Soundness of VI compared to MCMC for modelling gene expression

Theodoros Bitsakis, Tushar Goel, Siddhartha Jain, Wouter Saelens
École Polytechnique Fédérale de Lausanne, Switzerland

I. Introduction

All multicellular organisms are composed of cells. These cells can further be described using their *gene expressions*. Depending on the function of the cell, gene activity can differ significantly. Genes that are more active have more "copies" present in the cell.

Computational biologists often study the gene expression changes depending on the type of cell. The best approach for this is to use Bayesian modelling, but Markov Chain Monte Carlo (MCMC) can be prohibitively slow. In this project we study what the quality of the approximation provided by Variational Inference (VI) for MCMC.

II. Design

A. Model Creation

Before writing any code, the first step was to the read the relevant biology. It started with understanding the central dogma of molecular biology, stated popularly as "DNA makes RNA, and RNA makes protein." In the context of the project, we then read about single-cell transcriptomics - a technology which examines the gene expression level of individual cells. The dataset which we work with in this study represents the number of mRNAs inside a cell for each gene, which reflects how "active" those genes are for that given cell.

The objective of the project is to conclude the best way to model the effect of *perturbations* to the cells. For example, a perturbation can correspond to adding some chemicals or adding an extra gene to the cell. We implement parameterised models for the effect of the perturbation, and trained them using either MCMC or VI. Let D be the deviation in the gene expression, observed on application of perturbation p (normalised to be in $[0, 1]$). We used three models:

- *Nothing*: As the name suggests, this model assumes perturbations do not affect the count of the corresponding gene in the cell. This model then fits the best constant to the data.
- *Linear*: This model assumes there is a linear relation between perturbation and mRNA count. We then give the slope a prior.

  $$D = 1 + \beta p$$

  Where β is given the Negative Binomial distribution as a prior.
- *Switch*: This model would ideally assume there is a threshold after which the mRNA count shoots up, and the value does not change at all on either side of this threshold. However, this behaviour is not smooth and we needed to model it using a differentiable relaxation. So we used the *Sigmoid* function instead. We use γ to denote the switch threshold, and δ is a skew value to achieve a steep slope.

  $$D = 1 + \frac{\beta}{1 + exp(-\delta(p - \gamma))}$$

  We set β to have the same prior as the Linear model, δ to be a hyperparameter with value 50, and we give γ the prior of the Uniform distribution on $[0, 1]$.

To implement the models, we used a combination of `jax` and `numpyro`. These libraries are the best available combination for fast, accurate probabilistic programming. `jax` also has the advantage of implementing Autograd, which meant fast gradients without needing to implement them for each individual model.

B. Agenda

After we created the models, we had to formalise exactly what we wanted to test. We essentially want to know if MCMC and VI return similar posteriors,

1) What does the "exact" posterior look like?
2) Does VI produce a posterior similar to that of MCMC?

Further, the following questions have to be answered for each of the aforementioned questions.

1) Is this variable dependent?
2) Is this model dependent?
3) Is this gene dependent?

III. Methodology

A. Data Generation

For conducting this study, we generated a synthetic dataset of 400 cells with 30 genes per cell. Real world data would be too large and the time taken by MCMC would inhibit a careful study of the posterior distributions.

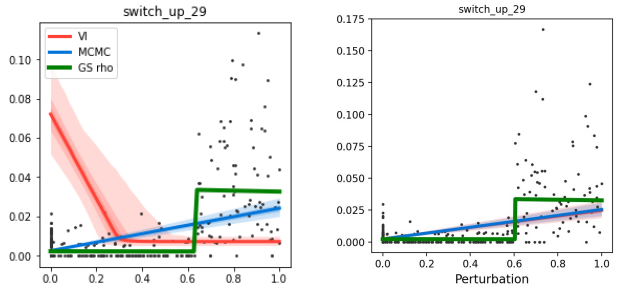
The 30 genes we generate have the following composition:

- 10 genes follow the Nothing model, in which genes which stay the same in all cells.
- 10 genes follow the Linear model (out of which 5 have positive slope and 5 have negative slope).
- 10 genes follow the Switch model (out of which 5 switch up and 5 switch down).

We standardised the dataset by saving a random sample using `pickle`.

B. Training

We observed multiple issues while training MCMC and VI, which significantly affect the quality of the posterior distributions.



(a) Bad initialisation (b) Good initialisatiom

Figure 1: Behaviour of VI depends heavily on initialisation

Initialisation played a crucial role in the quality of the solution of VI. We were initially using the `init_to_feasible` method, which initialises to an arbitrary feasible point, ignoring the distribution parameters. We observed that this resulted in VI getting stuck in a local minima very often, as illustrated in Fig 1a. This outcome was avoided by instead using the `init_to_median` method which samples 15 points from the distribution of the prior and initialises on their median.

MCMC also converged to local minima, for eg for the `switch_up_30` gene as shown visually in Fig 2. To overcome this we would use more warmup steps in the future.

C. Visualisation

We consider three visualisations of the results of the training to make inferences. One option is to directly visualise the posterior distributions returned by both algorithms.

To go beyond simply visualising the posterior distributions returned by MCMC and VI, we plot some statistics of the parameters returned by both MCMC (on the X axis) and VI (on the Y axis) and draw the identity line to see if VI is clearly underestimating or overestimating any parameter.

Finally, we can also visualise the marginals of each parameter in the distributions.

IV. Results

A. Posterior distributions

We see cases where VI underestimates the variance of the posterior distribution significantly. An exmaple of this is shown in Fig 3.

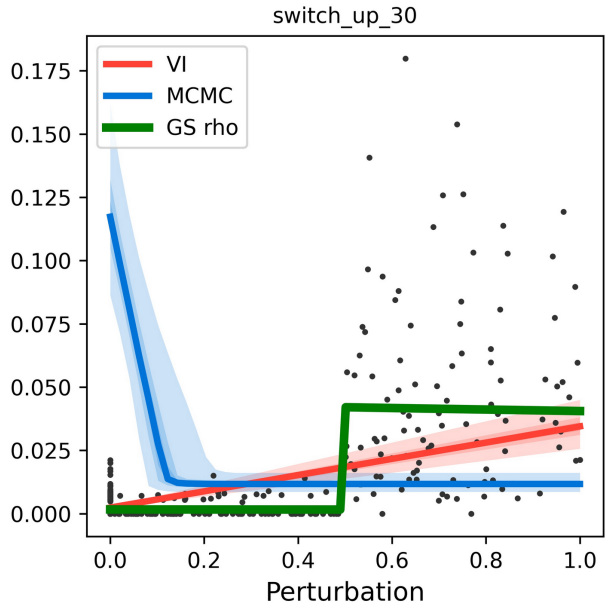


Figure 2: A scatterplot of the gene expression values, with the 'Gold Standard' fit (GS rho) and the predictions returned by MCMC and VI using a Linear model. An example where MCMC converges to a local minima.

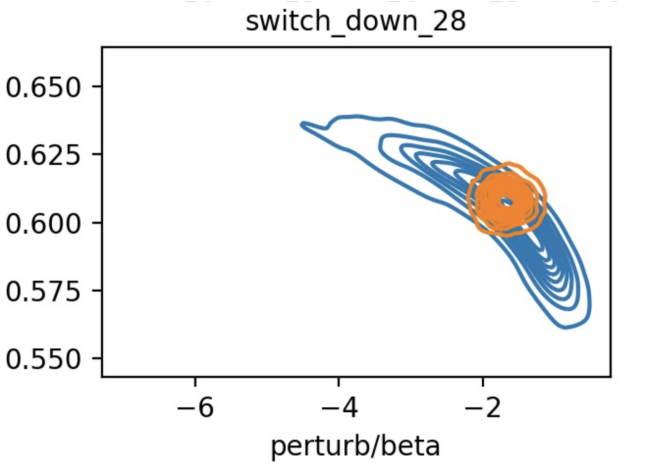


Figure 3: The "exact" posteriors returned by MCMC and VI for a particular gene using the Switch model

B. Parameter statistics

Plotting the mean of the parameters learned by MCMC and VI (see Fig 4), we make the following observations:

- For the Switch model it underestimates in one case, but is quite comparable otherwise. This event can be considered an outlier.
- For the Linear model, we see that it overestimates and underestimates an even number of times, indicating again that there is no bias in VI.

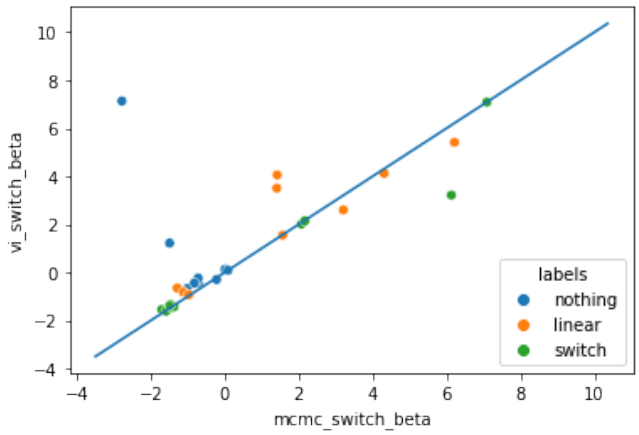


Figure 4: Mean of the β parameter returned by MCMC and VI

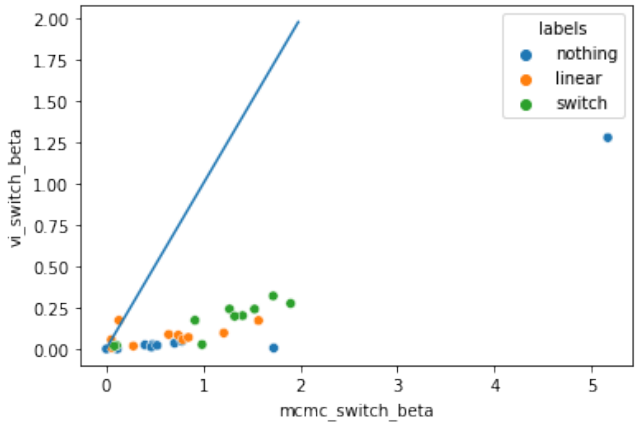


Figure 5: Variance of the β parameter returned by MCMC and VI

- For the Nothing model we see a trend of overestimation.

Moreover, we see that across the board VI heavily underestimates the variance. This can be seen in Fig 5.

C. Marginal distributions

We see that VI (orange •) estimates the parameters of the models very well, as the results are very close to those of MCMC (blue •). For eg, in Fig 6 we see a comparison of the β distributions for the Linear model.

V. Conclusion

We see that the posterior distribution returned by VI serves as a reasonable approximation to one returned by MCMC. Considering the large difference in time complexity of the two methods, VI is the more practical option.

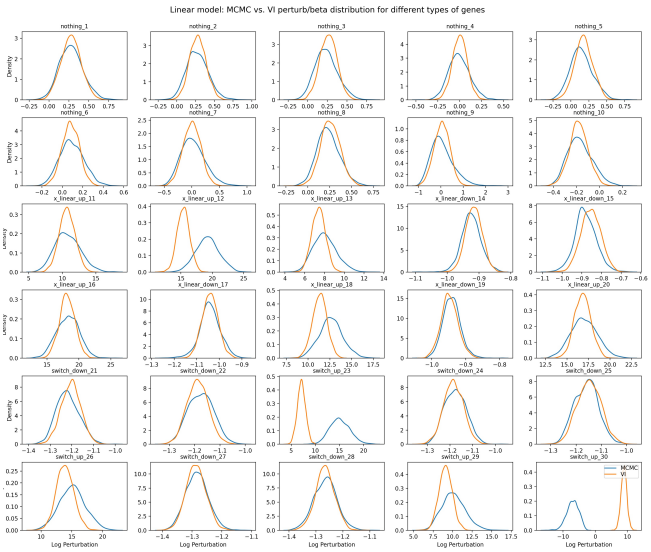


Figure 6

A. Future work

A natural extension of this study would be to answer the questions we consider for other models as well, for eg an Exponential or a Spline model.

References