Lecture 13 Distribution Shift

IEMS 402 Statistical Learning

Northwestern

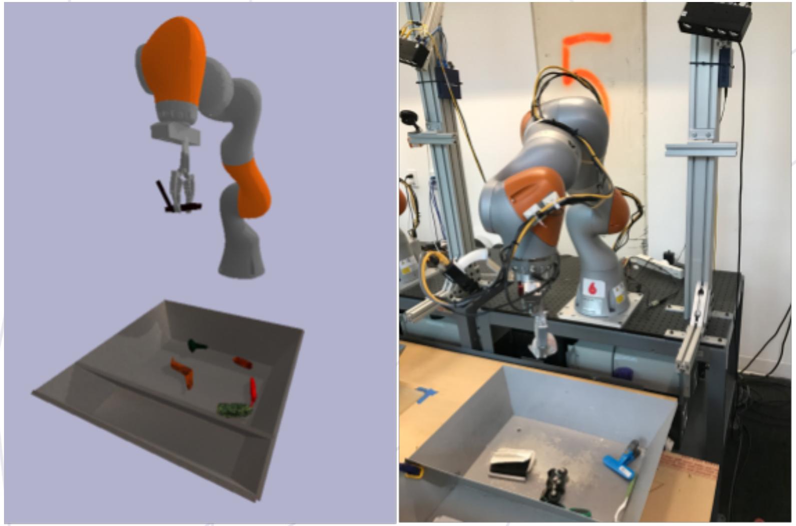
References

https://hsnamkoong.github.io/assets/html/b9145/index.html

Distribution Shift

Reconsider the ML Theory…

However…

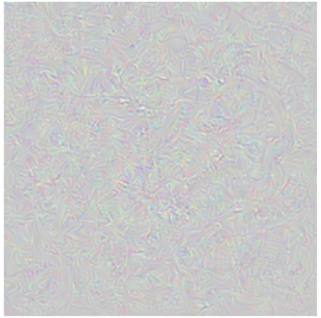


抖音

TikTok

Elephant or Cat

Shortcut learning



Article: Super Bowl 50

Paragraph: "*Peython Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had a jersey number 37 in Champ Bowl XXXIV.*"

Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"

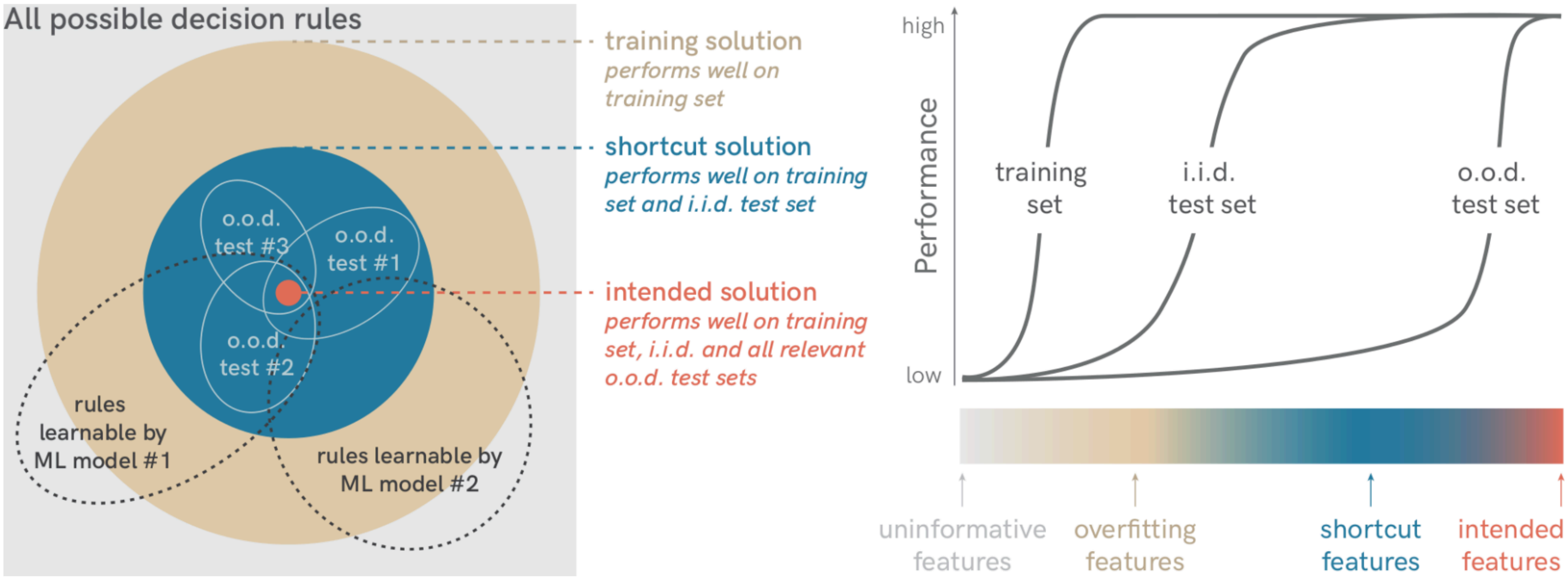
Original Prediction: John Elway

Prediction under adversary: Jeff Dean

Task for DNN	Caption image	Recognise object	Recognise pneumonia	Answer question
Problem	Describes green hillside as grazing sheep	Hallucinates teapot if certain patterns are present	Fails on scans from new hospitals	Changes answer if irrelevant information is added
Shortcut	Uses background to recognise primary object	Uses features irrecognisable to humans	Looks at hospital token, not lung	Only looks at last sentence and ignores context

spurious correlation

From i.i.d to o.o.d



Importance Weighting

Importance Weighting

How do we deal with covariate / label shifts?

What we have

$E_{p_{train}}[\ell(z;\theta)]$

What we want

$E_{p_{test}}[\ell(z;\theta)]$

Most basic approach: reweight the loss

$$\underbrace{E_{p_{train}}\left[\frac{p_{test}(z)}{p_{train}(z)}\ell(z;\theta)\right]}_{\text{Weighted loss over the training distribution}} = E_{p_{test}}[\ell(z;\theta)]$$

(also possible: resample the dataset)

Importance weighting

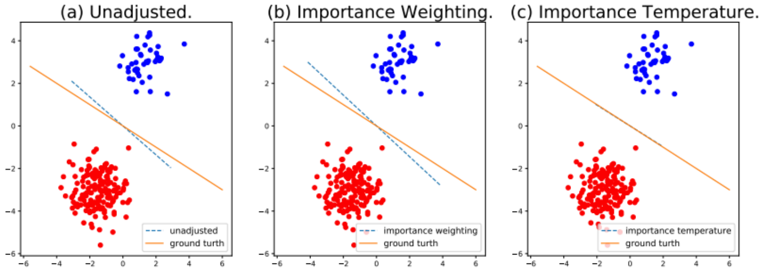
An alternative algorithm: use a classifier that separates p_{train} and p_{test}

1. Estimate a classifier $f(x) \approx \frac{p_{train}(x)}{p_{test(x)}+p_{train}(x)}$
2. Reweight by $h(x) = \frac{1}{f(x)} - 1$
3. Fit a model by minimizing the loss $h(x)\ell(x, y; \theta)$

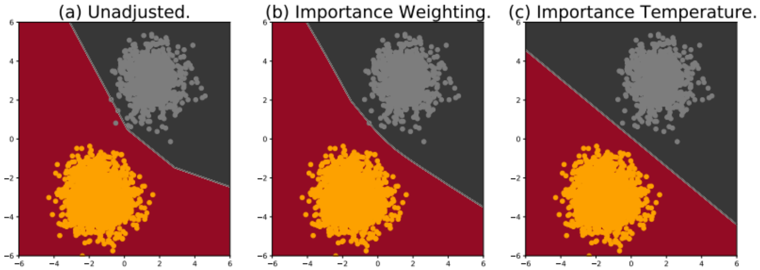
Discriminative Learning for Differing
Training and Test Distributions

Steffen Bickel BICKEL@MPI-INF.MPG.DE
Michael Brückner BRUM@MPI-INF.MPG.DE
Tobias Scheffer SCHEFFER@MPI-INF.MPG.DE
Max Planck Institute for Computer Science, Saarbrücken, Germany

Not Working for Over-parameterized Model



(a) Linear Model for Separable Data



(b) Multilayer Perceptron with two hidden layers of size 200

Byrd J, Lipton Z. What is the effect of importance weighting in deep learning? International conference on machine learning. PMLR, 2019: 872-881.

IPM

Background material: integral probability measures

To state this clearly, we need to first go into some background.

Definition (IPM):

For two probability distributions p and q, the integral probability metric (IPM) for a family of functions $\mathcal{F}$ is defined as

$$d_{\mathcal{F}}(p,q) = \sup_{f \in \mathcal{F}} |E_p[f(x)] - E_q[f(x)]|$$

Intuition: $\mathcal{F}$ are 'test functions' that can distinguish p and q

If two have the same function value for all $\mathcal{F}$, then they are similar

IPM and distribution shift

What we want | What we have | Domain distance

$$E_{p_{test}}[\ell(x,y,\theta)] = E_{p_{train}}[\ell(x,y,\theta)] + \Delta$$

From the trivial restatement

$$\Delta = E_{p_{test}}[\ell(x,y,\theta)] - E_{p_{train}}[\ell(x,y,\theta)]$$

This looks like an IPM! (if $\ell(x,y,\theta) \in \mathcal{F}$ for all θ)

$$\Delta \leq \sup_{f \in \mathcal{F}} E_{p_{test}}[f(x,y)] - E_{p_{train}}[f(x,y)] = d_{\mathcal{F}}(p_{train}, p_{test})$$

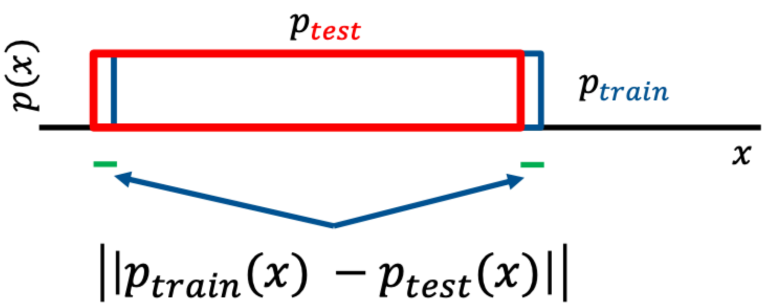
Takeaway: IPMs bound excess error under transfer

Example: L1 distance

We can now bound test performance in terms of IPMs

For $0 \leq \ell(x, y, \theta) \leq 1$ and under covariate shift,

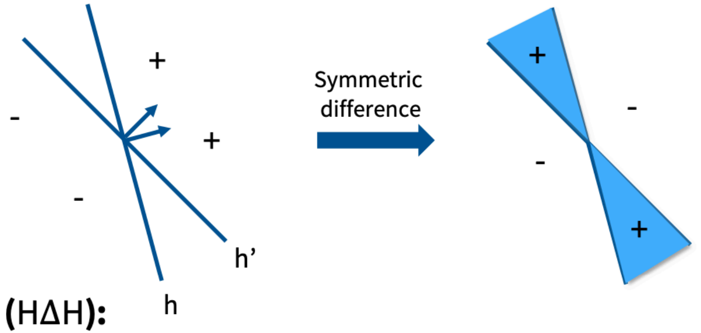
$$E_{p_{test}}[\ell(x, y, \theta)] \leq E_{p_{train}}[\ell(x, y, \theta)] + ||p_{train}(x) - p_{test}(x)||_1$$



	Reweighting	IPM
Goals	Correct train-test mismatch	Estimate train-test mismatch
Assumptions	Overlap	Boundedness
Training	Weighted/modified loss	No change
Costs	More samples (variance)	Inaccurate models (bias) Curse of dimensionality (next lecture)

Defining HΔH

For a hypothesis class $\mathcal{H}$, the HΔH set is defined as the symmetric difference



Definition (HΔH):

For a hypothesis class $\mathcal{H}$, the symmetric difference set $H\Delta H$ is defined as

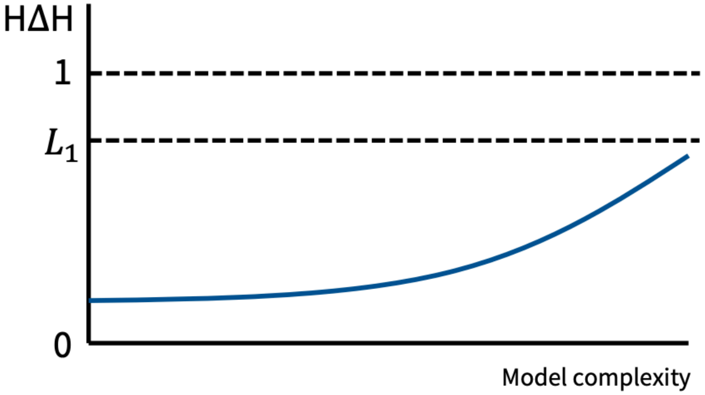
$$\text{H}\Delta\text{H} \coloneqq \{g: g(x) = \text{XOR}\big(h(x), h'(x)\big) \text{ and } h, h' \in \mathcal{H}\}$$

Dependency on Hypothesis Space

For a hypothesis class $\mathcal{H}$, the HΔH-divergence is

$$d_{H\Delta H}(p_{train}, p_{test}) = 2 \sup_{g \in H\Delta H} \left| E_{p_{train}}[g(x)] - E_{p_{test}}[g(x)] \right|$$

HΔH: $\frac{1}{2} d_{H\Delta H}(p_{train}, p_{test})$



$d_{H\Delta H}$ is upper bounded by the L_1 distance

$d_{H\Delta H}$ increases monotonically with model complexity. If $H \subset H'$,

$$d_{H\Delta H} \leq d_{H'\Delta H'}$$

Another trade-off

Let's walk through the main bound.

$$E_{p_{test}}[\ell(x,y,h)]$$
$$\leq E_{p_{train}}[\ell(x,y,h)] + \frac{1}{2} d_{H\Delta H}(p_{train}, p_{test}) + \lambda$$

Different answer on two domains

same answer but
Both are wrong

Training domain error

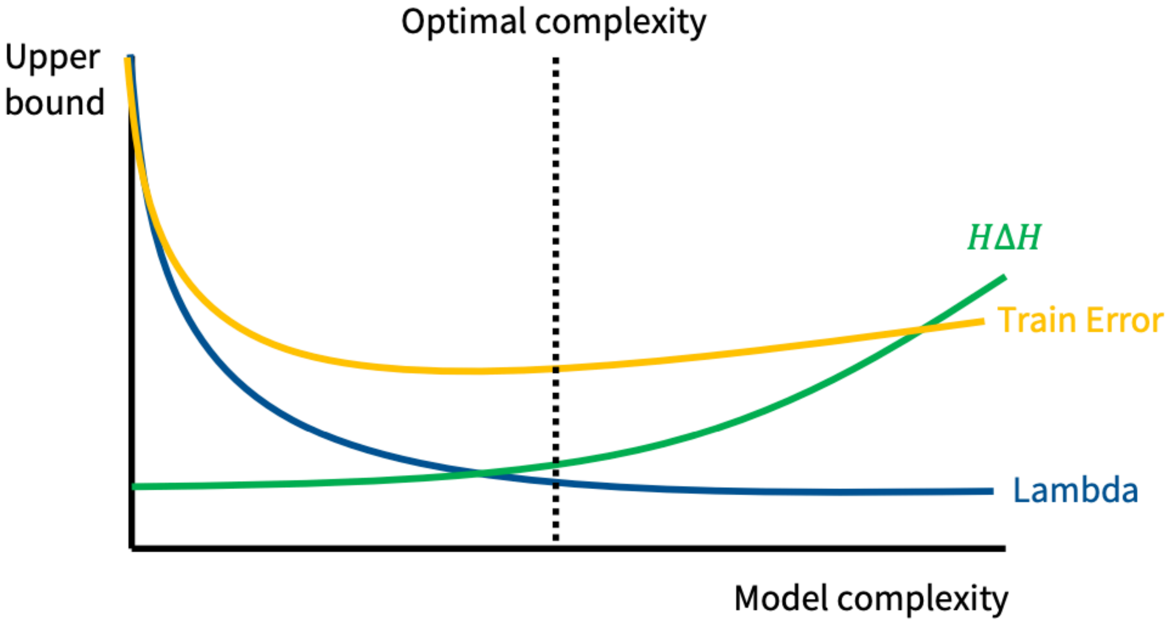
Domain distinguishability

Minimal error of a classifier on both domains

$$\lambda = \inf_{h \in \mathcal{H}} p_{train}\big(y \neq h(x)\big) + p_{test}(y \neq h(x))$$

HΔH claim: Low training domain error + low $H\Delta H$ divergence + rich $\mathcal{H}$
= good generalization to target domain

Another tradeoff



Distributionally Robust Optimization

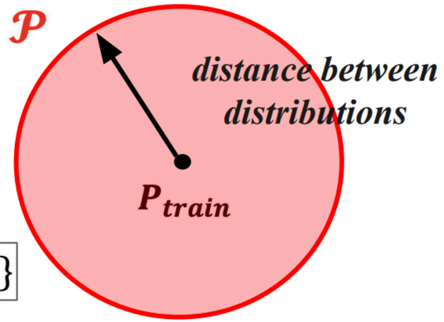
F-divergence

Distributionally Robust Optimization

Empirical Risk Minimization $\min_{\theta \in \Theta} \mathbb{E}_{Z \sim P_{train}}[\ell(\theta; Z)]$

DRO $\min_{\theta \in \Theta} \sup_{Q \in \mathcal{P}} \mathbb{E}_{Z \sim Q}[\ell(\theta; Z)]$

$$\boxed{\mathcal{P} = \{Q : Dist(Q, P_{train}) \leq \rho\}}$$



Instead of minimizing loss over training distribution, minimize loss over distributions ***near*** it

Generalization of DRO

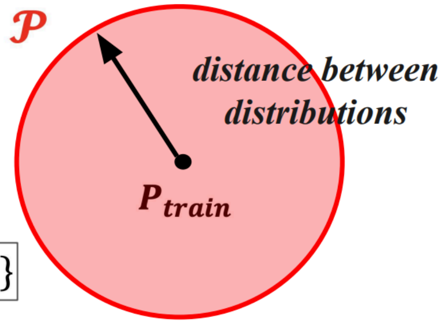
Empirical Risk Minimization

$$\min_{\theta \in \Theta} \mathbb{E}_{Z \sim P_{train}}[\ell(\theta; Z)]$$

DRO

$$\min_{\theta \in \Theta} \sup_{Q \in \mathcal{P}} \mathbb{E}_{Z \sim Q}[\ell(\theta; Z)]$$

$$\boxed{\mathcal{P} = \{Q : Dist(Q, P_{train}) \le \rho\}}$$



Instead of minimizing loss over training distribution, minimize loss over distributions ***near*** it

Duality of DRO

$$R_f(\theta; P) = \inf_{\lambda \geq 0, \eta \in \mathbb{R}} \left\{ \lambda \mathbb{E}_P \left[f^* \left(\frac{\ell(\theta; Z) - \eta}{\lambda} \right) \right] + \lambda \rho + \eta \right\}$$

$$f^*(s) := \sup_t \{st - f(t)\}.$$

$$= \sup_{L \geq 0} \inf_{\lambda \geq 0, \eta \in \mathbb{R}} \{\mathbb{E}_P[L(Z)\ell(\theta; Z)] + \lambda(\rho - \mathbb{E}_P[f(L(Z))] - \eta(\mathbb{E}_P[L(Z)] - 1)\}$$

Duality of DRO

$$R_f(\theta; P) = \inf_{\lambda \geq 0, \eta \in \mathbb{R}} \left\{ \lambda \mathbb{E}_P \left[f^* \left(\frac{\ell(\theta; Z) - \eta}{\lambda} \right) \right] + \lambda \rho + \eta \right\}$$

$$f^*(s) := \sup_t \{ st - f(t) \}.$$

$$= \sup_{L \geq 0} \inf_{\lambda \geq 0, \eta \in \mathbb{R}} \{ \mathbb{E}_P[L(Z)\ell(\theta; Z)] + \lambda(\rho - \mathbb{E}_P[f(L(Z))] - \eta(\mathbb{E}_P[L(Z)] - 1) \}$$

$$= \inf_{\lambda \geq 0, \eta \in \mathbb{R}} \sup_{L \geq 0} \left\{ \lambda \mathbb{E}_P \left[\frac{L(Z)(\ell(\theta; Z) - \eta)}{\lambda} - f(L(Z)) \right] \right\} + \lambda \rho + \eta.$$

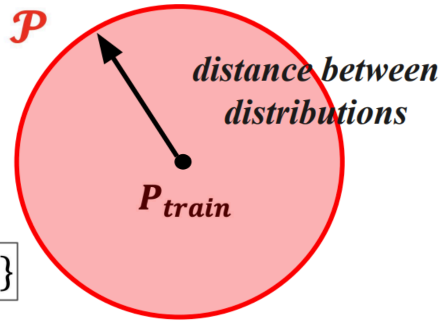
$$= \mathbb{E}_P \left[f^* \left(\frac{\ell(\theta; Z) - \eta}{\lambda} \right) \right].$$

Variance Regularization

Generalization of DRO

Empirical Risk Minimization	$\min_{\theta \in \Theta} \mathbb{E}_{Z \sim P_{train}}[\ell(\theta; Z)]$
DRO	$\min_{\theta \in \Theta} \sup_{Q \in \mathcal{P}} \mathbb{E}_{Z \sim Q}[\ell(\theta; Z)]$

$$\boxed{\mathcal{P} = \{Q : Dist(Q, P_{train}) \leq \rho\}}$$

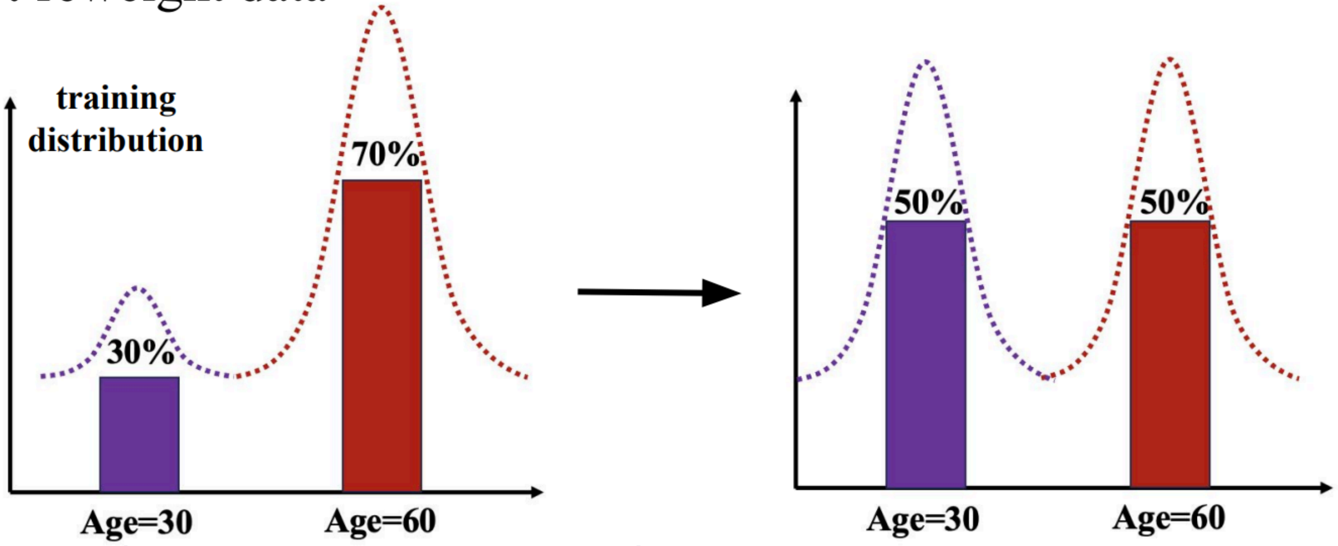


Instead of minimizing loss over training distribution, minimize loss over distributions ***near*** it

Is DRO Working?

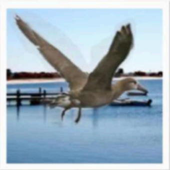
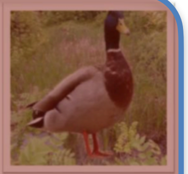
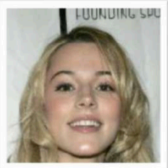
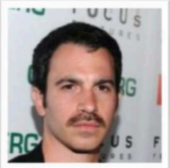
F-divergence DRO only reweighting

***f*-DRO:** reweight data

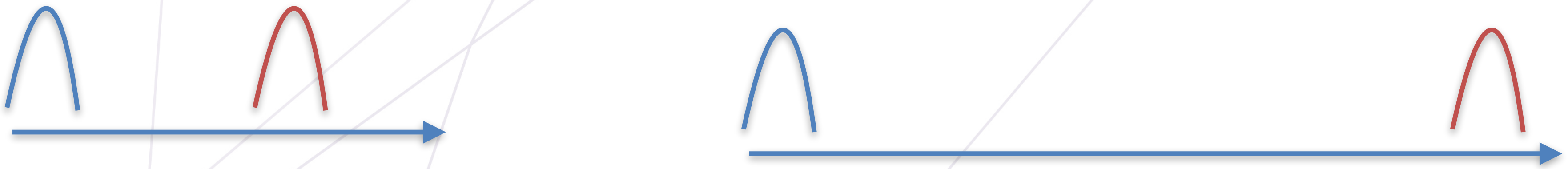


spurious correlation

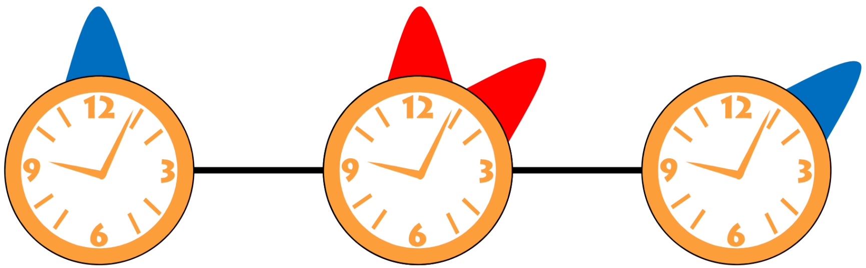
Weights more on rare data!

	Common training examples		Test examples
Waterbirds	y: waterbird a: water background 	y: landbird a: land background 	y: waterbird a: land background 
CelebA	y: blond hair a: female 	y: dark hair a: male 	y: blond hair a: male 
MultiNLI	y: contradiction a: has negation (P) The economy could be still better. (H) The economy has never been better.	y: entailment a: no negation (P) Read for Slate's take on Jackson's findings. (H) Slate had an opinion on Jackson's findings.	y: entailment a: has negation (P) There was silence for a moment. (H) There was a short period of time where no one spoke.

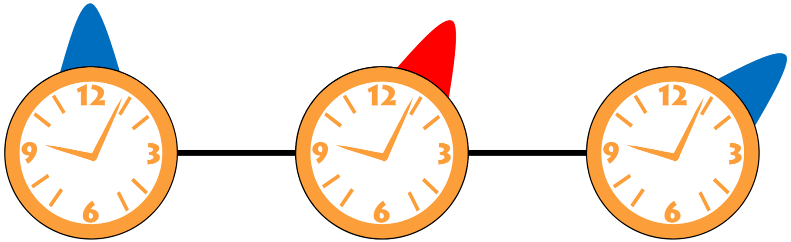
What’s wrong about f-divergence



What's wrong about f-divergence



Or

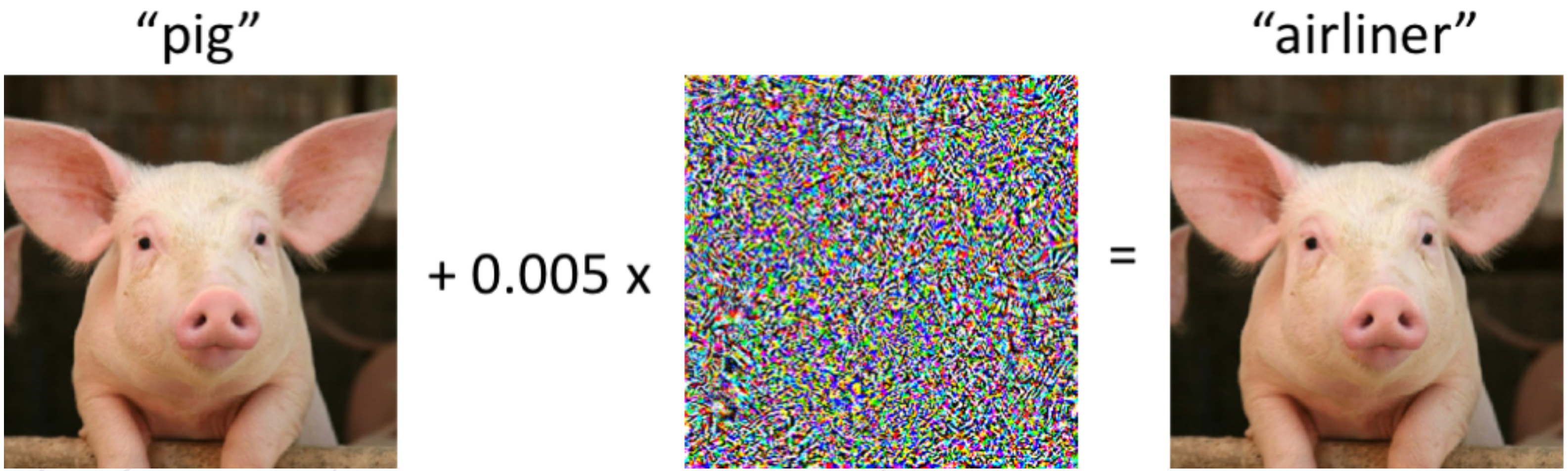


Over-parameterization?

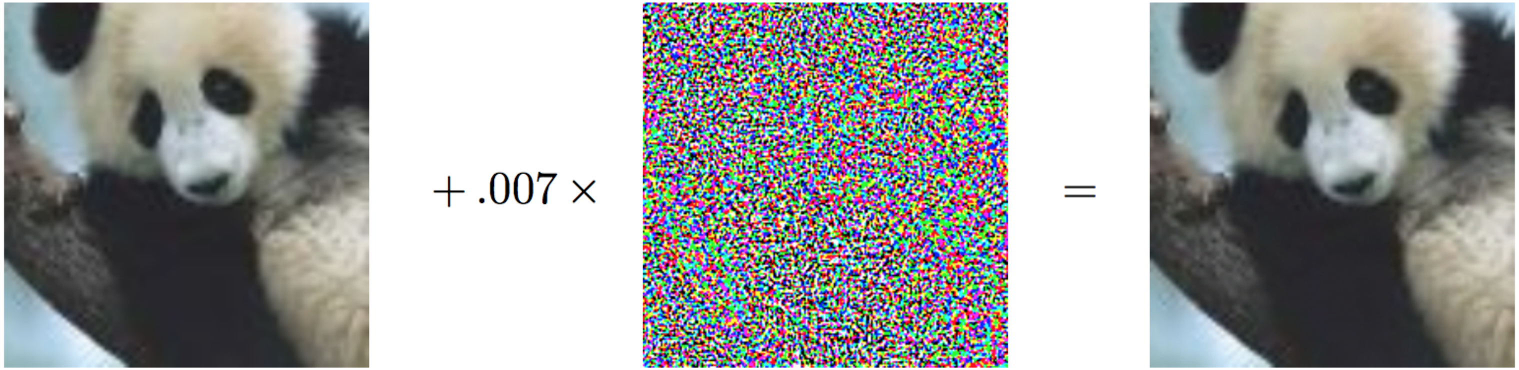
			Average Accuracy		Worst-Group Accuracy	
			ERM	DRO	ERM	DRO
Standard Regularization	Waterbirds	Train	100.0	100.0	100.0	100.0
		Test	97.3	97.4	60.0	76.9
	CelebA	Train	100.0	100.0	99.9	100.0
		Test	94.8	94.7	41.1	41.1
	MultiNLI	Train	99.9	99.3	99.9	99.0
		Test	82.5	82.0	65.7	66.4
Strong ℓ_2 Penalty	Waterbirds	Train	97.6	99.1	35.7	97.5
		Test	95.7	96.6	21.3	84.6
	CelebA	Train	95.7	95.0	40.4	93.4
		Test	95.8	93.5	37.8	86.7

Adversarial Learning

adversarial training



How to find Adversarial Examples?



$\boldsymbol{x}$	$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$	$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"panda"	"nematode"	"gibbon"
57.7% confidence	8.2% confidence	99.3 % confidence

Optimization that maximize the loss

Adversarial Training

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\delta\in\mathcal{S}} L(\theta, x+\delta, y)\right].$$

https://arxiv.org/pdf/1706.06083

Adversarial Training Can Hurt Generalization

	Standard training	Adversarial training
Robust test	3.5%	45.8%
Robust train	-	100%
Standard test	95.2%	87.3%
Standard train	100%	100%

Real World?

Lots of progress on ImageNet over the past 10 years, but models are still not robust.

Evaluation: **new test sets**

ImageNetV2

[Recht, Roelofs, Schmidt, Shankar '19]

ObjectNet

[Barbu, Mayo, Alverio, Luo, Wang, Gutfreund, Tenenbaum, Katz '19]

ImageNet-Sketch

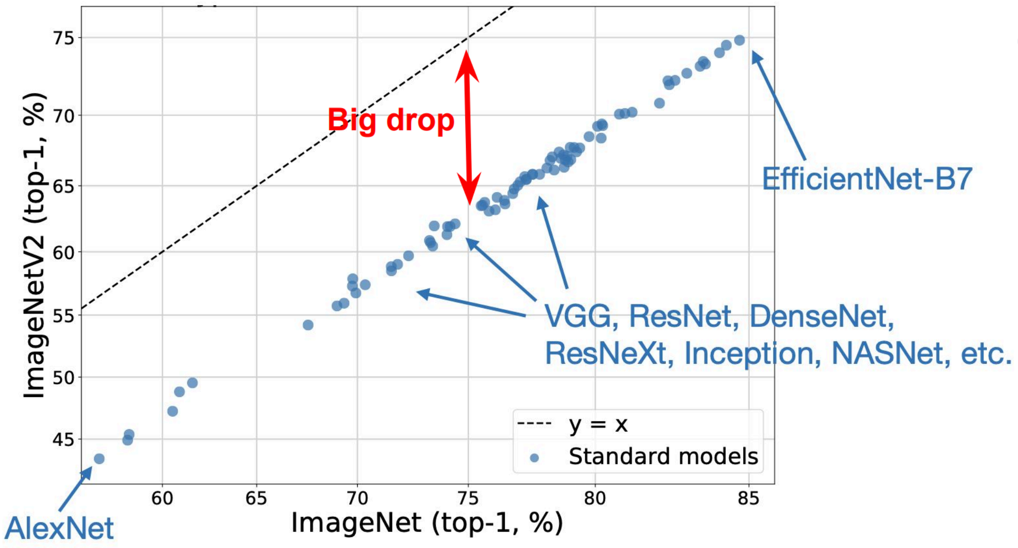
[Wang, Ge, Lipton, Xing '19]

ImageNet-R

[Hendrycks, Basart, Mu, Kadavath, Wang, Dorundo, Desai, Zhu, Parajuli, Guo, Song, Steinhardt, Gilmer '20]

Agree on the line!

Recht B, Roelofs R, Schmidt L, et al. Do imagenet classifiers generalize to imagenet?[C]// International conference on machine learning. PMLR, 2019: 5389-5400.



[Taori, Dave, Shankar, Carlini, Recht, Schmidt '20]

Why?