Thesis Proposal

Bayesian Nonparametric Model in Active Learning

Shaobo Dang (3384106)
sdang@cse.unsw.edu.au

Supervisor:
Dr. Xiongcai Cai

August 10, 2015

THE UNIVERSITY OF
NEW SOUTH WALES

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052, Australia

*Contents

1 Introduction

Data Mining has been an heating area in Machine Learning. It is especially important in the information age since the seemingly meaningless data often containing potential opportunity, as can be seen from large Internet companies such as Google, Facebook and Amazon [9]on their investment on large data processing project.

The primary task in data mining is to find the underlying distribution that generates the observed data to guide the computer to process the similar data automatically, saving the cost of both time and expense. As the data often consists of multi-modal distribution, modeling the distribution as a mixture of simpler distribution is found to be a well approximation. In parametric settings, the number of sub-distribution is fixed value which may cause the model fail to funtion as the complexity is not synchronized that of the data. Instead, Bayesian Nonparametric model, especially the Dirichlet Process employs a infinite countably number of components to handle the analysis by placing a prior distribution for mixing distribution. The nature of this set of problem bypasses the need to "determine" the number of comopnents in a finite mixture model.

Though a very promising and relatively simpler way to model, it is often difficult to process the following inference which is the main part of analysis. Except for placing conjugate prior, the desirable posterior distribution of most non-conjugate prior model is hard to obtain.

Use of Dirichlet process mixture models has been computationally feasible with the development of Markov Chian Monte Carlo methods for sampling from the posterior distribution of the parameters of the component distritution given observed data. But there need more research work on addressing the slow convergency problem. Though conjugate prior is preferable for it allows to introduce Gibbs sampling methods, it narrows the generalization of Dirichlet Process to address more complicated problem. In most cases, the non-conjugate prior is placed, thus it should be researched on how to use MCMC under such circumstances. Also research on applications based on DPMM that requires fast convergency will be conducted.

2 Aim Of Research

The primary objective of this research is to introduce the Bayesian Nonparametric(BNP) theory to the general framework of Active Leaning. We aim to design a new learning framework enabling the growing of model complexity without a great compromising in accuracy and a growth in cost. The initial motivation is take advantages of the merits , as for active learning the efficiency and for BNP the flexibility, to make up for their disadvantage, as for active learning low sampling procedure and and for BNP low convergency rate[4, 2, **?**]. In this proposal, we will exploit the general problems in this two distinct areas and the potential performance when combined together. The main aims of this research consist of the following parts(will be in addressed in detail in later part)

1. **Develop a new sampling methods in active learning**
   Sampling strategy lies in the core of BNP inference scheme. Most approaches select either informative or representative unlabeled instances. But it is usually challenging to find the querying instances that are both informative and representative. Thus in this research, a new approach is to be proposed to provide a systematic way to select samples having both features, and the effectiveness will be validated.

2. **Application of Baysian Nonparametric model in offline condition assessment and failure prediction**

3. **Application of Baysian Nonparametric model in online prediction problem**

3 Literature Review

This section overview of the basic concept of Dirichlet Process, including the representation, the main application-Dirichlet Process Mixture Model and the inference methods.

3.1 Background

Large Data sets are often heterogeneous from underlying sub-populations. When doing analysis on such data sets, it often involves grouping the data to make the original data more heterogeneous. Traditionally, parametric models using a fixed and finite number of parameters could be used, but they suffer from over- or under-fitting of data when there is a misfit between the complexity of the model and the amount of data available. Thus, model complexity selection is often an important issue in parametric modeling. But whether we use cross validation or marginal probabilities as the basis for selection, model selection is an operation that is fraught with difficulties, especially when the data set grows larger and larger.

The Bayesian Nonparametric approach is an alternative to parametric modeling and selection. Under this scheme, the model comes with an unbounded complexity, and under-fitting and over-fitting are mitigated. Typically, we assume that the observed data set $x_1, x_2, \cdots, x_i$ are i.i.d(Independent Identical Distributed) sampled from some underlying unknown distribution $\mathbb{F}$. In Bayesian approach, a prior is placed over $\mathbb{F}$ then he posterior over $\mathbb{F}$ given data is computed. This prior over distributions is given by a parametric family. But constraining distributions to lie within parametric families limits the scope and type of inferences that can be made. Instead, the nonparametric approach used a prior over distributions with wide support, typically the support being the space of all distributions. The Dirichlet Process is currently one of the most popular Bayesian Nonparametric Models. It was first formalized in [1] for general Bayesian statistical modeling.,as a prior over distributions with wide support yet tractable posteriors. But the Dirichlet Process is limited by the fact that draws from it are often discrete distributions, and tractable posterior is hard to obtian when it generalized to more general non-conjugate priors until MCMC techniques became available in the area.

3.2 Dirichlet Process

3.2.1 Dirichlet Distribution

A Dirichlet distribution is defined on the $(k-1)$-dimensional probability simplex, whic is a surface in $\mathbb{R}^k$ denoted by Δ_k and defined to be the set of vectors

whose k components are non-negative and sum to 1, that is

$$\Delta_k = \{q \in \mathbb{R}^k | \sum_{i=1}^{k} q_i = 1, q_i \geq 0 \ for \ i = 1, 2, \cdots, k\} \tag{3.1}$$

While the set Δ_k lies in a k-dimensional space, Δ_k is itself a $(k-1)$-dimensional object. Each point q in the simplex can be thought of as a probability mass function(pmf) in its own right. The Dirichlet distribution can be thought of as a probability distribution over the $k-1$-dimensional probability simplex Δ_k; that is, as a distribution over pmfs of length k.

Dirichlet Distribution: Let $Q = [Q_1, Q_2, \cdots, Q_k]$ be a random pmf, i.e. $Q_i \geq 0$ for $i = 1, 2, \cdots, k$ and $\sum_{i=1} kQ_i = 1$. In addition, suppose that $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_k]$ with $\alpha_i > 0$ for each i, and let $\alpha_0 = \sum_{i=1^k \alpha_i}$. Then Q is said to have a Dirichlet distribution with parameter α, which we denote by $Q \sim Dir(\alpha)$, if it has $f(a; \alpha) = 0$ then q is not a pmf, else if q is a pmf then

$$f(q; \alpha) = \frac{\Gamma(\alpha_0)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} q_i^{\alpha_i - 1} \tag{3.2}$$

where $\Gamma(s)$ denotes the gamma function, a generalization of the factorial function, for $s > 0, \Gamma(s+1) = s\Gamma(s)$, and for positive integers n, $\Gamma(n) = (n-1)!, \Gamma(1) = 1$. Fig. 3.1 shows the plots of the density fo the Dirichlet distri-

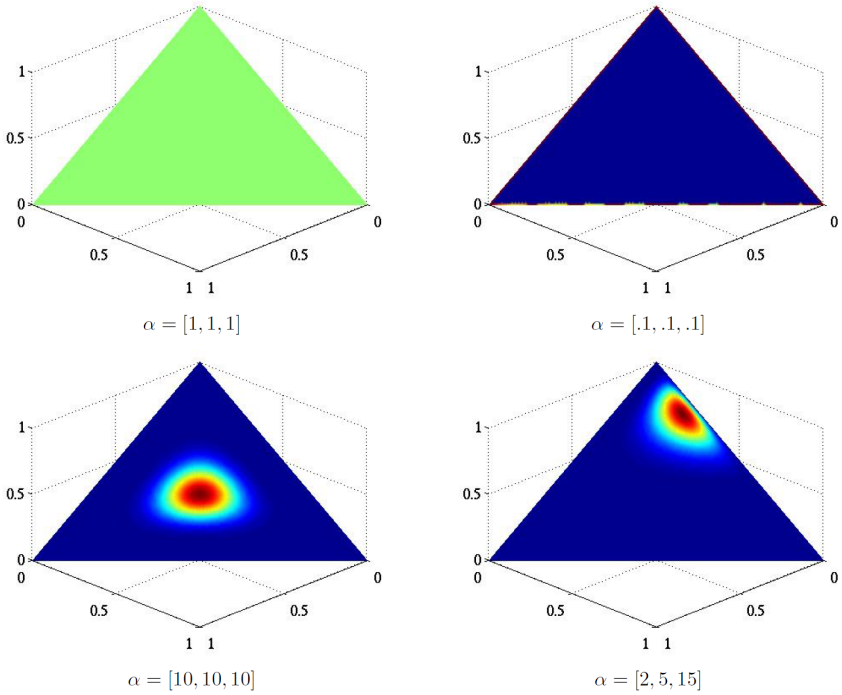


Figure 3.1: Dirichlet distribution density plots(blue=low, red=high) over the probability simplex in $\mathbb{R}^3$ for various values of the parameter α

bution over the two-dimensional probability simplex for $k = 3$ events lying in three-dimensional Euclidean space over a variety of parameter vector α. When $\alpha = [1, 1, 1]$, the Dirichlet distribution reduces to uniform distribution over the simplex. When all the components of α satisfy $\alpha_i > 1$, the density is monomodal with its mode somewhere in the interior of the simplex. And when $\alpha_i < 1$, the plots has sharp peaks at the vertices of the simplex. Another important feature should be mentioned, as can be seen from the definition of Dirichlet distribution, is that the support is open thus does not include the vertices or edge of the simplex, which in reality means that no component of a pmf drawn from a Dirichlet will ever be zero. Fig. 3.2 shows plots of samples drawn i.i.d from different Dirichlet distributions.

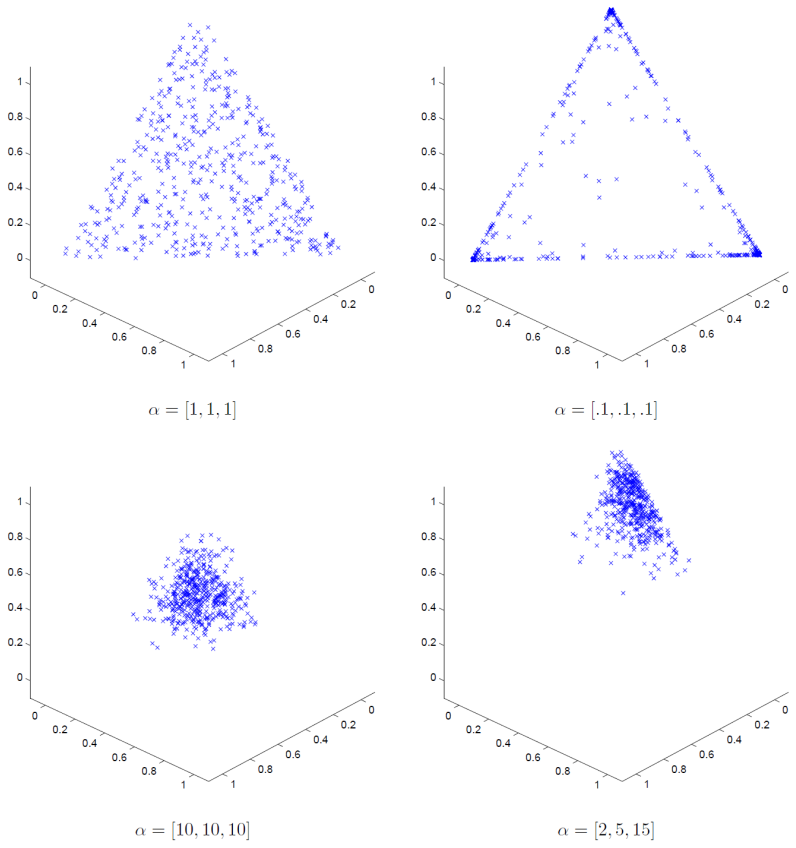


Figure 3.2: Plots of sample pmfs drawn from Dirichlet distributions over the probability simplex in $\mathbb{R}^3$ for various values of parameter α

3.2.2 Dirichlet Process

A Dirichlet Process is a distribution over probability distributions. Suppose that G is a probability distribution over a measurable space Θ, then G is a probability distribution over Θ and a DP is a distribution over all such distributions. A

DP is parameterized b a concentration parameter α and a base measure(base distribution) H. So, the formal definition of a DP would be

$$G \sim DP(\alpha, H) \tag{3.3}$$

it means for a finite set of measurable paritions $A_1 \cup \cdots \cup A_k = \Theta$,

$$(G(A_1) \cdots, G(A_k)) \sim Dir(\alpha H(A_1), \cdots, \alpha H(A_k)) \tag{3.4}$$

which means the probabilities that G assigns to any finite partition of Θ follow a Dirichlet distribution with parameters $(\alpha H(A_1), \cdots, \alpha H(A_k))$.

Since G is a random distribution we can inturn draw samples from G itself. Let $\theta_1, \theta_2, \cdots, \theta_n$ be a sequence of independent draws from G. What is the posterior distribution of G given observed values of $\theta_1, \theta_2, \cdots, \theta_n$. In other words, what is the posterior predictive distribution for a new item $p(\theta_{N+1}|\theta_1, \theta_2, \cdots, \theta_n) = \int p(\theta_{N+1}|G)p(G|\theta_1, \theta_2, \cdots, \theta_n)dG$? It is shown in ... that

$$(G(A_1), \cdots, G(A_k))|(\theta_1, \theta_2, \cdots, \theta_n) \sim Dir((\alpha H(A_1) + n_1, \cdots, \alpha H(A_k) + n_k) \tag{3.5}$$

where $n_k = \#\{i : \theta_i \in A_k\}$ is the number of observed values in A_k. Rewriting the posterior DP, we have:

$$G|(\theta_1, \theta_2, \cdots, \theta_n) \sim DP(\alpha + n, \frac{\alpha}{\alpha + n} H + \frac{n}{\alpha + n} \frac{\sum_{i=1}^{n} \delta(\theta_i)}{n}) \tag{3.6}$$

and for the posterior distribution,

$$\begin{aligned} p(\theta_{N+1}|\theta_1, \theta_2, \cdots, \theta_n) &= E[G(A)|\theta_1, \theta_2, \cdots, \theta_n] \\ &= \frac{1}{\alpha + n} \left(\alpha H + \sum_{i=1}^{n} \delta_{\theta_i} \right) \end{aligned} \tag{3.7}$$

Therefore the posterior base distribution given $\theta_1, \theta_2, \cdots, \theta_n$ is also the predictive distribution of θ_{n+1}. The sequence of predictive distribution 3.7 for $\theta_1, \theta_2, \cdots, \theta_n$ is very important and different interpretations on it indicate different properties of the result. To better understand it, we will introduce the most famous three representations of 3.7: **Pólya urn, Chinese Restaurant Process, Stick-breaking Process**.

- **Pólya urn**
  In this analogy, suppose we are drawing colored balls from an urn, θ_i represents the color of the i-th ball drawn . For each ball drawn, we place it back and add another one in the same color into the urn. In the beginning, we pick a color drawn from H, paint a ball with that color and drop it into the urn. In the following nth step, we will either, pick a new color with

probability $\frac{\alpha}{\alpha+n}$, or with probability $\frac{n}{\alpha+n}$ pick a random ball out of the urn. This process induces a "rich get richer" property on the frequencies of colors inside the urn. Also, it should be noticed that the predictive distribution has point masses located at the previous draws $\theta_1, \theta_2, \cdots, \theta_n$. Thus the distribution G itself has point masses. When sample size grows larger, the value of any draw will be repeated by another draw, implying that G is composed only of a weighted sum of point masses and thus it is a discrete distribution.

- **Chinese Restaurant Process**
  Another representation of Dirichlet Process-Chinese Restaurant Process(CRP) implies a clustering property. Equation 3.7 could be rewrite as :

$$p(\theta_{N+1}|\theta_1, \theta_2, \cdots, \theta_n) = \frac{1}{\alpha + n}\left(\alpha H + \sum_{i=1}^{n} n_k \delta_{\theta_i^\star}\right) \tag{3.8}$$

  where $\theta_i^\star$ is the unique values among $\theta_1, \theta_2, \cdots, \theta_n$, and n_k is the number of repeats of $\theta_i^\star$. $\theta_i^\star$ will be repeated by θ_{n+1} with probability proportional to n_k. The larger n_k is, the hight the probability that it will grow. This is similar to the "rich get richer" scheme in **Pólya urn**, where large clusters grow larger. We can see that the unique values of $\theta_1, \theta_2, \cdots, \theta_n$ induce a partitioning of the set $[n] = \{1, 2, \cdots, n\}$ into clusters such that within some cluster k, the θ_i's take on the same value $\theta_k^\star$. The random partion encapsulates all the properties of the DP. If we invert the generative process, we can reconstruct the joint distribution over $\theta_1, \theta_2, \cdots, \theta_n$ by first drawin ga random partion on $[n]$, then each cluster $k$in the partition draw a $\theta_k^\star \sim H$, and finally assign $\theta_i = \theta_k^\star$ for each i in cluster k.

  The distribution over partitions is called the Chinese Restaurant Process(CRP) in which we have a Chinese restaurant with infinite tables, each of which can seat an infinite number of customers. The first customer enters the restaurant and sits at the first table. And in the following, when the $n+1$th customer comes, he will either joins a tale k with probability propotional to the number n_k of people already sitting there or sts at a new table with probability propotional to α. The CRP define a distribution over partitions of $[n]$ and a distribution over permutations of $[n]$.

- **Stick-breaking Construction** The third representation of DP is very intuitive and the most widely used one to generate a sample from it. By knowing the fact that draws from a DP consists actually of a weighted sum of point masses, ...provides a constructive and straightforward definition

of the DP. It simply following the flowwin steps.

$$\begin{aligned} \beta_k &\sim Beta(1,\alpha) & \theta_k^\star &\sim H \\ \pi_k &= \beta_k \prod_{l=1}^{k-1}(1-\beta_k) & G &= \sum_{k=1}^{\infty} \pi_k \delta_k^\star \end{aligned} \tag{3.9}$$

Then we have $G \sim DP(\alpha, H)$. The construction of π could be interpreted as breaking a stick of length 1 step by step. First break it at β_1, assigning π_1 to be the length of segment we just got. Then recursively breaking the remaining portion to obtain $\pi_2, \pi_3, \cdots, \pi_{n-1}$. THis is a very straightforward and simple procedure.

3.2.3 Dirichlet Process Mixture Models

In this section, we will briefly describe the Dirichlet Process Mixture Model Most machine learning problem is targeted to learn a set of parameters describing the model from training data set. This sort of learning is often evaluated in two terms, with the first one being how well the model fits the data, expressed as accuracy or squared error, and the second one being a complexity penalty(favoring simpler models)[**?**, 2], also referred to as Ocam's Razor. In practical problem, the model complexity is hard to evaluate. Improper model complexity will lead to over-fitting or under-fi tting[4, 10], which will then affect the model generalization, i.e to be applied to practical use. Although through rigid training, desirable models could be trained but this is basically a trial and error process[**?**]. That is the movivation of discovering adaptive model complexity selection methods, among which Bayesian Nonparametric Method is the most widely used.

DP, as the most widely used methods among Bayesian Nonparametric Models, has found applications in both statistics and machine learning, including Bayesian model validation, density estimation and clustering via mixture models, among which the last one is the most salient when talking about DP.

Model validation is to evaluate whether a model gives a good fit to observed data. Under Bayesian approach, we would usually compute the marginal probability of the data under the model and compare the marginal probability to that of other candidate model. The one with the highest probability will be chosen as the best fitting to the observed data. Here arises an issue that how to choose the models to be compared. Usually, a set of candidate models as large as possible would be desirable. But it would be easier if we re-think this in a Bayesian Nonparametric way, i.e. to use the space of all possible distribution as our comparison class, with a prior over distributions. The DP is always the first priority for its similar nature to this problem. The approach is to use the given parametric model as the base distribution of the DP, with DP serving as

a nonparametric relaxation around this parametric model. If the parametric model performs as well or better than the DP relaxed model, we are convinced that the model is valid.

For density estimation, the aim is to modeling the latent density from which the observed data is drawn. To avoid the poor performance caused by the limitation in parametric model, we again employ a Nonparametric prior over all densities. If we drawn samples from a DP, which is distribution over distribution, we will obtain a random distribution which is discrete,thus has no densities. The solution is to smooth out draws form the DP with a kernel. Let $G \sim DP(\alpha, H)$ and $f(x|\theta)$ be a family of densities indexed by θ. Then

$$p(x) = \int f(x|\theta)G(\theta)d\theta \tag{3.10}$$

The most common application of the Dirichlet process is to cluster data using mixture models. The traditional finite mixture model assumes that there are K clusters, each associated with a parameter θ_k. Each observation y_n is assumed to be generated by first choosing a sluster c_n according to $P(c_n)$ and then generating the observation from its corresponding observation destribution parameterized by θ_{c_n}. Finite model can accomodate many kinds of data by changing the data generating distribution.

Bayesian mixture models further contain a prior over the mixing distributions. The nature of Dirichlet process will translate the mixing model to a countably infinite number of components. we model a set of observations $\{x_1, \cdots, x_n\}$ using a set of latent parameters $\{\theta_1, \cdots, \theta_n\}$, each θ_i is drawn I.I.d from G, while each x_i has distribution $F(\theta_i)$ parameterized by θ_i:

$$\begin{aligned} x_i|\theta_i &\sim F(\theta_i) \\ \theta_i|G &\sim G \\ G|\alpha, H &\sim DP(\alpha, H) \end{aligned} \tag{3.11}$$

Because G is discrete, multiple θ_i's can take on the same value simultaneously, and the model above can be seen as a mixture model, where x_i's with the same value of θ_i belong to the same cluster. The mixture perspective can be made more in agreement with the usual representation of mixture models using the stick-breaking construction. Let z_i be a cluster assignment variable, which takes on value k with probability π_k. Then equation 3.12 could be expressed as:

$$\begin{aligned} \pi|\alpha \sim GEM(\alpha) \qquad & \theta_k^\star|H \sim H \\ z_i|\pi \sim Mult(\pi) \qquad & x_i|z_i, \{\theta_k^\star\} \sim F(\theta_{z_i}^\star) \end{aligned} \tag{3.12}$$

with $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^\star}$, $\theta_i = \theta_{z_i}^\star$, π being the mixing proportion, $\theta_k^\star$ being the

cluster parameters, $F(\theta_k^\star)$ being the distribution over data in cluster k and H the prior over cluster parameters.

From above expression, it is seen that DP mixture model is an infinite mixture model-a mixture model with a countably infinite number of clusters. Different from finite mixture model using a fixed number of clusters, π_k's decrease exponentially quickly, and only a small number of clusters will be used to model the data a priori. In the DP mixture model, the actual number of clusters used to model data is not fixed, and can be automatically inferred from data using the usual Bayesian posterior inference framework. The equivalent operation for finite mixture models would be model averaging or model selection for the appropriate number of components.

3.3 Inference

We have described three different representations of Dirichlet Process, and all of them posit a generative probabilistic process of a collection of observed (and future) data that includes hidden structure. And the observed data is analyzed by examining the posterior distribution of the hidden structure given the observations; this gives us a distribution over which latent structure likely generated our data.

Thus, the basic computational problem in DP modeling (as in most of Bayesian statistics) is computing the posterior. Unfortunately, for many models posterior is not available in closed form. But there are several ways to approximate it. The most widely used posterior inference methods in Bayesian nonparametric models are Markov Chain Monte Carlo (MCMC) methods. The idea MCMC methods is to define a Markov chain on the hidden variables that has the posterior as its equilibrium distribution [**?**]. By drawing samples from this Markov chain, one eventually obtains samples from the posterior. A simple form of MCMC sampling is Gibbs sampling, where the Markov chain is constructed by considering the conditional distribution of each hidden variable given the others and the observations. CRP mixtures are particularly amenable to Gibbs sampling due to the exchangability property, and each observation can be considered to be the "last" one and the distribution of Equation 3.2 can be used as one term of the conditional distribution. [**?**] provides an excellent survey of Gibbs sampling and other MCMC algorithms for inference in CRP mixture models.

An alternative approach to approximating the posterior is variational inference [**?**]. This approach is based on the idea of approximating the posterior with a simpler family of distributions and searching for the member of that family that is closest to it. Although variational methods are not guaranteed to recover the true posterior (unless it belongs to the simple family of distribu-

tions), they are typically faster than MCMC [**?**]and convergence assessment is straightforward. These methods have been applied to CRP mixture models.

Both MCMC and variational strategies for posterior inference provide a data-directed mechanism for simultaneously searching the space of models and finding optimal parameters. This is convenient mixture modeling because we avoid needing updating on models for each candidate number of components. It is essential in more complex settings where the algorithm searches over a space that is diffcult to effciently enumerate and explore. In this research, we will focus on the MCMC algorithm on Dirichlet Process.

3.3.1 Variational Inference

Variational Inference Methods have been introduced for the inference of DP in [**?**]. Under this scheme, variational methods approximate a posterior distribution $p(\theta|X)$ with a distribution $q(\theta)$ belonging to a more manageable family of distributions and try to find the best approximation, usually by minimizing the *Kullback-Leibler* divergence between $p(\theta|X)$ and $q(\theta)$.

$$KL(q||p) = E_q \frac{q(\theta)}{p(\theta)} \tag{3.13}$$

Then the desired q could be formulated as an optimazation problem:

$$q^\star = \arg\min_{q \in Q} KL(q(\theta)||p(\theta|X)) \tag{3.14}$$

The variational inference framework gives a principled way of finding an approximate distribution which is as close (as measured by KL) to the posterior. This will allow us to tackle posterior computations for models such as mixtures.

A typical approach to selecting the family fo approximation distributions is to assume independencies that may not be present in the true posterior. The quality of the approximation depends on Q: the bigger the Q, the better. Two of the familiar classical algorithms (hard EM and EM) are based on a particular kind of Q. The assumption will allow the possibility of parallelization. However, the assumptions constrains the posterior distribution in a restricted class of models, thus the expressiveness of true model might be lost leading to a less accurate results.

3.3.2 Markov Chain Monte Carlo Sampling

One method for sampling from an arbitrary target distribution is to use a Markov chain Monte Carlo (MCMC) algorithm. MCMC methods simulate a Markov chain with a particular transition distribution such that the stationary distribution of the chain is exactly the target distribution of interest. Certain conditions must be specified to ensure that this condition holds. A sample from

the target distribution can then be generated by simulating a Markov chain until it converges to its stationary distribution, followed by taking the value of the chain when it is terminated.

The state of a Markov chain at iteration t is denoted as z^t. Suppose that the Markov chain evolves according to some transition distribution,$q^\star(z^{(t+1)}|z^{(t)})$. A stationary distribution of a Markov chain is a distribution over states that is invariant under the transition distribution $q^\star$. A distribution is the stationary distribution of a Markov Chain if it satisfies

$$f_z(z^{(t+1)}) = \int f_z(z^{(t)})q^\star(z^{(t+1)}|z^{(t)})dz^{(t)} \tag{3.15}$$

Stationarity of a Markov chain with respect to a transition distribution essentially means that if the chain is currently in the stationary distribution, simulating a transition from $q^\star$ will not alter the distribution.

Multiple such distribution can exist for a Markov chain. MCMC sampling algorithms must consequently ensure uniqueness by enforcing *ergodicity* of the Markov chain, i.e. the Markov chain must satisfy the condition of being *irreducible and aperiodic*. An ergodic Markov chains will converge to a unique stationary distribution regardless of the initial state.

- **Metropolis-Hasting Sampling**
  The idea of MCMC sampling is to simulate a Markov chain that has the target distribution as a stationary distribution. Different from the Variational Inference methods introduced in3.3.1, MCMC methods sampls from the true posterior distribution asymptotically. Ergodicity ensures convergence to the chain, but one has to use additional methodologies to ensure the correct stationary distribution. The Metropolis-Hastings algorithm is one such method. It is supposed that there is a proposal distribution $p(z^{(t+1)}|z^{(t)})$ from which we can sampling candidate state. Metropolis et al.[**?**] developed an algorithm that constructs a transition distribution, $q\star$, from a symmetric proposal distribution, p, such that the stationary distribution is exactly the target distribution. Hastings[**?**] later generalized this algorithm to allow for non-symmetric proposal distributions. The latter algorithm is commonly referred to as the Metropolis-Hastings (MH) algorithm. The concept underlying the MH algorithm the the notion of *detailed balance* which shows that if a Markov chain is constructed with a transition distribution, q, that satisfies

  $$f_z(z_1)q(z_2|z_1) = f_z(z_2)q(z_1|z_2) \tag{3.16}$$

  then the chain is said to satisfy the detailed balance condition. Furthermore, $f_z(z)$ is guaranteed to be a stationary distribution of the chain. Detailed balance is a sufficient condition to ensure that $f_z(z)$ is a station-

ary distribution, but it is not necessary. Once we get the candidate new state from the proposal distribution p, MH algorithm shows that if the transition distribution $q^\star$ is constructed according to

$$q^\star(z^{(t+1)}|z^{(t)}) = min[1, \frac{f(z^{(t+1)})}{f(z(t))} \frac{q(z(t)|z^{(t+1)})}{q(z^{(t+1)}|z^{(t)})}] \tag{3.17}$$

then the resulting Markov chain satisfies the detailed balance condition. The constructed transition distribution subjects the newly proposed sample to an accept or reject step. The Metropolis-Hastings algorithm therefore guarantees that the target distribution is a stationary distribution of the chain. Furthermore, Markov chain theory states that the resulting Markov chain is guaranteed to converge uniquely to the stationary distribution if it is ergodic.

- **Gibbs Sampling** Gibbs sampling [40] is a special case of Metropolis-Hastings where the transition distribution only acts on a subset of the variables (or, a subset of dimensions of a multidimensional variable). In particular, the Gibbs sampler chooses the proposal distribution to be the true posterior distribution of the subset of variables conditioned on all variables. Gibbs sampling precludes the need of an accept/reject step because it results in a Hastings ratio that evaluates to 1, i.e. the Gibbs sampling algorithm can accept all proposed samples while still satisfying detailed balance.

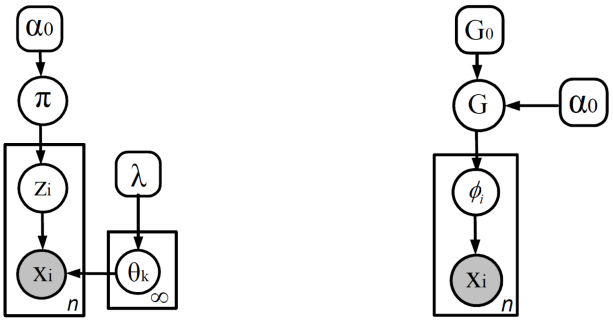
  Gibbs sampling is often preferred over Metropolis-Hastings because it does not re- quire one to specify a proposal distribution. However, it can only be used when the conditional posterior distributions are known. Furthermore, Gibbs sampling may result in slow convergence because only a single random dimension of the latent variable is sampled at a time. This procedure can explore the space very slowly due to the local changes proposed by the sampling algorithm.

3.4 Markov Chain Monte Carlo for DPMM

DPMM as widely used in the the industrial and reserch area of machine learning as the model allow for an automatic model selection in clustering problem, one of the basic yet important procudure of data mining. As can be seen from previous chapter, despite the powerful representation of the model, the infinite and unlimited component number make it hard to represent them explicitly, leading to work on developing methods of posterior inference. .

The most widely used methods in posterior inference in DPMM is to draw samples of those latent variables using a Markov Chain Monte Carlo scheme. But posterior sampling in such complex models using MCMC is often difficult

because samplers proposing local changes exhibit poor convergence. To address the convergence problem, many algorithms are poposed. In this section, a brief introduction to these methods will be given.



(a) Graphic model for Póly Urn Presentation

(b) Graphic model for Póly Urn Presentation

Figure 3.3: Graphic Model of two DPMM representations

- **Gibbs Sampling based on Pólya urn**
  In the Pólya urn representation of DPMM (3.12) as shown in figure the only unknown variables are $\{\phi_i\}_{i=1}^n$. This leads to a simple Gibbs sampling methods: alternatively draw ϕ_i from its posterior distribution conditioned on the other variables ϕ_{-i} and all the observations. Thus we have:

$$\phi_i = \phi|\phi_{-i} \sim \frac{\alpha_0 G_0(\phi)}{\alpha_0 + n - 1} + \frac{\sum_{j\neq i}\delta(\phi - \phi_j)}{\alpha_0 + n - 1} \tag{3.18}$$

  combined with the likelihood, we get the posterior of ϕ_i conditioned on ϕ_{-i}:

$$\begin{aligned} p(\phi|\phi_{-i}, x_i) &= b\alpha_0 q_0 H(\phi_i|x_i) + b\sum_{j\neq i} F(x_i|\phi_j)\delta(\phi_i - \phi_j) \\ H(\phi_i|x_i) &= \frac{G_0(\phi_i)F(x_i|\phi_i)}{\int_\phi G_0(\phi)F(x_i|\phi)} \\ q_0 &= \int_\phi G_0(\phi)F(x_i|\phi) \\ b &= \left(\alpha_0 q_0 + \sum_{j\neq i} F(x_i|\phi_j)\right)^{-1} \end{aligned} \tag{3.19}$$

  When G_0 is a conjugate prior for $F(x_i|\phi_i)$, the posterior distribution of $\phi_i, H(\phi_i|x_i)$ and the marginal distribution of x_i, q_0, have analytical forms and the Gibbs sampling can be easily performed.

- **Gibbs sampling using latent indicator variables**

Though the Gibbs sampling based on Pólya urn representation is very simple to implement, it is very inefficient. In each iteration, we need to sample the cluster parameter for n times and each time we only change the parameter for a single data point. As we know, there are usually lots of data points share the same cluster parameter. A more efficient way is obvious to operate the data points belonging the same cluster simultaneously. To do this, we need to employ the DPMM in stick-breaking representation, where cluster parameters are moved outside of the plate of x_i and the indicator variables are used to identify the cluster x_i associated to.

For each indicator variable z_i, the conditional posterior is:

$$\begin{aligned} & p(z_i = k | z_{-i}, x, \pi, \{\theta_k\}_{k=1}^K, \alpha_0, \lambda) \\ & = p(z_i = k | x_i, \pi, \{\theta_k\}_{k=1}^K) \\ & \propto p(z_i = k | \pi, \{\theta_k\}_{k=1}^K) p(x_i | z_i = k, \pi, \{\theta_k\}_{k=1}) \\ & = p(z_i = k | \pi) p(x_i | \theta_k) \\ & = \pi_k F(x_i | \theta_k) \end{aligned} \tag{3.20}$$

For the mixture weight π, we need to derive its conditional posterior:

$$\begin{aligned} p(\pi | z, x, \{\theta_k\}_{k=1}^K, \alpha_0, \lambda) &= p(\pi | z, \alpha_0) \\ &= Dir(n_1 + \alpha_0/K, \cdots, n_k + \alpha_0/K) \end{aligned} \tag{3.21}$$

For the cluster parameters, the conditional posterior is :

$$\begin{aligned} p(\theta_k | \theta_{-k}, x, z, \pi, \alpha_0, \lambda) &= p(\theta_k | \theta_{-k}, x, z, \lambda) \\ &= p(\theta_k | x_k, \lambda) \\ &\propto G_0(\theta_k | \lambda) L(x_k | \theta_k) \end{aligned} \tag{3.22}$$

The mixture weight π is explicitly sampled from a Dirichlet distribution. However, such sampling is difficult when K goes to infinite. One option is to integrate π out. This requires us to derive z_is conditional posterior:

$$\begin{aligned} & p(z_i = k | z_{-i}, x, \pi, \{\theta_k\}_{k=1}^K, \alpha_0, \lambda) \\ & = p(z_i = k | x_i, z_{-i}, \theta, \alpha_0) \\ & \propto p(z_i = k | z_{-i}, \theta, \alpha_0) p(x_i | z_i = k, z_{-i}, \theta_k, \alpha_0) \\ & = p(z_i = k | z_{-i}, \alpha_0) p(x_i | \theta_k) \\ & = \frac{n_{k,-i} + \alpha_0/K}{n + \alpha_0 - 1} F(x_i | \theta_k) \end{aligned} \tag{3.23}$$

- **Collpsed Gibbs Sampling** When using conjugate prior, we can often

integrate out the cluster parameters θ_k and then we need samplez_i only. This method is called as collapsed Gibbs sampling. The justification of integrating out the cluster parameters is due to the Rao-Blackwell Theorem, which states that marginalization of some variables from a joint distribution always reduces the variance of later estimates. Then the posterior distribution of z is:

$$\begin{aligned} & p(z_i = k | z_{-i}, x, \alpha_0, \lambda) \\ & = p(z_i = k | x_i, z_{-i}, x_{-i}, \alpha_0, \lambda) \\ & \propto p(z_i = k | z_{-i}, x_{-i}, \lambda, \alpha_0) p(x_i | z_i = k, z_{-i}, x_{-i}, \lambda, \alpha_0) \\ & = p(z_i = k | z_{-i}, \alpha_0) p(x_i | x_{k,-i}, \lambda) \end{aligned} \tag{3.24}$$

Despite the different expression of Dirichlet process, the general rules of sample-based inference is to estimate variables from its posterior distribution. However, it is difficult in practice because samplers that propose local changes exhibit poor convergence. And many adapted methods has been proposed to address this issues. The majority of the samplers fit into one of two categories: collapsed-weight samplers that marginalize over the mixture weights or instantiated-weight samplers that explicitly represent them.

Collapsed-weight (CW) samplers using both conjugate [1, 2, 7, 11] and non-conjugate[8, 12] priors sample the cluster labels iteratively one data point at a time. When a conjugate prior is used, one can also marginalize out cluster parameters. However, these methods often exhibit slow convergence. Additionally, due to the particular marginalization schemes, these samplers cannot be parallelized.

Instantiated-weight (IW) samplers explicitly represent cluster weights, typically using a finite approximation to the DP [6, 5]. Recently,[3] and [13] have eliminated the need for this approximation; however, IW samplers still suffer from convergence issues. If cluster parameters are marginalized, it can be very unlikely for a single point to start a new cluster. When cluster parame- ters are instantiated, samples of parameters from the prior are often a poor fit to the data. However, IW samplers are often useful because they can be parallelized across each data point conditioned on the weights and parameters. We refer to this type of algorithm as inter-cluster parallelizable, since the cluster label for each point within a cluster can be sampled in parallel.

3.5 Distance Metric Learning

Metric Learning is first introduced in [**?**]. The prime aim is to learn a proper distance metric given some data paris labeled as similar or dissimilar and the metric should make similar pairs close to each other whild separate dissimilar pairs apart. The mostly widely used distance metric is the Mahalanobis dis-

tance. Then the problem transforms to a optimization problem under certain constraints to minimize the distance of similar pairs and separate the dissimilar ones with a certain margin. Mathmatically, a Distance Metric Learning (DML) problem can be formulated as semi-supervised learning problem where, given side information in the form of data pairs that are determined to be similar or dissimilar, we learn a metric M, which places similar data pairs close to each other, and dissimilar data pairs as far apart as possible. This leads to a quadratic program whose size grows super-linearly with the size of the data and of the side information. Specifically, let M defines a Mahalanobis distance $(xy)^T M(xy)$, where x, y are d-dimensional feature vectors and $M \in R^{d\times d}$ is a positive semidefinite matrix (to be learned). If we denote $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{|\mathcal{S}|}$ and $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ as the set of similar and dissimliar labeled pairs repectively, mathmatically, the metric learning problem could be formulated as

$$\begin{aligned} \min_{M} \quad & \sum_{(x,y)\in\mathcal{S}} (x-y)^T M(x-y) \\ \text{s.t} \quad & (x-y)^T M(x-y) \geq 1, \forall (x,y) \in D \\ & M \geq 0 \end{aligned} \tag{3.25}$$

where $M \geq 0$ denotes that M is required to be positive semidefinite. Weinberger [**?**] employed a similar semidefinite formulation for K-nearest neighbor classification. The metric is trained with the goal that the k-nearest neighbors always belong to the same class while examples from different classes are separated by a large margin. Globerson and Roweis[**?**] proposed a formulation aiming to collapse all examples in the same class to a single point and push examples in other classes infinitely far away. Kostinger et.al [**?**] proposed a fast DML method based on likelihood-ratio test, which does not require iterative optimization procedures and hence is very efficient and scalable. However, distance metrics learned by this method yield poor performance empirically and this method is less flexible (can only deal with equivalence constraints). Because M is a $d \times d$ matrix, when the feature dimension d$d$is huge such as in web-scale problems where web pages are represented with bag-of-word (BOW) vectors that are millions of words long, or in computer vision where hundreds of thousands of features are routinely extracted from images- the size of M quickly becomes intractable for a single machine; e.g., if d contains 1 million features, then M contains 1 trillion parameters. Storing this M requires 4 terabytes of memory, to say nothing of the massive computational cost of learning so many parameters. Even worse, the number of labeled data pairs can easily exceed billions or even trillions, particularly in web data. For instance, on social website, photos are organized into many interests groups by the users if we simply regard photos in the same group as similar, and those in different groups as dissimilar, we can rapidly generate a huge number of similar/dissimilar pairs.

In this scenario related to big data volume, we could easily resort to the modern distribution system which has the great performance dealing with scalability. But for distance metric learning problem, it is challenging that the basic formulation requires both substantial redesian of the original expression and a parallel computing strategy for optimization which is not supported under the current bulk synchronization parallelism.

3.6 Summary

In this section, we briefly reviewed the concepts of Bayesian Nonparametric Models theory and metric learning basics. For BNP, we introduced its three different representations-Pólya urn, Chinese Restaurant Process and Stick-breaking Process, and each one shows the different properties of Dirichlet Process. Also, we delve into the inference strategies of Dirichlet Process, including Markov Chain Monte Carlo. And for distance metric learning, the basic schema is introduced and as our main goal, the adaption ability on big data set is discussed.

Generally, Dirichlet Process, as the most representative modeling in Bayesian Nonparametric analysis, are an emerging trend for buildin gflexible models whose strcture grows and adapts to data. But the easy and flexible modeling procedure does not guaranteen a relaxed inference procedure especially when the data set grows larger.

4 Methodology

This section will describe the methodology to fulfill the research target. As mentioned in the 2, the primary goal of this research is to introduce the bayesian learning method into active learning framework to handling the learning problem with variant size of data set and enable the model flexible enough to adapt the model complexity.

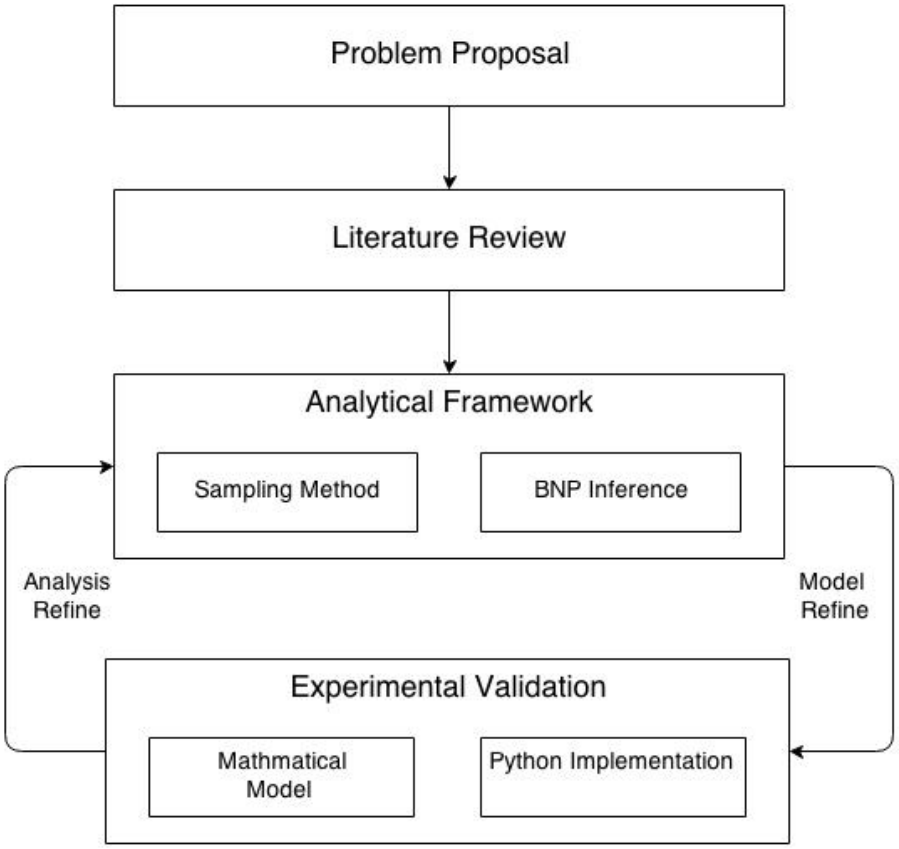


Figure 4.1: Research Flowchart

Fig4.1 is the main flowchart. briefing the detailed process, and in each main stage the major work should be:

1. **Literature Review**

   This part contains a comprehensive review on the machine learning literature, especially on the relevant topics in this proposal. Besides the classical learning theories, extensive reading on analysis of their characteristics and application should also be focused on. The goal is to gain a general understanding of the theory framework, relation to other learning theory, their history, development and current research.

2. **Design a metric learning framework that will support big data set**

   Currently, most of the existing distance metric learning approaches involve computationally expensive procedure in which it requires an iterative eigen-decomposition of metric matrix M, let alone when it comes to big data set. We are going to design a distributed computation framework to handle the problem. The basic idea is to redesign the original algorithm express to avoid the eigen-value decomposition which require expensive computation.

   We plan to design a distributed framework that implements parallel DML. Based on the Bayesian Nonparametric method, we partition the data onto different nodes. Each nodes uses the data it hods to update the model parameters to gain a local optimum. The challenge is the synchronization of the parameters trained on each nodes. We plan to use a central node to synchronize the parameters across machines.

3. **Propose a new Gibbs Sampling method to accelerate the convergency in DPMM**

   As shown in literature review, traditional Gibbs sampling in Dirichlet process mixture model demonstrate a low convergency, especially when applying on a large data set since the sampler will traverse all the observations to get the parameters updated according to the posterior distribution. Though several methods has been proposed by a representation of the basic DPMM model, such as introducing split-merge model to improve the probability of generating new cluster, super cluster and sub-cluster to make parallel computing possible to this framework. But this focuses on the sampling the procedure leaving the original data set intact. In order to overcome the curse of large data set, we are trying to propose a method that including a high quality preprocessing of the data set according to a criterion coupling with the sampling process. This idea is motivated by the shema of Active Learning theory.

   We plan to avoid the slow convergency rate caused by the size of data set by selecting a subset of the original observations as a initial training set and during the sampling procedure if a new cluster is generated we will jump into a procedure selecting observations that will support the new cluster. The first key point in this algorithm is to design a new criterion function and second one is to expolore whether the bias of the initial training set will affect the final training result.

4. **Metric Learning on multi-modal problem**

   Currently, the metric learning problem is mostly applied on the single

modality problem and for multi-modal problem, it required further research. McFee and Lanckriet[**?**] proposed a kernel-learning methods. But it is computationally costly because it involves optimizing over multiple high-dimensional positive semi-definite matrices and it is hard to scale to large date set. Another piece of work by [**?**] integrates multi-wing harmonium model and large margin learning to predictively learn a latent subspace for multi-view data by using the class labels.

We plan to propose a general supervised framewok of multi-modal distrance metric learning methods which can flexibly embed arbitrary number of features modalities using the "similar" and "sissimilar" pairs.

5. **Implementation and Experiment**

   This steps include implementing the proposed framework and test the algorithm on different data set regarding the specific problem this framework will be used to evaluate its performance. The validation should contain two stage, the first one being comparison with classical and state-of-art algorithm in active learning and bayesian nonparametric model leanring respectively, the second one being test the performance of the whole framework.

   The result will be compared to the state-of-art methods,in the area of the problem to be solved, such as classification, regression or density estimation, on the basis of both the efficiency and accuracy. The accuracy is to test how our proposed method performs regarding to the specific target of the problem. And the efficiency is to check whether this framework will decrease the rely on calculation resource, both time and space. The performance will be the feedback for analytical refinement.

5 Research Plan

In this section, a research plan is listed to achieve the previous proposed objective as well as some resources required for the research.

5.1 Research Timetable

In this section, a timetable of the research plan will be described. Basically, this research is planned to span the first one and a half year of my research.

Year 2014

- COMP9417 Data Mining and Machine Leanring course
- Basic Bayesian theory learning
- Review Bayesian Non-parametric Theory
- GSOE9400 Research Course
- Review Active Learning Theory
- Review sampling methods and inference theory on bayesian learning

Year 2015

- Develop the proposed sampling methods and implementation
- Experiment of the proposed learning theory and testing
- Reviewing the topic of Metric Learning

6 Current Progress

Currently, the progress related to each topic is listed below:

- **Active Learning**
  - Gained a general understanding of the theory
  - Diving into applications of different sampling framework in pratical problem
  - Primary framework programming and testing

  For the learning process, I employed the SVM and Bayesian frame- work. And for querying strategy, I used Random Sampling, uncertainty Sampling. The test dataset is used is the Digit11, the dataset used in The Semi-Supervised Learning Book. This dataset includes 1500 instances and each one has 241 features. Below is the result of the current programming.

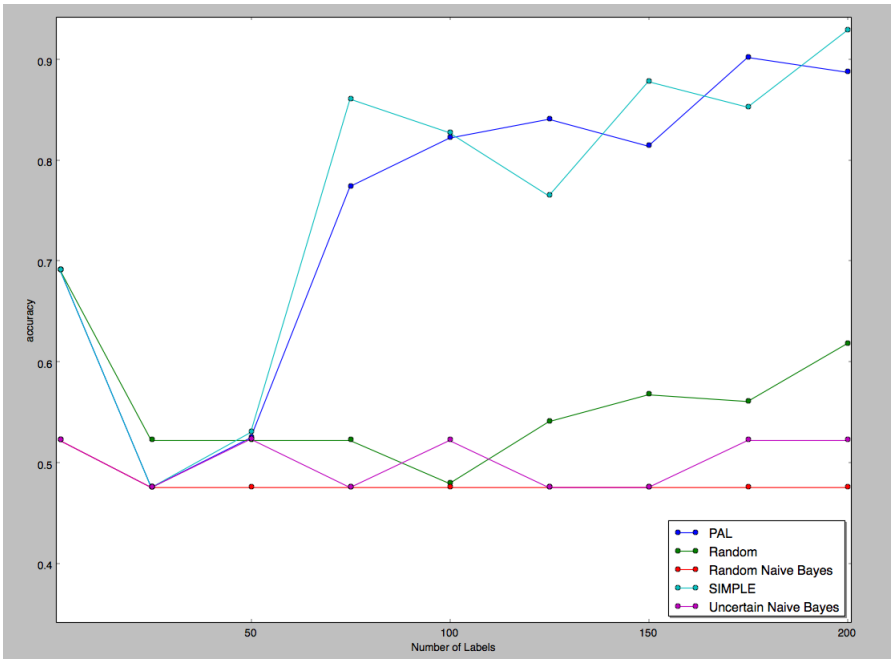


Figure 6.1: Active learning result

As can be seen from the figure, if active sampling procedure is applied, the performance is on average better than random sampling method(the base test of the test).

- **Bayesian Nonparametric Model**
  - Gained a general understanding of the theory

- Currently reviewing on the Gaussian process used for BNP regression
- Primary framework programming

7 Conclusion

In this proposal, we concerned about the shortcoming in traditional active learning and bayesian nonparametric model in order to make them more applicable on big data problem. To adress these problems, we propose to introduce bayesian nonparametric model to active learning framework. On one hand, active learning theory is motivated on the scenario that concerns learning cost. It actively queries samples that will mostly improve the performance at each step, reaching similar or better performance than traditional supervised learning methods but at less cost. On the other hand, BNP model tries to slove the constraints of parameters by combining the bayesian and non-parametric methods. It allows for a dynamic change of param- eters in the learning procedure. This flexibility is very important when trying to learn on ever-changing and complex data sets. As in big data set, abundancy and complexity are more likely to happen than on small data set. We aim to take advantages of the merits , as for active learning the efficiency and for BNP the flexibility, to make up for their disadvantage, as for active learning low sampling procedure and and for BNP low convergency rate, thus this whole framework will perform better both on efficiency and on accuracy in big dataset.

Bibliography

[1] C. A. Bush and S. N. MacEachern. A semiparametric bayesian model for randomised block designs. *Biometrika*, 83(2):275–285, 1996.

[2] M. D. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588, 1995.

[3] S. Favaro, Y. W. Teh, et al. Mcmc for normalized random measure mixture models. *Statistical Science*, 28(3):335–359, 2013.

[4] S. J. Gershman and D. M. Blei. A tutorial on bayesian nonparametric models. *Journal of Mathematical Psychology*, 56(1):1–12, 2012.

[5] H. Ishwaran and L. F. James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453), 2001.

[6] H. Ishwaran and M. Zarepour. Exact and approximate sum representations for the dirichlet process. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, pages 269–283, 2002.

[7] S. N. MacEachern. Estimating normal means with a conjugate style dirichlet process prior. *Communications in Statistics-Simulation and Computation*, 23(3):727–741, 1994.

[8] S. N. MacEachern and P. Müller. Estimating mixture of dirichlet process models. *Journal of Computational and Graphical Statistics*, 7(2):223–238, 1998.

[9] A. McAfee, E. Brynjolfsson, T. H. Davenport, D. Patil, and D. Barton. Big data. *The management revolution. Harvard Bus Rev*, 90(10):61–67, 2012.

[10] P. Müller and F. A. Quintana. Nonparametric bayesian data analysis. *Statistical science*, pages 95–110, 2004.

[11] R. M. Neal. Bayesian mixture modeling. In *Maximum Entropy and Bayesian Methods*, pages 197–211. Springer, 1992.

[12] R. M. Neal. Markov chain sampling methods for dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2):249–265, 2000.

[13] O. Papaspiliopoulos and G. O. Roberts. Retrospective markov chain monte carlo methods for dirichlet process hierarchical models. *Biometrika*, 95(1):169–186, 2008.