Friday, March 26, 2021 3:14 PM

Lecture 17

Methods for decorrelated classifiers

Modify training data
- remove mass from features X
- planing — reweight signal & backgd. mass dist'ns to look same
  train with weighted events.
  $$L = \sum_{i \in sig} w_i^{(sig)} \log y(x_i) + \sum_{i \in bg} w_i^{(bg)} \log(1 - y(x_i))$$

Modify training procedure

→ 2 approaches that are SOTA
- Adversarial decorrelation
  - Louppe, Kagan, Cranmer 1611.01046
  - Shimmin et al 1703.03507
- Distance Decorrelation — Kasieczka & DS 2001.05310

Friday, March 26, 2021 3:44 PM

Adversarial Decorrelation

"adversary"

Idea: train 2nd NN that tries to predict m from $f_{cl}(x)$

if it succeeds $\rightarrow$ m & f_{cl} are not st. indep.

fails $\rightarrow$ " are st. indep.

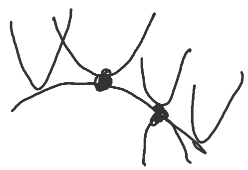
example of adversarial training

want: adversary to be as good as it can be and still fail

loss: $L = L_{cl}[f_{cl}] - \lambda L_{adv}[f_A(f_{cl}), m] \Big|_{bg}$

e.g. MSE $(f_A(f_{cl}) - m)^2$

goal: $\min\limits_{f_{cl}} \max\limits_{adv} L$



min-max objective!
nonconvex $\longleftrightarrow$ saddle pt.

Friday, March 26, 2021 3:54 PM

Pros: works well when it works

Cons: training is challenging

unstable

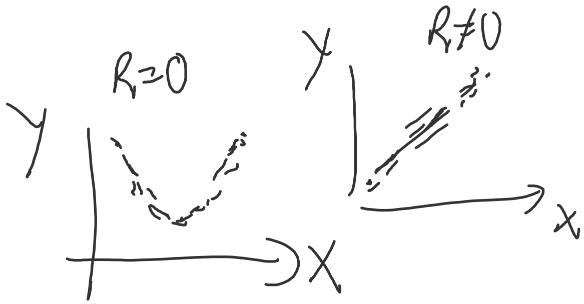
requires a lot tuning of learning rates, #epochs, ...

Alternative – what if we could add a regularizer term to loss which promotes decorrelation?

Want: $C(f_{cl}, m) \geq 0$

$= 0$ iff f_{cl} & m are statistically independent.

$$L = L_{cl} + \lambda C(f_{cl}, m)$$



Pearson correlation coeff. $R^{(X,Y)} = \langle (X-\bar{X})(Y-\bar{Y}) \rangle$ ...

only measures linear correlations!

Friday, March 26, 2021 3:59 PM

Statisticians have invented a measure of statistical dependence

Szekely & Rizzo et al (2007), ...

"distance correlation"

$$dCov^2(X,Y) = \langle |\vec{X}-\vec{X}'||\vec{Y}-\vec{Y}'| \rangle + \langle |\vec{X}-\vec{X}'| \rangle \langle |\vec{Y}-\vec{Y}'| \rangle - 2\langle |\vec{X}-\vec{X}'||\vec{Y}-\vec{Y}''| \rangle$$

(X,Y), (X',Y'), (X'',Y'') are iid from joint distribution $P(X,Y)$.

$$\boxed{dCorr^2(X,Y) = \frac{dCov^2(X,Y)}{dCov(X,X)\,dCov(Y,Y)}}$$

$0 \le dCorr \le 1$.

$= 0$ iff $P(X,Y) = P(X)P(Y)$

i.e. X & Y are st. indep.

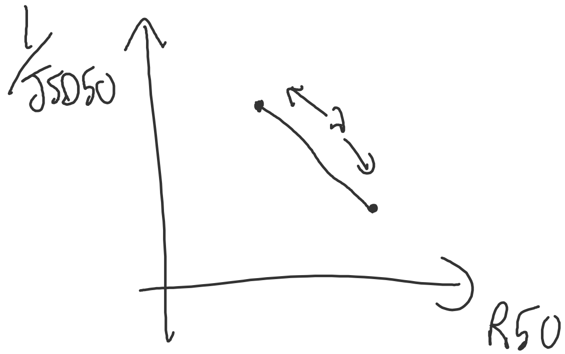
otherwise > 0.

use in loss term (dimensionless).

"Disco Decorrelation" $L = L_{cl}(f_{cl}) + \lambda\, dCorr^2(m, f_{cl})\Big|_{bg}$.

Friday, March 26, 2021 4:09 PM

Both adv. & disco decorr. have a tunable parameter λ.
→ λ large decorr > perf.
→ λ small perf > decorr



Friday, March 26, 2021 4:26 PM

Generative Modeling

- Given some data $\{x_i\}$: goal is to learn (explicitly or implicitly) prob density $p(x)$ that data was drawn from s.t. can sample from it and generate new examples.

3 major approaches:

- Generative Adversarial Networks (GANs)
- Variational Autoencoders (VAEs)
- Density Estimation

learn map from noise → data

$z \sim N(0,1)$ or uniform $\rightarrow G(z)$ ← data.

Friday, March 26, 2021 4:34 PM

For HEP, Generative Modeling has many applications

In general goal is to speed up simulations. → simulate data once and for all, train GAN, etc on it to learn dist'n, generate more examples

suppose simulations are expensive, time consuming

or train on data and bypass simulations altogether

Friday, March 26, 2021 4:38 PM

GANs

Idea: adversarial training but generator instead of classifier

train 2 NNs: G generator → $\overset{\text{noise}}{z} \longrightarrow \overset{\text{data}}{x}$

D critic / adversary / discriminator → classifier generated samples vs. data.

goal GAN: generator can fool the discriminator and discriminator is as good as it can be.

GAN loss: $L = \sum_{i \in \text{data}} \log D(x_i) + \sum_{i \in \text{gen}} \log(1 - D(G(z_i)))$

$\boxed{\min_G \max_D L}$

usual BCE (but keep in mind, usually use −BCE for classification)