Literature Report on DP-means

Aaron Maurer

STAT 302, Spring Quarter 2015

1 Overview

For this report, I chose to read “Revisiting k-means: New Algorithms via Bayesian Nonparametrics”[1]. This article introduces an alternative to the k-means clustering algorithm called DP-means, where instead of specifying a number of clusters at the start, then working to optimize their fit, new clusters are introduced when a point in the data set is too far from one of the existing clusters. This in itself is a ‘hard’ clustering algorithm, which outputs an assignment to clusters rather than a posterior distribution across clusters typical of a Bayesian model. However, this paper demonstrates that this algorithm is the limit of a Bayesian model. In the model, the total distribution is a mixture of Gaussian distributions, with the number of components coming from a Dirichlet process. This is the source of the name of the algorithm: Dirichlet process means or DP-means.

In addition to a description and derivation of this algorithm, the paper offers a few extensions and simulations. The most interesting of theses is an extension of DP-means called the “hard Gaussian hierarchical Dirichlet process”, over multiple data sets, local clusters are simultaneously fit on each data set to match a set of global clusters. This is again a hard clustering algorithm, but it can be derived as well in a Bayesian fashion by taking the limit of a Bayesian model. The next extension shows how DP-means extends to spectral clustering in a similar fashion to how k-means does. Similarly, the author also shows that DP-means can be extended to graph cut problems. The paper concludes with a few simulations demonstrating the effectiveness of DP-means and its multiple data set extension.

This article offers obvious extensions of how the classical Bayesian statistics we learned can be extended to a machine learning algorithm. The DP-means algorithm is a hierarchical model which uses a Dirichlet process as the prior on the number of clusters. The method for fitting the model is the limit of a Gibbs sampling algorithm. The hard Gaussian HDP extension is a typical Bayesian extension of the original model, where an additional level prior distributions is added to an existing model. All together, this

[1] Brian Kulis and Michael I. Jordan, “Revisiting k-means: New Algorithms via Bayesian Nonparametrics”, *CoRR* (2011): http://arxiv.org/abs/1111.0352

forms an interesting competitor to k-means, being similarly easy to compute, but built on classic Bayesian principles.

2 Paper Contents

2.1 DP-Means Algorithm

The idea of clustering arises, in statistical terms, from the idea that a set of random variables are drawn from a mixture distribution. In other words, a random variable X_i is drawn in a two stage process. First, a multinomial variable z_i with k possible outcomes is drawn. Then, $X_i \bigm| z_i \sim F_{z_i}$, where $\bigl\{F_j \bigm| j \in \{0, ..., k\}\bigr\}$ is some set of probability distributions. The goal of a clustering method is then to impute the z_i based on the X_i, revealing important underlying structure in the data. Bayesian models provide a natural way to fit probability distributions to z_i, in particular without choosing k a priori, but there issue is that they, classically, are complicated to compute on large data sets and don't necessarily scale well.

Thus, the k-means algorithm, where k must be specified beforehand, remains the most commonly used algorithm. This method can be thought of as designed for a model where z_i is multinomial k, for fixed k, and then $X_i \bigm| z_i \sim N_p(\mu_c, \sigma I_p)$ for some set of mean vectors $\bigl\{\mu_c \bigm| c \in \{0, ..., k\}\bigr\}$. To find $\{\mu_i\}$ using k-means, one picks initial guesses for the $\hat{\mu}_c$, then alternates assigning $\hat{z}_i = \arg\min_c \|X_i - \hat{\mu}_c\|$ and

$$\hat{\mu}_c = \bar{X}_c = \frac{1}{\bigl|\{i \bigm| z_i = c\}\bigr|} \sum_{i \bigm| z_i = c} X_i$$

until the μ_c and z_i have converged. This method allows little flexibility, but is relatively easy to compute, even for large data sets. The authors of the paper notes that the fitting process is the limit of what the EM algorithm would do for a mixture of k Gaussians if the variance of each, σ, goes to 0.

In designing the DP-means algorithm, the authors "attempt to achieve the best of both worlds by designing scalable hard clustering algorithms from a Bayesian nonparametric viewpoint." To this end, they start with a purely Bayesian model for how the data would arise. Here, a prior is put on k, the weights of each cluster, the distribution of the centers, and the conditional distributions of the X_i given the centers. Then, as with k-means, a hard clustering algorithm is the limit of a Gibbs sampler as a variance parameter goes to 0. The Bayesian model for how the data arises sequentially is something like this:

- The z_i arise from a Dirichlet process. This means that given a parameter α, for the ith draw, if we have k clusters already
  - $z_i = c$ for $c \le k$ with probability $\frac{|\{j|j<i, z_j=c\}|}{\alpha+i-1}$
  - $z_i = k + 1$ with probability $\frac{\alpha}{\alpha+i-1}$

- The cluster centers $\mu_1, ..., \mu_k \sim N(0, \rho I)$ for some variance parameter ρ.

- Finally, $X_i \mid z_i \sim N(\mu_{z_i}, \sigma I)$ for another variance parameter σ.

The authors fix ρ and σ, and then set $\alpha = \left(1 + \frac{\rho}{\sigma}\right)^{d/2} \exp\left(-\frac{\lambda}{2\sigma}\right)$, where d is the dimension of the X_i. To set up a Gibbs sampling algorithm, the authors calculate the conditional distributions of the unknown parameters $z_1, ..., z_n$ and $\mu_1, ..., \mu_k$. Let us say that there are k clusters, excluding observation i, at any given point. Also, let the number of points assigned to cluster c excluding point i be $n_{-i,c}$. This yields, for the conditional distribution of the z_i,

$$\begin{aligned}
\mathrm{P}(z_i = c \mid X_{1:n}, z_{-i}, \mu_{1:k}, \rho, \sigma) &= \mathrm{P}(z_i = c \mid X_i, z_{-i}, \mu_{1:k}, \rho, \sigma) \\
&\propto \begin{cases} \mathrm{P}(X_i \mid z_i = c, \sigma, \rho)\mathrm{P}(z_i = c \mid z_{-i}) & \text{if } c = k+1 \\ \mathrm{P}(X_i \mid z_i = c, \mu_c, \sigma)\mathrm{P}(z_i = c \mid z_{-i}) & \text{if } c \leq k \end{cases} \\
&\propto \begin{cases} (2\pi\sigma)^{-d/2} \exp\left(-\frac{\lambda}{2\sigma} - \frac{\|X_i\|^2}{2(\rho+\sigma)}\right) & \text{if } c = k+1 \\ n_{-i,c}(2\pi\sigma)^{-d/2} \exp\left(-\frac{\|X_i - \mu_c\|^2}{2\sigma}\right) & \text{if } c \leq k \end{cases} \\
&\propto \begin{cases} \exp\left(-\frac{\lambda}{2\sigma} - \frac{\|X_i\|^2}{2(\rho+\sigma)}\right) & \text{if } c = k+1 \\ n_{-i,c} \exp\left(-\frac{\|X_i - \mu_c\|^2}{2\sigma}\right) & \text{if } c \leq k \end{cases}
\end{aligned}$$

For the conditional distribution of the μ_c given everything else, it yields, where $\bar{X}_c$ is the mean over all $X_i \mid z_i = c$,

$$\begin{aligned}
\mathrm{P}(\mu_c = \tau \mid X_{1:n}, z_{1:n}, \mu_{-i}, \rho, \sigma) &\propto \mathrm{P}(\bar{X}_c \mid n_c, \mu_c, \sigma)\mathrm{P}(\mu_c = \tau \mid \rho) \\
&\propto \exp\left(\frac{1}{(\rho^{-2} + n\sigma^{-2})} \left\| \bar{X}_c - \frac{n_c \bar{X}_c}{\sigma^2(\rho^{-2} + n\sigma^{-2})} \right\|^2\right)
\end{aligned}$$

Now, as one sends σ to 0, both of these probabilities become point masses. For the first probability, $\mathrm{P}(z_i = c \mid X_{1:n}, z_{-i}, \mu_{1:k}, \rho, \sigma)$, it is a point mass on $z_i = c \leq k$ if $\|\mu_c - X_i\|$ is minimal and less than λ, otherwise it is a point mass on $z_i = k+1$. The second distribution, of $\mu_c \mid X_{1:n}, z_{1:n}, \mu_{-i}, \rho, \sigma$, is just a point mass on $\bar{X}_c$. Thus, the Gibbs sampling algorithm initialized with one cluster at the global mean turns into the DP-means algorithm, which is iterated till convergence:

- Initialize $k = 1$, $\mu_1 = \bar{X}$, and $z_i = 1 \ \forall i$

- For each X_i

  - **If X_i is too far from a cluster, introduce a new one:**

    if

    $$\lambda \leq \|X_i - \mu_c\|^2 \ \forall c$$

    set $k = k + 1$, $z_i = k$ and $\mu_k = X_i$

  - **Otherwise, have it join the nearest cluster:**

    Otherwise

$$z_i = \arg\min_c \|X_i - \mu_c\|^2$$

- **Set the cluster centers to the mean of the cluster:** For each c, set $\mu_c = \bar{X}_c$

2.2 Hierarchical DP-means

As is common with Bayesian models, a hierarchical model such as DP-means can be expanded to a more complicated model by adding an additional level of hierarchy by putting a probability distribution on some of the hyperparameters. This is exactly what the authors of this paper do to DP-means. They replace the original fixed Dirichlet process with a mixture of Dirichlet processes as determined by an additional Dirichlet process. This is called a hierarchical Dirichlet process. While this doesn't yield anything interesting for a single data set, this allows for a clustering algorithm that clusters over multiple data sets. The idea is that one might have multiple data sets of different observations, but with identical measurements taken for each. In this setting each data set might have a set of local clusters partially, but not fully, align with local clusters of each other data set. This method will allow the simultaneous identification of a set of global clusters while the data sets borrow strength from each other to identify the shared local clusters within each data set.

The "Revisiting k-means:..." paper doesn't go into the full derivation of this model from a Bayesian setup, instead pointing to the original paper on the hierarchical Dirichlet process[2] and indicating that "the limiting process described earlier for the standard DP can be straightforwardly extended to the HDP". To partially describe what this would look like, the Bayesian model for how the data arises goes something like this:

- The z_{ij}, which are the local indicators of the group for the ith observation in the jth data set, are drawn first. Lets say, up to the current z_{ij} being drawn, there are $k_{j.}$ local clusters in the data set, each group has n_{jc} observations in it, and z_{ij} is the $n_{j.}$th observation drawn. In this setting, z_{ij} is drawn locally by a Dirichlet process with parameter α.

  - As before, $z_{ij} = c \leq k_{j.}$ with probability $\frac{n_{jc}}{\alpha + n_{j.} - 1}$
  - $z_{ij} = k_j + 1$ with probability $\frac{\alpha}{\alpha + n_{j.} - 1}$

- v_{jc}, which is the association of the cth group in the jth data set with a global cluster, is drawn next. Let us say that there are g global clusters,$k_{..}$ is the total number of local clusters, and $k_{.p}$ is the

[2] Yee Teh, Micheal Jordan, Matthew Beal, David Blei, "Hierarchical Dirichlet Processes", JASA, 1001(476):1566-1581, 2006.

number of local clusters, in all data sets, for which $v_{jc} = p$. In this setting, v_{jc} is drawn globally by a second Dirichlet process with parameter γ:

  - $v_{jc} = p \leq g$ with probability $\frac{k_{.p}}{\gamma+k_{..}-1}$
  - $v_{jc} = g + 1$ with probability $\frac{\gamma}{\gamma+k_{..}-1}$

- All the global cluster centers μ_p are drawn iid from $N(0, \rho I)$ for some variance parameter ρ

- Finally, each observed data point $X_{ij} \mid z_{ij}, v_{jz_{ij}} \sim N(\mu_{v_{jz_{ij}}}, \sigma I)$ for another variance parameter σ

Its important to note that the particular local cluster matters only so far as which global cluster it maps to. All the X_i that are map to the same global cluster are iid.

After fixing σ and ρ in a similar manner to before, the conditionals can be constructed for a Gibbs sampler. Then α and γ are in terms of local and global parameter λ_l and λ_g. In the same way, as σ is sent to 0, this makes each conditional distribution becomes a point mass. This turns the Gibbs sampler into the "Hard Gaussian HDP", which is the hard clustering algorithm to fit the local and global clusters across data sets. The algorithm is this, iterated till convergence.

- Initialize $g = 1$, $k_j = 1$, $\mu_1 = \bar{X}$, $z_{ij} = 1$, $v_{j1} = 1$ for all i and j.

- For each X_{ij}:

  - **If the point is too far from a global cluster, introduce a new one:**

    If

    $$\lambda_l < \|X_{ij} - \mu_{v_{jc}}\|^2 \;\; \forall c < k_j \text{ and } \lambda_l + \lambda_p < \|X_{ij} - \mu_p\|^2 \;\; \forall p \leq g$$

    set $k_j = k_j + 1$, $g = g + 1$, $z_{ij} = k_j$, $v_{jk_j} = g$, $\mu_g = X_{ij}$

  - **If the point is too far away from a local cluster, but near a global cluster, add a new local cluster:**

    If

    $$\min_c \|X_{ij} - \mu_{v_{jc}}\|^2 > \min_p \|X_{ij} - \mu_p\|^2 + \lambda_l$$

    set $k_j = k_j + 1$, $z_{ij} = k_j$, and $v_{jz_{ij}} = \arg\min_p \|X_{ij} - \mu_p\|^2$

  - **Otherwise, match to one of the nearest clusters:**

    Set

    $$z_{ij} = \arg\min_c \|X_{ij} - \mu_{v_{jc}}\|^2$$

    There may be multiple c which fit this criteria, in which case one should be chosen randomly.

- For each local cluster c_j:

- **If the local cluster is too far from a global cluster, introduce a new one:** If

  $$\min_p \sum_{\{X_{ij} \,\big|\, z_{ij}=c_j\}} \|x-\mu_p\|^2 \;>\; \lambda_l + \sum_{\{X_{ij} \,\big|\, z_{ij}=c_j\}} \|x-\bar{X}_{jc_j}\|$$

  Set $g = g+1$, $v_{jc_j} = g$, and $\mu_g = \bar{X}_{jc_j}$

- **Otherwise, match each the local cluster to the nearest global cluster:** Set

  $$v_{jc_j} = \arg\min_p \sum_{\{X_{ij} \,\big|\, z_{ij}=c_j\}} \|x-\mu_p\|^2$$

- For each global cluster p:

  - **Recenter each global cluster:**

    Set

    $$\mu_p = \sum_{\{X_{ij} \,\big|\, v_{jz_{ij}}=p\}} x$$

As mentioned above, the local clustering is irrelevant in the end. This means that the partition of points which map to the same global cluster into local clusters is nonidentifiable. Thus, the important value is the global cluster association of each point, $v_{jz_{ij}}$.

2.3 Spectral Clustering with DP-means

A convieient way of thinking of fitting clusters is that one is trying find the set of cluster centers which subsititue for the original observations with minimal distortion. Another way of saying this is that we want to project our data onto a set of vectors in the fashion that reduces the sum of the norms, or sum of squares, the least. This is akin to linear regression, where we are looking to project our data onto a plane. Just as with linear regression, where this projection was performed with the hat matrix H, for each clustering we can use a weight matrix W to do this. Its definition is that $W_{ij} = \frac{1}{n_c}$ if X_i and X_j are to be placed in the same cluster, and 0 otherwise. This means WX replaces each row of X with its cluster mean. We can accordingly calculate a clustering's total distortion as $\text{trace}(X'X) - \text{trace}(W'X'XW)$. Since all the rows in WX will be identical within a given cluster, we can replace this by a weighted sum of the unique cluster centers. Thus, $\text{trace}(W'X'XW) = \text{trace}(Y'X'XY)$, where Y is the normalized $n \times k$ clustering matrix, which will have $Y_{ic} = \frac{1}{\sqrt{n_c}}$ if observation i is in cluster c, and 0 otherwise.

Both clustering algorithms, k-means and DP-means, attempt to minimize this distortion, or maximize $\text{trace}(Y'X'XY)$, under a certain criteria. For k-means, that criteria is over all normalized clustering matricies Y with k is fixed, while for DP-means, it turns out the criteria is to minimize $\text{trace}(Y'X'XY) - \lambda k$ over all normalized clustering matricies Y and corresponding k. A natural relaxation, or simplification of both of these targets is to let Y be any orthonormal matrix rather than just a normalized clustering matrix. This is where the spectral part comes in; let $X'X = Q\Lambda Q'$ be the eigen decomposition of $X'X$, where Q is the orthonormal matrix of eigenvalues and $\Lambda = \text{diag}(\lambda_1, ..., \lambda_n)$ is the diagonal matrix of eigenvalues in decreasing order. Its obvious that the optimal Y in both cases will be a subset of the columns of Q. For k-means, it will be the the first k columns, corresponding to the largest k eignvalues, while for DP-means, it will be all columns such that $\lambda_p > \lambda$.

The reason this relaxation is useful is two fold. First, when one has the relaxed matrix Y above, they can perform a potentially simpler clustering on it, treating it as a new data matrix. This clustering will also work very well on the original data set, since Y is a set of principle components from it. More importantly though, if one thinks of $X'X$ as a matrix meauring similarity between all the observations in a particular fashion, it can be replaced with another similarity matrix K which uses a different critiera. For instance $K_{ij} = \exp\left(-\|X_i - X_j\|^2\right)$ is common. Then, one similarly can make a matrix Y which is the first k eigenvectors for k means or all eigenvectors with eigenvalues greater than λ for DP-means. The advantage of this is when X is not a nice gaussian mixture, as needed for k-means or DP-means to work well, one can use a different measure of similarity to generate K and Y such that Y is a proper mixture.

2.4 Remaining Content

The remaining part of the paper had two sections. The first of these has to do with graph clustering. The paper only gives a very brief desciription of the result, without enough detail to understand any of it without knowing the area already. The gist is that there is a natural extension of DP-means to the graph cut problem, just as there is with k-means. It turns out that, as has been demonstrated similarlily in the three earlier sections, DP-means ends up searching for an equivalent optimal result as k-means given the right parameterization.

The final section deals with a number of simulations that the authors conducted comparing fitting models by k-means, DP-means, and a full Dirichlet process mixture with Gibbs sampling. On individual data sets, they all generally produce similar quality clusters. On large data sets though, Gibbs sampling becomes computationally infeasible while both k-means and DP-means run in similar time. The impressive result though come from the hard Gaussian HDP. To test it, they created 15 cluster centers, then created 50 data sets which each had 5 data points generated from 5 randomly chosen clusters. The hard Gaussian HDP clustering, fit accross all data sets, performed considerably better than k-means or DP-means fit on all the data sets pooled together or each data set individually.

3 Similarities to Class Material

The central idea from our class which this paper uses is that of the hiearchical model. In the basic DP-algorithm, the underlying Bayesian model treats the observations X_i as exchangable. This allows the X_i, although independent conditioned on their clusters, to lend strength to each other in infering both their cluster associations and the cluster means. This concept, where the observations share information, is a central one we saw in many contexts in class. The hard Gaussian HDP model displayed another precept of heiarchical models we saw; Bayesian approaches are accomodating to additional levels of hiearchy being added to models. This is exactly how the hard Gaussian HDP was developed. They added an additional level of hiearchy to the DP-means model by putting a prior on the Dirichlet process for each data set. This allows for different mixtures in each data set, while still lending strength accross all data sets on several levels.

The second important idea we saw before in class was fitting the model by a MCMC process, in particular Gibbs sampling. There model was complicated to the point that deriving an analytical solution would be at best infeasible. Since the marginal distributions of parameters could be expressed in closed form, this allowed a Gibbs sampler to be set up. An interesting twist I thought was stretching the Gibbs Sampler till it was deterministic so as to derive a hard clustering algorithm. While it wasn't like the other markov chains we saw which were irreducible, instead having an absorbing state, it still proved useful in fitting the model.

The final idea from class that stuck out was the idea of an uniformative prior. While this isn't how it was described in the paper, several of the priors struck me as being set up to be largely uninformative. For instance, by sending the within cluster variance to 0 while the cluster mean variance was fixed, the result was essentially a uniform prior on $\mathbb{R}^p$ for the cluster centers. The result was that the cluster centers were estimated as the mean, rather than a mix of the data mean and prior mean. Certainly, taking the limit of a distribution as a parmeter goes to 0 or infinity was a common trick we saw to set up uninformative priors.

4 My Thoughts

On the whole, I found this paper very interesting to read, and learned a lot from it and the related papers I looked up. However, as a practical matter, they established DP-means as a largely equivalent method to k-means rather than a superior method. Through out the paper, they showed that they the two methods acheived similar results in similar times, and were both applicable in the same contexts. The main difference was in how they are parameterized. In practice, with either one, I would want to try several different values of the parameter and compare the fit. I'm not sure why DP-means would make deciding on the proper parameter easier. I would just be looking for a number of parameters to map to the same number of clusters with DP-means, while with k-means I would be looking for a falloff in the marginal improvement in distortion. I like that its parameter, λ is akin to the complexity parameter we see in a lot of machine learning algorithms, but that is more from an aesthetic point of view than anything else.

The value of the Bayesian approach, for me, comes with the hard Gaussian HDP algorithm. As mentioned before, adding the hiearchy to Bayesian models is natural in a fashion it often isn't with other types of models. I assume this is why there is not, or I am unaware of, a k-means based algorithm which can similarly be fit simaltaneously accross many data sets. I would certainly choose this algorithm if I ran into a situation where I was trying to fit clusters accross many data sets and had reason to expect that not all clusters were within all data sets. That said, I think that situation is a somewhat rare occurance. Either way, it is a good tool to have, and learning about hiearchical Dirichlet processes through this algorithm was interesting.

An extension which I would have liked to see in the paper would be an algorithm for hiearchical clustering in the more traditional sense, where the base clusters within one data set are clustered into meta clusters within that one data set. For instance, one might imagine cluster centers $\mu_{ij} \sim N(\nu_i, \sigma I)$ with $\nu_i \sim N(0, \rho I)$. This is fertile grounds for a similar Bayesian algorithm based on DP-means, and it seems like something which wouldn't be hard to develop.

5 Big Picture

This is a recent paper, only published in 2012, which adds to the relatively recent area of Bayesian nonparametrics which I know at least one of the authors, Micheal Jordan, has been central to. The uniting thread of this field seems to be attempts to model objects which can have infinite dimmension with a countably infinite, nonparametric discrete distrubtion prior. The dirichlet process, which is the conjugate prior to this family of distributions, thus tends to pop up a lot, as it does here. Another example is that one can put a prior on the space of cumulative probability functions by drawing an countably infinite number of point masses and assigning weight to them based on a dirichlet process. This process, though a newer development, is based on the Dirichlet distribution, which has long been a tool of traditional Bayesian statistics. Bayesian nonparametrics is also closely tied to the more general statistics/computer science field of machine learning, with many of the methods of the former being adopted by the latter. This paper was, in large part, an attempt of the authors to push this process further by offering a bayesian nonparametric algorithm which has appealing properties to machine learning, in particular scalability.