Adaptive divergences

Maxim Borisyak, Tatiana Gaintseva and Andrey Ustyuzhanin

National Research University Higher School of Economics

October 21, 2020

Original paper

Adaptive divergence for rapid adversarial optimization

Maxim Borisyak[1], Tatiana Gaintseva[1] and Andrey Ustyuzhanin[1,2]

[1] Laboratory of Methods for Big Data Analysis, National Research University Higher School of Economics, Moscow, Russia

[2] Physics Department, Imperial College, London, United Kingdom

Problem statement

For:

- a parametrized family Q_ψ,
- ground-truth distribution P

find ψ^* such that:

$$Q_{\psi^*} = P \text{ (almost everywhere)};$$

given that Q_ψ and P are defined implicitly, either as:

- a **black-box** sampling procedure;
- a large data set.

Existing approaches

Heuristics:

- heavily rely on narrow assumptions;
- require specially constructed statistics[1];

$$\chi^2(P, Q) = \sum_i \frac{(n_P^i - n_Q^i)^2}{(\sigma_P^i)^2 + (\sigma_Q^i)^2}$$

- n_P^i, n_Q^i — estimated frequencies in i-th bin;
- σ_P^i, σ_P^i — uncertainties for i-th bin.

[1] The following example is from Ilten P., Williams M., Yang Y. Event generator tuning using Bayesian optimization

Existing approaches

General-purpose:

- ABC:
  - relies on summary statistics;
  - distribution of these statistics;
- adversarial:
  - rely on the underlying classifier model;
  - requires large number of samples.

$$\mathrm{JSD}(P, Q_\psi) \to \min_\psi;$$

$$\mathrm{JSD}(P, Q) = \log 2 + \frac{1}{2} \max_{f \in \mathcal{F}} \left[\mathop{\mathbb{E}}_{x \sim P} \log f(x) + \mathop{\mathbb{E}}_{x \sim Q} \log(1 - f(x)) \right].$$

Jensen-Shannon divergence

$$\mathrm{JSD}(P, Q) = \log 2 + \frac{1}{2} \max_{f \in \mathcal{F}} \left[\mathop{\mathbb{E}}_{x \sim P} \log f(x) + \mathop{\mathbb{E}}_{x \sim Q} \log(1 - f(x)) \right];$$

- maximization over all possible $f : \mathcal{X} \to [0, 1]$;
- replaced with $M \subset \mathcal{F}$ in practice:
    - typically, a powerful neural network;
    - requires large number of samples.

Pseudo-divergence

$$\mathrm{pJSD}_M(P, Q) = \log 2 + \frac{1}{2} \max_{f \in M} \left[\mathop{\mathbb{E}}_{x \sim P} \log f(x) + \mathop{\mathbb{E}}_{x \sim Q} \log(1 - f(x)) \right]$$

- M is high-capacity:
    - close approximation of JSD;
    - large number of sample for estimation;
- M is low-capacity:
    - $\exists P \neq Q : \mathrm{pJSD}_M(P, Q) = 0$;
    - small number of samples for estimation.

Adaptive divergence: main idea

Given P and Q:

- use low-capacity pseudo-divergences first:

$$\mathrm{pJSD}(P, Q) > 0 \implies \mathrm{JSD}(P, Q) > 0;$$

- increase capacity if low-capacity pseudo-divergence fails.

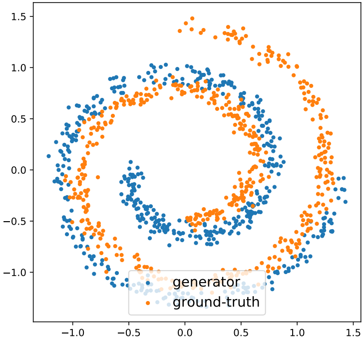
Adaptive divergence

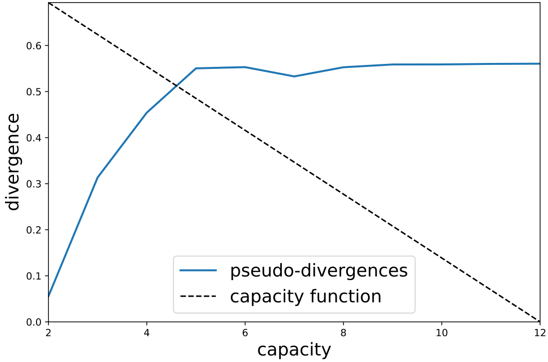
Definition (adaptive divergence)

If a family of pseudo-divergences $\mathcal{D} = \{D_\alpha \mid \alpha \in [0,1]\}$ is ordered and complete with respect to Jensen-Shannon divergence, then adaptive divergence $\mathrm{AD}_\mathcal{D}$ produced by $\mathcal{D}$ is defined as:

$$\mathrm{AD}_\mathcal{D}(P, Q) = \inf \left\{ D_\alpha(P, Q) \mid \mathrm{D}_\alpha(P, Q) \geq (1 - \alpha) \log 2 \right\}.$$

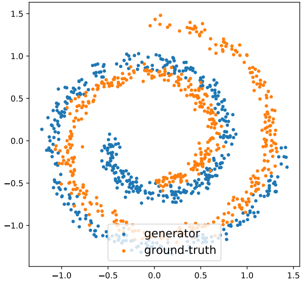
Adaptive divergence



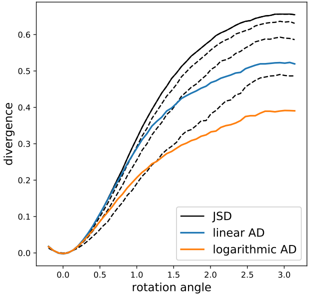


Discriminator: a 3-layer dense network with $2 \cdot N$, N and 1 units. N is the capacity parameter.

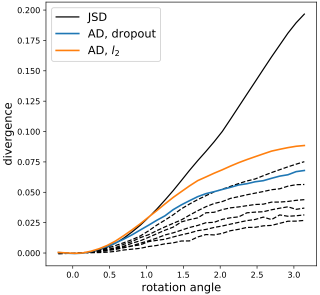
Adaptive divergence



A toy example, generator is a rotated version of the ground-truth.

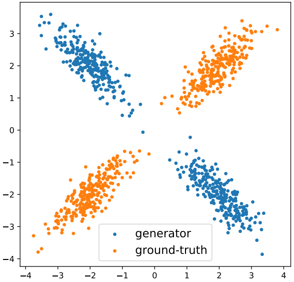


Boosted adaptive divergence, Gradient Boosting

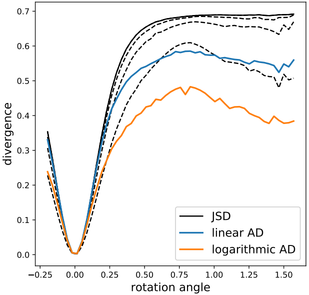


Regularized adaptive divergence, NN + dropout/ l_2

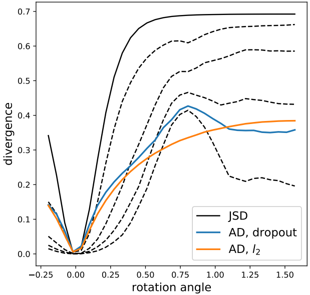
Adaptive divergence



A toy example, generator is a rotated version of the ground-truth.



Boosted adaptive divergence, Gradient Boosting

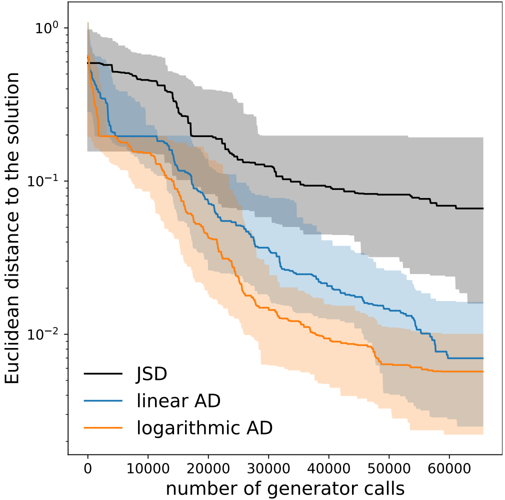


Regularized adaptive divergence, NN + dropout/ l_2

XOR experiment

XOR-like synthetic dataset:

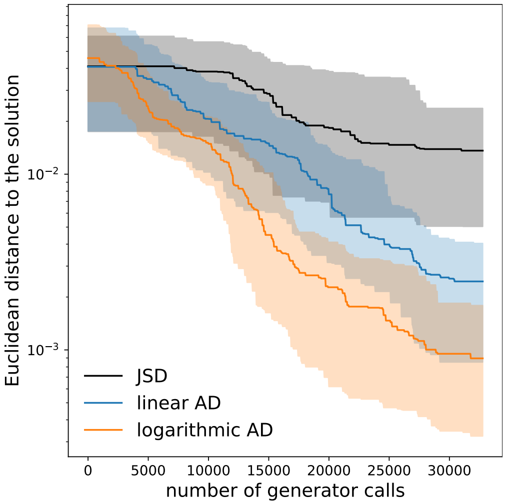
- optimizer: BO-GP;
- parameter: rotation angle;
- classifier: GBDT:
    - 100 trees of depth 3;
- family of pseudo-divergences: a boosted family.



Pythia tuning experiment

Pythia hyper-parameter tuning:

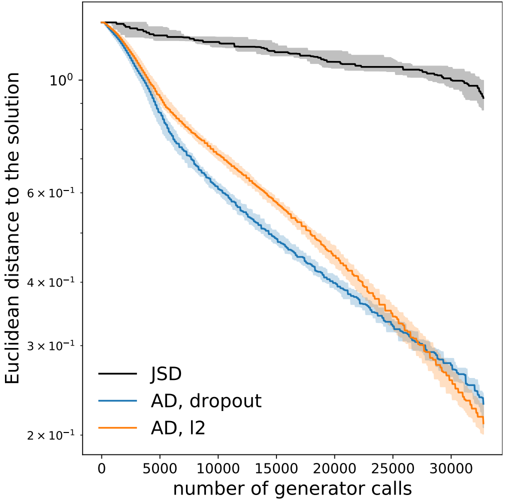
- features: Monash;
- parameter: `alphaSValue`;
- optimizer: Bayesian Optimization with Gaussian Processes;
- classifier: CatBoost:
  - 100 trees of depth 3;
- family of pseudo-divergences: a boosted family.



Pythia alignment experiment

Pythia hyper-parameter tuning:

- features: spherical toy tracker;
- parameters: tracker offset;
- optimizer: AVO;
- classifier: VGG-like CNN;
- families of pseudo-divergences:
  - dropout-regularized + $\mathrm{const}\ R_1$;
  - l_2-regularized + $\mathrm{const}\ R_1$.



Adaptive divergence

Adaptive divergence:

- *is a divergence*:
    - can be employed for fine-tuning;
- employs low-capacity pseudo-divergences when possible:
    - requires less samples for estimation;
- computationally efficient estimations algorithms:
    - for boosting-based classifiers, e.g., gradient boosting;
    - for regularized neural networks.

Extra

Pseudo-divergence

Definition (pseudo-divergence)

A function $D : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R}$ is a pseudo-divergence, if:

(P1) $\forall P, Q \in \Pi(\mathcal{X}) : D(P, Q) \geq 0;$

(P2) $\forall P, Q \in \Pi(\mathcal{X}) : (P = Q) \Rightarrow D(P, Q) = 0;$

where $\Pi(\mathcal{X})$ — set of all probability distributions on space $\mathcal{X}$.

Pseudo-divergence

Definition (ordered and complete family of pseudo-divergences)

A family of pseudo-divergences $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0,1]\}$ is ordered and complete with respect to Jensen-Shannon divergence if:

- (D0) D_α is a pseudo-divergence for all $\alpha \in [0,1]$;
- (D1) $\forall P, Q \in \Pi(\mathcal{X}) : \forall 0 \leq \alpha_1 < \alpha_2 \leq 1 : D_{\alpha_1}(P,Q) \leq D_{\alpha_2}(P,Q)$;
- (D2) $\forall P, Q \in \Pi(\mathcal{X}) : D_1(P,Q) = \mathrm{JSD}(P,Q)$.

Adaptive divergence

Theorem (on adaptive divergence)

If $\mathrm{AD}_{\mathcal{D}}$ is an adaptive divergence produced by an ordered and complete with respect to Jensen-Shannon divergence family of pseudo-divergences $\mathcal{D}$, then for any two distributions P and Q:

$$\mathrm{JSD}(P, Q) = 0 \iff \mathrm{AD}(P, Q) = 0.$$

Nested pseudo-divergences

Definition (nested pseudo-divergences)

A model family $\mathcal{M} = \{M_\alpha \subseteq \mathcal{F} \mid \alpha \in [0,1]\}$ is complete and nested, if:

- **(N0)** $(x \mapsto 1/2) \in M_0$;
- **(N1)** $M_1 = \mathcal{F}$;
- **(N2)** $\forall \alpha, \beta \in [0,1] : (\alpha < \beta) \Rightarrow (M_\alpha \subset M_\beta)$.

Theorem (on nested pseudo-divergences)

If a model family $\mathcal{M} = \{M_\alpha \subseteq \mathcal{F} \mid \alpha \in [0,1]\}$ is complete and nested, then the family $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0,1]\}$, where:

$$D_\alpha(P, Q) = \log 2 - \inf_{f \in M_\alpha} L(f, P, Q)$$

is a complete and ordered with respect to Jensen-Shannon divergence family of pseudo-divergences.

Regularization-based pseudo-divergences

Definition (regularized family of pseudo-divergences)

If M is a parameterized model family $M = \{f(\theta, \cdot) : \mathcal{X} \to [0,1] \mid \theta \in \Theta\}$, then a function $R : \Theta \to \mathbb{R}$ is a proper regularizer for the family M if:

(R1) $\forall \theta \in \Theta : R(\theta) \geq 0$;

(R2) $\exists \theta_0 \in \Theta : \left(f(\theta_0, \cdot) \equiv \frac{1}{2}\right) \wedge (R(\theta_0) = 0)$.

Regularization-based pseudo-divergences

Theorem (on regularized family of pseudo-divergences)

If M is a parameterized model family: $M = \{f(\theta, \cdot) \mid \theta \in \Theta\}$ and $M = \mathcal{F}$, $R : \Theta \to \mathbb{R}$ is a proper regularizer for M, and $c : [0, 1] \to [0, +\infty)$ is a strictly increasing function such, that $c(0) = 0$, then the family $\mathcal{D} = \{D_\alpha : \Pi(\mathcal{X}) \times \Pi(\mathcal{X}) \to \mathbb{R} \mid \alpha \in [0, 1]\}$:

$$
\begin{aligned}
D_\alpha(P, Q) &= \log 2 - \min_{\theta \in \Theta_\alpha(P,Q)} L(f(\theta, \cdot), P, Q);\\
\Theta_\alpha(P, Q) &= \operatorname*{Arg\,min}_{\theta \in \Theta} L^R_\alpha(\theta, P, Q);\\
L^R_\alpha(\theta, P, Q) &= L(f(\theta, \cdot), P, Q) + c(1 - \alpha) R(\theta);
\end{aligned}
$$

is a complete and ordered with respect to Jensen-Shannon divergence family of pseudo-divergences.

Boosted family

A boosting-based method is applicable for a discrete approximation:

$$\begin{aligned} D_{c(i)}(P, Q) &= \log 2 - L(F_i, P, Q); \\ F_i &= F_{i-1} + \rho \cdot \operatorname*{arg\,min}_{f \in B} L(F_{i-1} + f, P, Q); \\ F_0 &\equiv \frac{1}{2}; \end{aligned}$$

where:

- ρ — learning rate,
- B — base estimator,
- $c : \mathbb{Z}_+ \to [0, 1]$ — a strictly increasing function for mapping ensemble size onto $\alpha \in [0, 1]$.

Boosted Adaptive Divergence

Algorithm 2 Boosted adaptive divergence

Require: X_P, X_Q — samples from distributions P and Q, B — base estimator training algorithm, N — maximal size of the ensemble, $c : \mathbb{Z}_+ \to [0, 1]$ — capacity function; ρ — learning rate;

$F_0 \leftarrow 1/2$
$i \leftarrow 0$
$L_0 \leftarrow \log 2$
for $i = 1, \ldots, N$ **do**
  if $L_i > c(i)\log 2$ **then**
    $F_{i+1} \leftarrow F_i + \rho \cdot B(F_i, X_P, X_Q)$
    $L_{i+1} \leftarrow L(F_{i+1}, X_P, X_Q)$
    $i \leftarrow i + 1$
  else
    return $\log 2 - L_i$
  end if
end for
return $\log 2 - L_N$

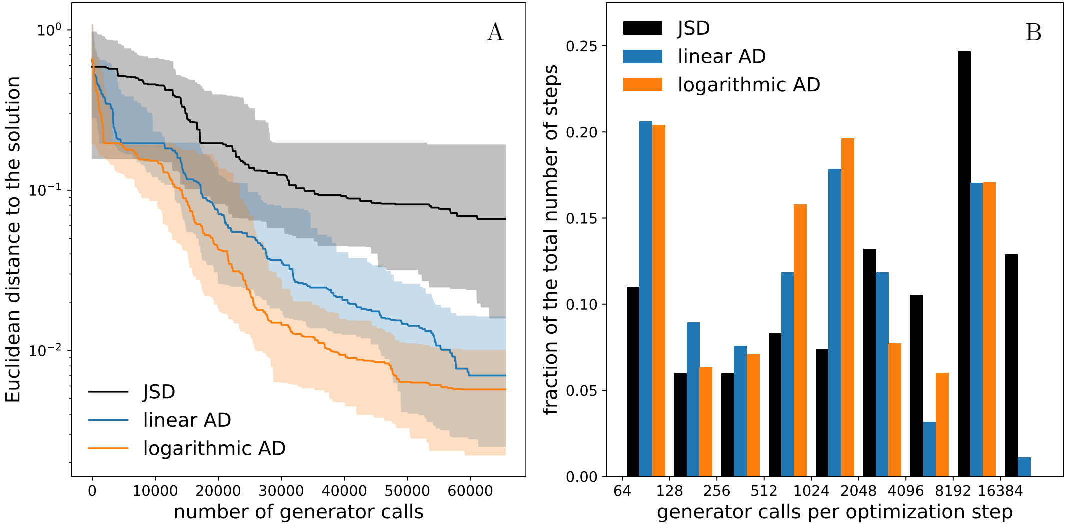
Explicitly Regularized Adaptive Divergence

Algorithm 4 Adaptive divergence estimation by a regularized neural network

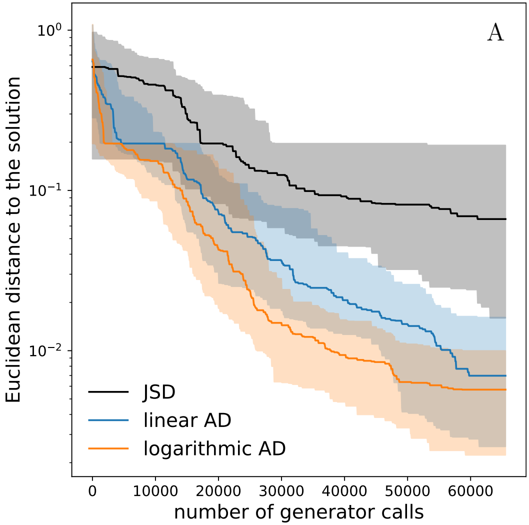
Require: X_P, X_Q — samples from distributions P and Q;
$f_\theta : \mathcal{X} \to \mathbb{R}$ — neural network with parameters $\theta \in \Theta$;
$R : \Theta \to \mathbb{R}$ — regularization function; c — capacity function;
ρ — exponential average coefficient;
β — coefficient for R_1 regularization;
γ — learning rate of SGD.

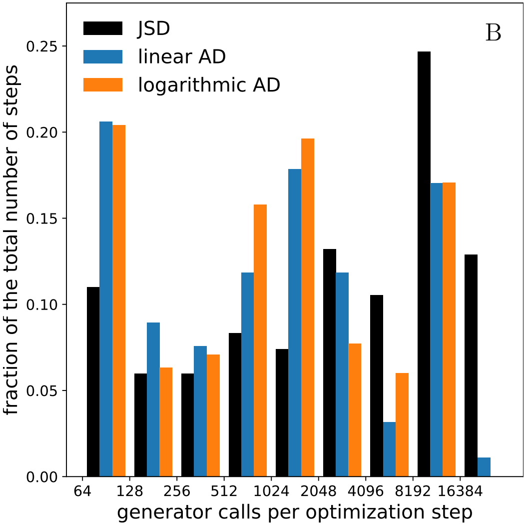
$L_{\text{acc}} \leftarrow \log 2$
while not converged **do**
  $x_P \leftarrow \text{sample}(X_P)$
  $x_Q \leftarrow \text{sample}(X_Q)$
  $\zeta \leftarrow c\left(1 - \frac{L_{\text{acc}}}{\log 2}\right)$
  $g_0 \leftarrow \nabla_\theta \left[L(f_\theta, x_P, x_Q) + \zeta \cdot R(f_\theta)\right]$
  $g_1 \leftarrow \nabla_\theta \|\nabla_\theta f_\theta(x_P)\|^2$
  $L_{\text{acc}} \leftarrow \rho \cdot L_{\text{acc}} + (1 - \rho) \cdot L(f_\theta, x_P, x_Q)$
  $\theta \leftarrow \theta - \gamma \left(g_0 + \beta g_1\right)$
end while
return $\log 2 - L(f_\theta, X_P, X_Q)$

XOR-like synthetic data

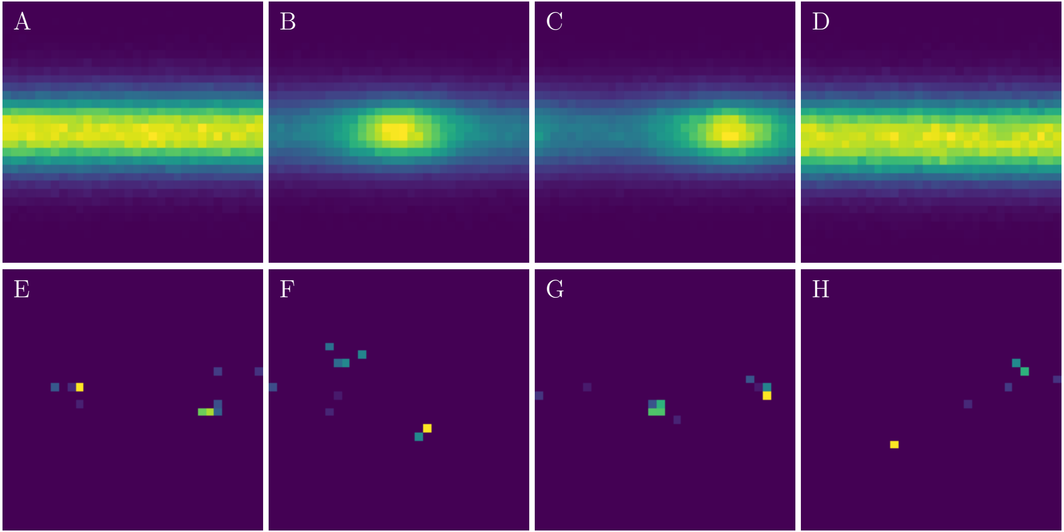


Pythia tuning

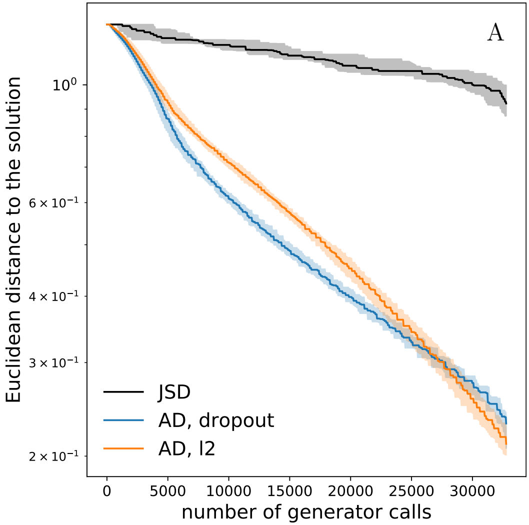


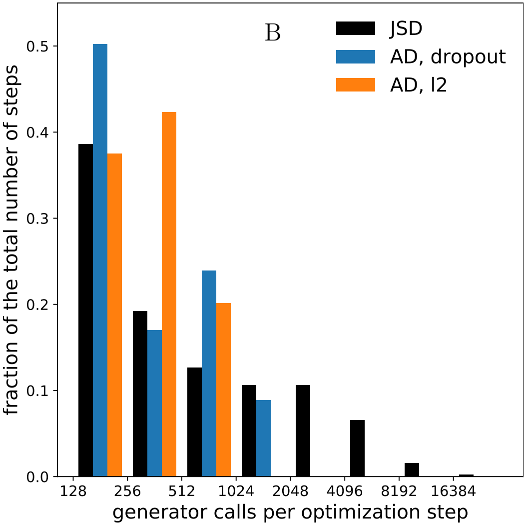


Pythia alignment



Pythia alignment





Source of acceleration