ZIPBN: A Novel Contribution to Causal Structural Learning

Reading Group 02/23/2021

Roman Kouznetsov

University of Michigan Department of Statistics

UNIVERSITY OF MICHIGAN

Table of Contents

1. Introduction/Prereqs

2. ZIPBN Model

3. Experiments w/ ZIPBN

4. Conclusion

1. Introduction/Prereqs

2. ZIPBN Model

3. Experiments w/ ZIPBN

4. Conclusion

Zero-Inflated Data

Zero-inflated data is data with a tremendously large amount of 0 values.

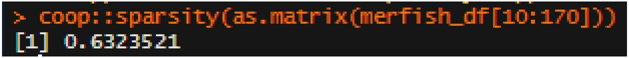
These include:

- Genetic Expression Data
- Insurance (especially w/ young people)
- Defect Counting (in small batches)
- Weekly School Attendance Absences of Students

In general, we don’t want to apply standard count models to events that are overwhelmingly likely to not occur (occur with value 0)

Close to Home

Zero-inflated data is data with a tremendously large amount of 0 values.



```
> coop::sparsity(as.matrix(merfish_df[10:170]))
[1] 0.6323521
```

The spatial transcriptomics data set I talked about last week has a lot of zeros in it!!

Motivation

"This paper is motivated by causal structural learning for zero-inflated count data which arise in a wide range of areas..."

ZIPBN is a bayesian network that makes conclusions about causal inference on *zero-inflated* count data.

Directed Acyclic Graphs (DAGs)

DAG: $\mathcal{G}$=(V,E)

- V: nodes (each usually representing a variable in **X**
- E: edges
  - $e_{jk} = 1$ if $k \to j$
  - $e_{jk} = 1$ implies that X_k causes X_j: an edge represents a causal relationship
  - Once the graph leaves k, it never returns (hence the name acyclic).

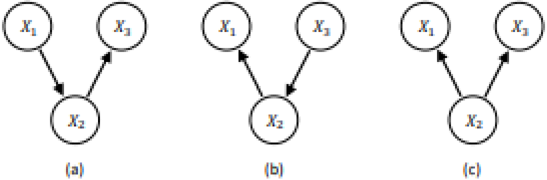
Bayesian Networks (BNs)

A BN consists of:

- a DAG ($\mathcal{G}$)
- DAG parameters (θ)
  - DAG parameters help determine how X_k causes the value of X_j to occur.
  - In ZIPBN, these parameters help determine whether X_k contributes to the zero-inflation mass estimator η_j or the mean estimator λ_j

Markov Equivalency Class (MEC)

MEC: a set of DAGs that encode the same set of conditional independencies
The example below showcases 3 different DAGs with $X_1 \perp X_3 | X_2$



The main takeaway is that DAGs in the same MEC are frequently indistinguishable. This is problematic because in graph (a), X_1 contributes to the value of X_2 that generates X_3, but does not in graph (c).

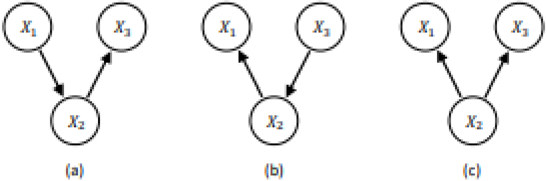
BN Identifiability

Def: *distribution equivalent*: 2 BNs $\mathcal{B}_1 = (\mathcal{G}_1, \theta_1)$ and $\mathcal{B}_2 = (\mathcal{G}_2, \theta_2)$ are distributionally equivalent if there exists θ_2 that represents the same distribution.

Def: *identifiable*: A BN is identifiable if its directional graph is recoverable (not just the MEC).

Result: If a BN is identifiable, then another DAG with the same skeleton (same as DAG in the same MEC), encodes a different distribution.

Ex: These DAGs would not be distributionally equivalent in ZIPBNs.



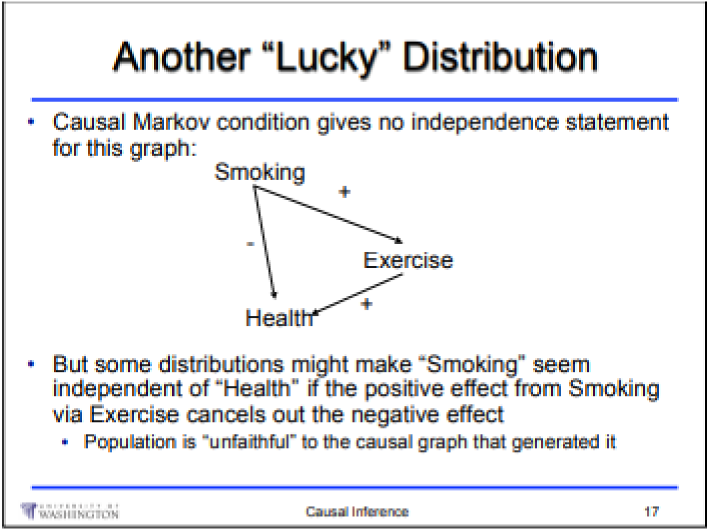
Takeaway: With ZIPBNs you are recovering causes, not just a conditional independence structure.

Causal Inference Definitions

- causal sufficiency: the assumption that all relevant variables have been observed
- causal faithfulness: the independence relations in the causal graph are the only true ones

Takeaway: Sufficiency says we have allowed the potential causes to exist in our model. Faithfulness assumes that a graph displays all the conditional independence relationships that exist.

Relevance of Faithfulness



Another "Lucky" Distribution

- Causal Markov condition gives no independence statement for this graph:

Smoking
+
-
Exercise
Health
+

- But some distributions might make "Smoking" seem independent of "Health" if the positive effect from Smoking via Exercise cancels out the negative effect
  - Population is "unfaithful" to the causal graph that generated it

Source: Michele Banko & Kevin Duh (University of Washington)

Novelty of ZIPBN

- ZIPBN are identifiable WITHOUT assuming causal faithfulness.
- Previous work establishes identifiability with distributional assumptions that frequently relies on causal faithfulness.

1 Introduction/Prereqs

2 ZIPBN Model

3 Experiments w/ ZIPBN

4 Conclusion

Sampling Model

$$p(\mathbf{X}) = \prod_{j=1}^{p} p(X_j | X_{pa(j)})$$

$$P(X_j = x | X_{pa(j)}) = \begin{cases} \eta_j + (1-\eta_j) exp(-\lambda_j) & x = 0 \\ (1-\eta_j)\frac{exp(-\lambda_j)}{x!} & x > 0 \end{cases}$$

Link Functions:

- $log(\frac{\eta_j}{1-\eta_j}) = \sum_{k \in pa(j)} \alpha_{jk} X_k + \delta_j$
- $log(\lambda_j) = \sum_{k \in pa(j)} \beta_{jk} X_k + \gamma_j$

Note that

- α_{jk} represent parameter estimates of how X_k impacts the zero-inflation mass estimate of X_j
- βjk represent parameter estimates of how X_k impacts the mean count estimate of X_j

Prior Model (Adjacency Matrix $\mathbf{E}$)

$$P(\mathbf{E}|\rho) = z(\rho)^{-1} \prod_{j \neq k} \rho^{e_{jk}} (1-\rho)^{1-e_{jk}} \mathbb{I}(\mathcal{G} \in \mathcal{D})$$

$$P(\rho) \propto z(\rho) \rho^{a_\rho - 1} (1-\rho)^{b_\rho - 1}$$

$$P(\mathbf{E}) = \int_0^1 p(\mathbf{E}|\rho) * P(\rho) d\rho$$

where

- $\underline{\rho}$: edge-inclusion probability for a given edge e_{jk}
- $\underline{\mathbb{I}(\mathcal{G} \in \mathcal{D})}$: an indicator function that ensures that the graph $\mathcal{G}$ created by $\mathbf{E}$ is actually a DAG

Prior Model (Graph Parameters θ)

Spike-and-Slab Distribution:

- *spike*: discrete point mass (at 0)
- *slab*: the standard continuous prior
- A fancy name for a mixture distribution that has one discrete and one continuous component

$$(\alpha_{jk}, \beta_{jk})|e_{jk}, \tau_1, \tau_2 \sim e_{jk}N_2(\mathbf{0}, \mathbf{P}^{-1}) + (1 - e_{jk})\delta_{\mathbf{0}}$$

Interpretation: If a causal relationship exists in the graph ($e_{jk} = 1$), then the contribution to the zero-inflation mass η_j and estimated Poisson mean λ_j value are normally distributed and independent. Otherwise, no edge exists at all, so $(\alpha_{jk}, \beta_{jk}) = (0, 0)$.

Posterior Distribution

$$P(\mathbf{E}, \theta, \tau, \rho | \mathbf{X}) \propto P(\mathbf{X}|\mathbf{E}, \theta) P(\theta|\mathbf{E}, \tau) P(\mathbf{E}|\rho) P(\tau) P(\rho)$$

- Term 1: Likelihood
- Terms 2-5: Prior (taking into account that τ and ρ are independent and conditionally independent.

Takeaway: With the posterior, we can now have a metric of uncertaintly about the existence of a causal relationship: $P(e_{jk} = 1|\mathbf{X})$

Parallel-Tempered MCMC

- We take samples from the posterior using MCMC.
- Standard MCMC using Gibbs samplers get trapped in local modes when collecting samples.
- This is problematic when our discrete distribution is multi-modal.
- **Enter Parallel-Tempered MCMC**

Fractional Flattening

An important concept of P-T MCMC is that raising distributions to the power of $\frac{1}{T}$, $T > 1$ flattens the distribution *allowing for samples outside of the mode more frequently*.

https://www.desmos.com/calculator/yrzyr1mxv2

Usefulness of Tempering

In the swapping step,

$$R_s = \frac{\pi(\boldsymbol{E}_\ell, \boldsymbol{\theta}_\ell, \boldsymbol{\psi}_\ell | \boldsymbol{X})^{1/T_m} \pi(\boldsymbol{E}_m, \boldsymbol{\theta}_m, \boldsymbol{\psi}_m | \boldsymbol{X})^{1/T_\ell}}{\pi(\boldsymbol{E}_\ell, \boldsymbol{\theta}_\ell, \boldsymbol{\psi}_\ell | \boldsymbol{X})^{1/T_\ell} \pi(\boldsymbol{E}_m, \boldsymbol{\theta}_m, \boldsymbol{\psi}_m | \boldsymbol{X})^{1/T_m}},$$

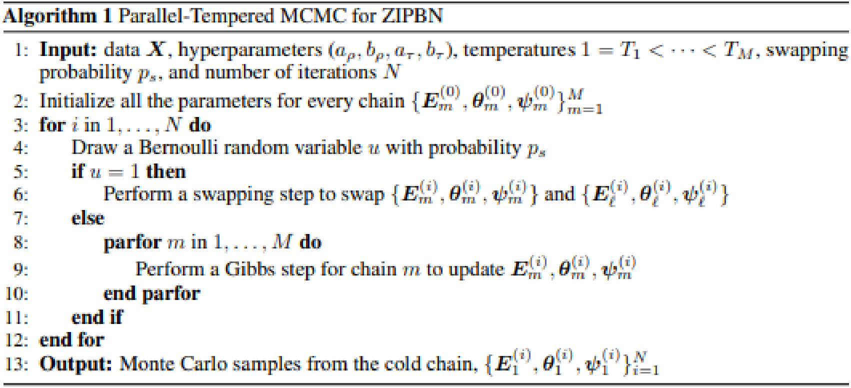
🟨 - t

🟩 - p

the ratio of proposals approaches 1 as $m \to \infty$. This means that the acceptance probability of a sample in HOTTER chain is still somewhat high.

Goal: Create a set of samples in E_1 that take some samples from regions of lower density by accepting values in $E_m, > 1$.

Algorithm



Algorithm 1 Parallel-Tempered MCMC for ZIPBN

```
Input: data $\boldsymbol{X}$, hyperparameters $(a_\rho, b_\rho, a_\tau, b_\tau)$, temperatures $1 = T_1 < \cdots < T_M$, swapping probability $p_s$, and number of iterations $N$
Initialize all the parameters for every chain $\{\boldsymbol{E}_m^{(0)}, \boldsymbol{\theta}_m^{(0)}, \boldsymbol{\psi}_m^{(0)}\}_{m=1}^{M}$
for $i$ in $1, \ldots, N$ do
    Draw a Bernoulli random variable $u$ with probability $p_s$
    if $u = 1$ then
        Perform a swapping step to swap $\{\boldsymbol{E}_m^{(i)}, \boldsymbol{\theta}_m^{(i)}, \boldsymbol{\psi}_m^{(i)}\}$ and $\{\boldsymbol{E}_\ell^{(i)}, \boldsymbol{\theta}_\ell^{(i)}, \boldsymbol{\psi}_\ell^{(i)}\}$
    else
        parfor $m$ in $1, \ldots, M$ do
            Perform a Gibbs step for chain $m$ to update $\boldsymbol{E}_m^{(i)}, \boldsymbol{\theta}_m^{(i)}, \boldsymbol{\psi}_m^{(i)}$
        end parfor
    end if
end for
Output: Monte Carlo samples from the cold chain, $\{\boldsymbol{E}_1^{(i)}, \boldsymbol{\theta}_1^{(i)}, \boldsymbol{\psi}_1^{(i)}\}_{i=1}^{N}$
```

Gibbs Updates

$\underline{\mathbf{E}, \theta | \psi}$: **E** and θ are updated via a M-H within Gibbs sampler where either an edge comes alive ($e_{jk} = 0 \rightarrow e_{jk} = 1$), dies ($e_{jk} = 1 \rightarrow e_{jk} = 0$), or reverses the direction of all edges in **E** ($e_{jk} = 1 \rightarrow e_{kj} = 1$)

$\underline{\theta | \mathbf{E}, \psi}$: $\alpha, \beta, \gamma, \delta$ updated via Gaussian random walk for all $j \neq k$ combinations

$\underline{\psi | \mathbf{E}, \theta}$: τ and ρ are updated by their full conditionals (dependent on $\mathbf{E}, \theta$, and hyperparameters

Summary of PT-MCMC

- Uses fractional powers of probability distributions to generate samples outside mode neighborhoods.
- Gathers samples from a wider range of the network space.
- Works best with T_m values in between 1 and e.
- The swapping probability is controlled enough so that no drastic swaps occur (i.e. if $M = 50$ it's unlikely that a sample from that chain would be swapped with one from the posterior ($m = 1$)).
- Convergence time of $O(pmax(n, p))$ for sparse models. (DQ: Why?)

Obtaining Point Estimates

Before we look at some experiments, it is important to answer the question...

Q: How are point estimates taken for the adjacency matrix?

A: If N represents the number of samples collected by PT-MCMC, then

- $p_{jk} = p(e_{jk}|\mathbf{X}) \approx \frac{\sum_{i=1}^{N} e_{jk}}{N}$
- $\hat{e}_{jk} = \mathbb{I}(p_{jk} > c)$ (usually $c = 0.5$)
  - In fact, the hyperparameter of c is how the FDR gets controlled!!!
- $\hat{\mathbf{E}} = [\hat{e}_{jk}]$

1. Introduction/Prereqs
2. ZIPBN Model
3. Experiments w/ ZIPBN
4. Conclusion

Applied Experiments

- DAG Simulation
- Transcription Factors to Targets
- Gene Regulatory Network (Pathway Analysis)

Simulation

- A sparse DAG was simulated with p nodes and p edges.
- All of the DAG parameters were generated so that the resulting count observations were 0 with roughly 50% frequency.
- The goal of the experiment is to maximize the number of edges correctly found by the model while controlling the FDR.

Simulation Results

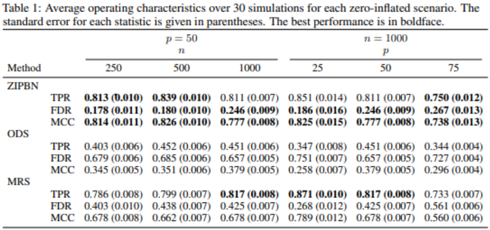


Table 1: Average operating characteristics over 30 simulations for each zero-inflated scenario. The standard error for each statistic is given in parentheses. The best performance is in boldface.

		$p = 50$			$n = 1000$		
			n			p	
Method		250	500	1000	25	50	75
ZIPBN							
	TPR	**0.813 (0.010)**	**0.839 (0.010)**	0.811 (0.007)	0.851 (0.014)	0.811 (0.007)	**0.750 (0.012)**
	FDR	**0.178 (0.011)**	**0.180 (0.010)**	**0.246 (0.009)**	**0.186 (0.016)**	**0.246 (0.009)**	**0.267 (0.013)**
	MCC	**0.814 (0.011)**	**0.826 (0.010)**	**0.777 (0.008)**	**0.825 (0.015)**	**0.777 (0.008)**	**0.738 (0.013)**
ODS							
	TPR	0.403 (0.006)	0.452 (0.006)	0.451 (0.006)	0.347 (0.008)	0.451 (0.006)	0.344 (0.004)
	FDR	0.679 (0.006)	0.685 (0.006)	0.657 (0.005)	0.751 (0.007)	0.657 (0.005)	0.727 (0.004)
	MCC	0.345 (0.005)	0.351 (0.006)	0.379 (0.005)	0.258 (0.007)	0.379 (0.005)	0.296 (0.004)
MRS							
	TPR	0.786 (0.008)	0.799 (0.007)	**0.817 (0.008)**	**0.871 (0.010)**	**0.817 (0.008)**	0.733 (0.007)
	FDR	0.403 (0.010)	0.438 (0.007)	0.425 (0.007)	0.268 (0.012)	0.425 (0.007)	0.561 (0.006)
	MCC	0.678 (0.008)	0.662 (0.007)	0.678 (0.007)	0.789 (0.012)	0.678 (0.007)	0.560 (0.006)

Figure: Performance of ZIPBN with $\approx$50% Zero Counts for Various (n, p) Paris

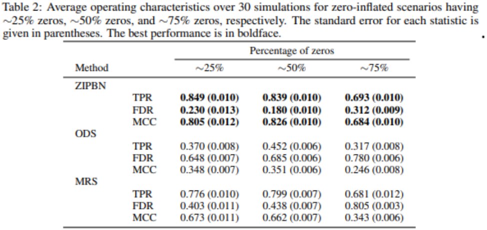


Table 2: Average operating characteristics over 30 simulations for zero-inflated scenarios having ~25% zeros, ~50% zeros, and ~75% zeros, respectively. The standard error for each statistic is given in parentheses. The best performance is in boldface.

		Percentage of zeros		
Method		~25%	~50%	~75%
ZIPBN				
	TPR	**0.849 (0.010)**	**0.839 (0.010)**	**0.693 (0.010)**
	FDR	**0.230 (0.013)**	**0.180 (0.010)**	**0.312 (0.009)**
	MCC	**0.805 (0.012)**	**0.826 (0.010)**	**0.684 (0.010)**
ODS				
	TPR	0.370 (0.008)	0.452 (0.006)	0.317 (0.008)
	FDR	0.648 (0.007)	0.685 (0.006)	0.780 (0.006)
	MCC	0.348 (0.007)	0.351 (0.006)	0.246 (0.008)
MRS				
	TPR	0.776 (0.010)	0.799 (0.007)	0.681 (0.012)
	FDR	0.403 (0.011)	0.438 (0.007)	0.805 (0.003)
	MCC	0.673 (0.011)	0.662 (0.007)	0.343 (0.006)

Figure: Performance of ZIPBN with $(n, p) = (500, 50)$ for Various 0 Count

Pathway Analysis

- This experiment took $p = 40$ Wnt genes and $n = 1025$ cells from one (AhR-knockout) mouse.
- Applying ZIPBN to this data revealed a gene regulatory network that was able to identify 3 hub genes - genes that contribute to many unrelated processes and, therefore, diseases.
- This showcases a VERY useful application of ZIPBN.

1. Introduction/Prereqs
2. ZIPBN Model
3. Experiments w/ ZIPBN
4. Conclusion

Conclusion

Major Takeaways:

1. ZIPBNs are bayesian networks specifically designed to perform well in problems of count data with high zero counts
2. ZIPBNs are identifiable and incredibly good at discovering causal relationships between its nodes.
3. ZIPBNs provide point and uncertainty estimates for a causal relationship between X_k and X_j.
4. ZIPBNs are highly parameterized AND hyperparameterized, suggesting the range of problems it can solve is incredibly wide.
5. ZIPBNs are incredibly successful in gene regulatory networks (a highly causal setting).