ISYE 6420

Bayesian Statistics

Michael Guo

April 21, 2022

Determining methods for using the mcp package to pinpoint change points in quality data

Background

At a biotech company, change points in our quality data are currently being identified manually. Manually determining change points is not ideal, because it is a highly variable process. When presented with the same data, individual reviewers are very likely to identify different change points, especially when the data are noisy like in quality data. Additionally, accurately identifying change points manually is difficult in noisy non-trivial situations like quality control.

In order to address these issues at the biotech company, I am investigating ways to use the mcp change point detection package in R to pinpoint the exact change points (https://lindeloev.github.io/mcp/). The mcp package allows users to do regression with one or multiple change points between generalized and hierarchical linear segments using Bayesian inference. The methods investigated will be evaluated on the quality_control_1, quality_control_2, quality_control_3, and quality_control_4 datasets from the Turing Change Point Dataset created by the Alan Turing Institute (https://github.com/alan-turing-institute/TCPD).

Datasets

Four datasets of example univariate quality control data were examined. The datasets were annotated by five human annotators to provide ground truth on the presence and location of change points. The 1st, 2nd, and 3rd datasets have single strongly defined change points at indices = 146, 97, and 179, respectively. The quality_control_4 dataset (**Figure 1**) has one strongly defined change point at 341 and the 3 relatively weakly defined change points at 157, 242, and 467. I couldn’t find the exact indices of the weakly defined change points, so I had to set those change points based on their approximate locations in the quality_control_4 graph in the van den Burg and Williams paper (https://arxiv.org/abs/2003.06222v3).

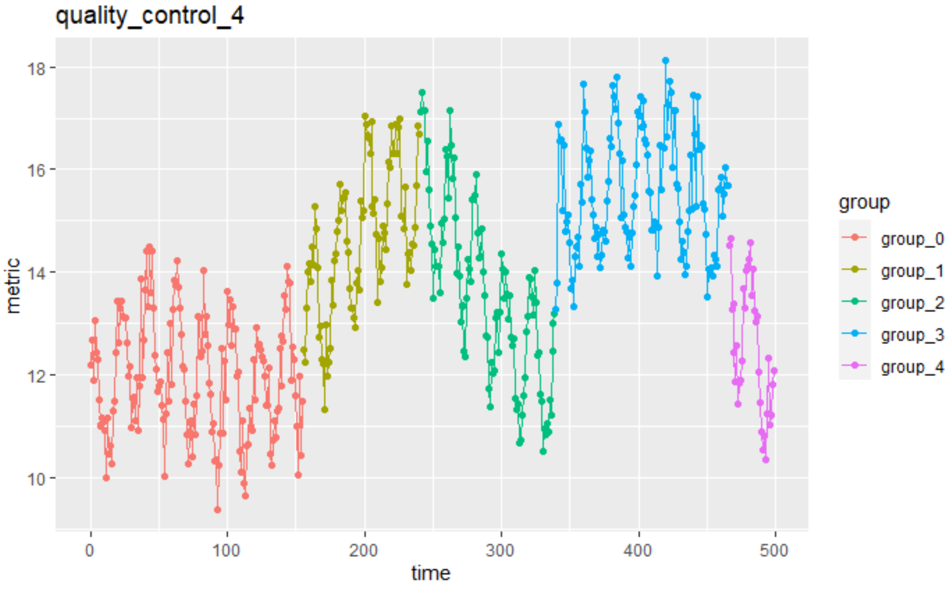

Figure 1: An example of True Groups

Methodology

Selecting likelihoods

When modeling the regressions for each of the segments, I used the gaussian family in mcp. This means that we model the likelihood of the quality data as $y_i \sim N(y_{fitted}, \frac{1}{\sigma^2})$ and assume an iid σ^2 across all of the segments. This is equivalent to fitting a normal linear regression with no link function for each section. Since we're assuming a linear relationship between index and the quality metric and a constant variance, we run the risk of getting linear segments that are not very well fit to the data, but that shouldn't be an issue as long as the models there's enough of a difference between the segments to accurately determine change points. These assumptions should also reduce the computational complexity of the models that will be created.

Selecting priors

We use the following prior distributions for the parameters of the linear regression segments:

- $Intercept \sim Unif(\min(y), \max(y))$
- $Slope \sim TDist(0, standard\ deviation(y)/(\max(x) - \min(y), df = 3)$
- $\sigma = dnorm\big(0, standard\ deviation(y)\big), where\ \sigma\ is\ non - negative$

I initially tried to use non-informative parameter distributions for the slopes and σ without respect to the data, but the misclass rates for the Dirichlet distribution and default student's t distribution were much lower when we used slightly more informative priors informed by the spread of y and the indices.

For the single change point models, we use the uninformative mcp default change point prior: $change\ point \sim Unif(\min(x), \max(x))$

Investigating methods for detecting multiple change points (quality_control_4 dataset).

For the quality_control_4 dataset, the Default t-Distribution Priors that mcp uses by default for models with multiple change points yielded significantly inaccurate results. Because of this, I also investigated using Dirichlet Priors, and Informative Uniform Priors as the change point priors. Dirichlet distribution priors are recommended in the mcp documentation as a more non-information distribution compared to the default t-tailed priors. The Informative Uniform priors were 41 index units long and are centered on the true change points.

I also tested the possibility of dividing the dataset up into subsets of data that have a single suspected change point and determining the change point of each subset individually. To create these subsets, I divided the data so the subsets contain the following indices: [1:241], [157,340], [242,466], [341,500]. The edges of the subsets to identify the cp_k (for the kth subset) are:

$[1, cp_{k+1}], when\ k = 1$

$[cp_{k-1}, cp_{k+1}], when\ k = 2,3$

$[cp_{k-1}, 500], when\ k = 4$

Evaluating method performance

For each method, the error in determining the change points is measured via misclassification rate. The true value of a datapoint is the “true” group it was assigned by the annotators, and the test value of a datapoint is the group it was assigned by the method in question. The time taken to generate each of the models via mcp package was also measured. The method(s) chosen for each dataset were run 5 times for reproducibility given the random nature of Gibbs Sampling.

Results

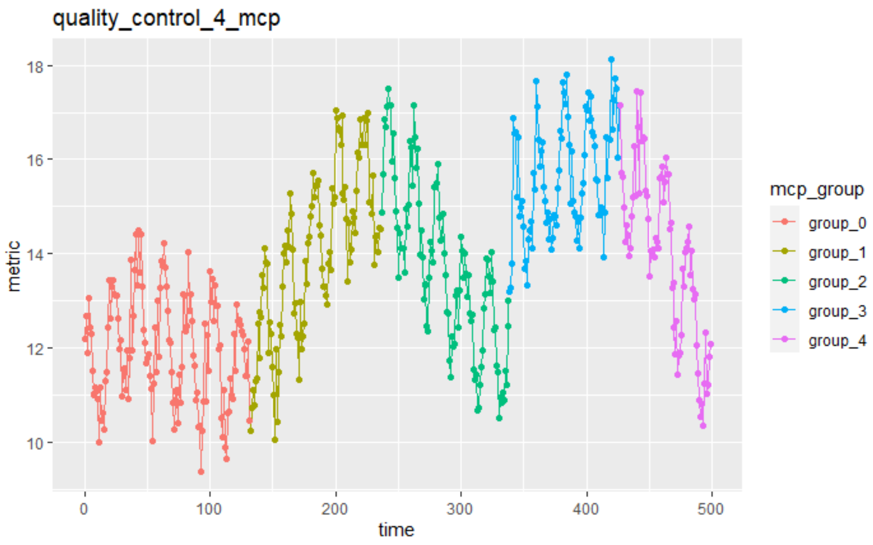

Figure 2: An Example of Assigned Groups

The resultant misclassification rates from testing on the datasets can be seen in **Table 1**. The low misclassification rates for the single change point datasets indicate that we can confidently use the mcp package to identify change points when there is only one change point in the data. The dataset methods used with the multi change point dataset had much worse performance. The performance of the methods listed from best performing to worst performing are: Informative Uniform Priors, Subsetting Method, Dirichlet Priors, and Default t-Distribution Priors. We should note that the best performing methods (Subsetting and using Informative Uniform Prior) do rely on some knowledge of where the change point is. This is especially true with the Informative Uniform Prior method that requires the user to identify a range that the change point is actually in. With the subsetting method, the user needs to be less precise with the subsets presented.

Iteration	**QC1**	**QC2**	**QC3**	**QC4_Dirichlet**	**QC4_Default**	**QC4_Informed**	**QC4_Subset**
1	0.0064	0.0035	0.057	0.3920	0.7320	0.0120	0.178
2	0.0064	0.0035	0.057	0.3580	0.7880	0.0180	0.162
3	0.0064	0.0035	0.055	0.3120	0.6740	0.0080	0.1700
4	0.0064	0.0035	0.055	0.3100	0.6380	0.0200	0.1600
5	0.0064	0.0035	0.055	0.2860	0.7240	0.0040	0.1360
Mean	**0.0064**	**0.0035**	**0.056**	**0.3316**	**0.7712**	**0.0124**	**0.1612**
STDev	**0.0000**	**0.0000**	**0.0015**	**0.0426**	**0.0575**	**0.0067**	**0.0158**

Table 1: Misclassification Rates

Looking at the execution times of the methods (**Table 2**), it is clear that the mcp package finishes creating its models with only a single change point much more quickly than when creating models with multiple change points. Even when 4 separate models are created in the Subsetting Method, the combined execution time (49.01 seconds) is still much lower than the times required to create single models with multiple change points (73.14, 100.83, and 108.11 seconds)

Iteration	QC1	QC2	QC3	QC4_Dirichlet	QC4_Default	QC4_Informed	QC4_Subset
1	18.92	17.30	21.50	70.88	103.49	107.07	49.11
2	18.92	16.99	22.19	70.17	98.76	106.29	49.02
3	22.38	17.27	21.82	71.77	94.68	103.93	49.05
4	18.76	17.01	21.95	72.26	95.91	106.25	48.95
5	19.07	17.21	24.99	73.14	100.83	108.11	48.93
Mean	**19.61**	**17.15**	**22.49**	**71.64**	**98.74**	**106.33**	**49.01**
STDev	**1.553**	**0.147**	**1.421**	**1.162**	**3.586**	**1.539**	**0.074**

Table 2: Execution Times

Conclusion

Based on the experiments using mcp to predict change points in our quality control datasets, I believe we can confidently use the package to pinpoint single change points in our Biotechnology company's quality control data without modifying the normal usage of the mcp package.

When it comes to data with multiple suspected change points, the mcp package performed much worse without some sort of user specified input. Subsetting the data (misclass rate: 0.1612) and specifying informative priors (misclass rate: 0.0124) did improve accuracy a great deal, but require the user to divide the dataset into subsets containing only one change point or identify credible intervals for each change point. Using the Subsetting and Informative Uniform Prior methods to identify multiple change points could be very useful going forward, but there should be more studies on their performance since only one dataset was examined in this study.

It should be noted that all of the single change point datasets had strongly defined change points and the one multi change point dataset had multiple more weakly defined change points. This may help explain why the performance on single change point data performed so much better than the multi change point data, and should be explored further in other studies.