Two Generator Game: Learning to Sample via Linear Goodness-of-Fit Test

Lizhong Ding, Mengyang Yu, Li Liu, Fan Zhu, Yong Liu, Yu Li, Ling Shao

Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, UAE.
Institute of Information Engineering, CAS, China.
King Abdullah University of Science and Technology (KAUST), Saudi Arabia.

Background

- Learning the probability distribution of high-dimensional data.
- GANs are a type of implicit generative models (IGMs).
- Existing GAN models are fundamentally two-sample test problems.
- We summarize existing GAN models into two categories:
  - Integral Probability Metric (IPM).

$$\delta(p,q)=\sup_{f\in\mathcal{F}}\left|\int_{\mathcal{X}} f(x)p(x)dx-\int_{\mathcal{X}} f(x')q(x')dx'\right|.$$

    Wasserstein GANs (WGANs), MMD-GAN

  - ζ-divergence

$$\delta_\zeta(p,q)=\int_{\mathcal{X}} q(x)\zeta\left(\frac{p(x)}{q(x)}\right)dx,$$

    where ζ is a convex, lower-semicontinuous, satisfying $\zeta(1)=0$.
    GAN, least squares GAN

- Analyzing and comparing distributions without imposing any parametric assumptions
  - Two-sample test:
    Whether two distributions **p** and **q** are different based on

$$\mathcal{D}_x=\{x_i\}_{i=1}^n\subset\mathcal{X}\subseteq\mathbb{R}^d\sim p$$
$$\mathcal{D}_y=\{y_j\}_{j=1}^m\subset\mathcal{Y}\subseteq\mathbb{R}^d\sim q$$

  - Goodness-of-fit test:
    How well a given model density **p** fits a set of given samples

$$\mathcal{D}_x=\{x_i\}_{i=1}^n\subset\mathcal{X}\subseteq\mathbb{R}^d\sim q$$

Contributions

- Deep energy adversarial network (DEAN):
  A new paradigm that casts the generative adversarial learning as a goodness-of-fit (GOF) test problem.
- A novel two generator game via GOF tests:
  - One explicit generator is designed to learn an energy-based distribution (EBD), which maps the real data to a scalar energy-based probability.
  - The other implicit generator is trained by minimizing the vFSSD between the EBD and the generated data.
- A two-level alternative optimization procedure to train the explicit and implicit generative networks, such that the hyper-parameters can also be automatically learned.

Energy Estimator Network

Energy-based models $\mathcal{E}_\theta(x):\mathcal{X}\to\mathbb{R}$ associate an energy value with a sample **x**, where θ are the parameters. We can obtain a distribution

$$p(x;\theta)=\frac{1}{Z_\theta}\exp(-\mathcal{E}_\theta(x)).$$

We define the loss function of the explicit generative network (EGN) of DEAN as follows:

$$\min_{\theta_e}\mathcal{E}(x;\theta_e)+\left[\gamma-\mathcal{E}\left(G(z;\theta_g^*);\theta_e\right)\right]^+,\qquad(1)$$

where $\mathcal{E}(x;\theta_e)$ is an *energy model* parameterized by θ_e, $[\cdot]^+=\max(\cdot,0)$ and γ is a given positive margin.

Energy Estimator Network

We consider a deep auto-encoder as a more complex energy model

$$\mathcal{E}(x;\theta)=\|x-\mathrm{AE}(x;\theta_e)\|,$$

where $\mathrm{AE}(x;\theta_e)$ denotes a deep auto-encoder parameterized by θ_e. For the optimized parameters θ_e^*, we can define

$$p(x;\theta_e^*)=\frac{1}{Z_{\theta_e^*}}\exp(-\mathcal{E}(x;\theta_e^*)).$$

GOF-driven Generator Network

Kernel Stein operator can be written as

$$(T_pf)(x)=\sum_{i=1}^d\left(\frac{\partial\log p(x)}{\partial x_i}f_i(x)+\frac{\partial f_i(x)}{\partial x_i}\right)=\langle f,\omega_p(x,\cdot)\rangle_{\mathcal{F}^d}.$$

Kernel Stein discrepancy (KSD) is formulated as

$$\mathrm{KSD}[\mathcal{F}^d,p,\mathcal{D}_x]=\sup_{\|f\|_{\mathcal{F}^d}\le 1}\langle f,E_{x\sim q}\omega_p(x,\cdot)\rangle:=\|g(\cdot)\|_{\mathcal{F}^d},\qquad(2)$$

where $g(\cdot)=E_{x\sim q}\omega_p(x,\cdot)$. The statistic of the finite set Stein discrepancy (FSSD) is defined as

$$\mathrm{FSSD}[\mathcal{F}^d,p,\mathcal{D}_x]=\frac{1}{dJ}\sum_{i=1}^d\sum_{j=1}^J g_i^2(v_j).$$

The unbiased estimator of FSSD is defined as

$$\widehat{\mathrm{FSSD}}^2[\mathcal{F}^d,p,\mathcal{D}_x]=\frac{2}{n(n-1)}\sum_{i<j}\Delta(x_i,x_j),$$

where $\Delta(x,y)=\tau(x)^{\mathrm{T}}\tau(y)$. To alleviate the impact of dimension and stabilize the statistic, we introduce

$$\mathrm{vFSSD}[p,\mathcal{D}_x]=\frac{1}{\hat{\sigma}_{H_1}}\widehat{\mathrm{FSSD}}^2[p,\mathcal{D}_x].$$

GOF-driven Generator Network

We define the loss function of the IGN as follows:

$$\min_{\theta_g}\max_{\xi}\mathrm{vFSSD}_\xi\left[p(x;\theta_e^*),\mathcal{D}_{x'}\right],\qquad(3)$$

where $\xi=\left\{\{v_i\}_{i=1}^J,\sigma_k\right\}$ denotes the hyper-parameters of vFSSD, including the kernel parameter σ_k and J test locations $\{v_i\}_{i=1}^J$.

We present the following two objectives, (4) and (5), to optimize Equation (3) and improve the test power of DEAN.

$$\max_{\xi}\mathrm{vFSSD}_\xi\left[p(y;\theta_e^*),\mathcal{D}_{x'^*}\right],\qquad(4)$$

where $\mathcal{D}_{x'^*}=\left\{x_i'^*:=G(z_i;\theta_g^*)\right\}_{i=1}^n$ and $G(z_i;\theta_g^*)$ is a deep network with the optimized parameter θ_g^*. The hyper-parameters $\xi=\left\{\{v_i\}_{i=1}^J,\sigma_k\right\}$ will be optimized in Equation (4).

$$\min_{\theta_g}\mathrm{vFSSD}_{\xi^*}\left[p(y;\theta_e^*),\mathcal{D}_{x'}\right],\qquad(5)$$

where $\xi^*=\left\{\{v_i^*\}_{i=1}^J,\sigma_k^*\right\}$ denotes the optimized hyper-parameters, and the parameters θ_g for $\mathcal{D}_{x'}=\left\{x_i':=G(z_i;\theta_g)\right\}_{i=1}^n$ will be optimized.

Theorem

We assume that $\mathcal{D}_{x'}$ is drawn from $p_{x'}$. If κ is a universal and analytic kernel; $E_{a\sim p_{x'}}E_{b\sim p_e}\left[s^{\mathrm{T}}(a)s(b)\kappa(a,b)+s^{\mathrm{T}}(b)\nabla_a\kappa(a,b)+s^{\mathrm{T}}(a)\nabla_b\kappa(a,b)+\sum_{i=1}^d\frac{\partial^2\kappa(a,b)}{\partial a_i\partial b_i}\right]<\infty$ *with* $s(a)=\nabla_a\log p_e(a)$; $E_{a\sim p_{x'}}\|\nabla_a\log p_e(a)-\nabla_a\log p_{x'}(a)\|^2<\infty$; $\lim_{\|a\|\to\infty}p_e(a)g(a)=0$, *where $g(\cdot)$ is given in Eq. (2) in Section 4.2; for any $J\ge 1$, almost surely $\mathrm{FSSD}[p_e,\mathcal{D}_{x'}]=0$ if and only if $p_{x'}=p_e$.*

Theorem

Let $\Lambda(\theta_e)=\mathcal{E}(x;\theta_e)+\left[\gamma-\mathcal{E}\left(G(z;\theta_g^*);\theta_e\right)\right]^+$. The minimum of $\Lambda(\theta_e)$ is achieved if and only if $p_e=p_x$. With the optimized θ_e^*, $\int_{x,z}\Lambda(\theta_e^*)p_x(x)p_z(z)dxdz=\gamma$.