Low-Variance Gradient Estimates for the Plackett-Luce Distribution

Artyom Gadetsky* artygadetsky@yandex.ru

Kirill Struminsky* k.struminsky@gmail.com

Christopher Robinson cfr20@sussex.ac.uk

Novi Quadrianto n.quadrianto@sussex.ac.uk

Dmitry Vetrov vetrovd@yandex.ru

arXiv · GitHub

p(B|A)yesgroup.ru

Higher School of Economics, National Research University

University of Sussex

Motivation & Overview

- Permutations occur in multiple tasks:
  - Causal Inference
  - Information Retrieval
  - Combinatorial Optimization
- At the same time, models with discrete latent variables are hard to train
- **Our goal** is to design gradient estimators for models with latent permutations
- We extend the gradient estimators [1,2] to the Plackett-Luce distribution, a distribution over permutations

The Plackett-Luce Distribution (PL)

- Consider a vector of logits $\theta = (\theta_1, \ldots, \theta_d) \in \mathbb{R}^d$
- To a permutation $b = (b_1, \ldots, b_d) \in S_d$ PL with parameters θ assigns probability

$$p(b \mid \theta) = \prod_{i=1}^{d} \frac{\exp \theta_{b_i}}{\sum_{j=i}^{d} \exp \theta_{b_j}}$$

- This **is equivalent to sampling d times w/o replacement** from categorical distribution with logits θ

- Note: the mode of PL is the sorting of θ
  - the product of denominators is minimized when $\theta_{b_1} \geq \ldots \geq \theta_{b_d}$
  - the product of numerators does not depend on b

Gumbel top-k Trick for PL

- Gumbel top-k is a generalization of Gumbel max trick, which allows sampling w/o replacement from categorical distribution with logits θ
- To obtain k samples w/o replacement
  1. Perturb θ with Gumbel noise: $z_i = \theta_i - \log(-\log(v_i)),\ v_i \sim U[0,1]$
  2. Take positions of top-k $z = (z_1, \ldots, z_d)$
- When $k = 1$ we get the Gumbel max trick
- When $k = d$ we obtain a sample from the Plackett-Luce distribution

- Note: the trick reduces sampling complexity from $O(d^2)$ to $O(d \log d)$

Use Cases

- **Variational Optimization:** replace discrete optimization w.r.t. $b \in S_d$ with continuous optimization w.r.t. θ

$$\min_{b \in S_n} f(b) \leq \min_{\theta \in \Theta} \mathbb{E}_{p(b|\theta)} f(b)$$

- **Variational inference:** approximate the posterior distribution for models with latent permutations

$$\max_{\theta} \mathbb{E}_{q(b|\theta)} \log \frac{p(X, b)}{q(b \mid \theta)}$$

- However, expectations are typically intractable and we need to use SGD to solve the tasks
- To use SGD efficiently we need low-variance gradient estimates

A Brief Tour of Gradient Estimation

For now, we consider optimization task $\min_{\theta} \mathbb{E}_{p(b|\theta)} f(b)$ and **an arbitrary** discrete $p(b \mid \theta)$

REINFORCE

For $b \sim p(b \mid \theta)$ the estimator is

$$\hat{g}_1(f) = (f(b) - C)\, \nabla_\theta \log p(b \mid \theta)$$

+ No bias, applicable to *almost* any distribution
- High variance if C is not carefully chosen

Reparametrized Gradients

For continuous relaxation $z = T(v, \theta)$ (e.g. Gumbel-Softmax) and $v \sim U[0,1]^d$ we have

$$\hat{g}_2(f) = \nabla f(b_{cont}) = \frac{\partial f}{\partial T} \cdot \nabla_\theta T$$

+ Low variance, extendable to permutations [3, 4]
- Cons: biased gradients due to relaxation, f must be defined for relaxed b

REBAR & RELAX

Rough idea:

1. *from REINFORCE estimator subtract the REINFORCE estimator for relaxed variable to reduce variance*
2. *Add the reparametrized estimator to compensate bias*

For relaxation $z \sim p(z \mid \theta)$, hard map $b = H(z)$ and conditional sample $\tilde{z} \sim p(z \mid b, \theta)$ we have

$$\hat{g}_3(f) = [f(b) - c_\phi(\tilde{z})]\, \nabla_\theta \log p(b \mid \theta) + \nabla_\theta c_\phi(z) - \nabla_\theta c_\phi(\tilde{z})$$

+ No bias, low variance, trainable control variate $c_\phi(\cdot)$ in RELAX
- Need to find suitable $p(z \mid \theta), H(z)$ and $p(z \mid b, \theta)$ for $p(b \mid \theta)$

REBAR & RELAX for PL

- [1] and [2] derive estimators for categorical $p(b \mid \theta)$
- In this section, we define the estimator for $p(b \mid \theta)$ from the Plackett-Luce distribution

- Need to define $p(z \mid \theta)$ and $H(z)$, s.t. for $p(z, b \mid \theta) = I[b = H(z)] \cdot p(z \mid \theta)$ the marginal over b is the PL distribution $p(b \mid \theta)$
- We define $p(z \mid \theta)$ and $H(z)$ using the Gumbel top-k trick. For $v_i \sim U[0,1]$

$$z_i := \theta_i - \log(-\log(v_i)),\ i = 1, \ldots, d$$
$$H(z) := \arg\operatorname{sort}(z_1, \ldots, z_d)$$

- Given $p(z \mid \theta)$ and $H(z)$ we derive conditional distribution $p(z \mid b, \theta)$

Proposition. Assume $\sum_{i=1}^{d} \exp \theta_i = 1$, then for $v_i \sim U[0,1],\ i = 1, \ldots, d$ and $\Theta_i = \sum_{j=i}^{k} \exp \theta_{b_j}$

$$z_{b_i} = \begin{cases} -\log(-\log v_i), & i = 1 \\ -\log\left(-\frac{\log v_i}{\Theta_i} + \exp(-z_{b_{i-1}})\right) & i \geq 2 \end{cases}$$

is a sample from $p(z \mid b, \theta)$

The Toy Experiment

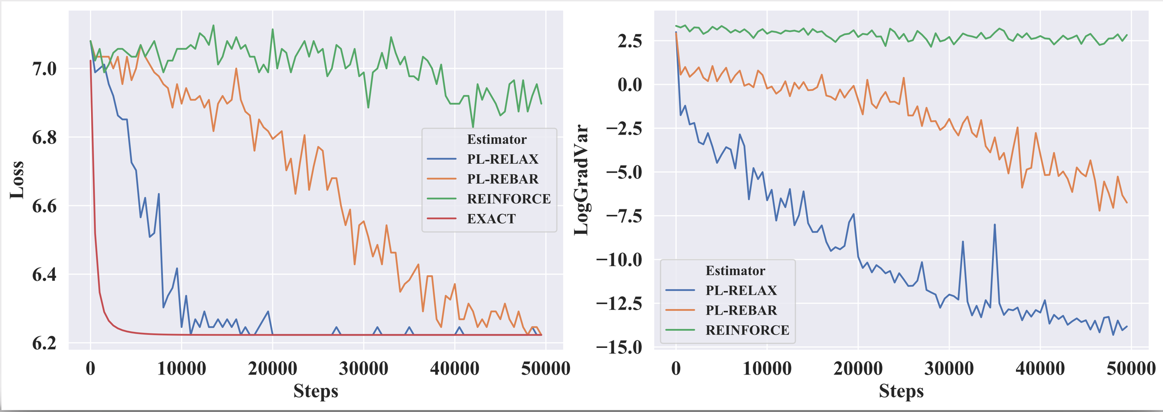
Consider a simple **linear sum assignment** problem with the specifically constructed doubly stochastic matrix P_t of size $d = 8$:

$$\min_{\theta} \mathbb{E}_{p(b|\theta)} \|P_b - P_t\|_F^2$$

Here P_t and P_b are defined as follows:

$$(P_t)_{ij} = \begin{cases} \frac{1}{d} + t, & i = j \\ \frac{1}{d} - \frac{t}{d-1}, & i \neq j \end{cases} \qquad (P_b)_{ij} = \begin{cases} 1, & j = b_i \\ 0, & j \neq b_i \end{cases}$$

- REINFORCE does not work even for simple task
- PL-RELAX converges almost as fast as with the exact gradient and significantly reduces variance



Causal Structure Learning

Consider **linear structural equation model**

$$X = W^T X + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I)$$

and corresponding **optimization problem**

$$\min_{W} \frac{1}{2n} \|X - W^T X\|_F^2 + \lambda \|vec(W)\|_1$$

where **W** is **the adjacency matrix of DAG**, which describes causal relations.

We parametrize W as $W = P_b A P_b^T$, where

- P_b is the permutation matrix of a topological sort of a DAG
- A is a strictly upper-triangular matrix

For each b we find the best A by optimizing

$$\hat{Q}(P_b, X) = \min_{A} \frac{1}{2n} \|X - P_b A P_b^T X\|_F^2 + \lambda \|vec(A)\|_1$$

Then we use PL-RELAX to solve

$$\min_{\theta} \mathbb{E}_{p(b|\theta)} \hat{Q}(P_b, X)$$

	Val $\widehat{Q} - \widehat{Q}^*$	SHD	SHD-CPDAG	SID
PL-RELAX	-1.8±1.3	19.2±6.9	20.6±7.8	103.2±55.5
SINKHORN$_{ECP}$	5.5±7.0	30.0±6.3	30.8±5.8	151.8±35.1
URS$_{ECP}$	10.3±4.7	41.0±2.4	40.0±2.7	177.6±17.1
SINKHORN	90.3±35.8	49.6±4.3	49.6±4.3	275.0±42.5
URS	90.3±35.8	49.6±4.3	49.6±4.3	275.0±42.5
GREEDY-SP	N/A	38.2±21.6	38.2±24.6	151.6±84.3
RANDOM	271.0±71.6	99.4±9.3	99.8±9.5	301.2±60.4

Fig 1. Results for Erdos-Renyi graphs with 50 nodes and 10% edges. See our paper more results, including different number of nodes and other graph types

References

[1] Tucker, George, et al. "Rebar: Low-variance, unbiased gradient estimates for discrete latent variable models." *NIPS 2017*

[2] Grathwohl, Will, et al. "Backpropagation through the void: Optimizing control variates for black-box gradient estimation." *ICLR 2018*

[3] Mena, Gonzalo, et al. "Learning latent permutations with gumbel-sinkhorn networks." *ICLR 2018*

[4] Grover, Aditya, et al. "Stochastic optimization of sorting networks via continuous relaxations." *ICLR 2019.*