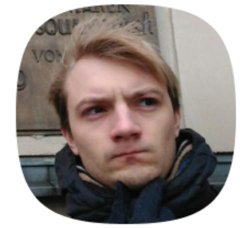
Vladislav Belavin, Maxim Borisyak



Variational Optimization

2021

MLHEP 2021

HIGHER SCHOOL OF ECONOMICS
NATIONAL RESEARCH UNIVERSITY

LAMBDA • HSE

Yandex

SCHOOL OF DATA ANALYSIS

EPFL

Schaffhausen Institute of Technology

Variational bound

Variational bound

Variational Optimization replaces problem:

$$\mathsf{f}(\theta) \to \min_{\theta};$$

with:

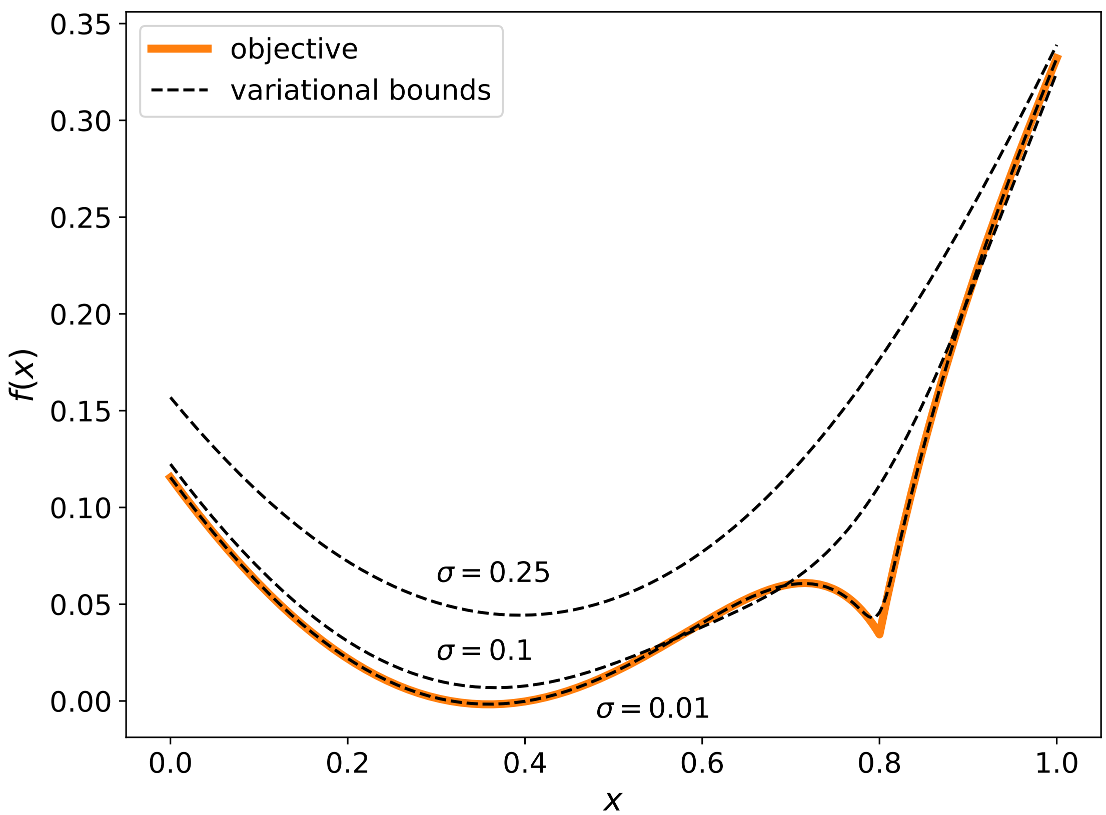
$$\mathsf{J}(\psi) = \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot|\psi)} \mathsf{f}(\theta) \to \min_{\psi}$$

where:

- $\mathsf{J}(\psi)$ - variational bound;
- $\mathsf{P}(\cdot \mid \psi)$ - search distribution.

This variational bound is not the only one, nevertheless, it is the most common in Varitional Optimization.

Variational bound: example



Properties

$$\mathsf{J}(\psi) = \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot \mid \psi)} \mathsf{f}(\theta) \to \min_{\psi}$$

- upper bound:
$$\forall \psi : \mathsf{J}(\psi) \geq \min \mathsf{f}(\theta);$$
- if $\mathsf{P}(\cdot \mid \psi)$ is allowed to (nearly) collapse into delta function, then
$$\mathsf{P}(\cdot \mid \psi^*) \approx \delta(\theta^*).$$

Gradient of the variational bound

$$
\begin{aligned}
\frac{\partial}{\partial \psi}\mathsf{J}(\psi) = & \frac{\partial}{\partial \psi} \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot|\psi)} \mathsf{f}(\theta) = \\
& \frac{\partial}{\partial \psi} \int_\theta \mathsf{d}\theta\ \mathsf{f}(\theta)\mathsf{P}(\theta \mid \psi) = \\
& \int\limits_\theta \mathsf{d}\theta\ \mathsf{f}(\theta)\ \frac{\partial}{\partial \psi}\mathsf{P}(\theta \mid \psi) = \\
& \int_\theta \mathsf{d}\theta\ \mathsf{f}(\theta)\ \mathsf{P}(\theta \mid \psi)\ \frac{\partial}{\partial \psi} \log \mathsf{P}(\theta \mid \psi) = \\
& \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot|\psi)} \mathsf{f}(\theta) \frac{\partial}{\partial \psi} \log \mathsf{P}(\theta \mid \psi)
\end{aligned}
$$

Gradient of the variational bound

$$\boxed{\nabla_\psi \mathsf{J}(\psi) = \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot|\psi)} \mathsf{f}(\theta)\ \nabla_\psi \log \mathsf{P}(\theta \mid \psi)}$$

$\nabla_\psi \mathsf{J}(\psi)$ does not depend on $\nabla_\theta \mathsf{f}(\theta)$

Gradient of the variational bound

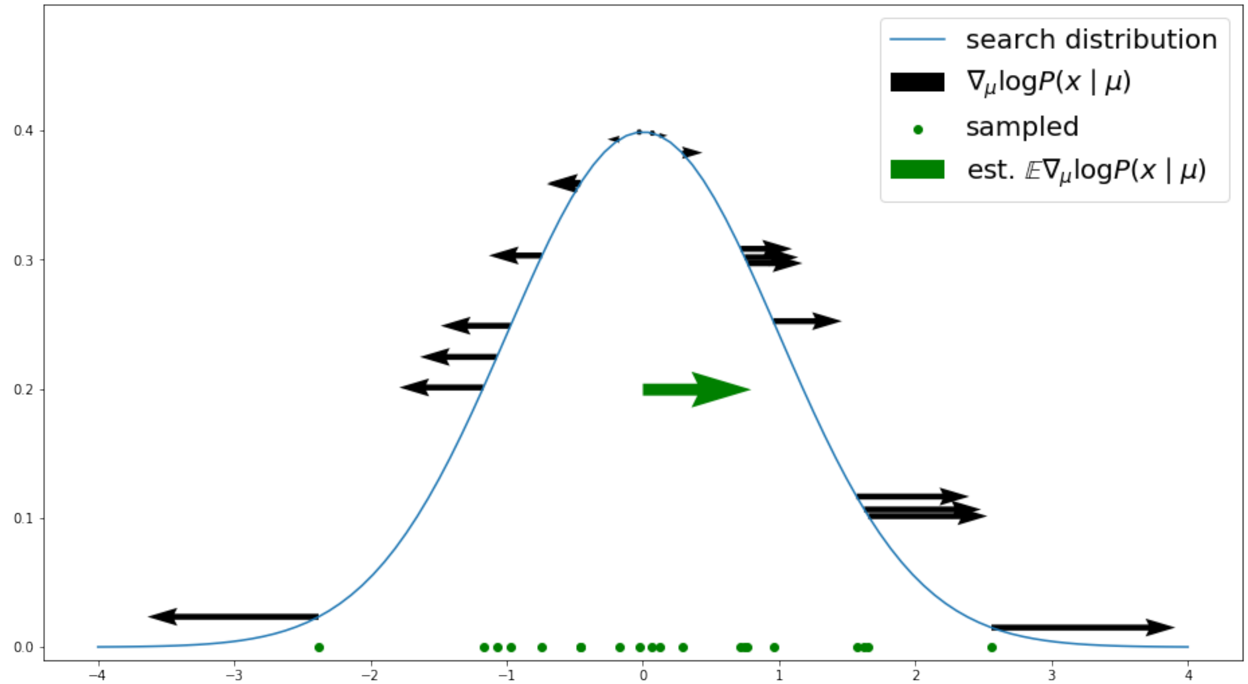


Figure 1: Gradient of $\mathcal{N}(\mathrm{x}; \mu, \sigma)$ w.r.t. μ, i.e. $\nabla_\mu \log \mathcal{N}(\mathrm{x}; \mu, \sigma)$

Gradient of the variational bound

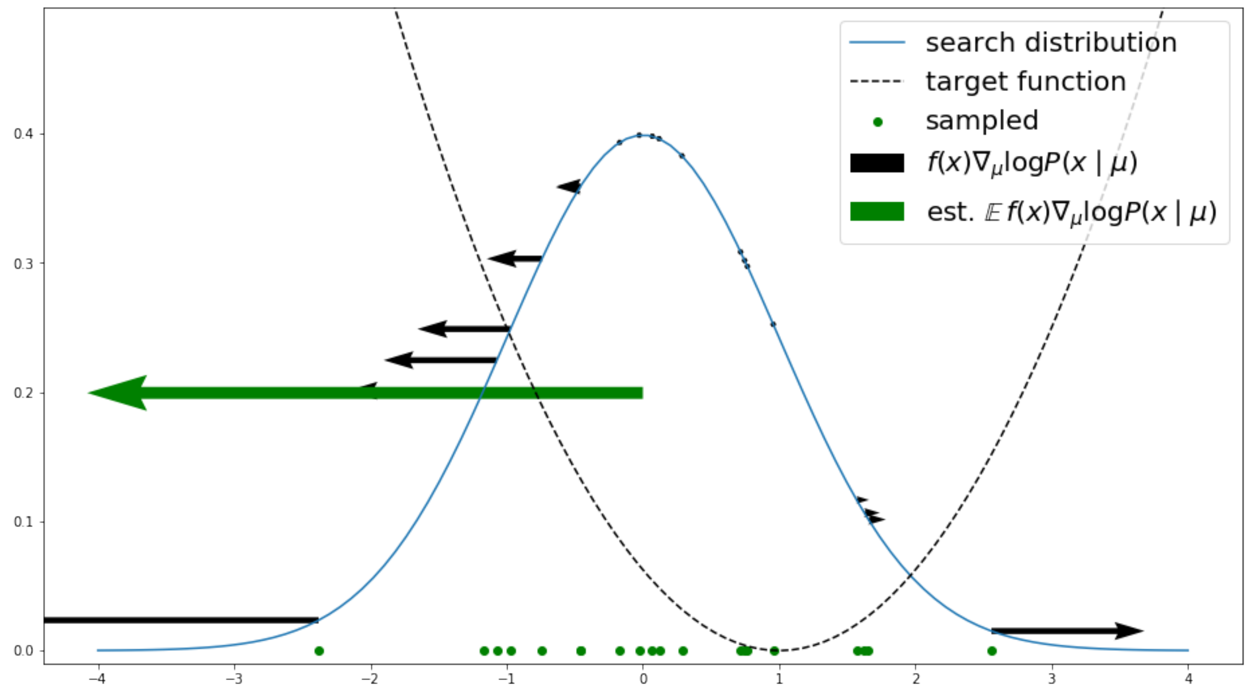
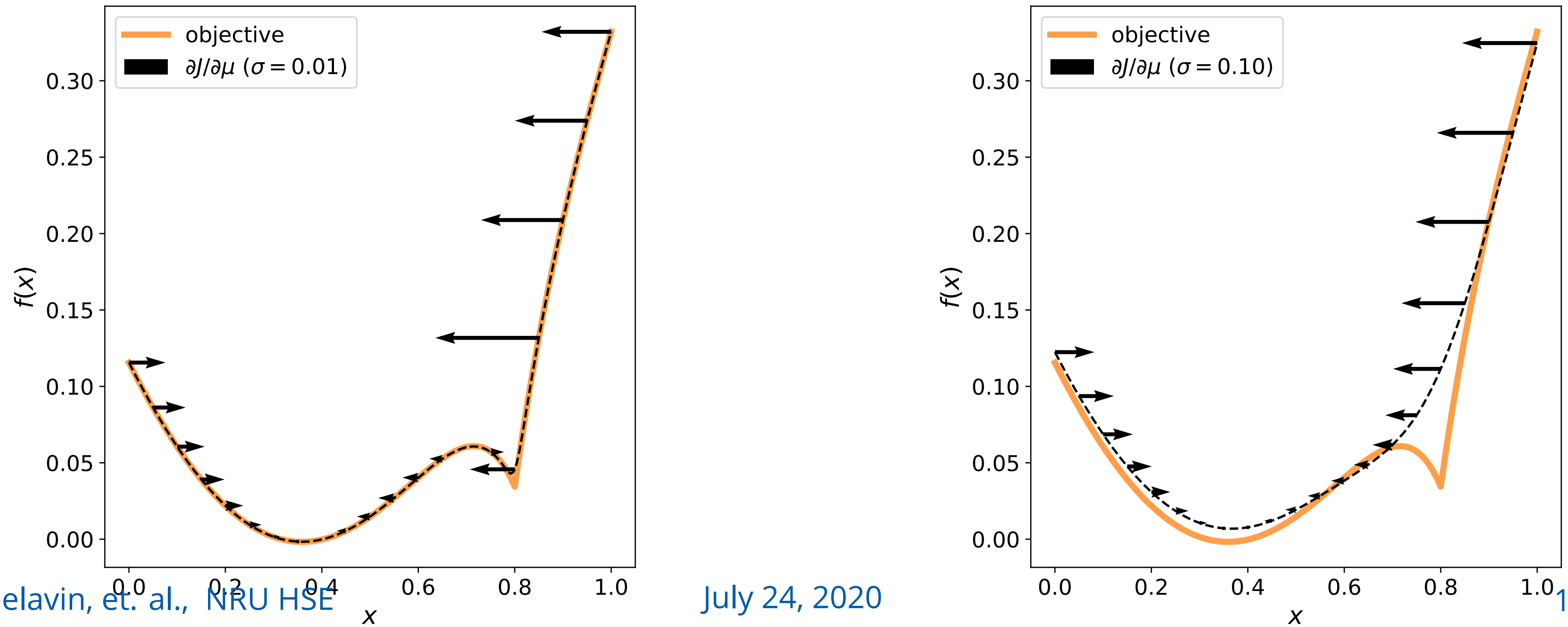


Figure 2: Gradient of variational bound with $\mathcal{N}(\mathrm{x}; \mu, \sigma)$ as search distribution w.r.t. μ, i.e. $\mathrm{f}(\mathrm{x})\nabla_\mu \log \mathcal{N}(\mathrm{x}; \mu, \sigma)$

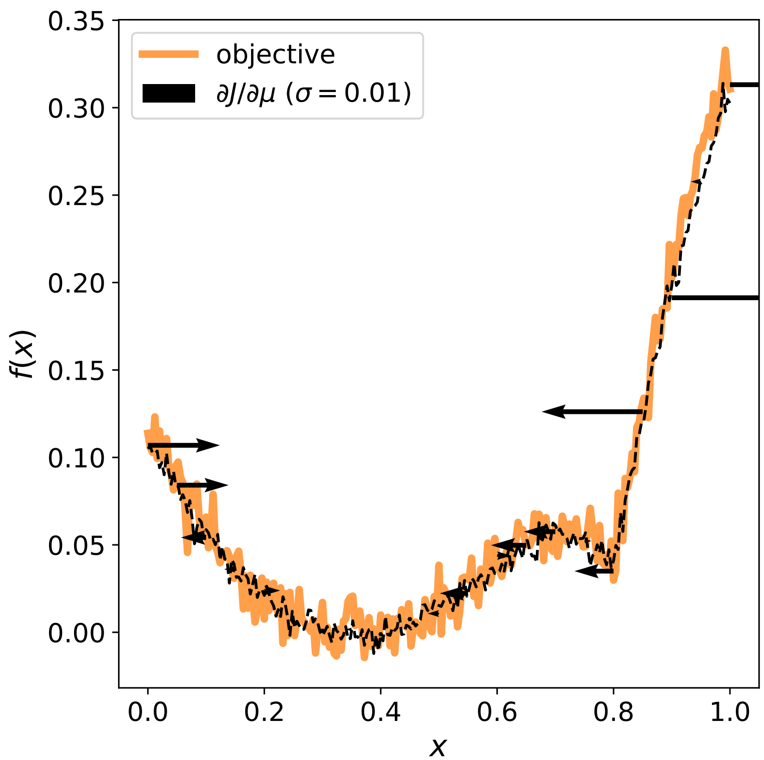
Gradient of the variational bound

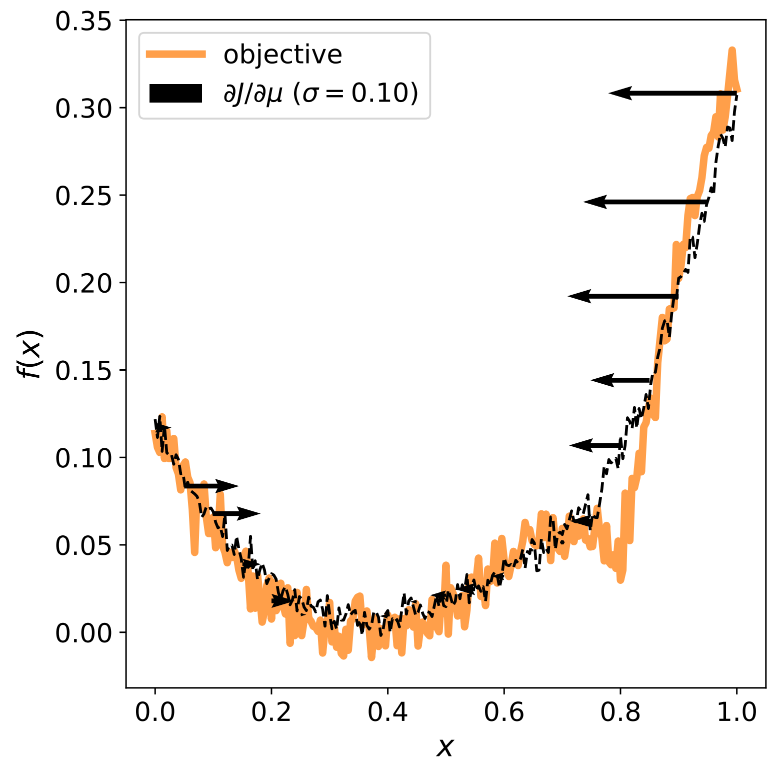
> Trade-off: higher σ – smooth gradients, ability to avoid local minima, lower σ – closer to original objective, but more prone to noise.



Gradient of the variational bound

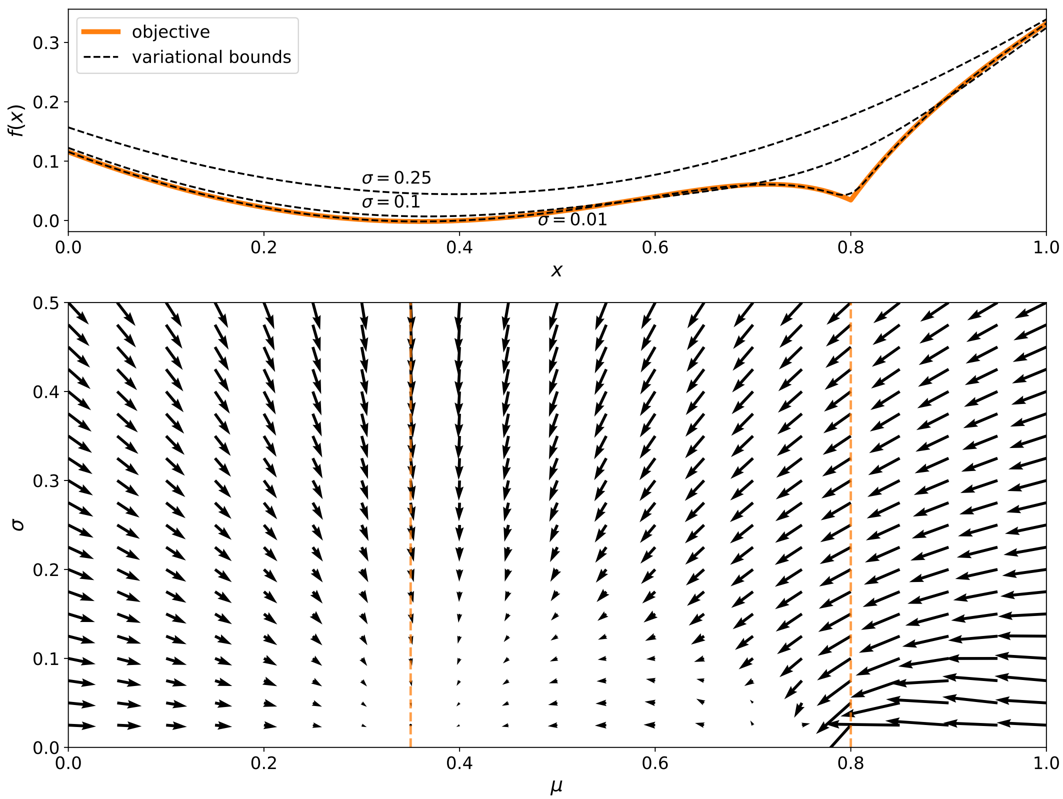
Higher σ also negates sampling noise effects.





Gradient of the variational bound

In practice, σ usually collapses from any initialization.



Variational optimization

SGD-VO

Algorithm 1 SGD-VO

initialize $\mathsf{P}(\cdot \mid \psi)$
while not converged **do**
  sample θ from $\mathsf{P}(\cdot \mid \psi)$;
  $\nabla_\psi \mathsf{J}(\psi) \leftarrow \mathsf{f}(\theta)\nabla_\psi \log \mathsf{P}(\theta \mid \psi)$;
  $\psi \leftarrow \psi - \gamma \nabla_\psi \mathsf{J}(\psi)$;
end while

Variational Optimization

- allows usage of stochastic gradient methods for black-box problems:
  - VO is much slower in contrast to using analytical gradient;
- search distribution is chosen to be simple:
  - e.g. normal distribution;
- dimensionality of the problem can be retained:
  - at least 1 additional parameter to allow the search distribution to collapse (σ);
  - $2 \cdot \mathrm{n}$ parameters for a normal distribution with diagonal covariance;
  - $\mathrm{O}(\mathrm{n}^2)$ for a full convariance matrix.

Discrete variables

Variational optimization

Variational optimization could be straightforwardly used to estimate gradient with the well know formula.

$$\nabla_\psi \mathsf{J}(\psi) = \mathop{\mathbb{E}}_{\theta \sim \mathsf{P}(\cdot|\psi)} \mathsf{f}(\theta)\ \nabla_\psi \log \mathsf{P}(\theta \mid \psi)$$

For example, imagine the simplest problem, where $\frac{\mathsf{df}}{\mathsf{dx}}$ do not exist:

$$\mathsf{f}(\mathsf{x}) = \begin{cases} 0.45, & \mathsf{x} = 0 \\ 0.53, & \mathsf{x} = 1 \end{cases}$$

Variational optimization

$$f(x) = \begin{cases} 0.45, & x = 0 \\ 0.53, & x = 1 \end{cases}$$

As a search distribution we will choose $p(x|\psi) = \mathrm{Bernoulli}(x|\psi)$:

$$J(\psi) = \mathop{\mathbb{E}}_{x \sim P(x|\psi)} f(x) = 0.53\psi + 0.45(1 - \psi) = 0.08\psi + 0.45$$

Voila! Even though f(x) can not be differentiated, we are stil able to compute gradient of variational bound ($\mathbb{E}_{x \sim P(x|\psi)} f(x)$) w.r.t. ψ and optimize upper bound.

Discrete variables in Deep Learning

Neural networks with discrete random variables are a powerful technique for representing processes encountered in language modeling, attention mechanisms, and robotics control. Discrete representations are often more interpretable.
However, networks with discrete variables are hard to train.

Cons of VO: Monte-Carlo estimate of $\nabla_{\psi} \mathbb{E}_{x \sim P(x|\psi)} f(x)$ has high variance and, consequently, slow convergence. The variance scales linearly with the number of dimensions of the sample vector*, making it especially challenging to use for categorical distributions.

*Source: `https://arxiv.org/abs/1401.4082`

Gumbel Max Trick

> Given probabilities $\{\pi_i\}_{i=1}^n$, $\sum \pi_i = 1$, we want to be able to sample from this discrete distribution.

Proposition. Following procedure:

$$z = \arg\max\{\log \pi_i + G_i\}, \quad G_i = -\log(-\log u_i), \quad u_i \sim U[0, 1],$$

generates samples from discrete probability distribution defined by $\{\pi_i\}_{i=1}^n$.

> Yet we still can't backprop through z w.r.t. π, but it's only a first step.

Gumbel Softmax Trick

> To be able to backprop through discrete sample we will relax our problem with the softmax operation:

$$\mathsf{y}_\mathsf{i} = \frac{\exp((\log \pi_\mathsf{i} + \mathsf{G}_\mathsf{i})/\tau)}{\sum\limits_\mathsf{j} \exp((\log \pi_\mathsf{j} + \mathsf{G}_\mathsf{j})/\tau)}, \quad \mathsf{G}_\mathsf{i} = -\log(-\log \mathsf{u}_\mathsf{i}), \quad \mathsf{u}_\mathsf{i} \sim \mathsf{U}[0,1],$$

Gumbel Softmax Trick

During forward pass:

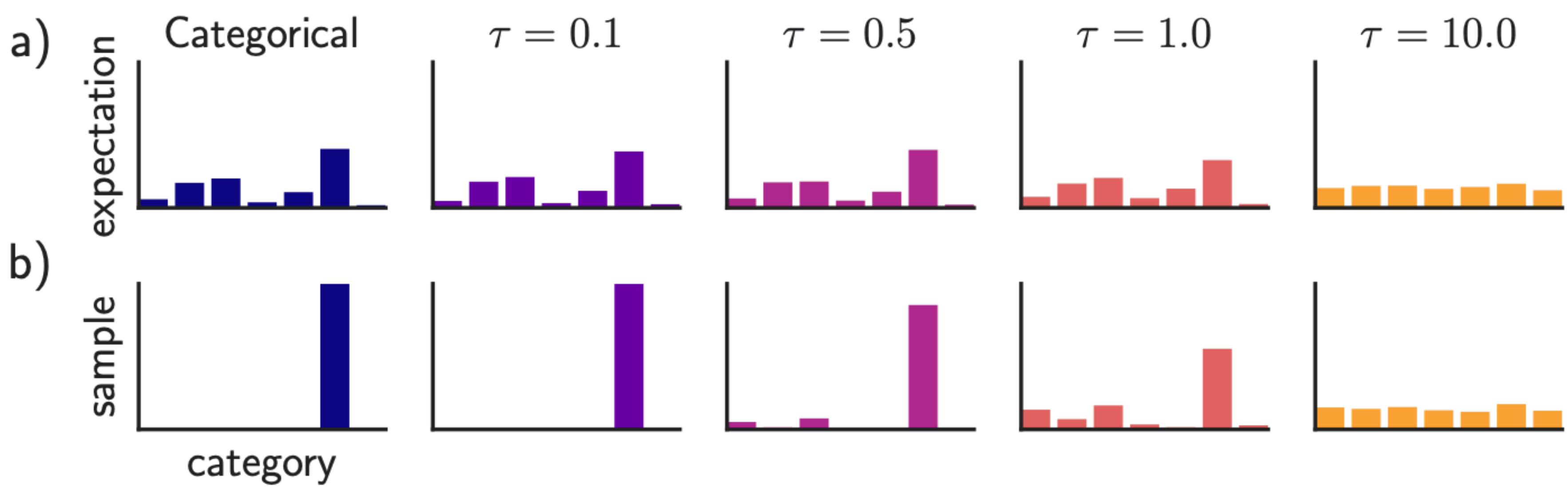
$$z = \text{one-hot}(\arg\max\{y_i\}),$$

During backward pass:

$$\frac{dy}{d\pi} \to \frac{dz}{d\pi}, \quad \tau \to 0$$

Gumbel Softmax Trick

Temperature annealing of gumbel softmax:



Gradient estimation for discrete variables overview

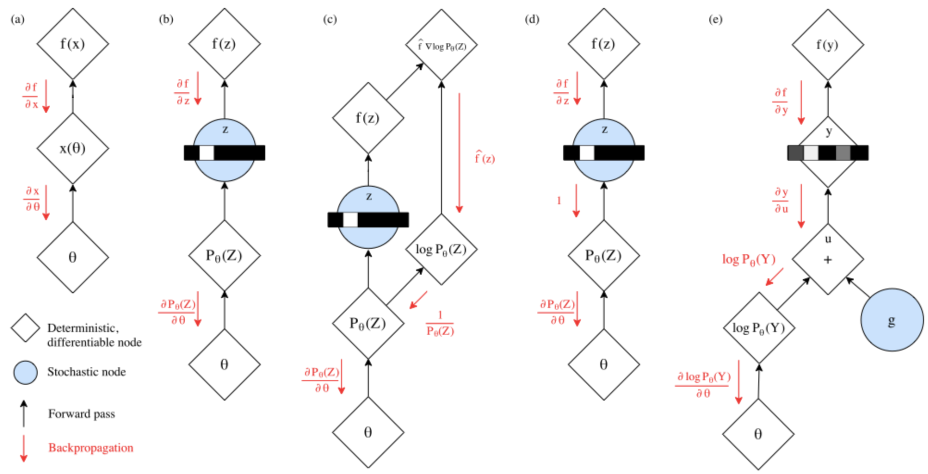


Figure 3: (1) Differentiable linear layer, (2) Non-differentiable discrete layer, (3) variational gradient, (4) Straight-Through estimator, (4) Gumbel-Softmax estimator

Source: `https://arxiv.org/abs/1401.4082`

Conclusion

Variational optimization:

- good for optimization of non-differentiable functions;
- local optimization algorithm:
  - suffer less from the curse of dimensionality then global algorithms;
  - might converge into local optima;
- sensitive to the choice of (hyper)parameters.

Discrete variables:

- variational optimization could be used;
  - high variance => slow convergence;
- relaxation methods, like Gumbel Softmax Trick, worth trying.

Quiz

> You want to estimate gradient of the variational bound for the function $f(x) = \sin x$ with normal search distribution $p(x|\mu, \sigma^2) = \mathcal{N}(x; \mu, \sigma^2)$. Choose the correct sample MC estimate $x_{\text{sample}} \sim \mathcal{N}(x; \mu, \sigma^2)$ of the gradient w.r.t. μ:

1. $\cos(x_{\text{sample}}) \cdot p(x_{\text{sample}}|\mu, \sigma^2)$
2. $\sin(x_{\text{sample}}) \cdot \frac{x_{\text{sample}} - \mu}{\sigma^2}$
3. $\cos(x_{\text{sample}}) \cdot \frac{x_{\text{sample}} - \mu}{\sigma^2}$
4. $\sin(x_{\text{sample}}) \cdot p(x_{\text{sample}}|\mu, \sigma^2)$

Thank you for your attention!

Vladislav Belavin

vbelavin@hse.ru

SchattenGenie

hse_lambda