Non-parametric Bayesian Methods in Machine Learning

Dr. Simon Rogers
School of Computing Science
University of Glasgow
simon.rogers@glasgow.ac.uk
@sdrogers

May 8, 2014

Outline

- **FIX ME AT THE END**
- (My) Bayesian philosophy
- Gaussian Processes for Regression and Classification
  - GP preliminaries
  - Classification (including semi-supervised)
  - Regression application 1: clinical (dis)-agreement
  - Regressopn application 2: typing on touch-screens
- Dirichlet Process flavoured Cluster Models
  - DP preliminaries
  - Idenfitying metabolites
  - (if time) Cluster models for multiple data views

About me

- I'm not a statistican by training (don't ask me to prove anything!).
- Education:
  - Undergraduate Degree: Electrical and Electronic Engineering (Bristol)
  - PhD: Machine Learning Techniques for Microarray Analysis (Bristol)
- Currently:
  - Lecturer: Computing Science
  - Research Interests: Machine Learning and Applied Statistics in Computational Biology and Human-Computer Interaction (HCI)

Lecture 6: Dirichlet Process Priors for Mixture Models

Dr. Simon Rogers
School of Computing Science
University of Glasgow
simon.rogers@glasgow.ac.uk
@sdrogers

May 8, 2014

Mixture Models

- A common strategy when faced with complex multi-modal data is to fit a *mixture model*.
- In general:

$$p(\mathbf{x}) = \sum_{k=1}^{K} P(k)p(\mathbf{x}|k)$$

- Where each component $p(\mathbf{x}|k)$ is some simple density, e.g. Gaussian
- Within this model, we must know K *a-priori*
- Can do inference with Expectation-Maximisation, Variational Bayes, Gibbs Sampling, etc
- Generative process for N data points:
  - For each datapoint, n:
    - Sample a component (k) according to $P(k)$.
    - Sample $\mathbf{x}_n \sim p(\mathbf{x}_n|k)$

Gibbs sampling for mixture models

- Assume that the kth mixture component has parameters θ_k.
- Define binary variables z_{nk} where $z_{nk} = 1$ if nth object is *in* kth component and zero otherwise.
- Define $\pi_k = P(k)$.
- Define prior density on θ_k: $p(\theta_k)$.
- For each iteration:
  - Sample each θ_k from $p(\theta_k | \ldots) \propto p(\theta_k) \prod_n p(\mathbf{x}_n|\theta_k)^{z_{nk}}$
  - For each object n:
    - Remove from its current component.
    - Sample a new component: $P(z_{nk} = 1| \ldots) \propto \pi_k p(\mathbf{x}_n|\theta_k)$

Being Bayesian

- We should treat $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_K]^T$ as a random variable.
- A suitable prior density is the Dirichlet:

$$p(\boldsymbol{\pi}_k) = \frac{\Gamma(\sum_k \beta)}{\prod_k \Gamma(\beta)} \prod_k \pi_k^{\beta_k - 1}$$

- (from now on, we'll assume $\beta_k = \alpha/K \quad \forall k$)

Being Bayesian

- We should treat $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_K]^T$ as a random variable.
- A suitable prior density is the Dirichlet:

$$p(\boldsymbol{\pi}_k) = \frac{\Gamma(\sum_k \beta)}{\prod_k \Gamma(\beta)} \prod_k \pi_k^{\beta_k - 1}$$

- (from now on, we'll assume $\beta_k = \alpha/K \quad \forall k$)
- We will also assume that (*a-priori*) the number of objects in each cluster ($c_k = \sum_n z_{nk}$) is multinomial with parameter $\boldsymbol{\pi}$:

$$p(\mathbf{c}|\boldsymbol{\pi}) \propto \prod_k \pi_k^{c_k}$$

Being Bayesian

- We can now compute the posterior density for $\boldsymbol{\pi}$. It's another Dirichlet:

$$p(\boldsymbol{\pi}|\mathbf{c}, \alpha) = \frac{\Gamma(\sum_k \alpha/K + c_k)}{\prod_k \Gamma(\alpha/K + c_k)} \prod_k \pi_k^{\alpha/K + c_k - 1}$$

Being Bayesian

- We can now compute the posterior density for $\boldsymbol{\pi}$. It's another Dirichlet:

$$p(\boldsymbol{\pi}|\mathbf{c}, \alpha) = \frac{\Gamma(\sum_k \alpha/K + c_k)}{\prod_k \Gamma(\alpha/K + c_k)} \prod_k \pi_k^{\alpha/K + c_k - 1}$$

- We can now also compute the probablity that some new observation would be placed in class j:

$$\begin{aligned} P(z_{*j} &= 1|\mathbf{c}, \alpha) = \int p(z_{*j} = 1|\boldsymbol{\pi}) p(\boldsymbol{\pi}|\mathbf{c}, \alpha)\ d\boldsymbol{\pi} \\ &= \int \pi_j p(\boldsymbol{\pi}|\mathbf{c}, \alpha) \\ &= \frac{c_j + \alpha/K}{\alpha + \sum_k c_k} \end{aligned}$$

- (Need to know that $\Gamma(z+1) = z\Gamma(z)$)

Gibbs sampling again

- Going back to our Gibbs sampling, we can replace π_k with this expression:

$$P(z_{nk} = 1 | \ldots) \propto \frac{c_k + \alpha/K}{\alpha + \sum_j c_j} p(\mathbf{x}_n | \theta_k)$$

- Where the point being sampled shouldn't appear in any c_j (i.e. $\sum_j c_j = N - 1$)

Sampling from the prior

- We can ignore the data $\mathbf{x}_n$ for a while and just sample partitions from this prior:
- Start with N objects, all in one cluster.
- For each iteration:
    - For each object n:
        - Remove from component it is in and re-assign with probability:

$$P(z_{nk} = 1 | \ldots) = \frac{c_k + \alpha / K}{\alpha + \sum_j c_k}$$

Sampling from the prior

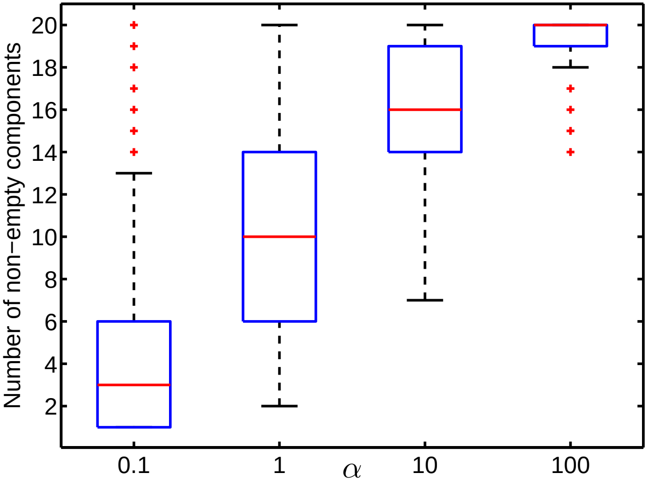


Figure 30 : Number of non-empty components as α is increased. $N = 100$ and $K = 20$. α controls how clustered the data are. Low α gives few populated clusters. Note: could have done this by sampling $\boldsymbol{\pi}$ and then sampling from $\boldsymbol{\pi}$)

$K \to \infty$

- What if we don't want to fix K?
- i.e. set $K = \infty$.

$$P(Z_{nk} = 1 | \ldots) = \frac{c_k + \alpha/K}{\alpha + \sum_j c_j} = \frac{c_k}{\alpha + \sum_j c_j}$$

- This is the probability for one of the *finite* number of clusters that are currently occupied.

$K \to \infty$

- What if we don't want to fix K?
- i.e. set $K = \infty$.

$$P(Z_{nk} = 1 | \ldots) = \frac{c_k + \alpha/K}{\alpha + \sum_j c_j} = \frac{c_k}{\alpha + \sum_j c_j}$$

- This is the probability for one of the *finite* number of clusters that are currently occupied.
- The probability of assigning to one of the *infinite* number of unoccupied clusters must be:

$$P(Z_{nk*} = 1 | \ldots) = 1 - \sum_k \frac{c_k}{\alpha + \sum_j c_j} = \frac{\alpha}{\alpha + \sum_j c_j}$$

$K \to \infty$

- What if we don't want to fix K?
- i.e. set $K = \infty$.

$$P(Z_{nk} = 1| \ldots) = \frac{c_k + \alpha/K}{\alpha + \sum_j c_j} = \frac{c_k}{\alpha + \sum_j c_j}$$

- This is the probability for one of the *finite* number of clusters that are currently occupied.
- The probability of assigning to one of the *infinite* number of unoccupied clusters must be:

$$P(Z_{nk*} = 1| \ldots) = 1 - \sum_k \frac{c_k}{\alpha + \sum_j c_j} = \frac{\alpha}{\alpha + \sum_j c_j}$$

- A nice paper describing this: Rasmussen 2000

Sampling from the $K = \infty$ prior

- Start with N objects all in one cluster
- For each iteration:
  - For each object n:
    - Remove from current component (if it leaves an empty component, delete it)
    - Re-assign according to:

$$P(z_{nk} = 1 | \ldots) = \begin{cases} \frac{c_k}{\alpha + \sum_j c_j} & k \text{ currently occupied} \\ \frac{\alpha}{\alpha + \sum_j c_j} & \text{new } k \end{cases}$$

    - if object in a new component, create one.

Sampling from the $K = \infty$ prior

- Start with N objects all in one cluster
- For each iteration:
  - For each object n:
    - Remove from current component (if it leaves an empty component, delete it)
    - Re-assign according to:

$$P(z_{nk} = 1 | \ldots) = \begin{cases} \frac{c_k}{\alpha + \sum_j c_j} & k \text{ currently occupied} \\ \frac{\alpha}{\alpha + \sum_j c_j} & \text{new } k \end{cases}$$

    - if object in a new component, create one.
- The prior we are sampling from is a *Dirichlet Process*

Sampling from the $K = \infty$ prior

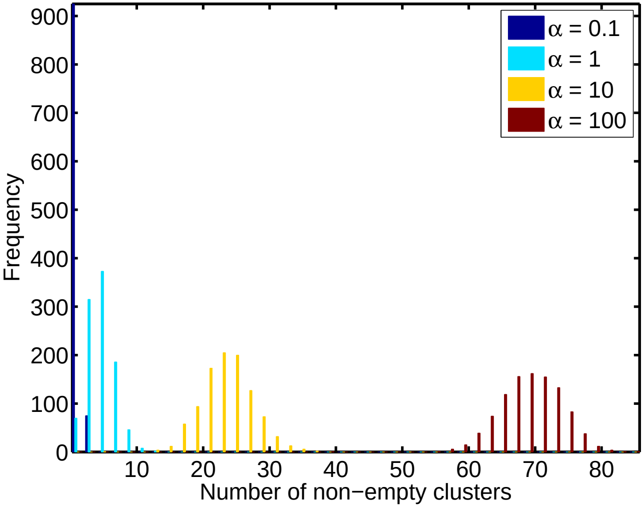


Figure 31 : Number of non empty clusters as α is varied. 1000 samples for $N = 100$ objects.

Including data

- In general, we want to fit the model to objects $\mathbf{x}_n$.
- Assume each component has parameters θ_k.
- Assume a likelihood $p(\mathbf{x}_n|\theta_k)$ and a prior $p(\theta_k)$

Including data

- In general, we want to fit the model to objects $\mathbf{x}_n$.
- Assume each component has parameters θ_k.
- Assume a likelihood $p(\mathbf{x}_n|\theta_k)$ and a prior $p(\theta_k)$
- Gibbs sampling:
  - Given a current clustering $\mathbf{Z}$ and parameters $\theta_1, \ldots, \theta_k$
  - For each object $\mathbf{x}_n$ in each iteration:
    - Remove $\mathbf{x}_n$ from the model (might require deleting a component).
    - Re-assign with probabilities:

$$P(z_{nk} = 1|\mathbf{x}_n, \ldots) \propto \begin{cases} c_k p(\mathbf{x}_n|\theta_k) & k \text{ currently occupied} \\ \alpha \int p(\mathbf{x}_n|\theta)p(\theta)\, d\theta & \text{new } k \end{cases}$$

    - Sample θ_k for each component:

$$p(\theta_k|\ldots) \propto p(\theta_k) \prod_n p(\mathbf{x}_n|\theta_k)^{z_{nk}}$$

Including data

- In general, we want to fit the model to objects $\mathbf{x}_n$.
- Assume each component has parameters θ_k.
- Assume a likelihood $p(\mathbf{x}_n|\theta_k)$ and a prior $p(\theta_k)$
- Gibbs sampling:
  - Given a current clustering $\mathbf{Z}$ and parameters $\theta_1, \ldots, \theta_k$
  - For each object $\mathbf{x}_n$ in each iteration:
    - Remove $\mathbf{x}_n$ from the model (might require deleting a component).
    - Re-assign with probabilities:

$$P(z_{nk} = 1|\mathbf{x}_n, \ldots) \propto \begin{cases} c_k p(\mathbf{x}_n|\theta_k) & k \text{ currently occupied} \\ \alpha \int p(\mathbf{x}_n|\theta)p(\theta)\, d\theta & \text{new } k \end{cases}$$

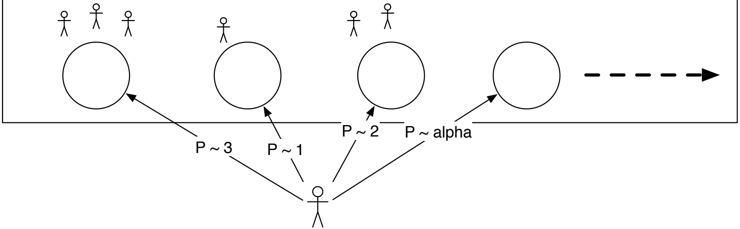
    - Sample θ_k for each component:

$$p(\theta_k|\ldots) \propto p(\theta_k) \prod_n p(\mathbf{x}_n|\theta_k)^{z_{nk}}$$

- If everything conjugate, can integrate θ_k out completely (better convergence).

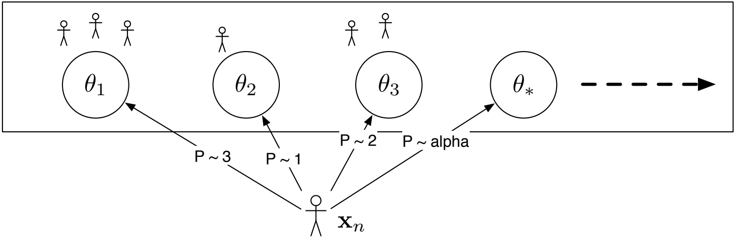
Chinese Restaurant Process (CRP)

- A popular way of thinking about DPs is via the 'Chinese Restaurant Process'
- Prior:
  - N people enter a Chinese Restaurant with infinite tables.
  - The first person sits at the first table.
  - The nth person sits at an occupied table with probability proportional to the number of people already at the table, or a new table with probability proportional to α.



CRP

- With data:
  - Each table has one dish. Table choice also depends on dish preference.
  - Table dishes updated according to preference of people at table (here the analogy starts(!) to get a bit tenuous)



Example

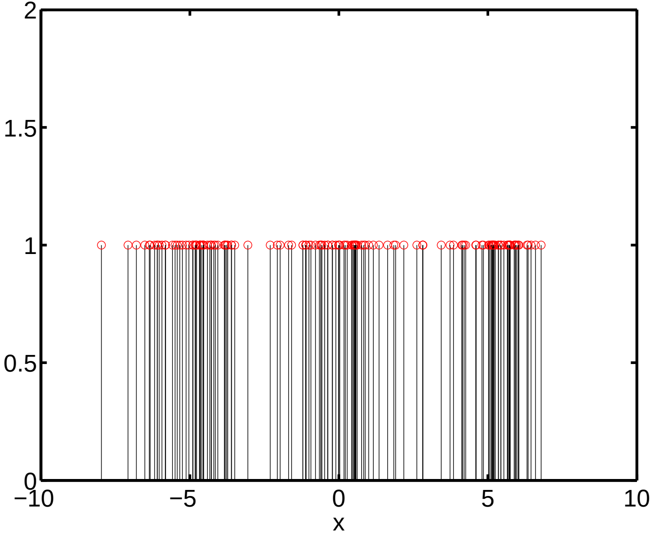


Figure 32 : Example output from CRP with Gibbs sampling. The data has three clusters. Visualising the θ_k is impossible, but we can look at

Example

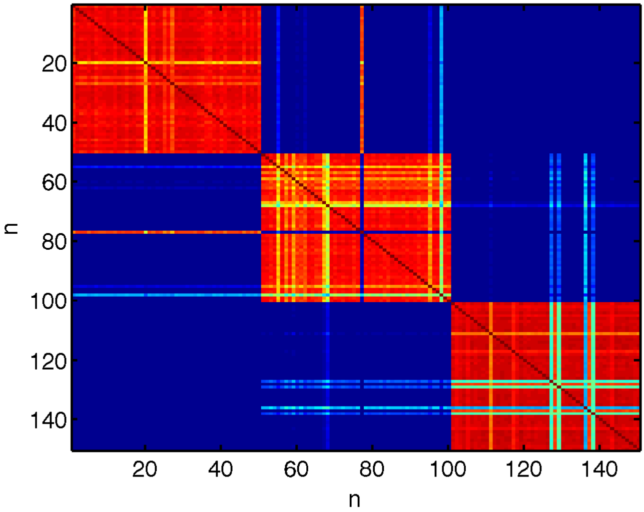


Figure 32 : Example output from CRP with Gibbs sampling. The data has three clusters. Visualising the θ_k is impossible, but we can look at

Example

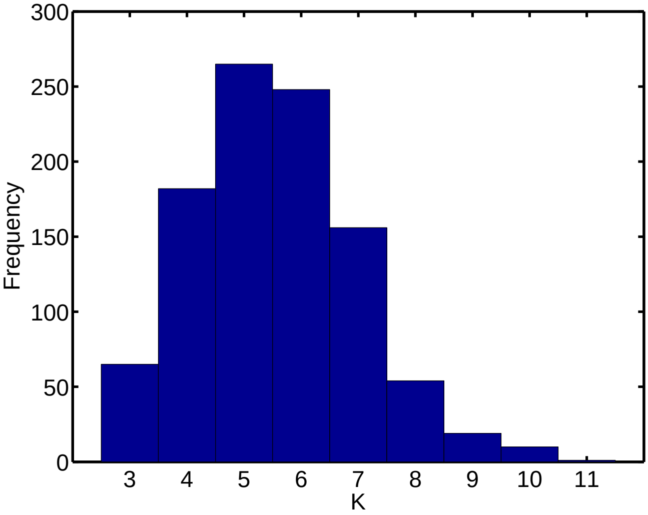


Figure 32 : Example output from CRP with Gibbs sampling. The data has three clusters. Visualising the θ_k is impossible, but we can look at the probablility that two objects are in the same cluster, and the number