Model_section

Danny Luo

4/29/2021

Nonparametric Learning for cluster size K

We assume the score vector for each individual in the survey comes from a mixture Gaussian distribution with unknown number of components K in the mixture. The normal assumption could be largely justified by the symmetric bell shape distribution of 3 dimensions in the personality data as shown in EDA.

To find the unknown number of components or cluster number K, we adopt a Dirichlet process Gaussian mixture model(DPGMM).It is a widely used clustering tool documented in literature.[@gorur_dirichlet_2010] The key motivation behind our adoption of this model is that Dirichlet Process Gauaussian mixture is nonparametric, in that it assumes a nonfixed number of clusters K. It has the following advantages: 1) it learns the number of clusters K from the data. It is extremely convenient especially provided that we do not have strong belief of exact number of personality types underlying this population. 2) it eliminates the necessity of the model selection procedure if we were to use parametric models. If a parametric model is adopted, optimal number of clusters would have to be tested via different runs of model with vaired K using criterion like BIC.[@gerlach_robust_2018] With DPGMM, the model itself returns the optimal "posterior" size.

The sampling model of DPGMM has the form below:

$$\begin{aligned} y_i &\sim N(y|\boldsymbol{\theta_i}), \\ \boldsymbol{\theta_i} = \{\boldsymbol{\mu_i}, \Sigma_i\} &\sim G, \\ G &\sim DP(\alpha, G_0) \end{aligned}$$

To give a brief overview, the process works by first drawing a distribution G from Dirichlet Process DP with concentration paramter α and a base distirbution of G_0. G_0 is a joint distribution of Gaussian paramters $\boldsymbol{\mu}, \Sigma$, which we assume all Gaussian mixture paramters come from. The hierachial process first draw a distribution G from the DP, where $G = \sum_{K=0}^{\infty} \pi_k \delta_{\boldsymbol{\theta_k}}$. That is, we can understand G as $K \to \infty$ random discrete probability measure, where $\delta_{\boldsymbol{\theta_k}}$ is a point mass centered on $\boldsymbol{\theta_k}$.[@teh_dirichlet_2010]. A stick-breaking property construction of the DP process suggests that most probability mass is concentrated on a few values, that is, when $\boldsymbol{\theta_i}$ is being simulated from G, it will mostly likely take on only a few discrete values given appropriate concentration value α and those few values become our cluster parameters $\boldsymbol{\theta_i}$.

We place the following priors on the paramters α and G_0:

$$\alpha \sim Gamma(a = 2, b = 4)$$
$$G_0(\boldsymbol{\mu}, \Sigma) \sim N(\boldsymbol{\mu}|\boldsymbol{\mu_0} = 0, \Sigma_0) IW(\nu_0, \Phi_0)$$

```
prior_val <- 0.5*log(10000)
```

Our prior choices are justified as follows. We chose a Gamma prior since the postive support matches that of α, and $Gamma(2, 4)$ gives us an expected value of $alpha = 0.5$. Literature has shown that the prior expeceted number of clusters can be expressed using concentration paramter $alpah$ as follows: $\alpha log(N) = 0.5\, timeslog(10000) = 4.6$.[@raykov2016simple]This matches the reporting of a meaingful cluster size of 4 on the same personality data we used in a recent study,[@gerlach_robust_2018] so it makes sense for us to set the $Gamma(2, 4)$ as prior for α. For the base distribution, since the data is scaled, we will set the prior paramter μ_0 to 0, ν_0 to be 1 and $\Phi_0 = I$ to represent a non-informative prior belief.

Semi-conjugacy and full conditionals can be established for G_0 since we do not assume any dependency between mean and variance. For posterior sampling of $alpha$, we adopted the MCMC sampling scheme as described by West (1992). [@west_hyperparameter_1992] This scheme is used in r-package $DirichletProcess$ [@R-dirichletprocess]. Due to the complexity of the full sampling scheme and the output specification of the package $DirichletProcess$, the full details and posteriors will not be discussed here at length.

```
cluster_size_list <- readRDS("cluster_size_MCMC.rds")
mean_estimator <- mean(cluster_size_list)
```

Since the process is very computationally costly, we chose to run the model with a random sample of 10,000 individuals out of over 1,000,000 data in total. The chain included 1000 iterations. We provided the summary statistics from the chain as follows. For each iteration, we estimated the number of clusters through 1) use posterior draw of α to perform stick breaking process in getting exact number of clusters (they can be quite large, i.e. 300-400 total clusters).(we used $PosteriorClusters$ method) [@R-dirichletprocess] 2) since we are only interested the major personality types, we choose to only retain clusters with size proportion (out of 10,000) greater than $\epsilon = 0.1$. This truncated the number of clusters to a magnitude of less than 10. We used this truncated "number of clusters" to derive a mean estimate of cluster number. (The traceplot of truncated number of clusters is in Fig.) The mean estimator is $rmean_estimator$. We rounded up to $K = 5$.We picked the last iteration for our posterior estimate of cluster paramters $\boldsymbol{\theta_i}$ due to the complication of taking the average over cluster paramters of different size.

```
#plot MCMC

plot(1:1000,cluster_size_list,type="l",
     ylab=" truncated cluster size",xlab="Iterature")
```

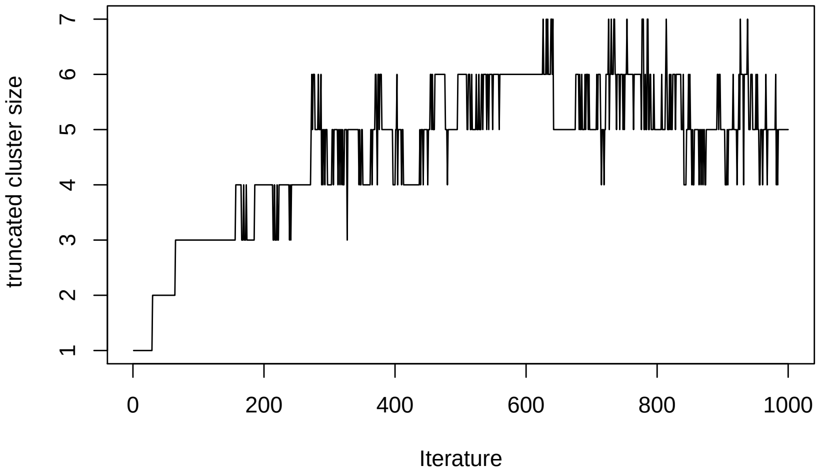

Figure 1: Traceplot for number of cluster

Gaussian mixture Model

To allow our model to capture more data in population, we will use the posterior five clusters to specify a prior to the Gaussian mixture model with fixed cluster size K and run a paramteric Gaussian mixture model on 200,000 individuals. The motivation is that since this process is less costly, we can run on more data to further update the "belief" on personality types and thus achieve more model generatlization. The key assumption we implicitly made here is that DPGMM on random 10,000 individuals is representative of the whole population so that at least $K = 5$ is fairly accurate assumption to feed into paramteric Gaussian mixture. Note that it does not have the same "accuracy" requirment on cluster parameters since Bayesian learning will keep updating them using 200,000 data while K is fixed throughout this phase. This process is justified since it inherently uses the "Bayesian" philosophy of using new information to update prior belief coming from limited data.

The fixed K mixture model can be described by the following:

$$\begin{aligned} y_i | z_i = j &\sim N(\boldsymbol{\mu_j}, \Sigma_j), \\ P(z_i = j) &= p_j, \end{aligned}$$

We assigned the following prior:

$$\begin{aligned} (\boldsymbol{\mu_j}, \Sigma_j) &\sim N(\boldsymbol{\mu_{0j}, \Phi_j}) \times Wishart(n, V) \forall j = 1, ..., K, \\ \boldsymbol{p} &\sim Dirichlet(\boldsymbol{\alpha}), \end{aligned}$$

where for each individual i, z_i is a latent unobserved component membership variable indicating which component in the mixture it belongs to.

We inference on the posterior cluster paramters by MCMC sampling, using the r package mixAK.[@R-mixAK], which does exert some additional prior constraints. We plug in the values of $(\boldsymbol{\mu_j}, \Sigma_j)$ using cluster paramters we derive from DPGMM. However, α vector is required to be uniform. Thus we place a small $\boldsymbol{\alpha} = \mathbf{1}$, representing weak prior belief, allowing the model to learn the posterior weighted towards data itself. mixAK also allows a uniform parameter V and places a hyper Gamma priors on V. (See our elicitation of hyper prior in Appendix) Our MCMC estimator is obtained by simply taking the cluster parameters' mean across all iterations.