Probability Functional Descent:
A Unifying Perspective on GANs, VI, and RL

Casey Chu <caseychu@stanford.edu>    Jose Blanchet    Peter Glynn
Deep generative models
Deep generative models

Variational inference

Deep reinforcement learning
Probability functional

\[ J : \mathcal{P}(X) \rightarrow \mathbb{R} \]
Probability functional

\[ J : \mathcal{P}(X) \rightarrow \mathbb{R} \]

“gradient” \( \nabla J \)
Probability functional

\[ J : \mathcal{P}(X) \rightarrow \mathbb{R} \]

von Mises influence function

\[ \Psi : X \rightarrow \mathbb{R} \]
Gradient descent on $f: \mathbb{R}^n \rightarrow \mathbb{R}$

0. Initialize $x \in \mathbb{R}^n$ arbitrarily

1. Compute the gradient $g = \nabla f(x)$

2. Choose $x'$ such that $x' \cdot g < x \cdot g$ (usually, we set $x' = x - \alpha g$)
Gradient descent on $f : \mathbb{R}^n \rightarrow \mathbb{R}$

0. Initialize $x \in \mathbb{R}^n$ arbitrarily
1. Compute the gradient $g = \nabla f(x)$
2. Choose $x'$ such that $x' \cdot g < x \cdot g$ (usually, we set $x' = x - \alpha g$)

Probability functional descent on $J : \mathcal{P}(X) \rightarrow \mathbb{R}$

0. Initialize a distribution $\mu \in \mathcal{P}(X)$ arbitrarily
1. Compute the influence function $\Psi$ of $J$ at $\mu$
2. Choose $\mu'$ such that $\mathbb{E}_{x \sim \mu'}[\Psi(x)] < \mathbb{E}_{x \sim \mu}[\Psi(x)]$
Generative modeling

\[ J_G(\mu) = D(\mu \| \nu_0) \]

where \( D \) is e.g. Jensen–Shannon, Wasserstein

1. Optimize the **discriminator**, which approximates the influence function of \( J_G \)
2. Update the **generator** \( \mu \)

PFD recovers:
- Minimax GAN
- Non-saturating GAN
- Wasserstein GAN

**Probability functional descent**

1. Compute the **influence function** \( \Psi \) of \( J \) at \( \mu \)
2. Choose \( \mu' \) such that
\[ \mathbb{E}_{x \sim \mu'}[\Psi(x)] < \mathbb{E}_{x \sim \mu}[\Psi(x)] \]
Variational inference

\[ J_{VI}(q) = KL(q(\theta) \parallel p(\theta|x)) \]

1. Compute the **ELBO**, \( \log(q(\theta)/p(x,\theta)) \), the influence function for \( J_{VI} \)
2. Update the **approximate posterior** \( q \)

PFD recovers:
- Black-box variational inference
- Adversarial variational Bayes
- Approximate posterior distillation

**Probability functional descent**

1. Compute the **influence function** \( \Psi \) of \( J \) at \( \mu \)
2. Choose \( \mu' \) such that \( E_{x \sim \mu'}[\Psi(x)] < E_{x \sim \mu}[\Psi(x)] \)
Reinforcement learning

\[ J_{RL}(\pi) = \mathbb{E}_\pi [\sum_t \gamma^t R_t] \]

1. Approximate the advantage \( Q^\pi(s, a) - V^\pi(s) \), the influence function for \( J_{RL} \)
2. Update the policy \( \pi \)

PFD recovers:
- Policy gradient
- Actor-critic
- Dual actor critic

Probability functional descent

1. Compute the influence function \( \Psi \) of \( J \) at \( \mu \)
2. Choose \( \mu' \) such that \( \mathbb{E}_{x \sim \mu'}[\Psi(x)] < \mathbb{E}_{x \sim \mu}[\Psi(x)] \)
Probability functional descent is a unifying perspective that enables the easy development of new algorithms.
Probability functional descent is a unifying perspective that enables the easy development of new algorithms.
Probability Functional Descent: 
A Unifying Perspective on GANs, VI, and RL

Casey Chu <caseychu@stanford.edu>  Jose Blanchet  Peter Glynn