

**Submission:** Copy the Jupyter notebook available in the Github repository https://github.com/APMLA/apmla_material/tree/master/L4 and complete the exercises proposed below. You will need to submit electronically the complete version of the Jupyter (together with the future exercises for Block I) by December 13th.

## Exercise 1: Derive the Gibbs sampling Algorithm for the iMM with categorical likelihood

Given the dataset and the probabilistic model described in the previous section, complete the following tasks in latex in the corresponding section of the Jupyter notebook:

1. **Algorithm 1:** Use the (collapsed) Gibbs sampling algorithm to approximate (using samples) the posterior distribution $p(\boldsymbol{Z}, \boldsymbol{\Theta}|\boldsymbol{X})$. Derive the posterior.

---

**Algorithm 1:** $\pi$ collapsed Gibbs sampling algorithm

---

Initialize $K^+ = 1$, $\{z_i = 1\}_{i=1}^N$ and $\boldsymbol{\theta}_1 \sim p(\boldsymbol{\theta}|\boldsymbol{\gamma})$;
**while** *not converged* **do**
  **for** $n = 1, \ldots, N$ **do**
    Sample $z_n \sim p(z_n|\boldsymbol{X}, \boldsymbol{Z}_{-n}, \boldsymbol{\Theta}) = p(z_n|\mathbf{x}_n, \boldsymbol{Z}_{-n}, \boldsymbol{\Theta})$;
    **if** $z_n = K_{new}$ **then**
      $K^+ += 1$;
      $\boldsymbol{\theta}_{K^+} \sim p(\boldsymbol{\theta}|\mathbf{x}_n)$
    **end**
    If necessary, remove empty clusters;
  **end**
  **for** $k = 1, \ldots, K^+$ **do**
    Sample $\boldsymbol{\theta}_k \sim p(\boldsymbol{\theta}_k|\boldsymbol{X}, \boldsymbol{Z})$ ;
  **end**
**end**

---

2. **Algorithm 2:** Use the (collapsed) Gibbs sampling algorithm to approximate (using samples) the posterior distribution $p(\boldsymbol{Z}|\boldsymbol{X})$. Derive the posterior.

---

**Algorithm 2:** $\pi, \boldsymbol{\Theta}$ collapsed Gibbs sampling algorithm

---

Initialize $K^+ = 1$ and $\{z_i = 1\}_{i=1}^N$;
**while** *not converged* **do**
  **for** $n = 1, \ldots, N$ **do**
    Sample $z_n \sim p(z_n|\boldsymbol{X}, \boldsymbol{Z}_{-n})$;
    **if** $z_n = K_{new}$ **then**
      $K^+ += 1$;
    **end**
    If necessary, remove empty clusters;
  **end**
**end**

---

## Exercise 2: Algorithms implementation in Python

1. Implement in Python the two Gibbs samplers derived in Exercise 1.
2. Let us consider the log-likelihood as the measure of convergence. Run each of the two Gibbs samples until convergence, i.e. until the log-likelihood does not improve in 10 iterations, with $\alpha \in \{0.1, 10\}$. Then:
   (a) Show the evolution of the log-likelihood per iteration.
   (b) Take a sample from the posterior after convergence and show the 10 most representative words for each topic using a cloud of words.
   (c) How does the number of "seen" clusters, i.e. $K^+$, vary with $\alpha$?

   Hint: $\log p(\boldsymbol{X}|\boldsymbol{\Theta}, \boldsymbol{\pi}, \boldsymbol{Z})$