Problem and Motivation

In this project we will investigate how to perform Bayesian hierarchical modeling (BHM), and apply BHM to a dataset concerning photon counts for a total of 1332 stars in 39 galaxies. Hierarchical Bayes is a model written in hierarchical form that is estimated using Bayesian methods. A hierarchical model is one that is written in terms of an umbrella modern containing sub – models. The sub – models combine to form the hierarchical model, and Bayes theorem is used to integrate the pieces together and account for all the uncertainty that is present. We will use this model in order to analyze the variability across galaxies. In essence we can compute summary statistics of the stars across galaxies, i.e. how the mean, median, and mode of the brightness of stars vary across galaxies. Using a Bayesian hierarchical model is important in this case because some galaxies only have a handful of data points. With such few data points it is hard to extract information from them, and we need to use the power of prior information in order to ascertain meaningful information on each galaxy. This is also important because perhaps we can use this data in order to begin to discover what the stars are really trying to tell us.

Model Details

For this report we are going to implement and apply a hierarchical Poisson model to a dataset concerning photon counts for stars across galaxies. We are making the following assumptions:

1. The observations in each of their respective categories all follow a Poisson distribution ~ Pois(λ_j)

2. Each of the lambdas follow a gamma distribution ~ Gamma(α,β)

3. Both the parameters of the gamma equation follow Gamma distributions :

α ~ Gamma(a_α,b_α)

β ~ Gamma(a_β,b_β)

Hierarchical models can combine several simple distributions into a layered model to build a very rich overall model. Hierarchical models also naturally reflect the logic of 'trade-off between levels' of information. For instance, if we only focus on the 'local information' the n_i will be modeled as sample mean for each galaxy. If however we combined the data from all schools into a single set then your estimate for the sample mean would simply be μ. However, the hierarchical model stands as a sort of a 'middle ground' which reflects both local information and universal information. This is called 'partial pooling', which means when modeling one specific school you rely on both 'borrowed information' from other galaxies and the local information from that specific school.

Finally, with hierarchical models one can sometimes split the posterior sampling into several easily – handled sample steps, such as a Gibbs – sampler or Metropolis – within – Gibbs sampler. In addition, due to the nature of conditional independence structures in hierarchical models, sometimes the sampling can be parallelized which can significantly accelerate the calculation.

The prior parameters of this particular model are a_α,b_α, a_β,b_β. These parameters influence the distribution of the alpha and beta in how the Poisson values are distributed. Because we are planning on using the metropolis method in order to find the values of alpha and beta, we can initially set these variables to an arbitrary number. As the metropolis sampling method runs, the values of alpha and beta will continue to approach their true values.

Computational Details

Before we even started working on the algorithm to get the posterior samples, we first have to calculate the joint posterior distribution for alpha, beta, and lambdas 1 through 36 given the observations (attached in the appendix). Once we calculated the joint posterior distributions we then proceeded to derive the conditional densities of each of the variables. We were able to calculate the conditional density of the lambdas (1 - 36), which turned out to be a gamma distribution:

$$\text{Gamma}(\alpha + \Sigma Y_{ij} , \beta + n_j)$$

We then were able to calculate the conditional density of the Beta which turned out to be a gamma distribution:

$$\text{Gamma}(a_\beta + J\alpha, b_\beta + \Sigma\lambda_j)$$

However, we were not able to calculate or find a closed for distribution for alpha. I will explain how we worked around this problem when I get to sampling the data. Now that we solved for all the conditional distributions that are of closed form we can move onto the algorithm to draw samples from the posterior distribution.

The first thing is set the prior parameters of the model which are outlined in the above section. We chose to set all these values to the value 1 for arbitrary reasons. The next thing we do is set arbitrary values for the parameters of alpha, beta, and lambdas(1 – 36). In this case we set them all also equal to 2 as well. As we run iterations of the algorithm these values will use the information gleamed from the data to “correct” themselves and reach their true values. After setting these values we start off with sampling a single value from the conditional distribution of Beta. This now becomes the new value we will use going forth of Beta. We then sample 36 separate values from the conditional distribution of lambdas. Because there are 36 distinct galaxies, we have to go through and sample a different lambda for each galaxy using the sum of observations and number of observations pertaining to each galaxy.

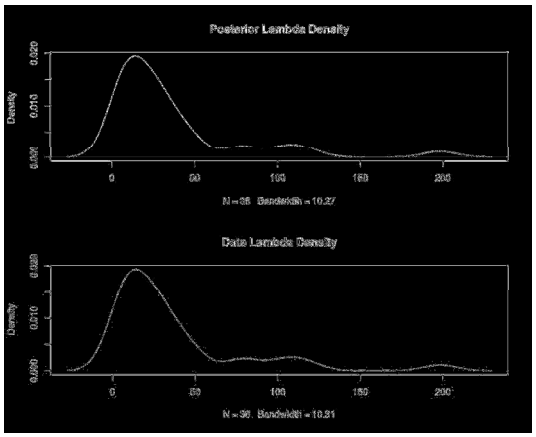
Now we have to tackle sampling the parameter of alpha. Because we could not calculate or come to a closed form distribution that would easily describe alpha, we have to use another method in order to get an alpha sample. We considered doing an inverse – CDF method, but quickly came to the realization that it was very computationally intensive and would be used if other methods failed. We then came to the conclusion that it would be easier to go the route of Metropolis – in – Gibbs sampling in order to find alpha. Also, not only would this method be easier to code, but it would not be too computationally expensive as we are only using it to sample one variable.

In order to carry out the Metroplis – in – Gibbs sampling, we first sample a proposed alpha from a symmetric normal proposal distribution with the current alpha we have and a standard deviation that we set and tuned. We then calculate the log densities of both the current alpha as well as the proposed alpha. To calculate the density we use the entire joint posterior density calculated before starting this algorithm. We use these values to calculate a ratio, coincidently called alpha-ratio with the log alpha proposed density over the log alpha current density. We then sample a single value from a uniform distribution. If the alpha-ratio is greater than the log of the value gotten from the uniform distribution, we replace the current alpha we have with the proposed alpha. This entire process is done within each iteration of the algorithm.

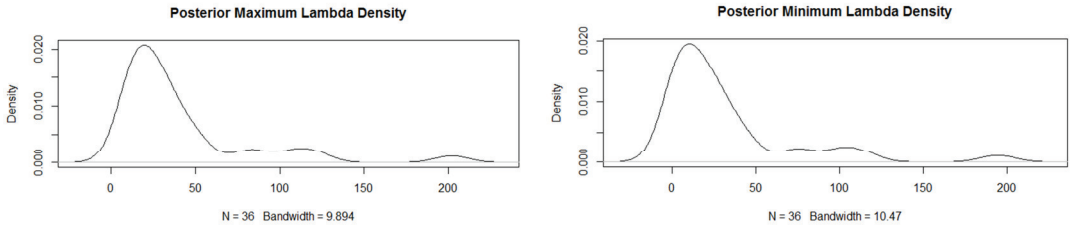
We run this entire process 22,000 times and with each iteration we use the newly gotten values of lambdas (1-36), beta, and alpha. Even though we know that iterating a large amount using the Gibbs sampler will eventually cause the variables to reach their true value, since all these values are dependent on each other we need to take certain precautions. The first thousand or so values are considered 'burnin' because they are too dependent on each other to independently give good estimates of the variables. Running the algorithm 22,000 times we discard the 'burnin' which in this case we set to 2,000. This means we discard the first 2,000 values that we get through the Gibbs sampling method. We then use the remaining 20,000 values in order to compute posterior summary statistics and analyze the data.

Data Analysis

As mentioned before the dataset we are looking at contains photon counts for a total of 1332 stars in 36 distinct galaxies (Galaxies 13, 30, 33 are missing from the data). The 1332 rows correspond to a photon count for a different star. The higher the photon count, the brighter the star. The first column of the dataset in essence serves as an indicator and lists the galaxy numbers that correspond to each star. The second column of the data lists the photon count corresponding to the star. The main goal of the analysis we conducted was to look for the variability across galaxies. We do this by looking at both the difference in the lambda variables (which correspond to the mean of the photon count or the average brightness of each galaxy) across each distinct galaxy. A good way to start is by looking at the distribution density of the lambdas.

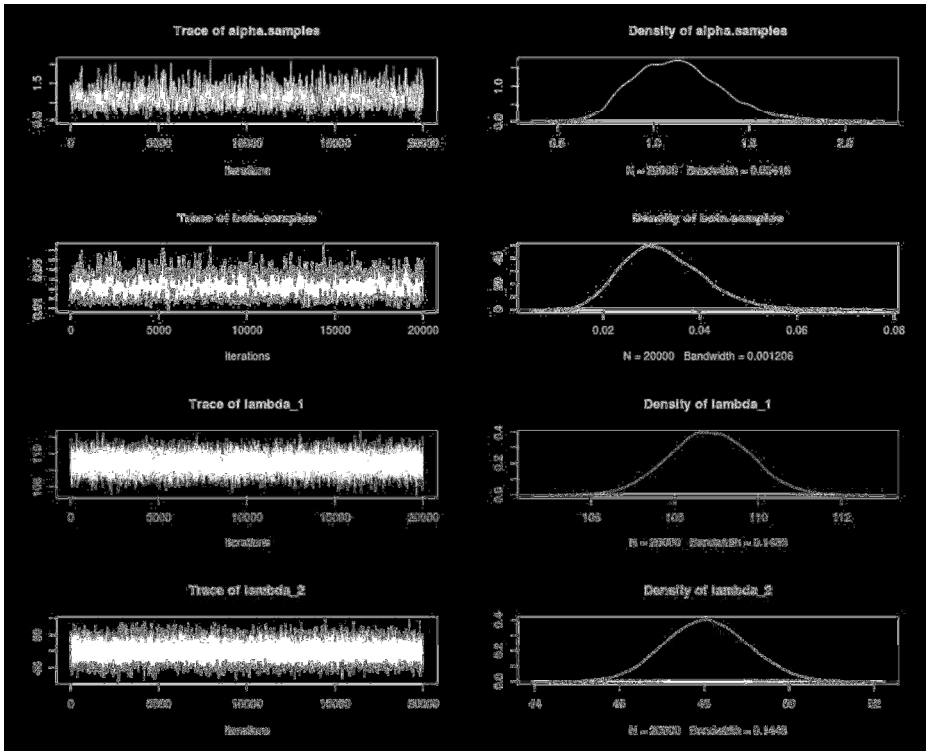


The graphs above outline the distribution of the lambda densities for both the posterior lambdas we predicted as well as the lambdas gotten by calculated the means of each of the galaxies from the data. We can see from the graphs above that lambdas are definitely skewed very much to the right. There are a large number of galaxies that are not particularly bright, and galaxies corresponding to a photon count above 50 exist but the number is very minimal. Next we take a look at the posterior minimum and maximum values of lambda to see if they tell a different story.



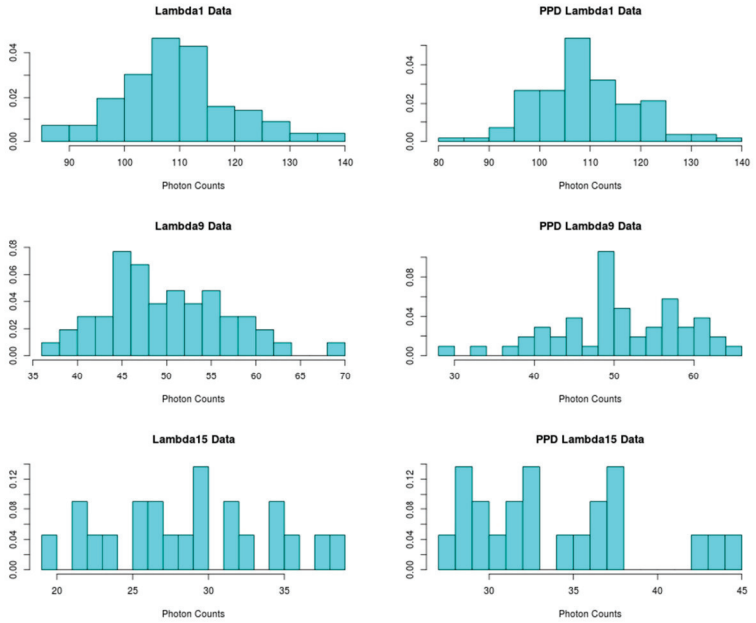
Looking at the two graphs above we see they correspond very closely to the average posterior lambda density. This helps confirm the conclusions that we came to in the previous paragraph. It is very likely that a galaxy has a photon count, or brightness, below 50. The number of galaxies with a brightness above 50 drop dramatically. This means that the brightness of galaxies overall tend to be on the lower side of the spectrum.

We then looked into posterior predictive checking in order to determine whether the model assumptions are reasonable. We first look at the trace plots generated by the data:



Looking at the above trace plots for the first few parameters we see that they converge, and thus look fairly good. Additional trace plots can be found in the appendix.

Next we will look at the posterior predictive data plots as another check:



We picked three lambdas at random to show here. The column on the right corresponds to the predicted values while the column on the left corresponds to the actual data distribution. Additional posterior predictive data sets are available in the appendix. We can see that visually the posterior predictive data sets match pretty well with the distribution of the actual data.

Lambda	PPD max	Data max	P-value	PPD min	Data min	P-value	PPD Median	Data Median	P-value
1	135	136	0.53	81	85	0.65	107	108.5	0.53
9	65	68	0.76	29	36	0.77	49.5	48.5	0.25
15	44	38	0.01	27	19	0.02	32	28.5	0.00
18	33.3	34	0	14.3	14	0	22	22	0
35	16	8	0.00	3	4	0.04	12	6	0.00

Lambda	PPD Mean	Data Mean	p-value	PPD Sd	Data sd	P- value
1	108.42	108.85	0.51	9.78	10.34	0.47
9	50.03	49.40	0.49	7.87	6.811	0.37
15	33.54	28.27	0.00	5.02	5.36	0.3
18	22	22.1	0	4	4.41	0
35	11.2	6	0.00	5.167	1	0.05

Looking at the above tables we can see that although the p-values are large for some variables overall the values gotten from the posterior predictive distribution are very similar to the values ascertained from the data. This is positive news, and shows us that the posterior predictive datasets are relatively accurate.

We then conducted "sensitivity" analysis to see if changing the value of the priors would have an effect on changing the posterior samples. We started off by using 1 as the value for all prior parameters. We then tested the setting the values of the priors to 3, .01, and 10 to see if they had any effect on the parameters.

Priors	Alpha mean	Beta mean	Lambda1 mean	Lambda2mean
1	1.13	.03	108.9	48.05
3	1.20	.03	108.8	48.05
.01	1.08	.03	108.8	48.06
10	1.34	.04	108.8	48.06

As you can see from the table above changing the prior values does not really have an effect on all the parameters. In essence, the values of the priors have very little influence on the overall results. Because the data set is fairly large, it ends up dominating and we gleam most of our information from the dataset itself.

Lessons Learned

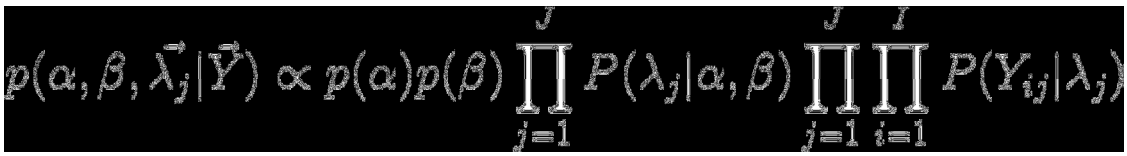
There were many lessons we learned during the project. The first among these was that you should always seek to find the closed form distributions of whatever variable it is that you are attempting to sample. This will not only make finding the samples easier to code, but will also save on computational energy. Also that although Gibbs sampling tends to be a little slow, it was relatively easy to implement and after disregarding the initial dependent samples gave us very good results.

We also learned that although the inverse – CDF method of sampling is relatively computationally inexpensive, it is fairly difficult to solve analytically. Although the Metropolis within Gibbs method is computationally expensive, it is a much easier way of tackling the same problem. We were able to use Metropolis within Gibbs in this case easily because we were only using it to sample one particular variable.

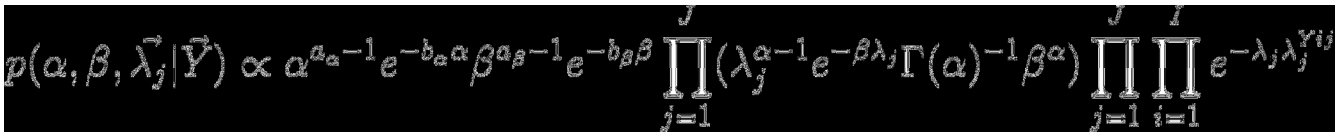
Furthermore, we saw that although the posterior predictive data sets may visually match up with the actual data set, the p-values corresponding to our posterior predictive datasets can tend to be large. We learned that this is relatively normal, as the p-value is used of more as a guideline of results rather than an end all be all indicator. For instance, the p-value can be influenced greatly by small variations in galaxies with relatively small numbers of observations. Finally, we discovered that our data was not very sensitive at all to the prior parameters. We could modify them to largely different values and the results ended up being extremely similar.

APPENDIX 2

Joint Derivation



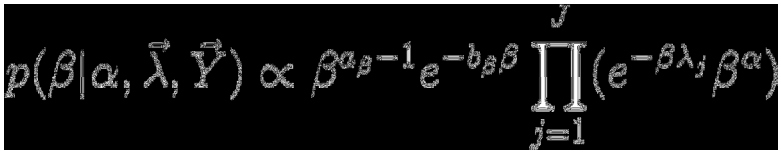
$$p(\alpha,\beta,\vec{\lambda_j}|\vec{Y}) \propto p(\alpha)p(\beta)\prod_{j=1}^{J} P(\lambda_j|\alpha,\beta)\prod_{j=1}^{J}\prod_{i=1}^{I} P(Y_{ij}|\lambda_j)$$



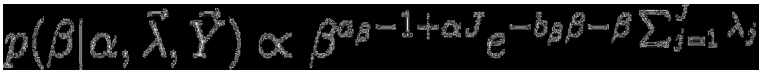
$$p(\alpha,\beta,\vec{\lambda_j}|\vec{Y}) \propto \alpha^{a_\alpha-1}e^{-b_\alpha\alpha}\beta^{a_\beta-1}e^{-b_\beta\beta}\prod_{j=1}^{J}(\lambda_j^{\alpha-1}e^{-\beta\lambda_j}\Gamma(\alpha)^{-1}\beta^{\alpha})\prod_{j=1}^{J}\prod_{i=1}^{I} e^{-\lambda_j}\lambda_j^{Y_{ij}}$$

Conditional Derivations

Beta



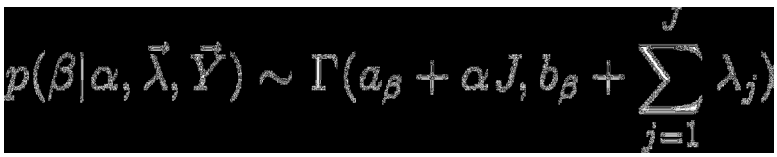
$$p(\beta|\alpha,\vec{\lambda},\vec{Y}) \propto \beta^{a_\beta-1}e^{-b_\beta\beta}\prod_{j=1}^{J}(e^{-\beta\lambda_j}\beta^{\alpha})$$



$$p(\beta|\alpha,\vec{\lambda},\vec{Y}) \propto \beta^{a_\beta-1+\alpha J}e^{-b_\beta\beta-\beta\sum_{j=1}^{J}\lambda_j}$$

$$p(\beta|\alpha,\vec{\lambda},\vec{Y}) \propto \beta^{(a_\beta+\alpha J)-1}e^{-(-b_\beta+\sum_{j=1}^{J}\lambda_j)\beta}$$

Note: Please ignore the negative sign in front of b_{\beta}.

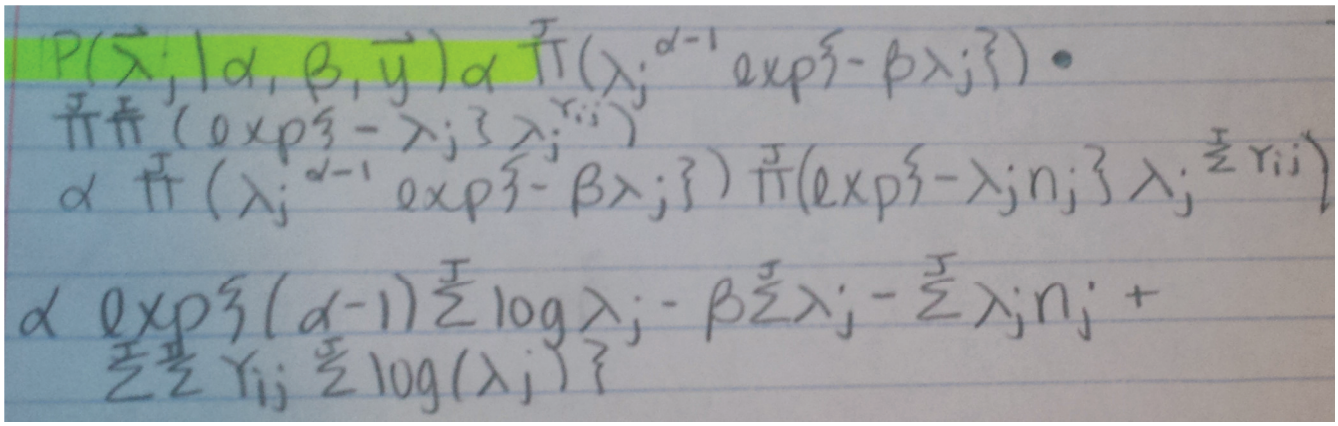


$$p(\beta|\alpha,\vec{\lambda},\vec{Y}) \sim \Gamma(a_\beta+\alpha J, b_\beta+\sum_{j=1}^{J}\lambda_j)$$

Alpha

$$p(\alpha|\beta,\vec{\lambda},\vec{Y}) \propto \alpha^{a_\alpha-1}e^{-b_\alpha\alpha}\prod_{1}^{j}(\lambda_j^{\alpha-1}\Gamma(\alpha)^{-1}\beta^{\alpha}$$

$$\propto \frac{1}{\Gamma(\alpha)^J}\alpha^{a_\alpha-1}e^{-\alpha b_\alpha-\sum log(\lambda_j(\alpha-1)-Jlog(\beta)\alpha)}$$



Lambda