The Peculiar Optimization and Regularization Challenges

in Multi-Task Learning and Meta-Learning

Chelsea Finn

Stanford

training data

Braque

Cezanne

test datapoint

By Braque or Cezanne?

How did you accomplish this?

Through previous experience.

How might you get a machine to accomplish this task?

Modeling image formation

Geometry

SIFT features, HOG features + SVM

Fine-tuning from ImageNet features
Domain adaptation from other painters

???

Fewer human priors,
more data-driven priors

Greater success.

Can we explicitly learn priors from previous experience
that lead to efficient downstream learning?

Can we learn to learn?

Outline

1. Brief overview of meta-learning

2. A peculiar yet ubiquitous problem in meta-learning
(and how we might regularize it away)

3. Can we scale meta-learning to broad task distributions?

How does meta-learning work? An example.

Given 1 example of 5 classes:

training data $\mathcal{D}_{\text{train}}$

Classify new examples

test set $\mathbf{x}_{\text{test}}$

How does meta-learning work? An example.

meta-training

$\mathcal{T}_1$

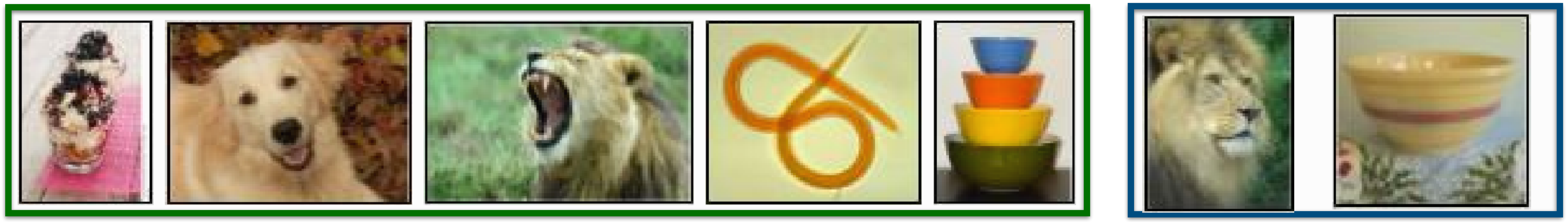
$\mathcal{T}_2$

training classes

meta-testing $\mathcal{T}_{\text{test}}$

Given 1 example of 5 classes:

Classify new examples



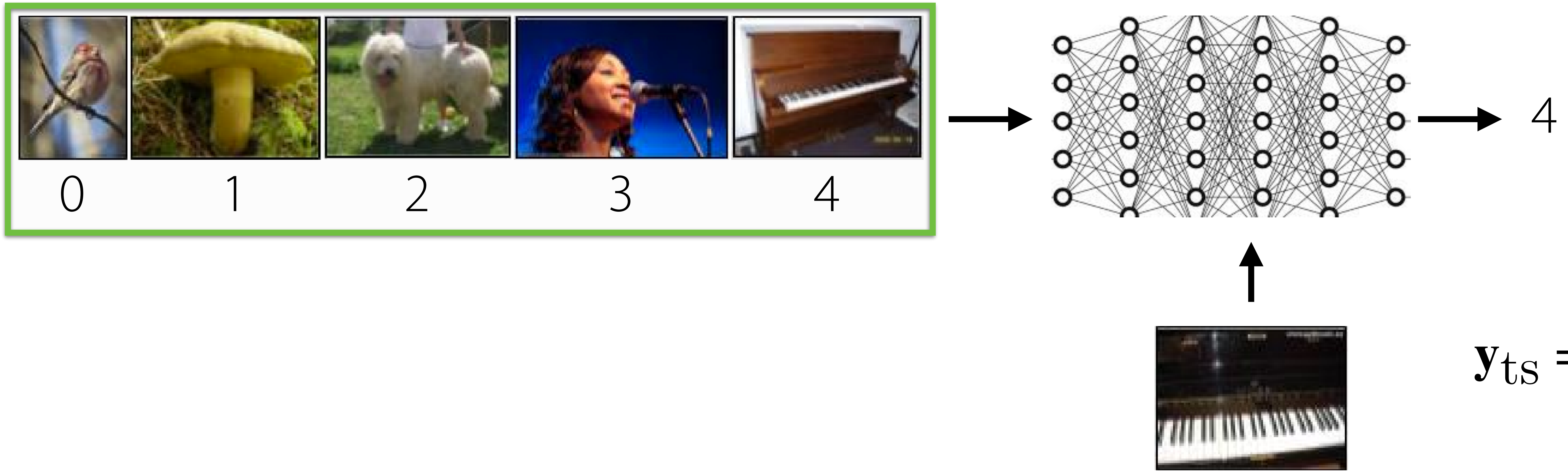
training data $\mathcal{D}_{\text{train}}$

test set $\mathbf{x}_{\text{test}}$

How does meta-learning work?

$\mathcal{T}_1$

One approach: parameterize learner by neural network



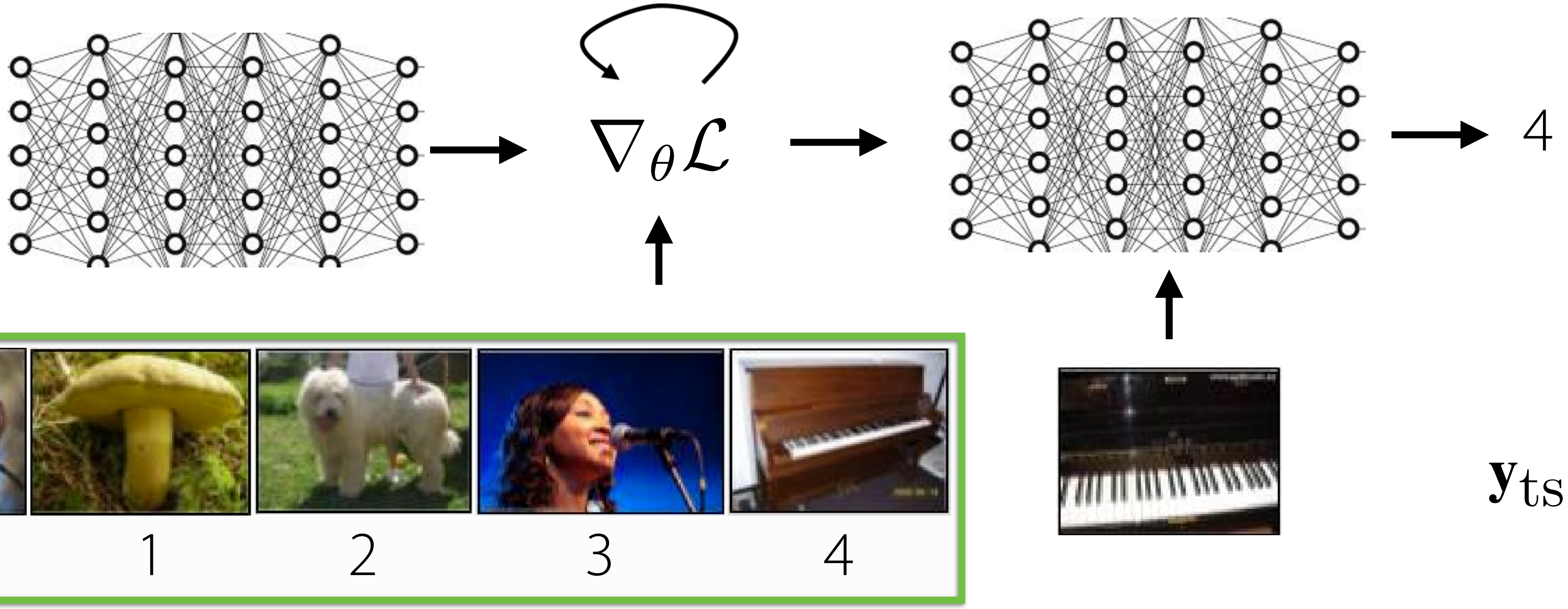
$$\mathbf{y}_{\text{ts}} = f(\mathscr{D}_{\text{tr}}, \mathbf{x}_{\text{ts}}; \theta)$$

(Hochreiter et al. '91, Santoro et al. '16, many others)

How does meta-learning work?

$\mathcal{T}_1$

Another approach: embed optimization inside the learning process

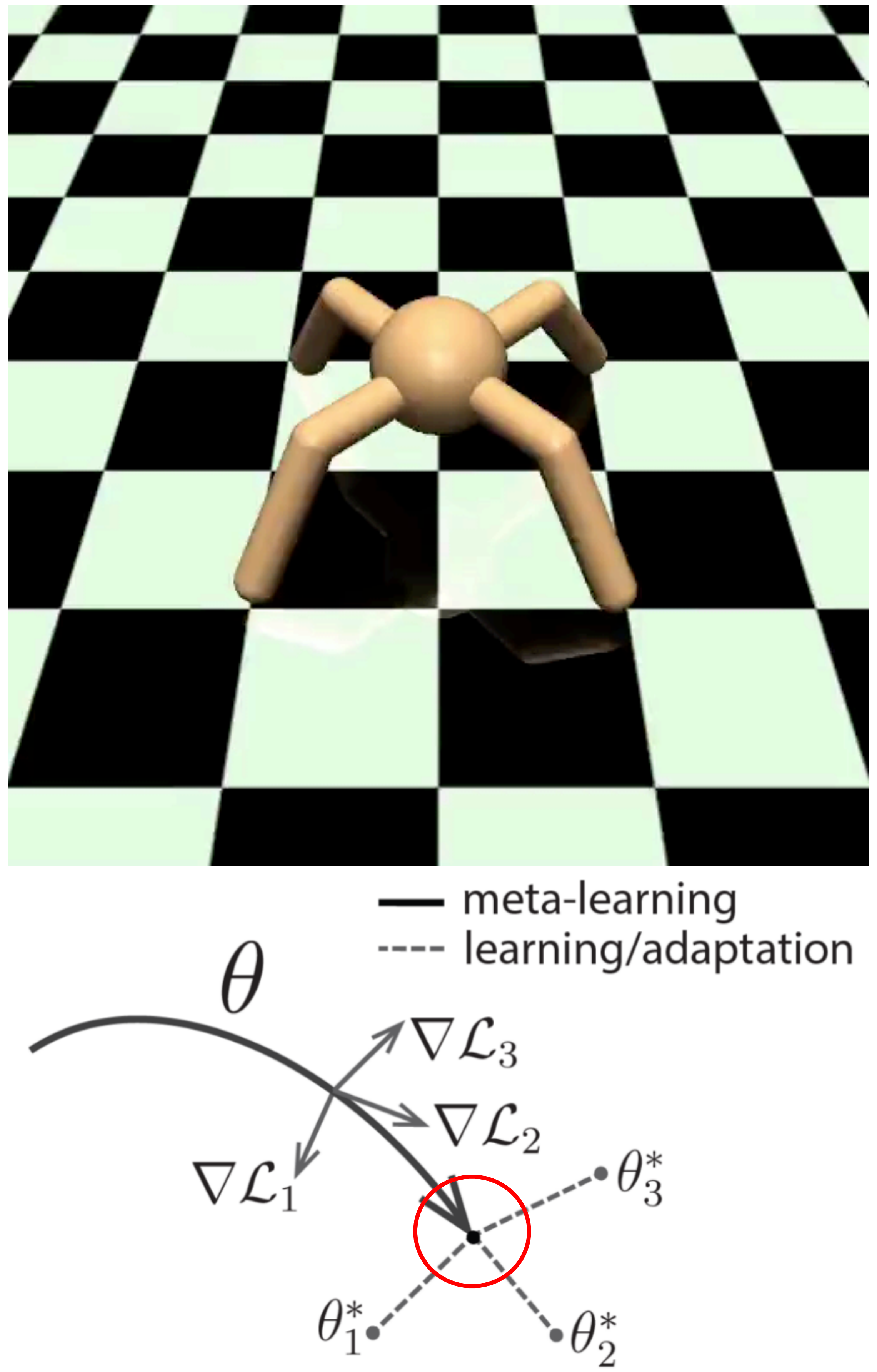


$$\mathbf{y}_{\text{ts}} = f(\mathcal{D}_{\text{tr}}, \mathbf{x}_{\text{ts}}; \theta)$$

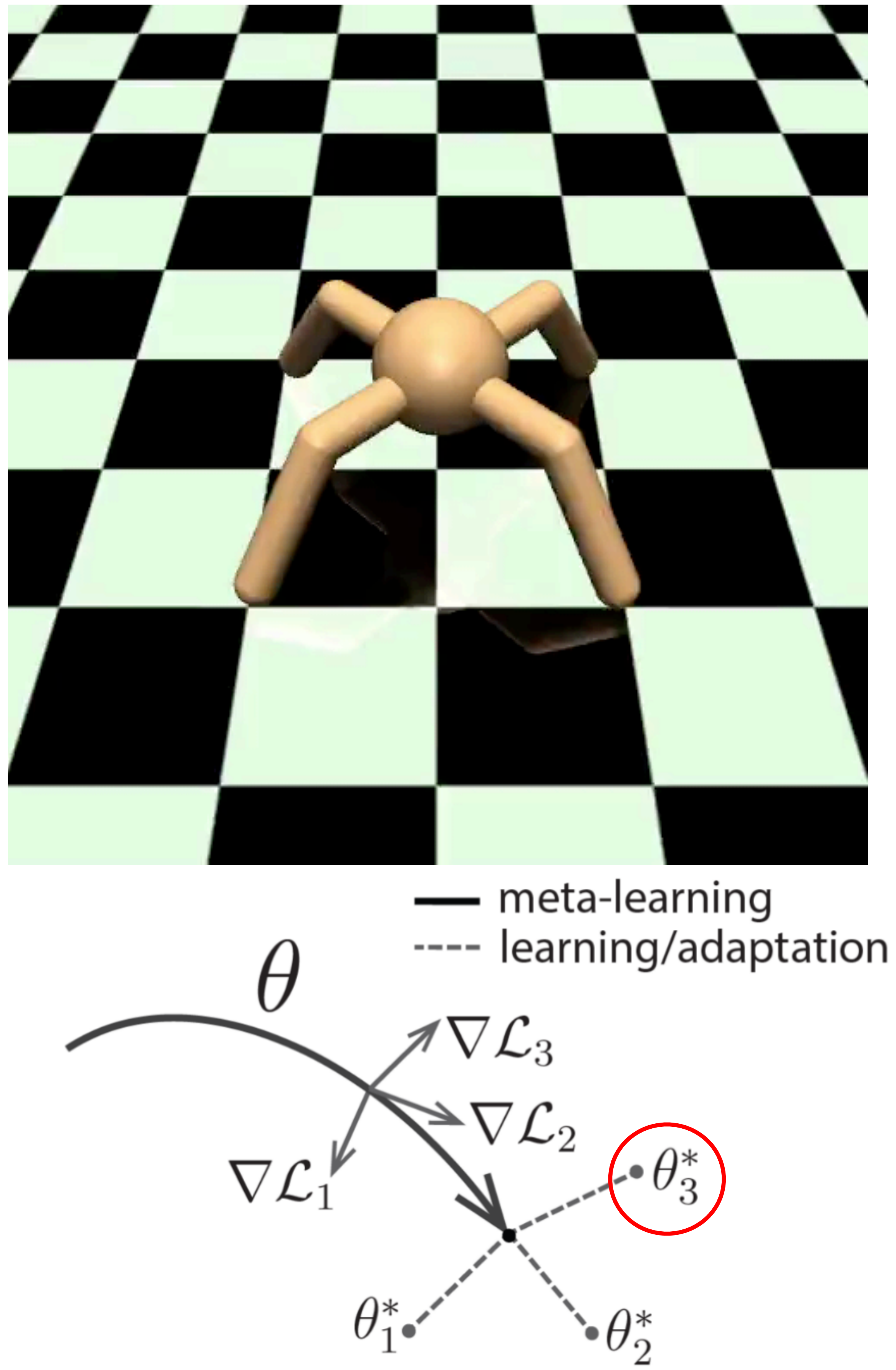
(Maclaurin et al. '15, Finn et al. '17, many others)

Can we learn a representation under which **RL** is fast and efficient?

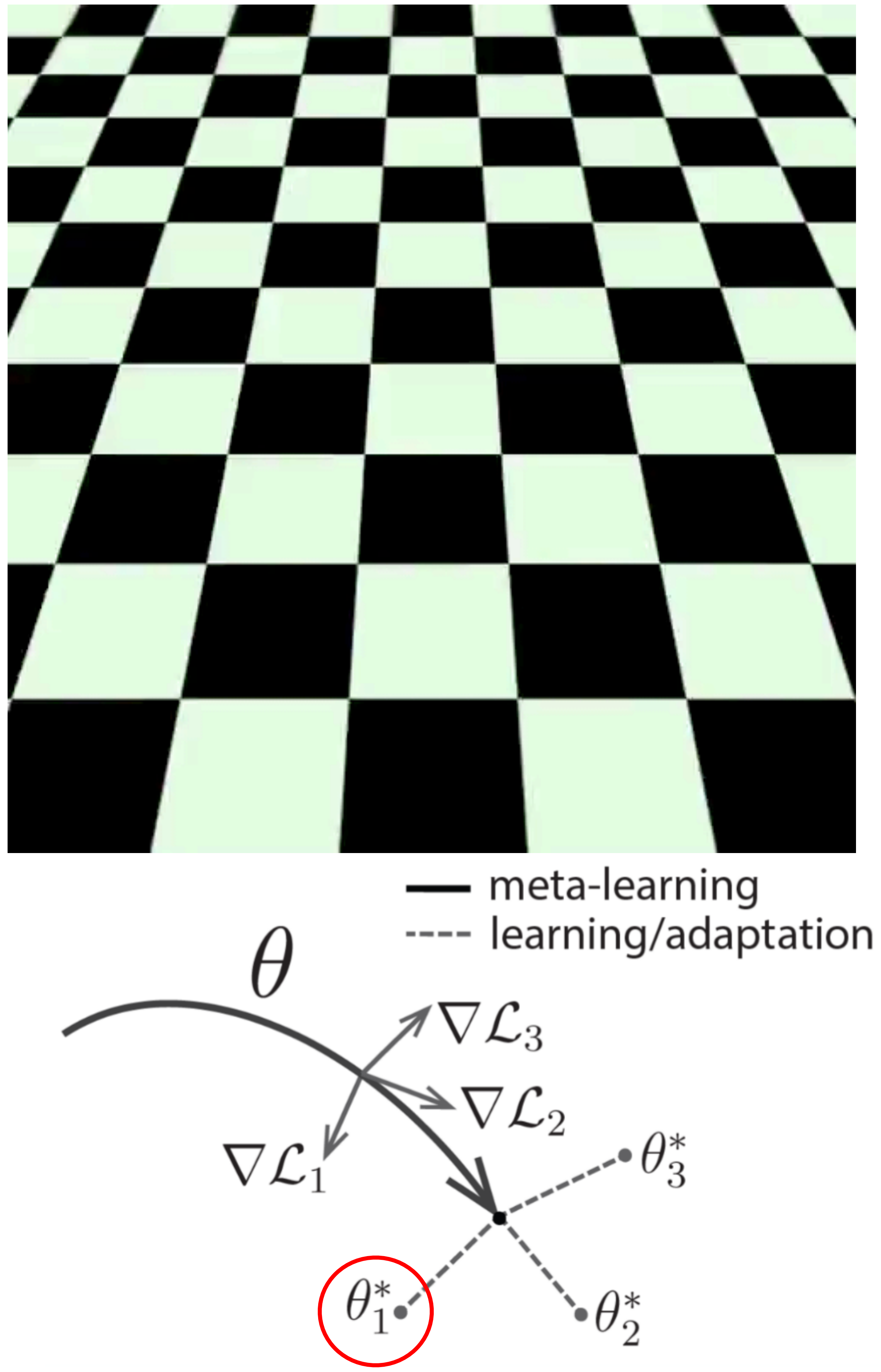
after MAML training



after 1 gradient step (forward reward)



after 1 gradient step (backward reward)



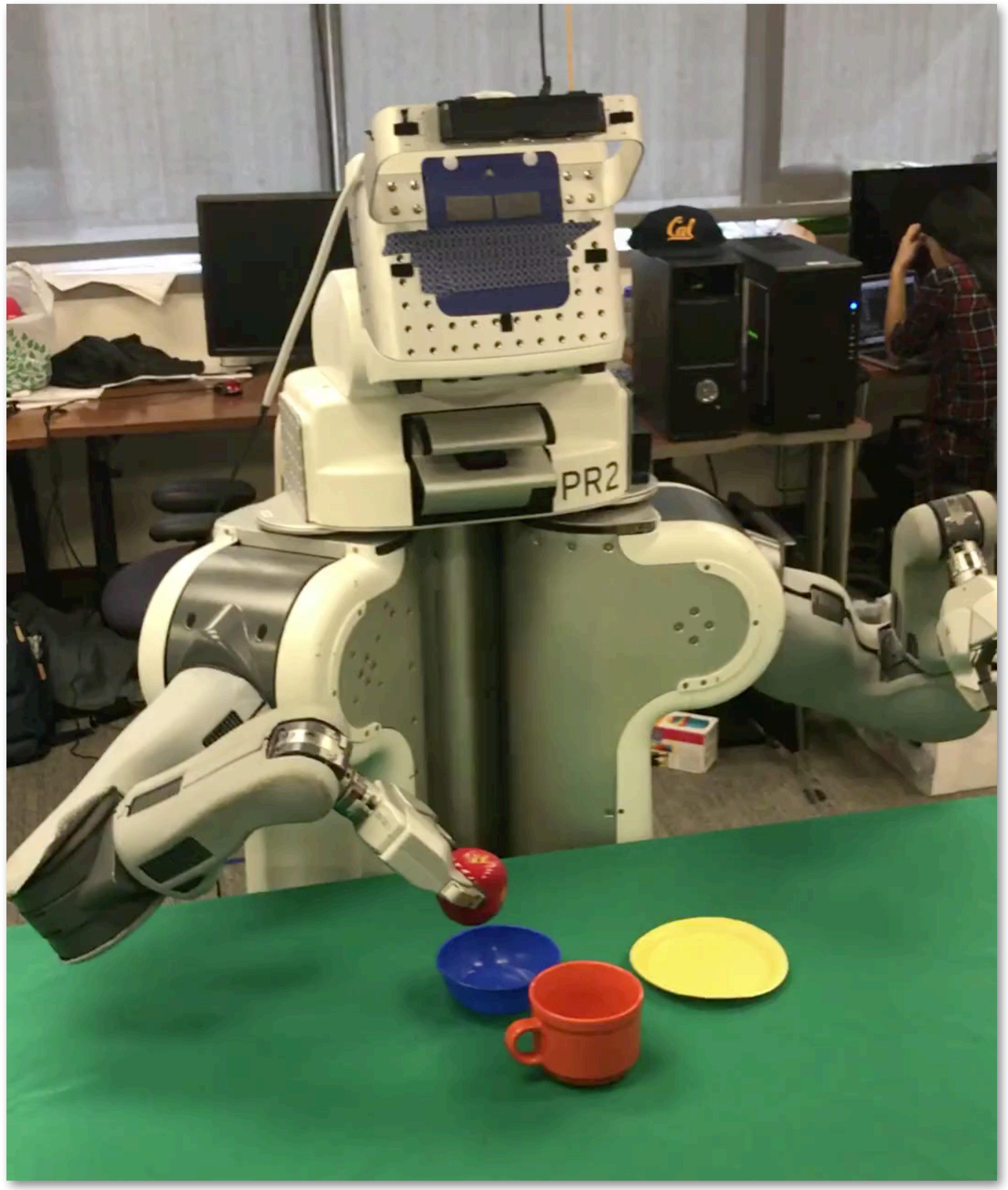
Finn, Abbeel, Levine. *Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks*. ICML '17

Can we learn a representation under which imitation is fast and efficient?

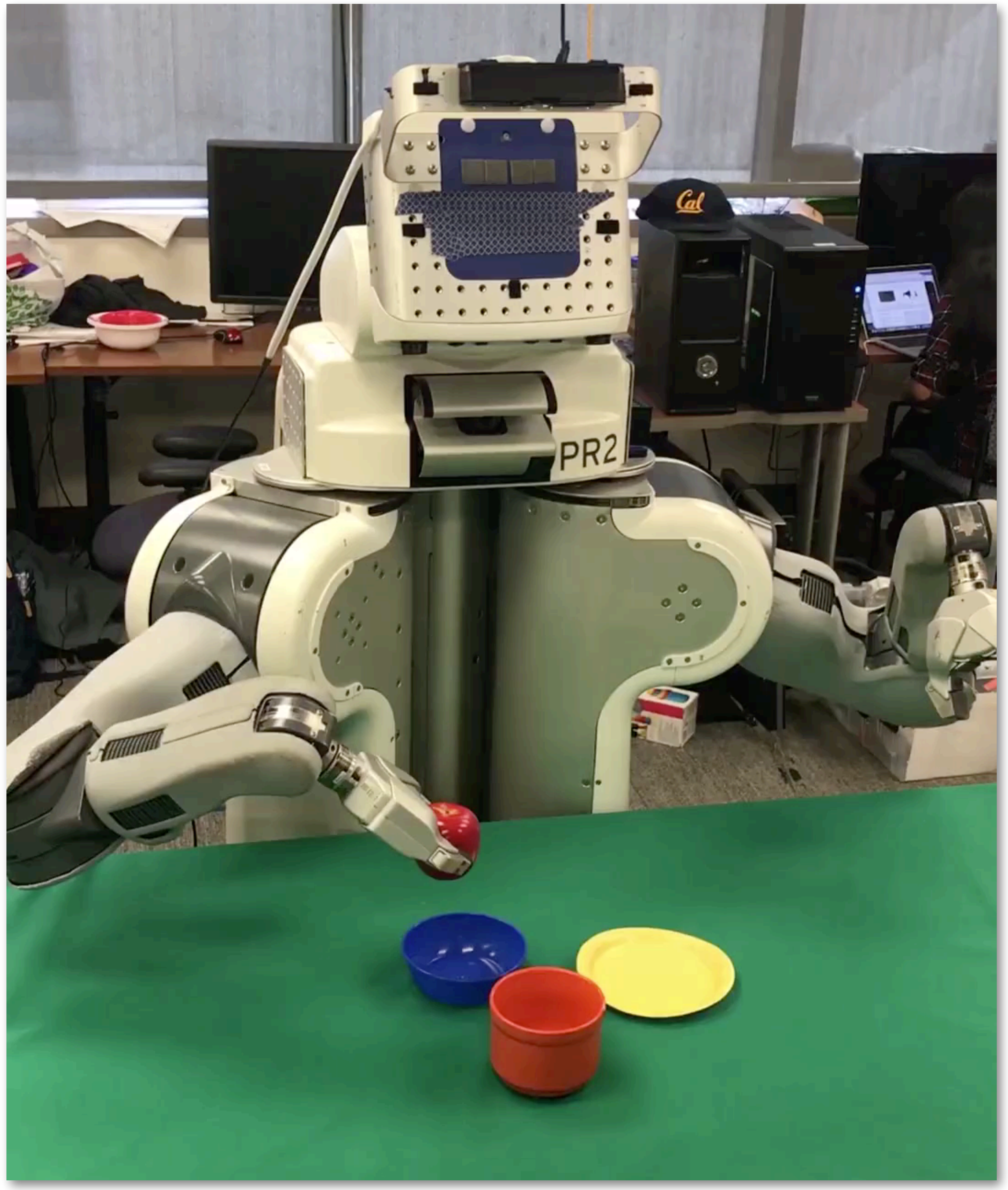
subset of training objects

held-out test objects

input demo
(via teleoperation)



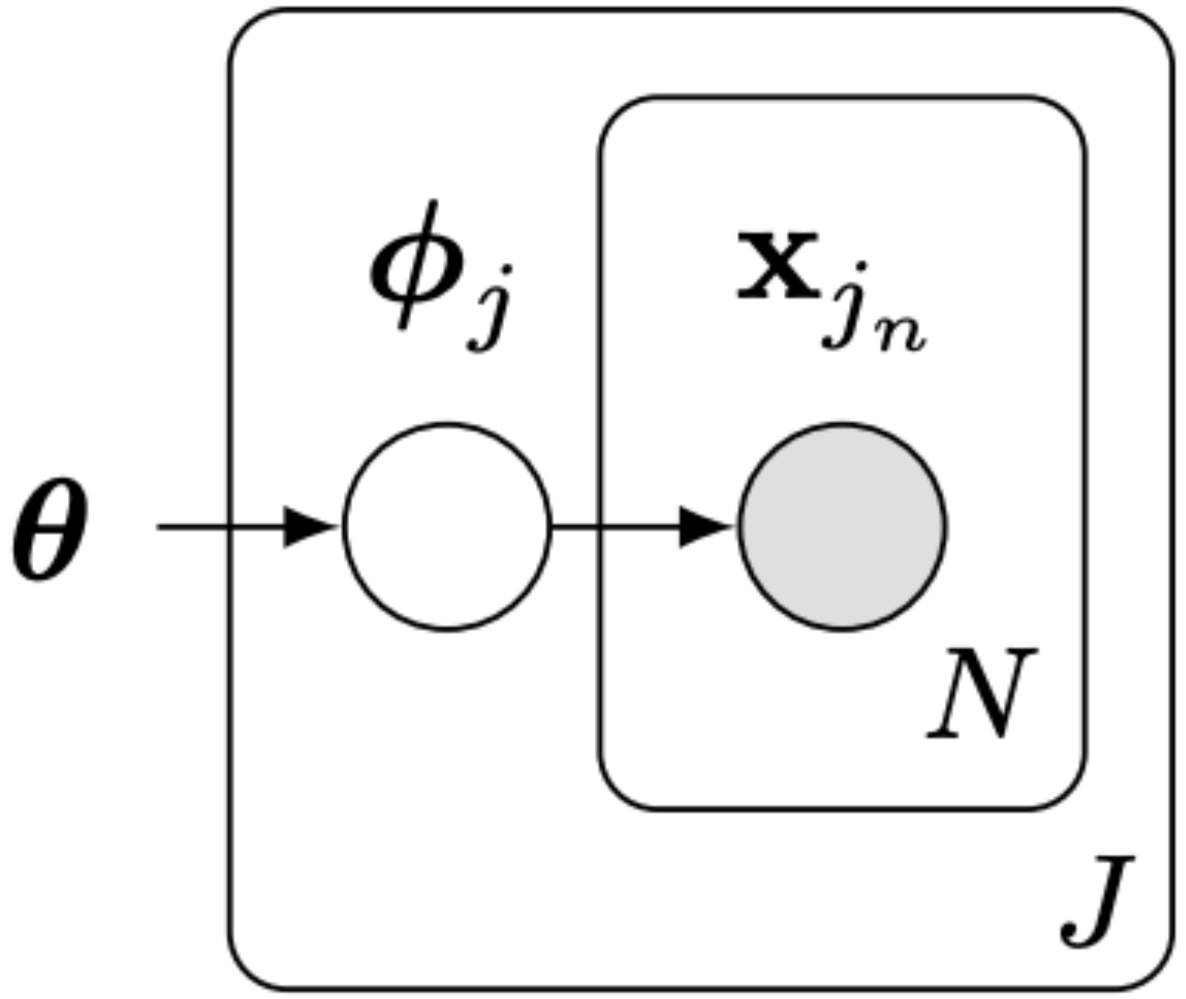
resulting policy



[real-time execution]

Finn*, Yu*, Zhang, Abbeel, Levine. *One-Shot Visual Imitation Learning via Meta-Learning.* CoRL '17

The Bayesian perspective



meta-learning <~> learning priors $p(\phi \mid \theta)$ from data

(Grant et al. '18, Gordon et al. '18, many others)

Outline

1. Brief overview of meta-learning

2. **A peculiar yet ubiquitous problem in meta-learning**
(and how we might regularize it away)

3. Can we scale meta-learning to broad task distributions?

How we construct tasks for meta-learning.

Randomly assign class labels to image classes for each task —> Tasks are *mutually exclusive*.

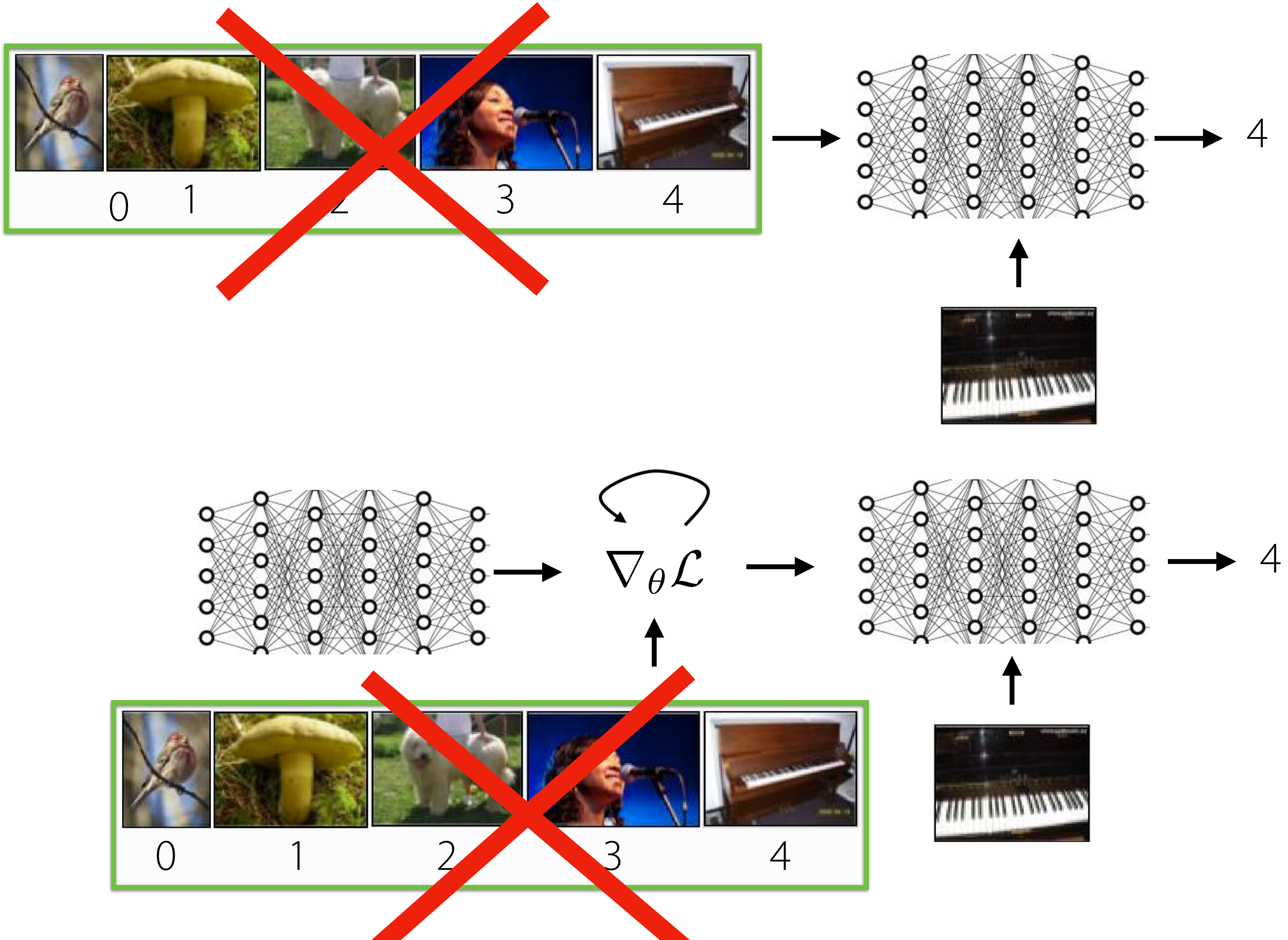
Algorithms **must** use training data to infer label ordering.

What if label order is consistent?

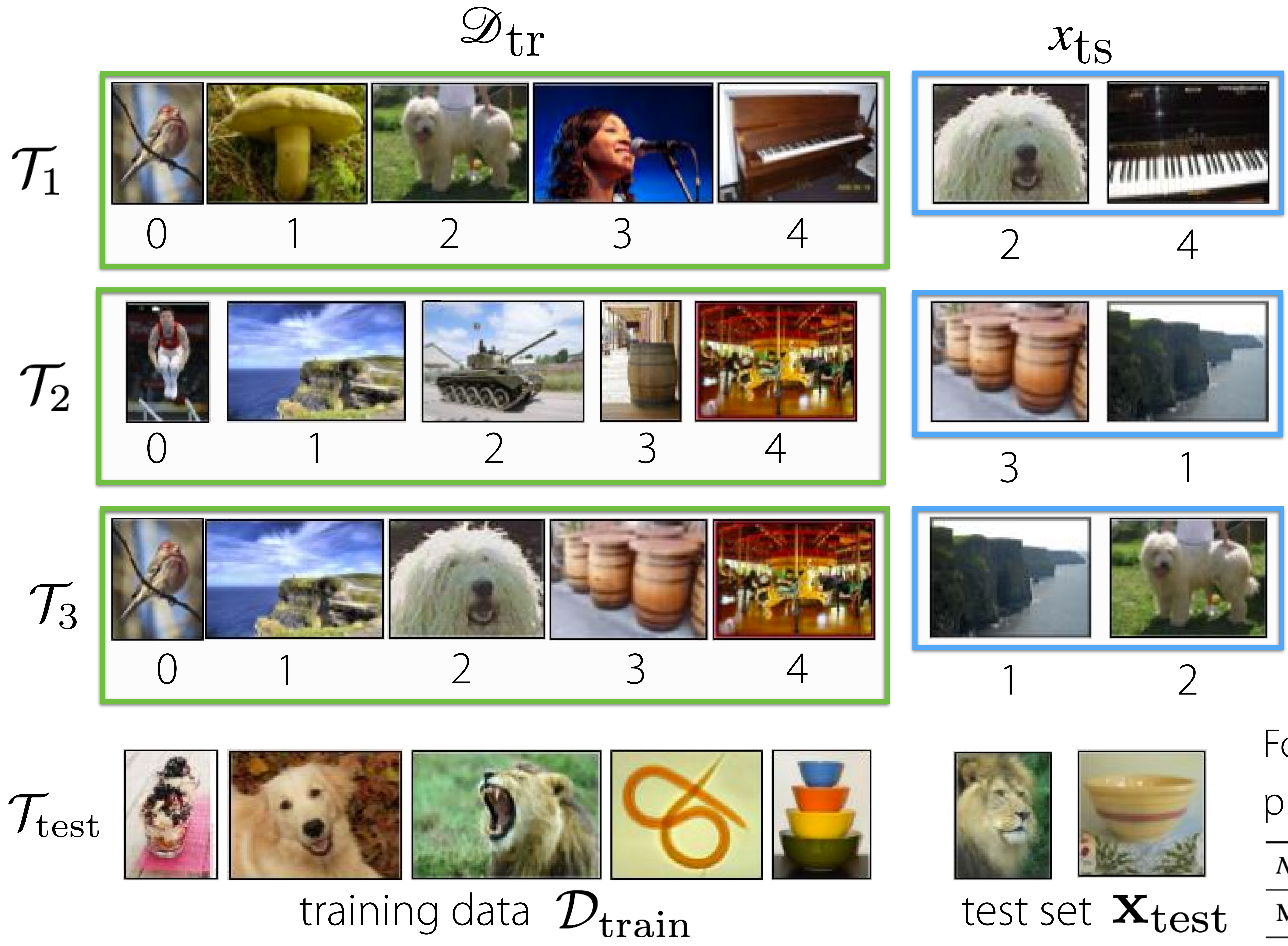
Tasks are **non-mutually exclusive**: a single function can solve all tasks.

The network can simply learn to classify inputs, irrespective of $\mathcal{D}_{\text{tr}}$

The network can simply learn to classify inputs, irrespective of $\mathscr{D}_{\mathrm{tr}}$



What if label order is consistent?



For new image classes: can't make predictions w/o $\mathscr{D}_{\mathrm{tr}}$

NME Omniglot	20-way 1-shot	20-way 5-shot
MAML	7.8 (0.2)%	50.7 (22.9)%

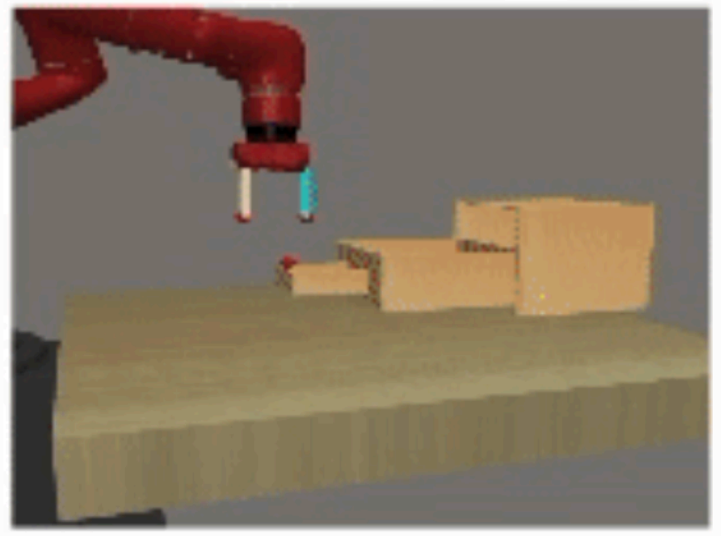
Is this a problem?

- **No**: for image classification, we can just shuffle labels*
- **No**, if we see the same image classes as training (& don't need to adapt at meta-test time)
- But, ***yes***, if we want to be able to adapt with data for new tasks.

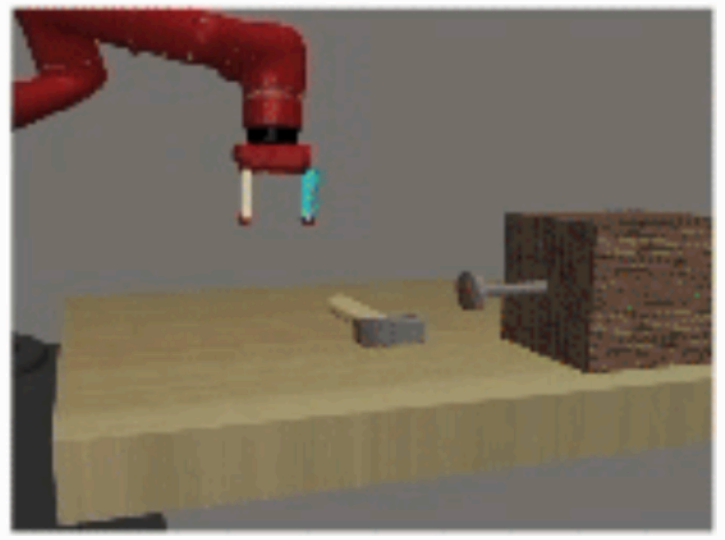
Another example

meta-training

“close drawer”

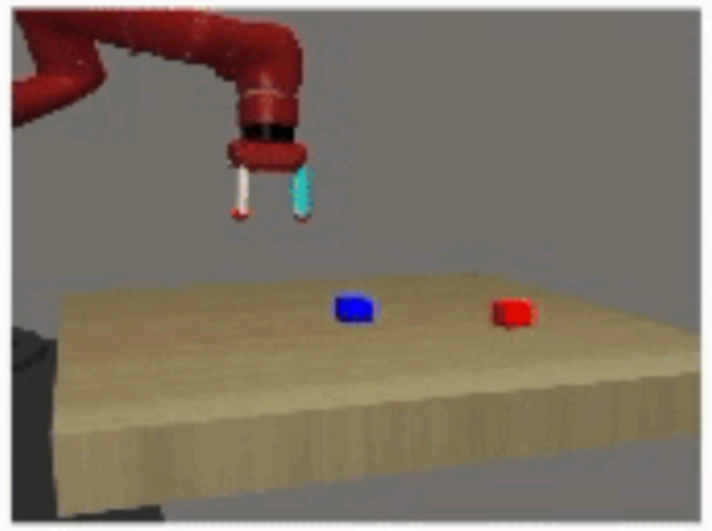
$\mathcal{T}_1$ 

“hammer”

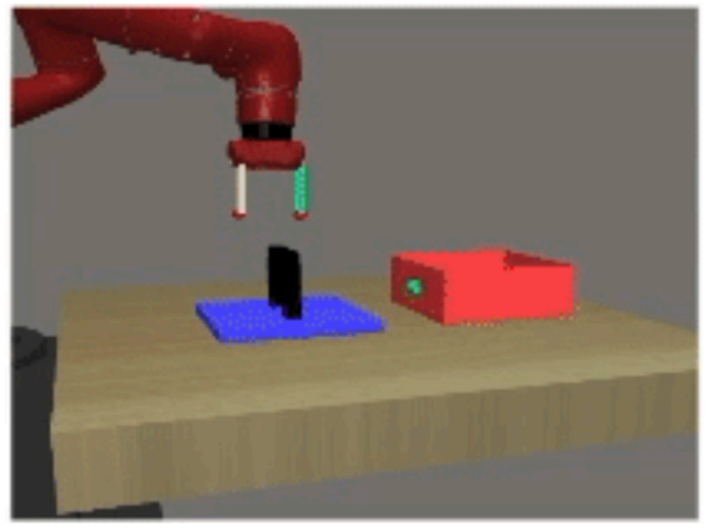
$\mathcal{T}_2$ 

…

“stack”

$\mathcal{T}_{50}$ 

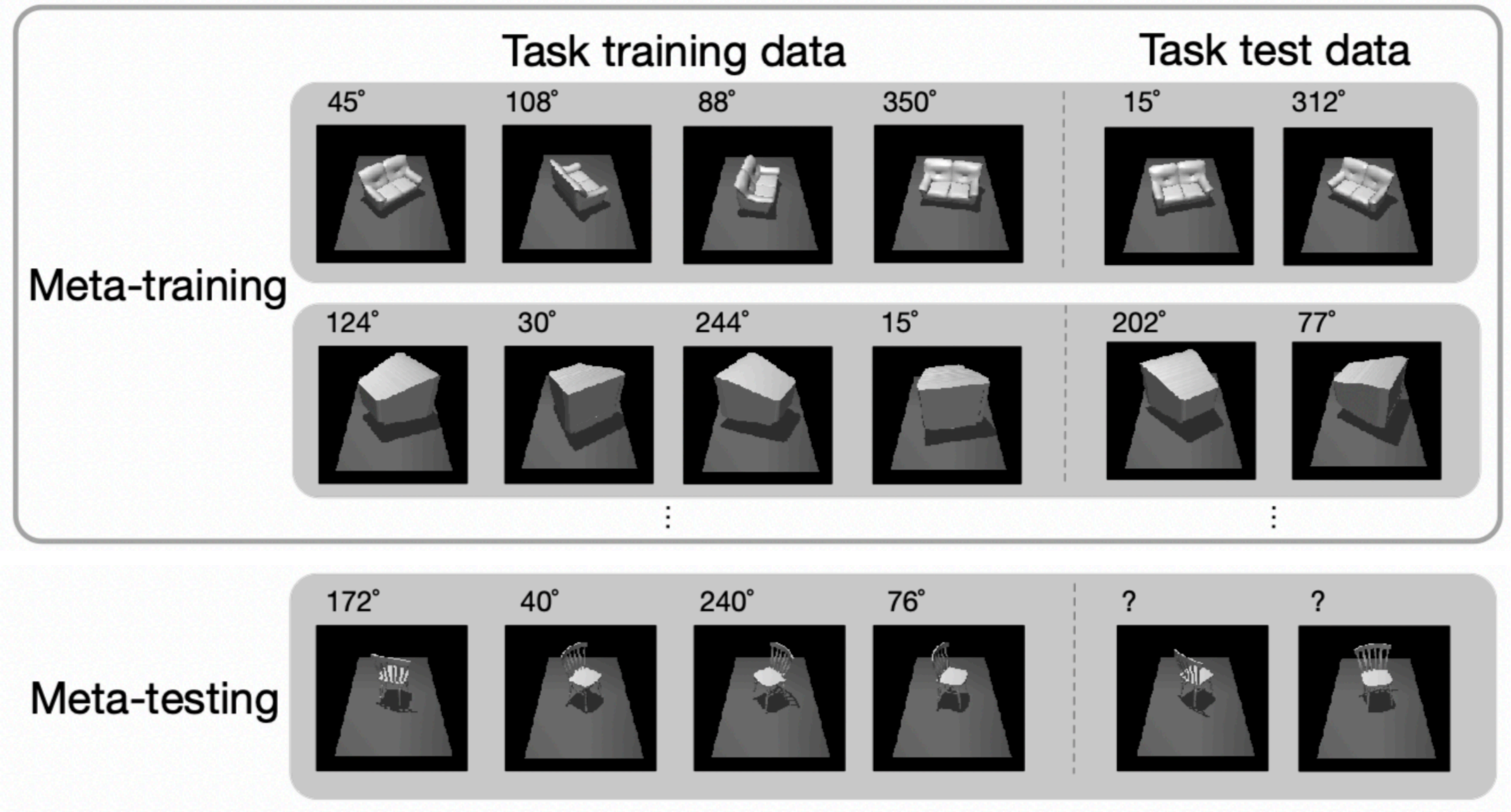
“close box”

$\mathcal{T}_{\text{test}}$ 

If you tell the robot the task goal, the robot can **ignore** the trials.

T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. ***Meta-World***. CoRL ‘19

Another example



Model can memorize the canonical orientations of the training objects.

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

Can we do something about it?

If tasks ***mutually exclusive*****:** single function cannot solve all tasks

(i.e. due to label shuffling, hiding information)

If tasks are ***non-mutually exclusive*****:** single function can solve all tasks

multiple solutions to the meta-learning problem

$$y^{\mathrm{ts}} = f_\theta(\mathcal{D}_i^{\mathrm{tr}}, x^{\mathrm{ts}})$$

One solution: memorize canonical pose info in $\boldsymbol{\theta}$ & ignore $\mathscr{D}_i^{\mathrm{tr}}$

Another solution: carry no info about canonical pose in $\boldsymbol{\theta}$, acquire from $\mathscr{D}_i^{\mathrm{tr}}$

An entire spectrum of solutions based on how **information** flows.

Suggests a potential approach: control information flow.

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

If tasks are *non-mutually exclusive*: single function can solve all tasks

multiple solutions to the meta-learning problem

$$y^{\mathrm{ts}} = f_\theta(\mathcal{D}_i^{\mathrm{tr}}, x^{\mathrm{ts}})$$

One solution: memorize canonical pose info in $\boldsymbol{\theta}$ & ignore $\mathscr{D}_i^{\mathrm{tr}}$

Another solution: carry no info about canonical pose in $\boldsymbol{\theta}$, acquire from $\mathscr{D}_i^{\mathrm{tr}}$

An entire spectrum of solutions based on how **information** flows.

Meta-regularization one option: $\max I(\hat{\mathbf{y}}_{\mathrm{ts}}, \mathscr{D}_{\mathrm{tr}} \,|\, \mathbf{x}_{\mathrm{ts}})$

minimize meta-training loss + information in $\boldsymbol{\theta}$

$$\mathscr{L}(\theta, \mathscr{D}_{meta-train}) + \beta D_{KL}(q(\theta; \theta_\mu, \theta_\sigma) \| p(\theta))$$

Places precedence on using information from $\mathscr{D}_{\mathrm{tr}}$ over storing info in $\boldsymbol{\theta}$.

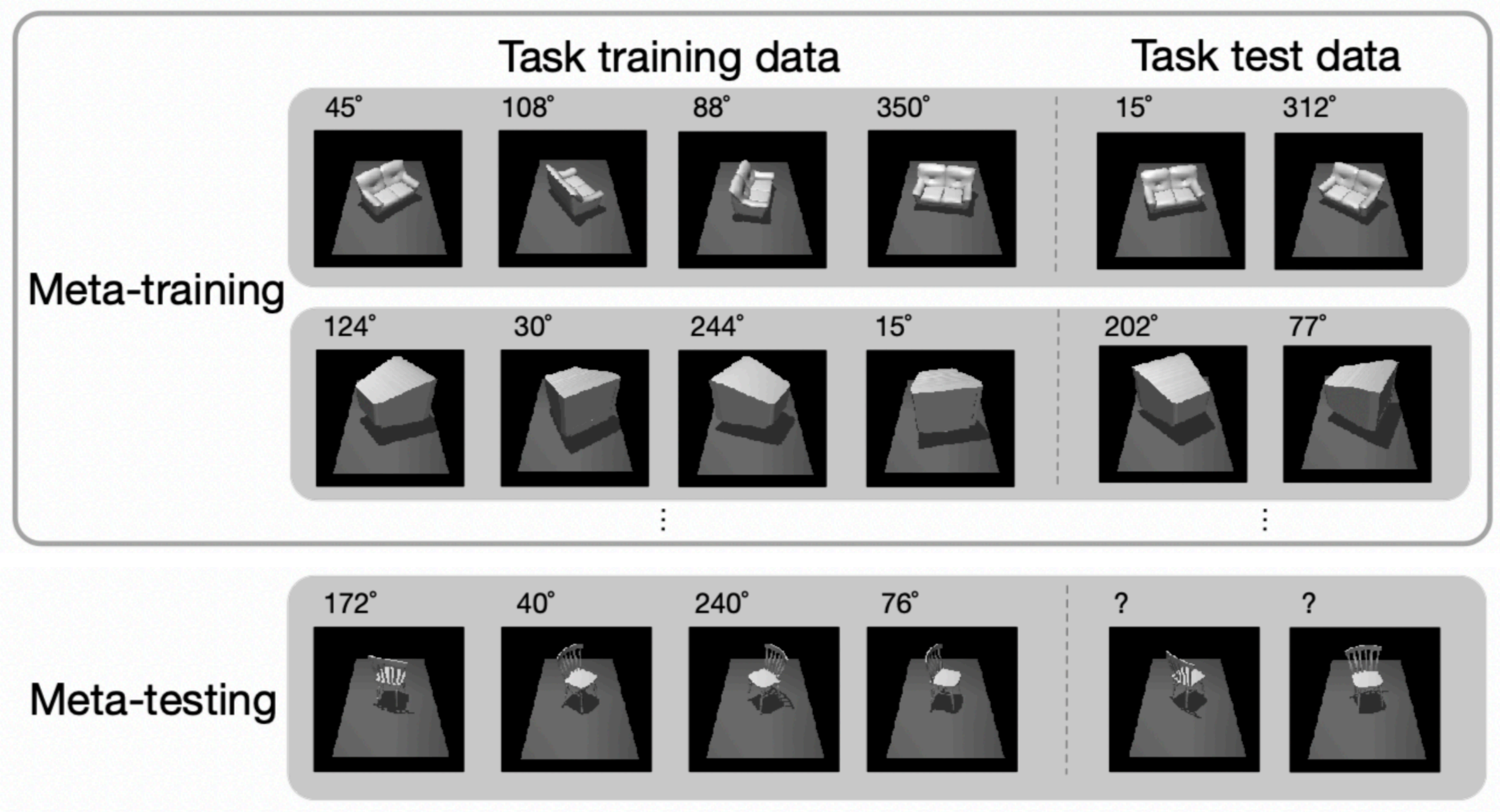
Can combine with your favorite meta-learning algorithm.

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

Omniglot without label shuffling: "non-mutually-exclusive" Omniglot

NME Omniglot	20-way 1-shot	20-way 5-shot
MAML	7.8 (0.2)%	50.7 (22.9)%
TAML	9.6 (2.3)%	67.9 (2.3)%
MR-MAML (W) (ours)	**83.3 (0.8)**%	**94.1 (0.1)**%

On **pose prediction** task:



Method	MAML	MR-MAML(W) (ours)	CNP	MR-CNP(W) (ours)
MSE	5.39 (1.31)	**2.26 (0.09)**	8.48 (0.12)	2.89 (0.18)

(and it's not just as simple as standard regularization)

CNP	CNP + Weight Decay	CNP + BbB	MR-CNP (W) (ours)
8.48 (0.12)	6.86 (0.27)	7.73 (0.82)	**2.89 (0.18)**

TAML: Jamal & Qi. **Task-Agnostic Meta-Learning for Few-Shot Learning**. CVPR '19

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

Does meta-regularization lead to better generalization?

Let $P(\theta)$ be an arbitrary distribution over θ that doesn't depend on the meta-training data.

(e.g. $P(\theta) = \mathcal{N}(\theta; \mathbf{0}, \mathbf{I})$)

For MAML, with probability at least $1 - \delta$,

$$\underbrace{er(\theta_\mu, \theta_\sigma)}_{\text{generalization error}} \le \underbrace{\frac{1}{n}\sum_{i=1}^{n} \hat{er}(\theta_\mu, \theta_\sigma, \mathcal{D}_i, \mathcal{D}_i^*)}_{\text{error on the meta-training set}} + \left(\sqrt{\frac{1}{2(K-1)}} + \sqrt{\frac{1}{2(n-1)}}\right)\sqrt{\underbrace{D_{KL}(\mathcal{N}(\theta; \theta_\mu, \theta_\sigma) \| P)}_{\text{meta-regularization}} + \log\frac{n(K+1)}{\delta}},$$

$\forall \theta_\mu, \theta_\sigma$

With a Taylor expansion of the RHS + a particular value of β —> *recover the MR MAML objective*.

Proof: draws heavily on Amit & Meier '18

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

2. **A peculiar yet ubiquitous problem in meta-learning**

(and how we might regularize it away)

Intermediate Takeaways

meta overfitting ⟷ **standard overfitting**

memorize training functions f_i corresponding to tasks in your meta-training dataset

memorize training datapoints (x_i, y_i) in your training dataset

meta regularization

controls information flow

regularizes description length of meta-parameters

standard regularization

regularize hypothesis class

(though not always for DNNs)

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization.** ICLR '19

Outline

1. Brief overview of meta-learning

2. A peculiar yet ubiquitous problem in meta-learning
(and how we might regularize it away)

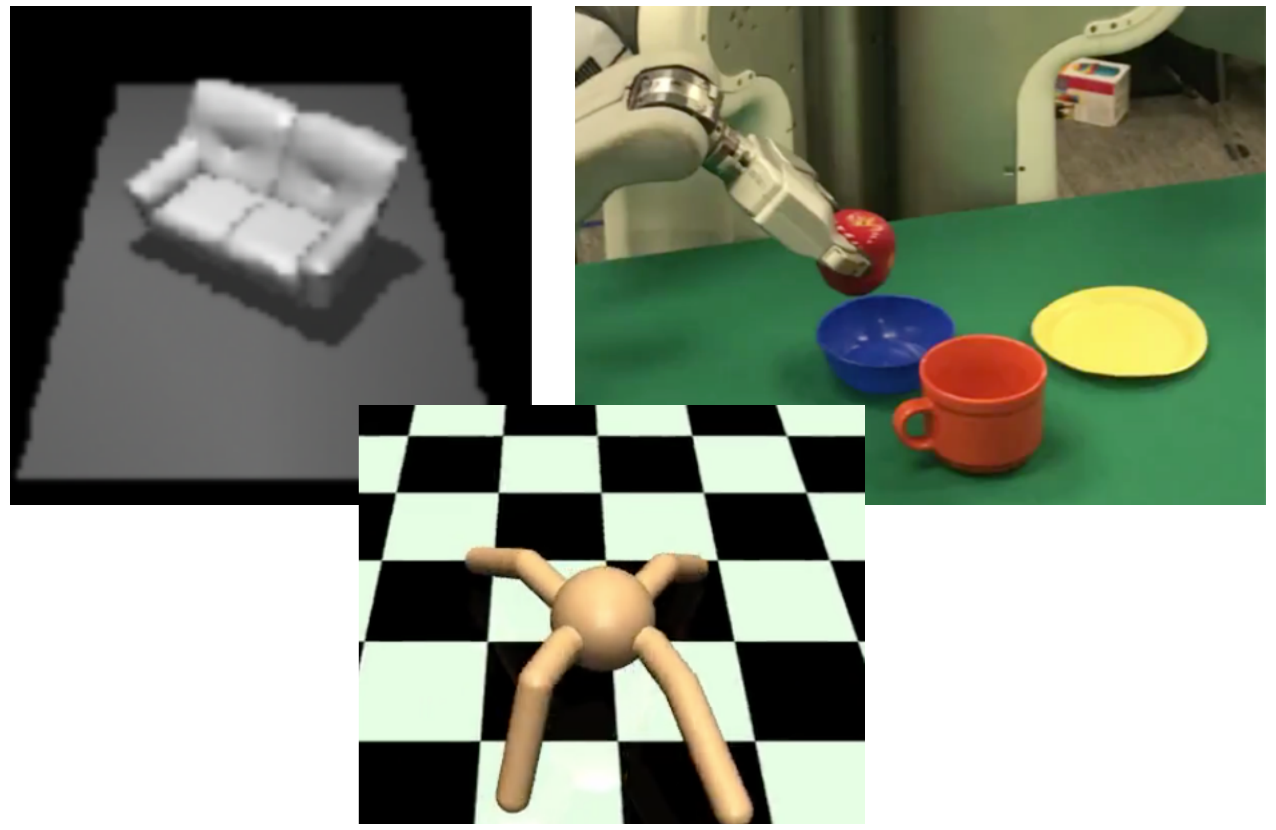
3. **Can we scale meta-learning to broad task distributions?**

Has meta-learning accomplished our goal of making adaptation fast?

Sort of…

Can adapt to:

- new objects
- new goal velocities
- new object categories



Can we adapt to entirely *new* tasks or datasets?

Can we adapt to entirely *new* tasks or datasets?

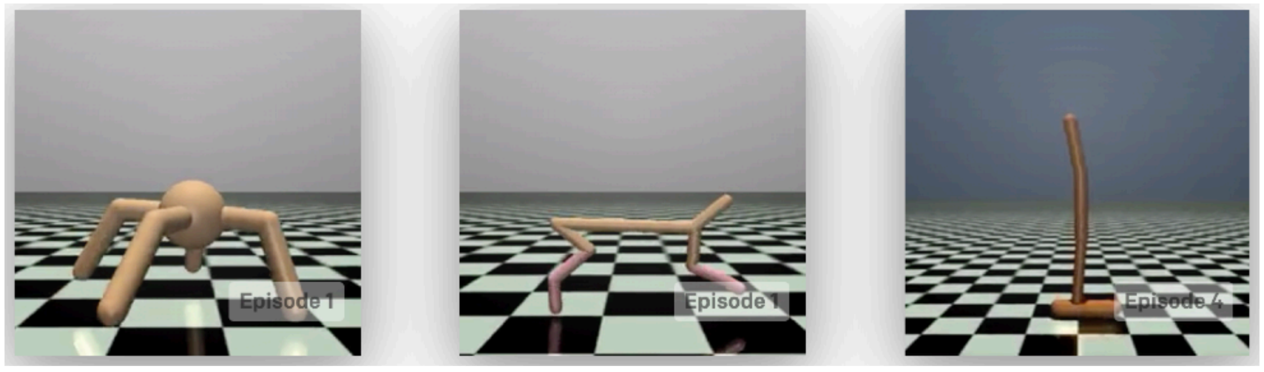
meta-train task distribution

=

meta-test task distribution

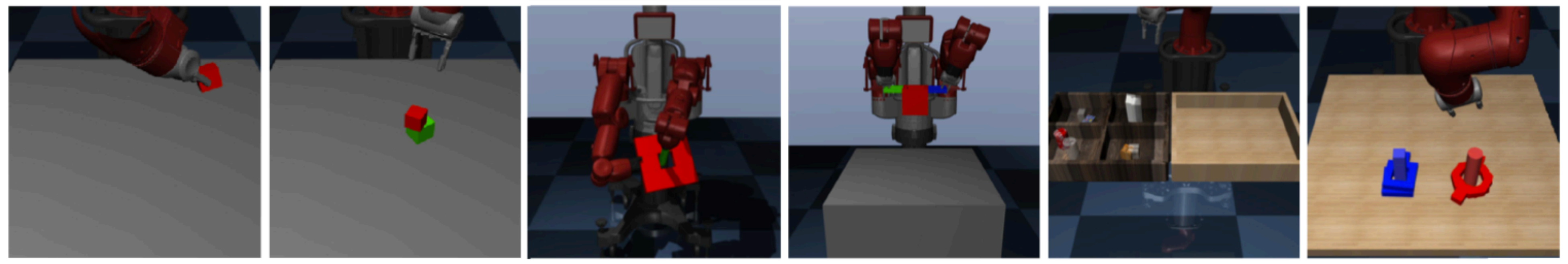
—> Need **broad** distribution of tasks for meta-training

Can we look to RL benchmarks?



Brockman et al. *OpenAI Gym*. 2016

Bellemare et al. *Atari Learning Environment*. 2016



Fan et al. *SURREAL: Open-Source Reinforcement Learning Framework and Robot Manipulation Benchmark*. CoRL 2018

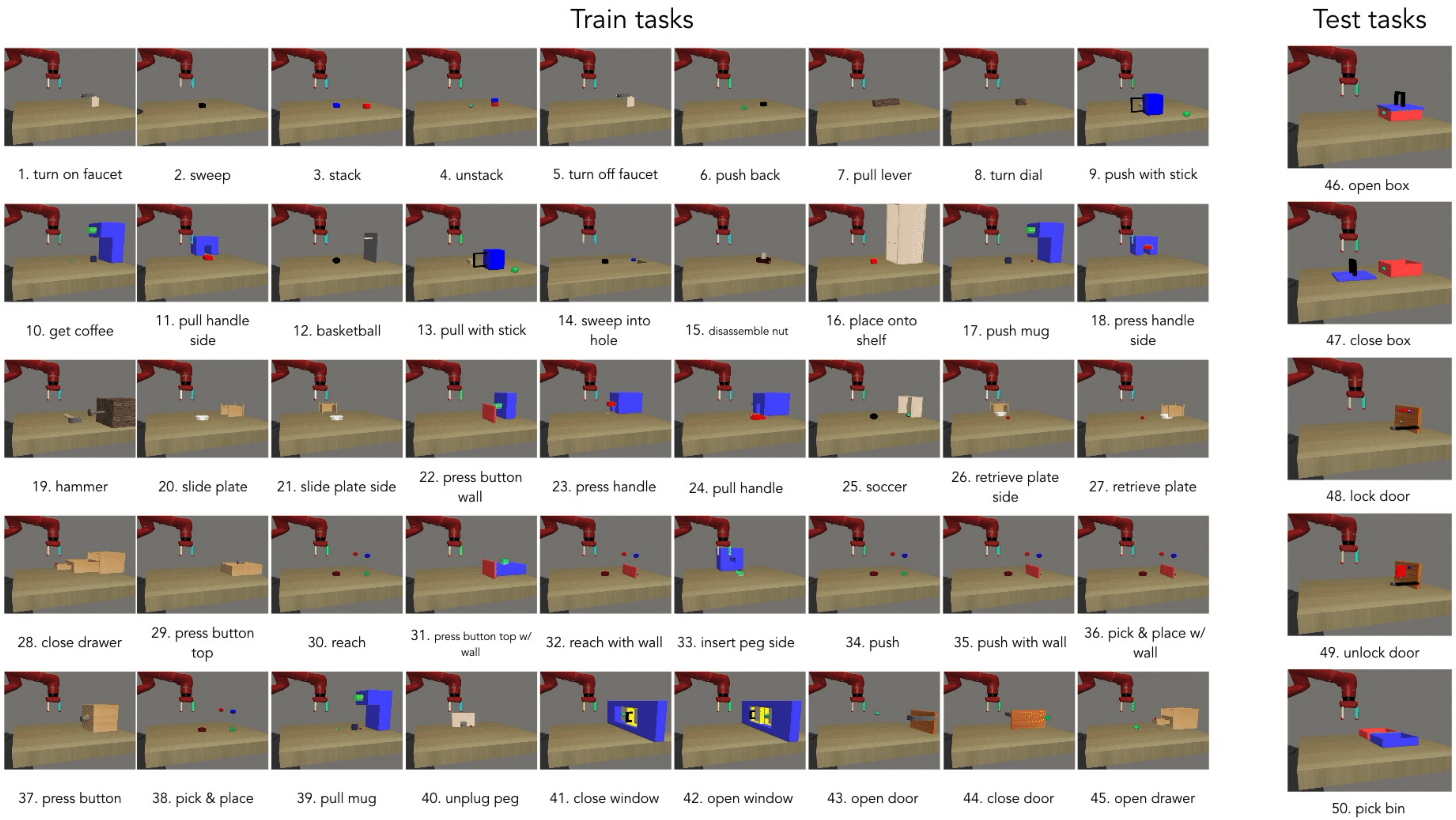
Our desiderata

50+ qualitatively distinct tasks

shaped reward function & success metrics

All tasks individually solvable (to allow us to focus on multi-task / meta-RL component)

Unified state & action space, environment (to facilitate transfer)



Meta-World Benchmark

T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. **Meta-World**. CoRL '19

Results: Meta-learning algorithms seem to struggle…

Methods	ML45	
	meta-train	meta-test
MAML		
RL^2		
PEARL		

…even on the 45 meta-training tasks!

Multi-task RL algorithms ***also*** struggle…

Methods	MT50
Multi-task PPO	8.98%
Multi-task TRPO	22.86%
Task embeddings	15.31%
Multi-task SAC	28.83%
Multi-task multi-head SAC	**35.85%**

T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. **Meta-World**. CoRL '19

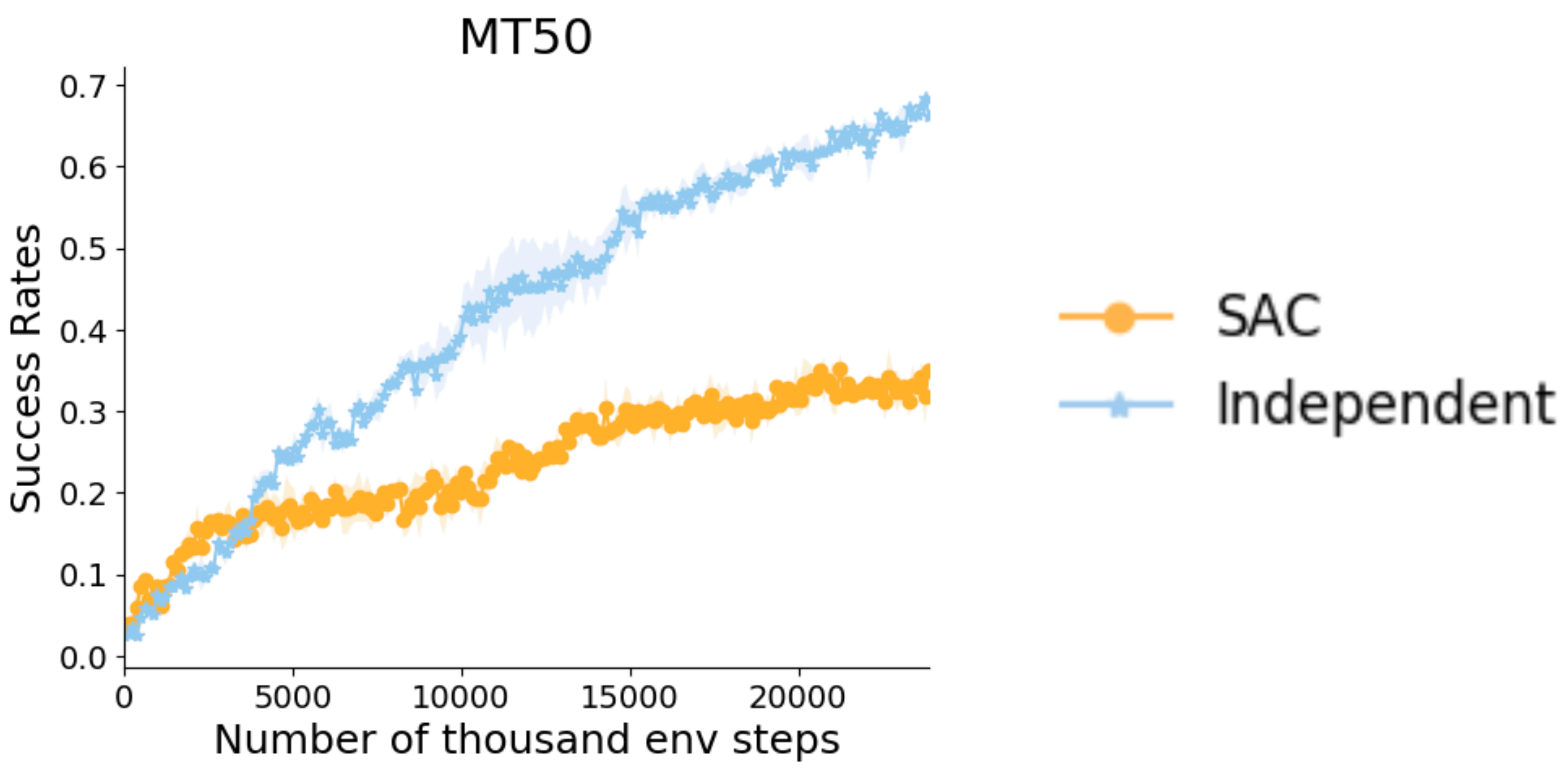
Why the poor results?

Exploration challenge? All tasks individually solvable.

Data scarcity? All methods given budget with plenty of samples.

Limited model capacity? All methods plenty of capacity.

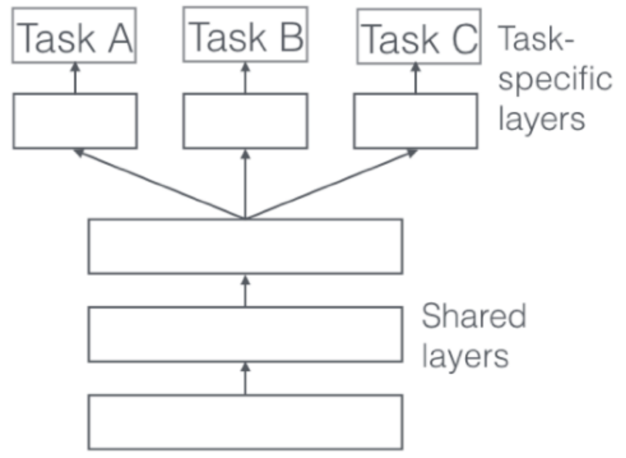
Training models *independently* performs the best.



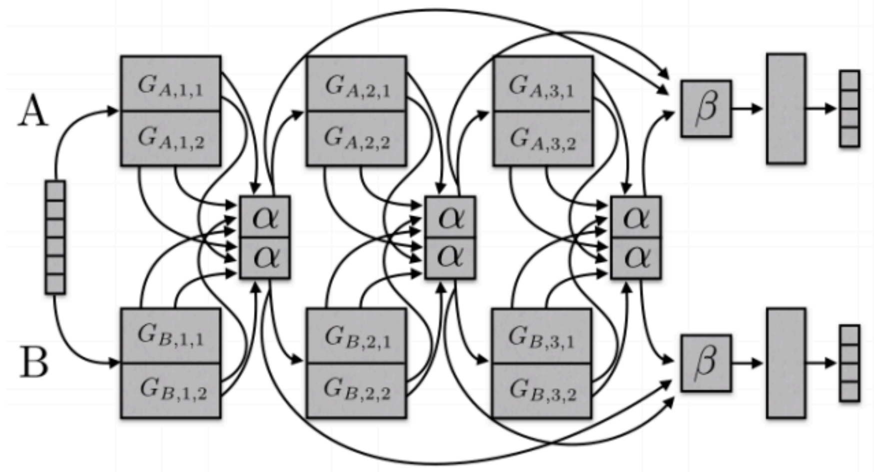
Our conclusion: must be an ***optimization*** challenge.

Prior literature on multi-task learning

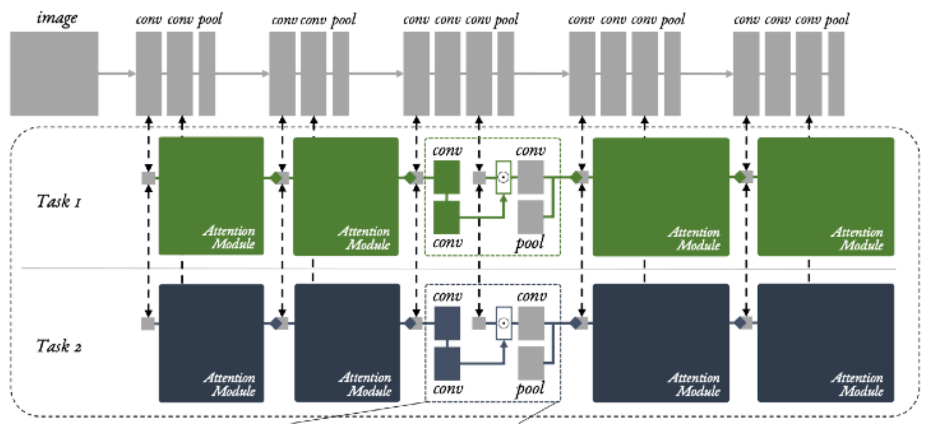
Architectural solutions:



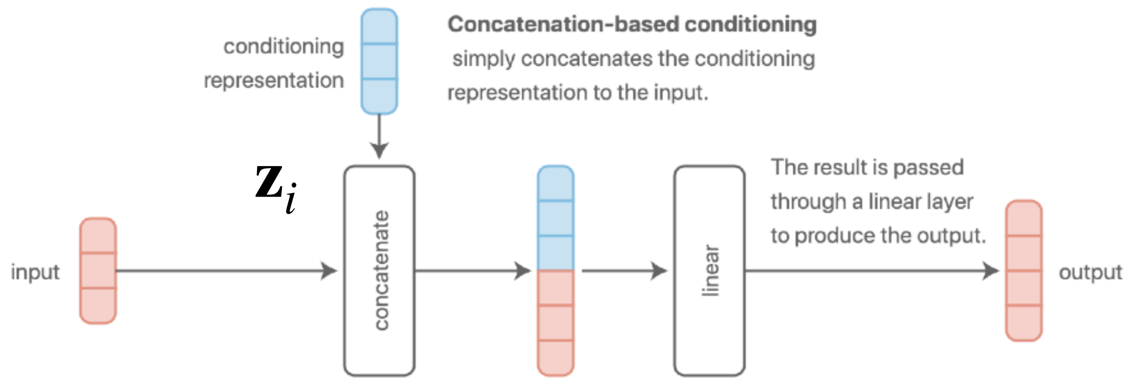
Multi-head architectures



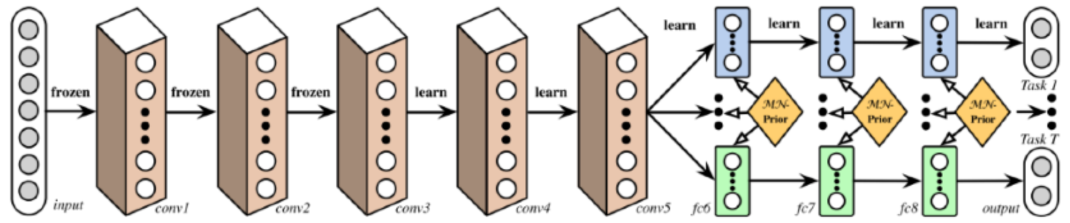
Sluice Networks. Ruder, Bingel, Augenstein, Sogaard '17



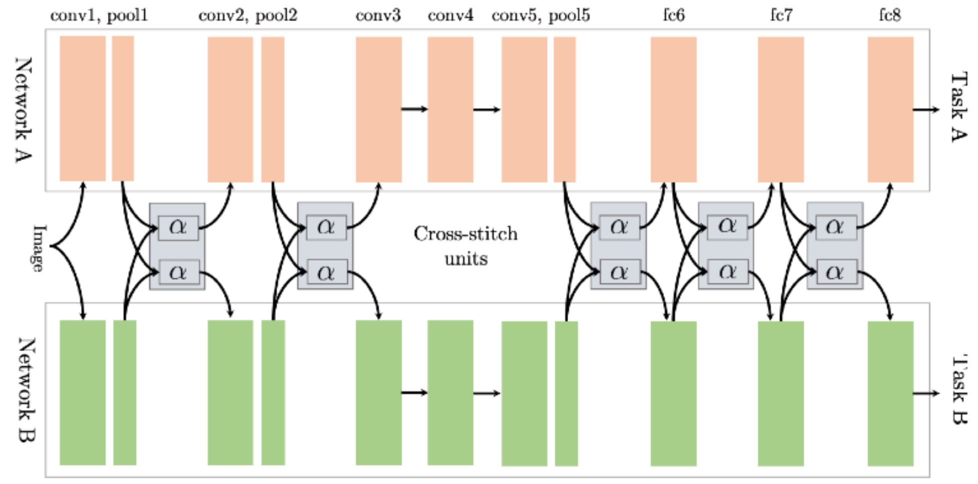
Multi-Task Attention Network. Liu, Johns, Davison '18



FiLM: Visual Reasoning with a General Conditioning Layer. Perez et al. '17



Deep Relation Networks. Long, Wang '15



Cross-Stitch Networks. Misra, Shrivastava, Gupta, Hebert '16

Task weighting solutions:

$$L_{\text{tot}} = w_{\text{depth}} L_{\text{depth}} + w_{\text{kpt}} L_{\text{kpt}} + w_{\text{normals}} L_{\text{normals}}$$

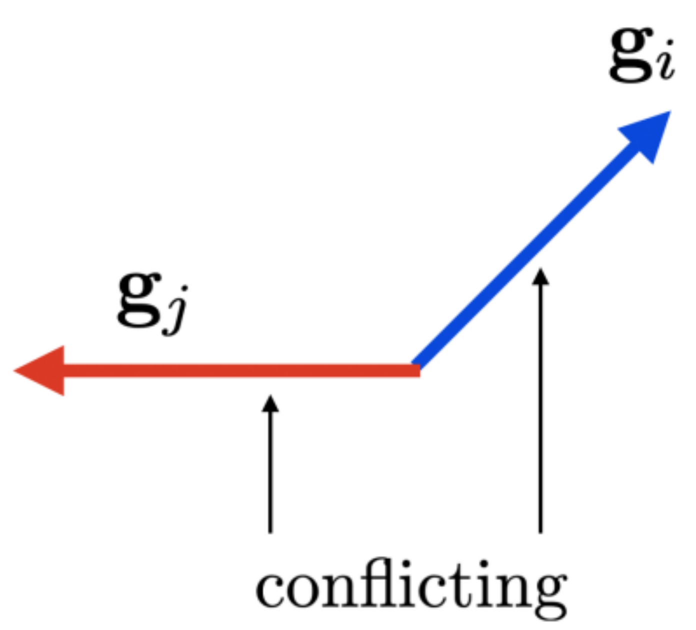
GradNorm. Chen et al. '18

$$\min_{\substack{\boldsymbol{\theta}^{sh}, \\ \boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^T}} \sum_{t=1}^{T} c^t \hat{\mathcal{L}}^t(\boldsymbol{\theta}^{sh}, \boldsymbol{\theta}^t)$$

MT Learning as Multi-Objective Optimization. Sener & Koltun. '19

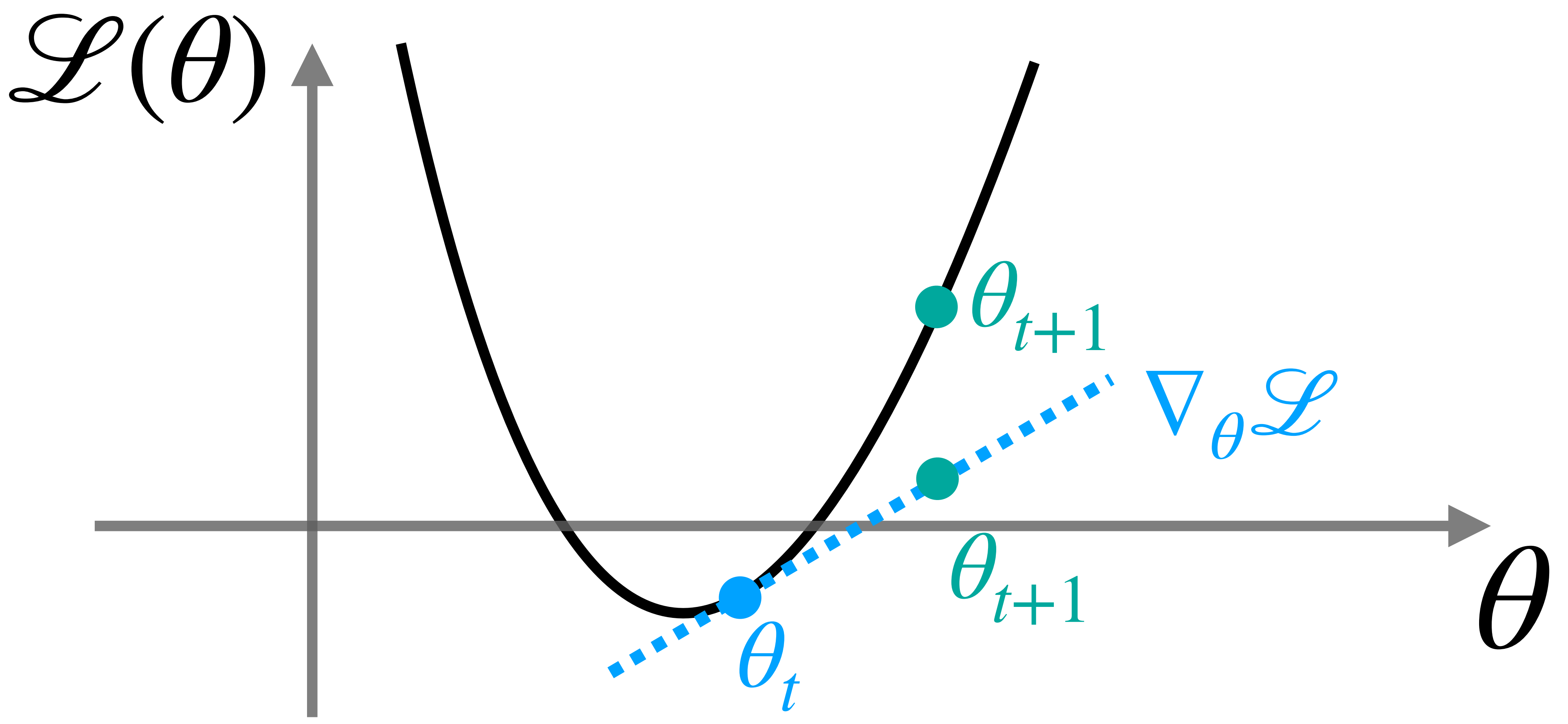
Hypothesis 1: Gradients from different tasks often conflict

If so: would see negative inner product of gradients



Hypothesis 2: When they do conflict, they cause more damage than expected.

i.e. due to high curvature & difference in grad magnitude

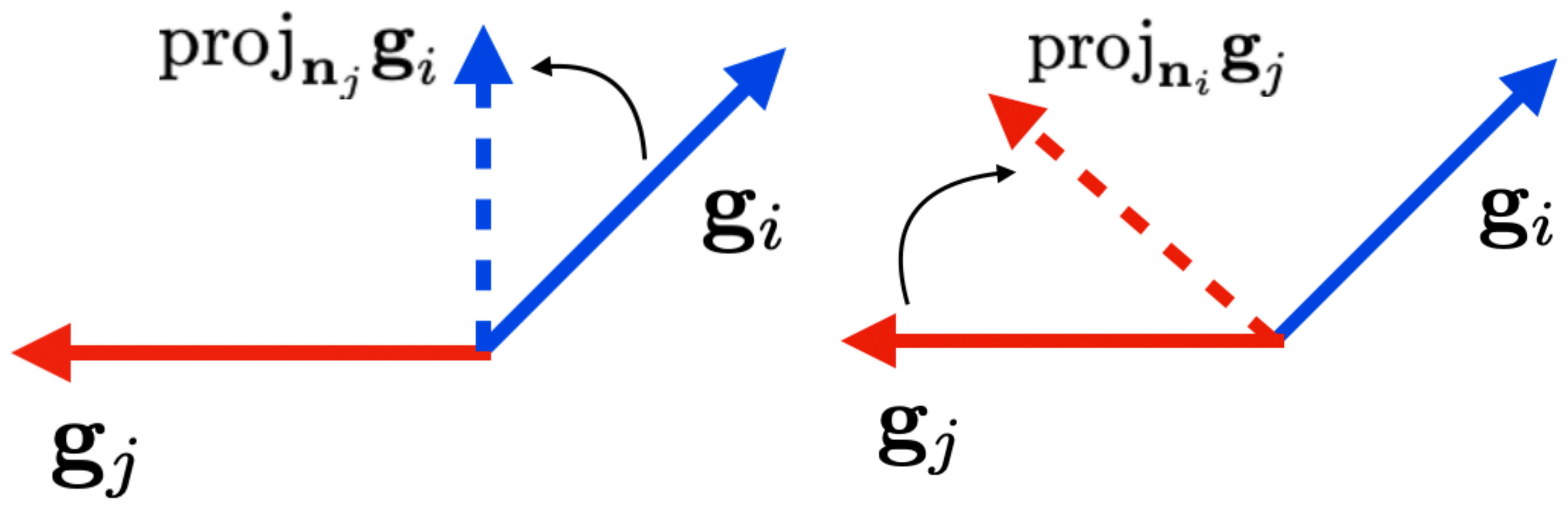


T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

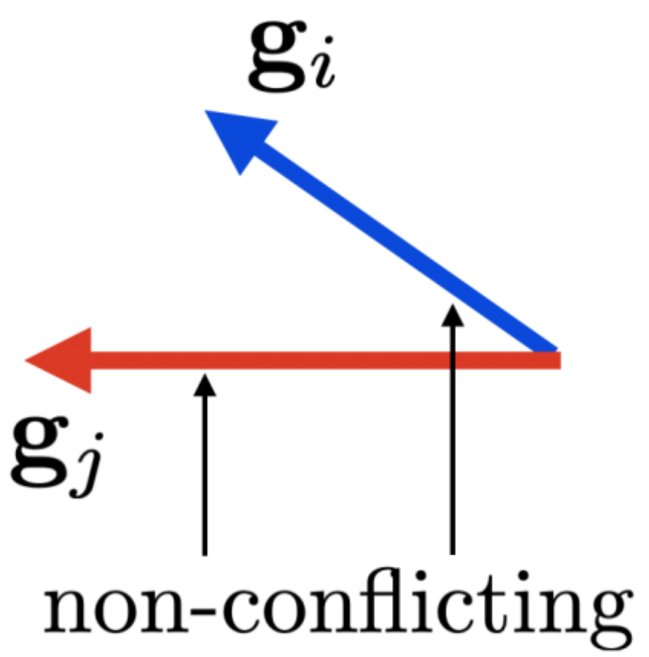
Idea: try to avoid making other tasks worse, when taking gradient step

Algorithm:

If two gradients *conflict*:
project each onto the normal plane of the other



Else:
leave them alone

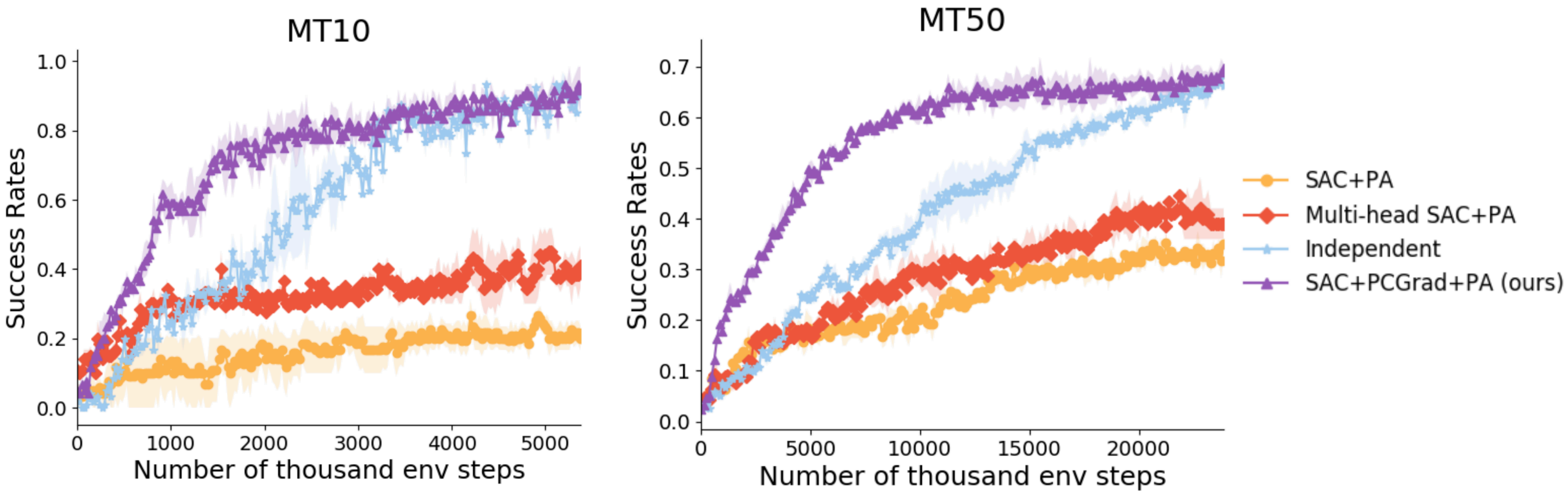


i.e. project conflicting gradients

"PCGrad"

T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

Multi-Task RL on Meta-World:



T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

Multi-Task CIFAR-100

	% accuracy
task specific-1-fc (Rosenbaum et al., 2018)	42
task specific-all-fc (Rosenbaum et al., 2018)	49
cross stitch-all-fc (Misra et al., 2016b)	53
routing-all-fc + WPL (Rosenbaum et al., 2019)	74.7
independent	67.7
PCGrad (ours)	71
routing-all-fc + WPL + PCGrad (ours)	$\boxed{\mathbf{77.5}}$

+ also helps multi-task **supervised** learning
+ complementary to multi-task **architectures**

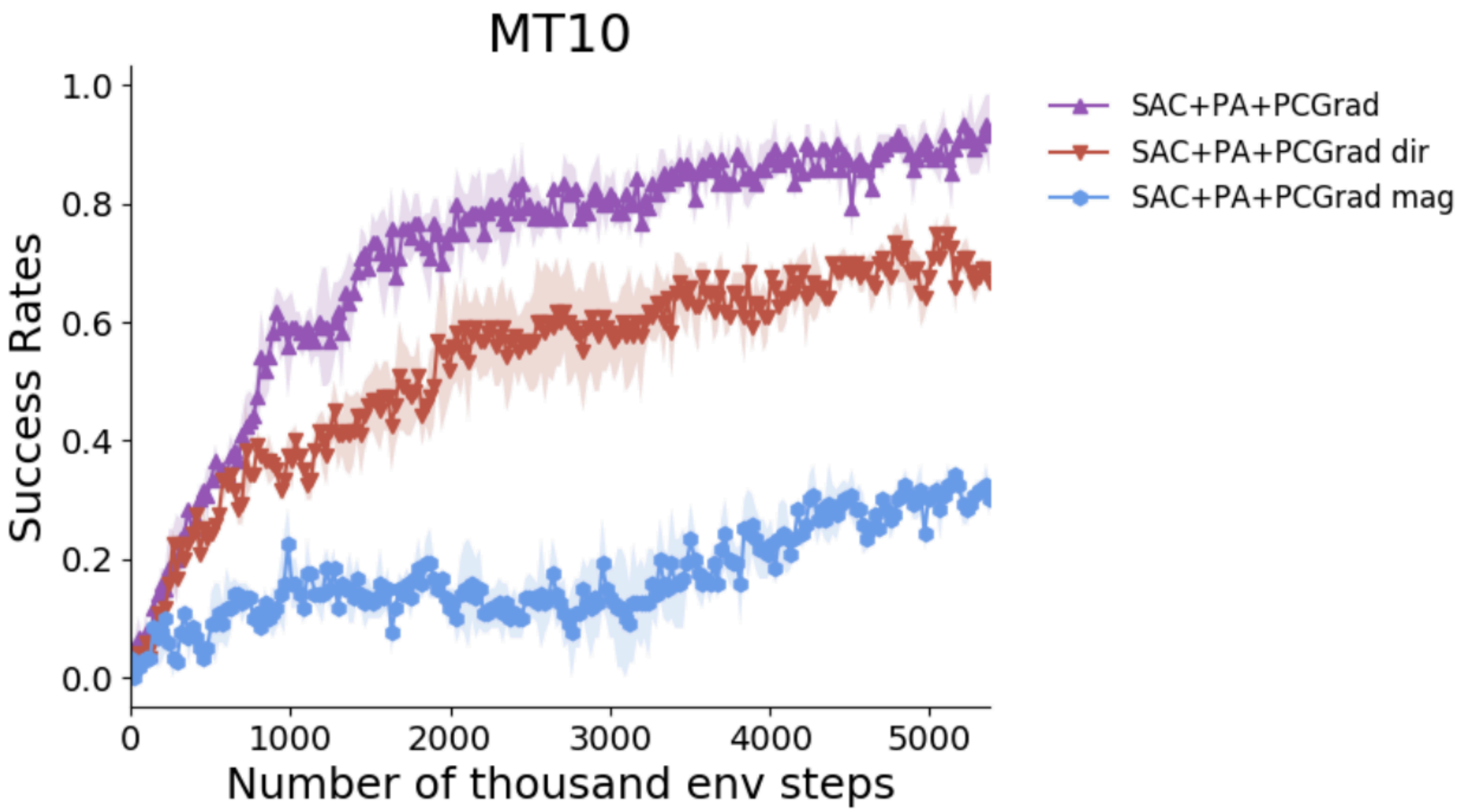
Multi-Task NYUv2

#P.	Architecture	Weighting	Segmentation (Higher Better)		Depth (Lower Better)		Surface Normal: Angle Distance (Lower Better)		Surface Normal: Within t° (Higher Better)		
			mIoU	Pix Acc	Abs Err	Rel Err	Mean	Median	11.25	22.5	30
≈3	Cross-Stitch‡	Equal Weights	14.71	50.23	0.6481	0.2871	33.56	28.58	20.08	40.54	51.97
		Uncert. Weights*	15.69	52.60	0.6277	0.2702	32.69	27.26	21.63	42.84	54.45
		DWA†, $T = 2$	**16.11**	**53.19**	**0.5922**	**0.2611**	**32.34**	**26.91**	**21.81**	**43.14**	**54.92**
1.77	MTAN†	Equal Weights	**17.72**	55.32	**0.5906**	0.2577	31.44	**25.37**	$\boxed{\mathbf{23.17}}$	45.65	57.48
		Uncert. Weights*	17.67	**55.61**	0.5927	0.2592	**31.25**	25.57	22.99	**45.83**	**57.67**
		DWA†, $T = 2$	17.15	54.97	0.5956	**0.2569**	31.60	25.46	22.48	44.86	57.24
1.77	MTAN† + PCGrad (ours)	Uncert. Weights*	$\boxed{\mathbf{20.17}}$	$\boxed{\mathbf{56.65}}$	$\boxed{\mathbf{0.5904}}$	$\boxed{\mathbf{0.2467}}$	$\boxed{\mathbf{30.01}}$	$\boxed{\mathbf{24.83}}$	22.28	$\boxed{\mathbf{46.12}}$	$\boxed{\mathbf{58.77}}$

T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

Why does it work?

(Part 1)



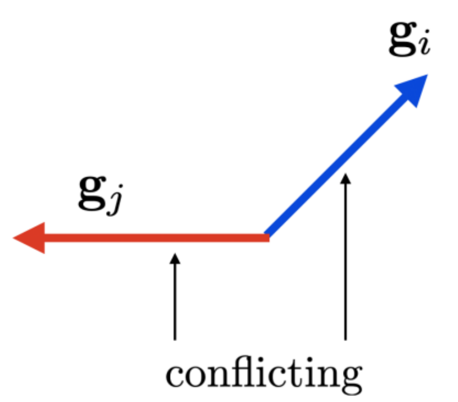
T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

Why does it work?

(Part 2)

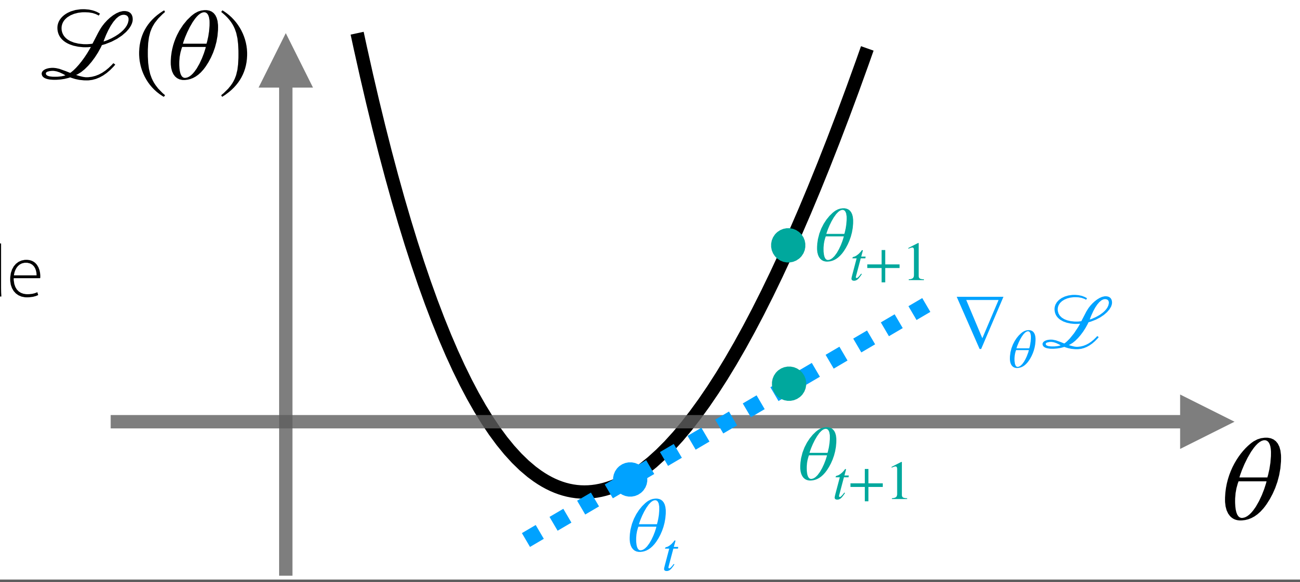
Hypothesis 1: Gradients from different tasks often conflict

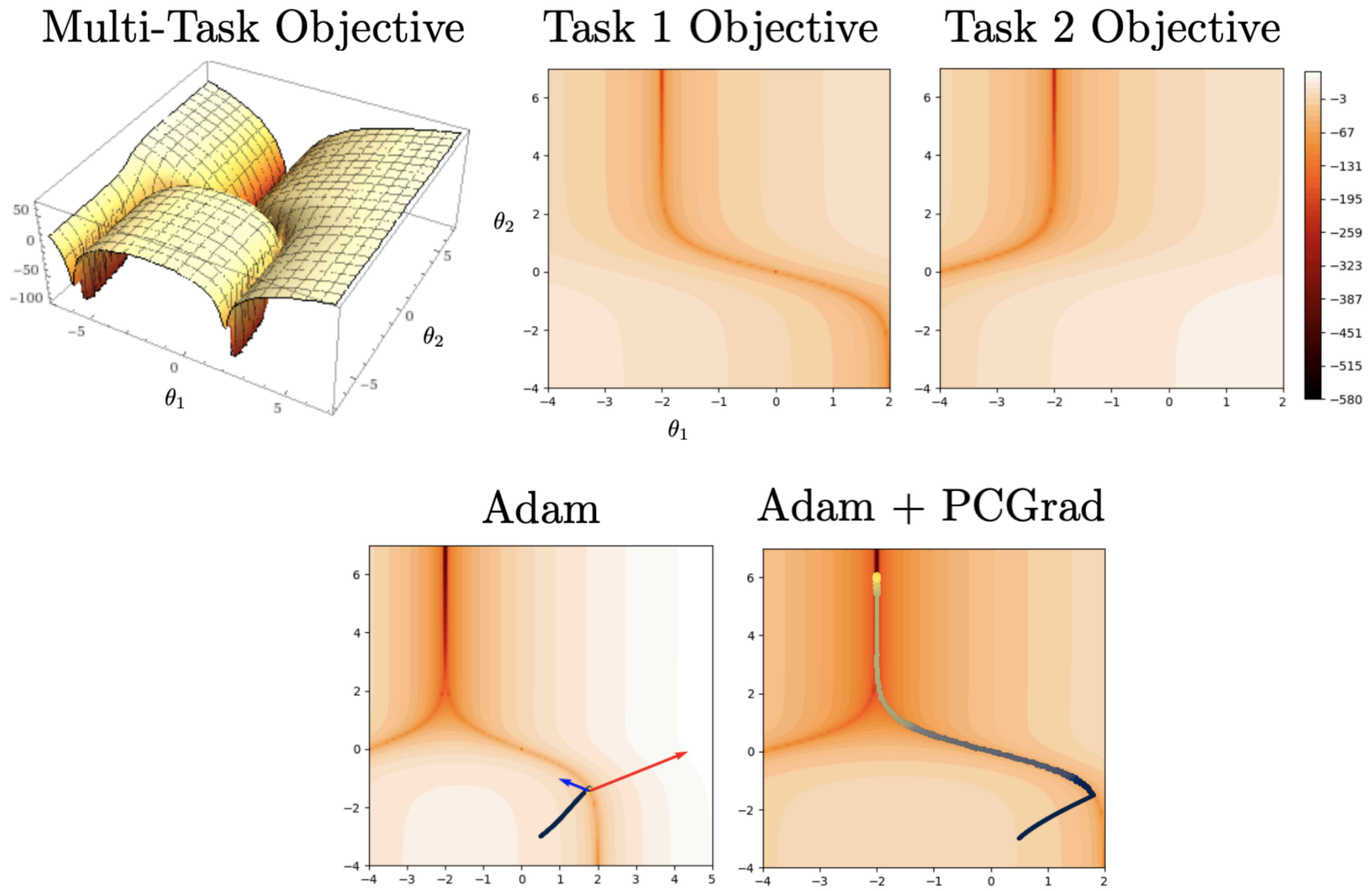
If so: would see negative inner product of gradients



Hypothesis 2: When they do conflict, they cause more damage than expected.

i.e. due to high curvature & difference in grad magnitude





1. conflicting gradients
2. large positive curvature
3. difference in gradient magnitude

"tragic triad"

Is PCGrad ***provably*** better under these three conditions?

Are these three conditions actually ***why*** we see improvements on large-scale problems?

T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

Why does it work?

(Part 2)

"tragic triad"

1. conflicting gradients
2. large positive curvature
3. difference in gradient magnitude

Is PCGrad ***provably*** better under these three conditions?

short answer: yes, if large enough conflict, curvature, gradient magnitude difference

(for two tasks)

long answer:

Theorem 2. *Suppose $\mathcal{L}$ is differentiable and the gradient of $\mathcal{L}$ is Lipschitz continuous with constant $L > 0$. Let θ^{MT} and θ^{PCGrad} be the parameters after applying one update to θ with $\mathbf{g}$ and PCGrad-modified gradient $\mathbf{g}^{PC}$ respectively, with step size $t > 0$. Moreover, assume $\mathbf{H}(\mathcal{L};\theta,\theta^{MT}) \geq \ell\|\mathbf{g}\|_2^2$ for some constant $\ell \leq L$, i.e. the multi-task curvature is lower-bounded. Then $\mathcal{L}(\theta^{PCGrad}) \leq \mathcal{L}(\theta^{MT})$ if*

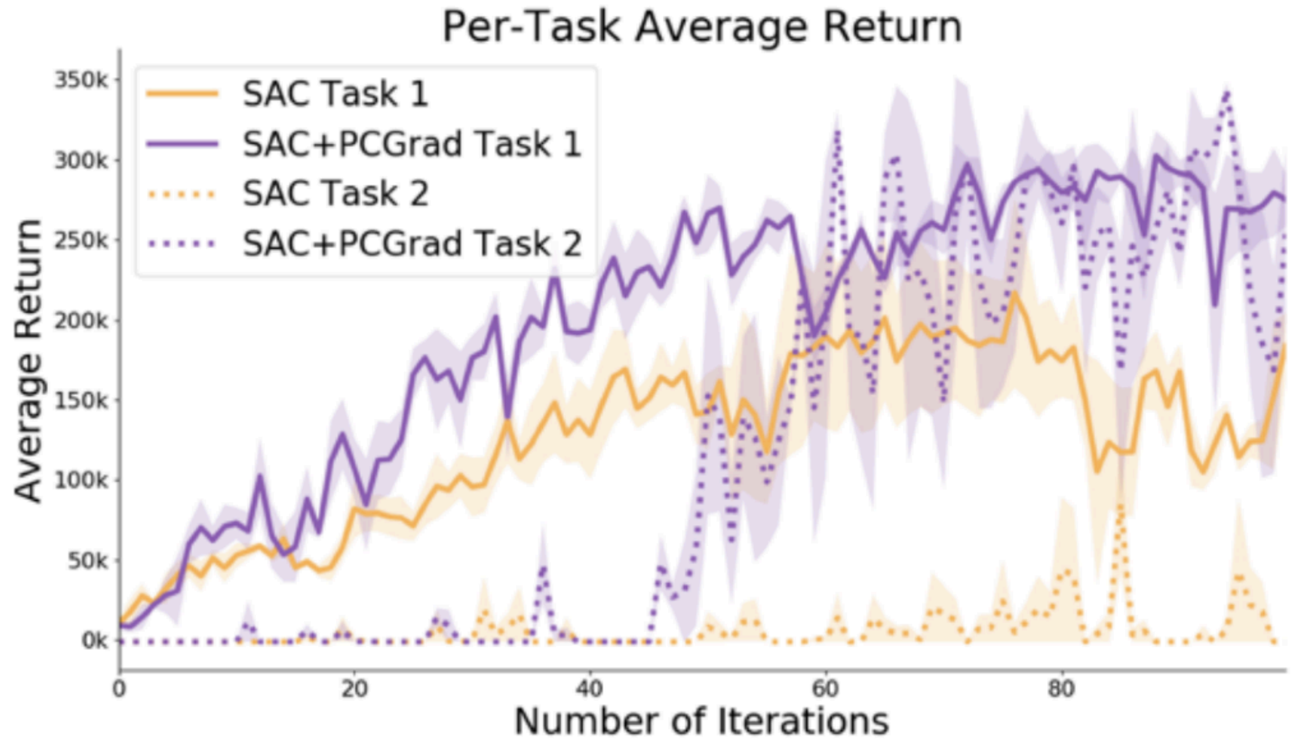
(a) $\cos\phi_{12} \leq -\Phi(\mathbf{g}_1,\mathbf{g}_2)$,

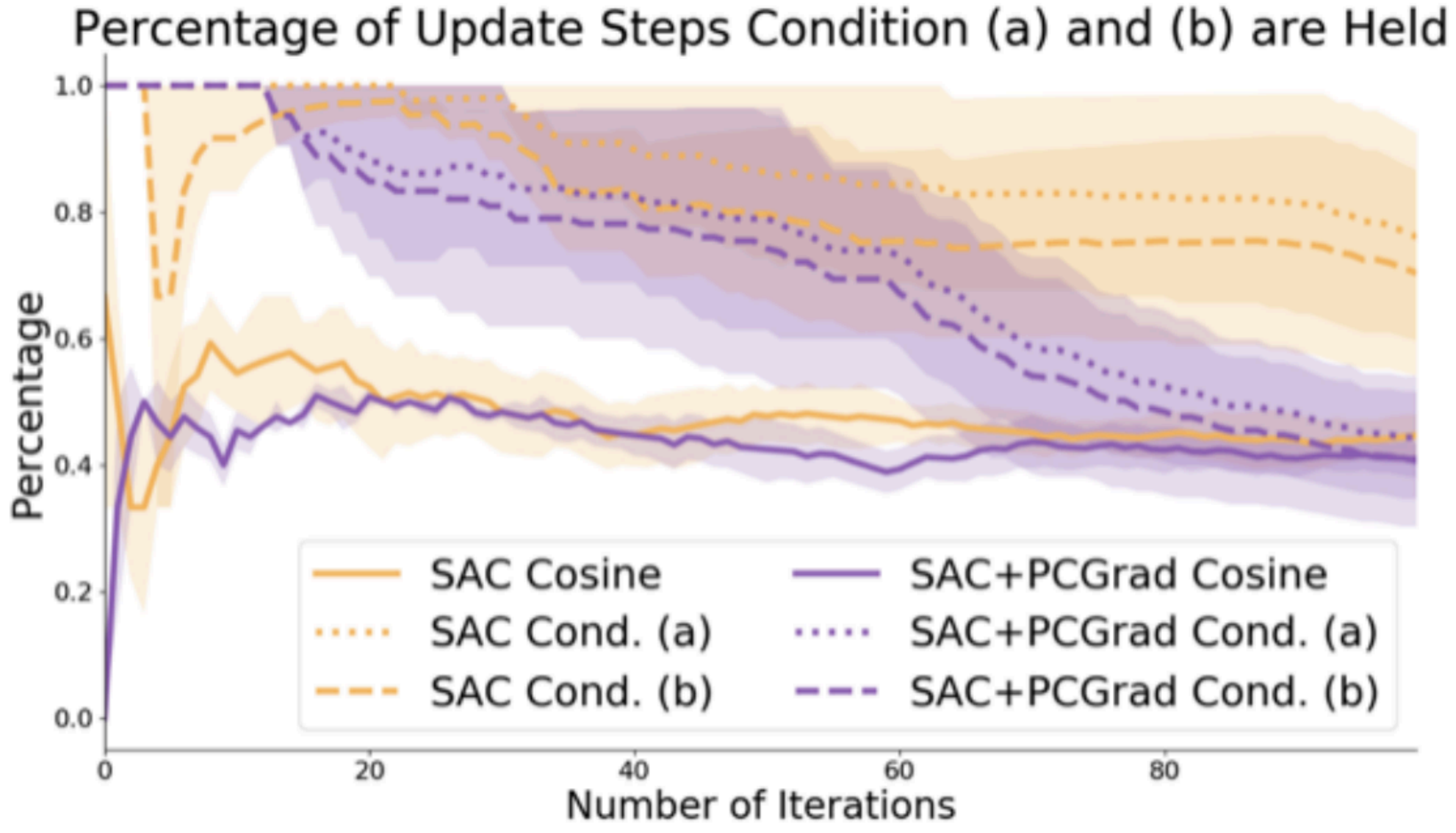
(b) $\ell \geq \xi(\mathbf{g}_1,\mathbf{g}_2)L$, *and*

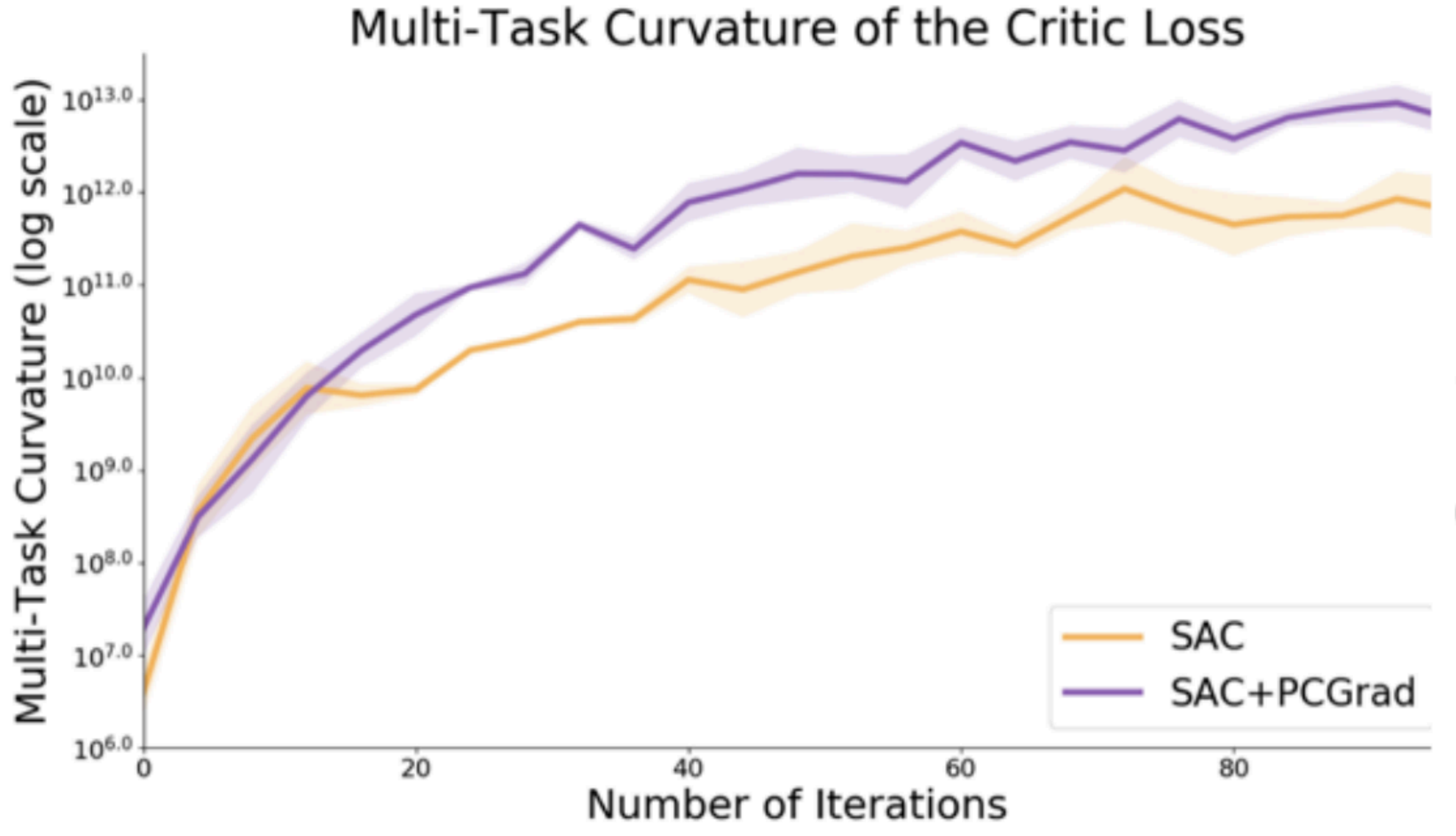
(c) $t \geq \frac{2}{\ell-\xi(\mathbf{g}_1,\mathbf{g}_2)L}$.

Proof. See Appendix B. □

Are these three conditions actually ***why*** we see improvements on large-scale problems?







T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19

3. **Can we scale meta-learning to broad task distributions?**

Scaling to **broad task distributions** is hard,
can't be taken for granted

Lack of good benchmarks —> **Meta-World** with broad, dense task distribution

scaling primarily hindered by *optimization* challenges in MTL

Optimization challenges —> three conditions seem to plague MTL, MTRL

a solution: project conflicting gradients (**PCGrad**)

Remaining questions:

Does this solution translate back to meta-learning?

Is this problem unique to multi-task learning?

Takeaways

2. **A peculiar yet ubiquitous problem in meta-learning**

(and how we might regularize it away)

meta overfitting

memorize training functions f_i

corresponding to tasks in your meta-training dataset

meta regularization

controls information flow

regularizes description length of meta-parameters

3. **Can we scale meta-learning to broad task distributions?**

Lack of good benchmarks —> **Meta-World** with broad, dense task distribution

scaling primarily hindered by *optimization* challenges in MTL

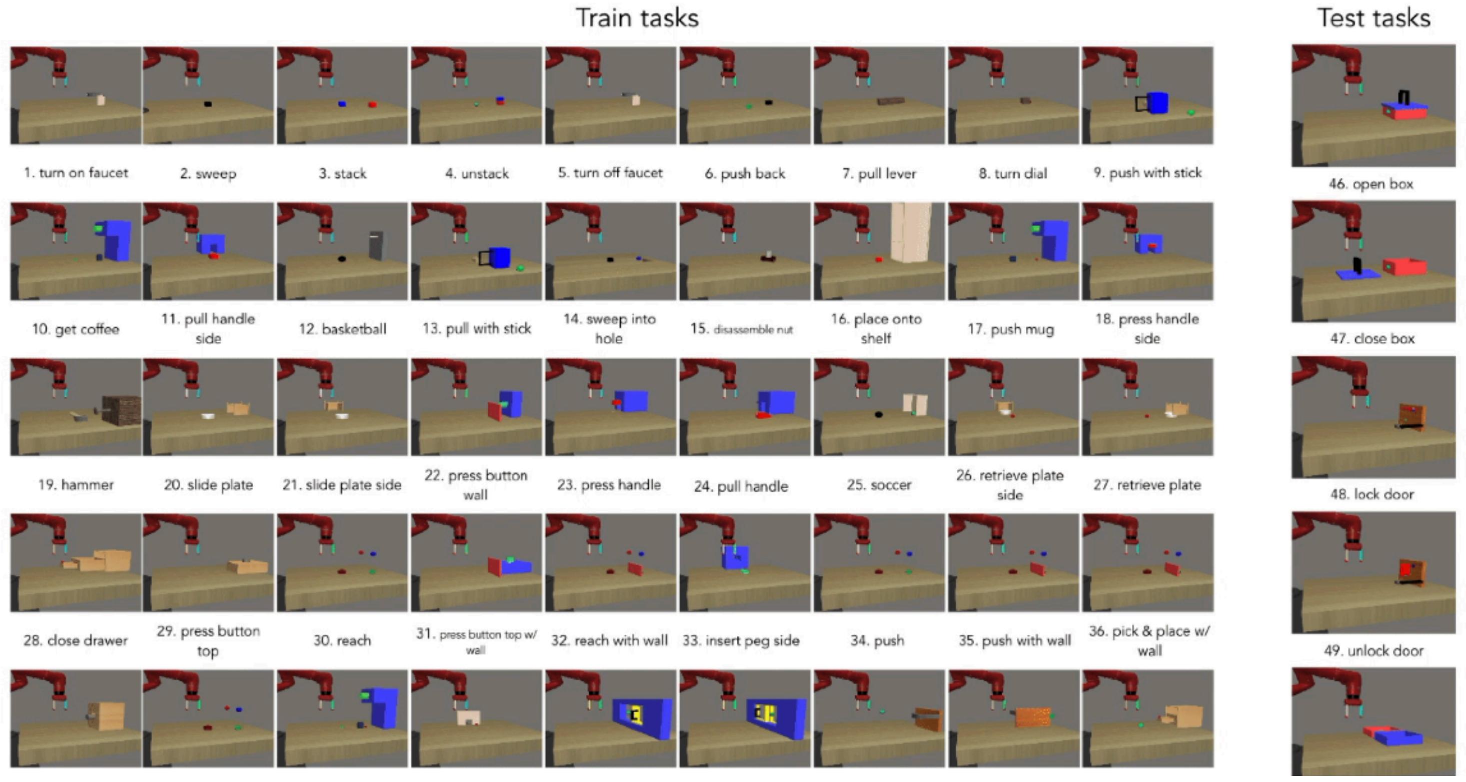
Optimization challenges —> three conditions seem to plague MTL, MTRL

a solution: project conflicting gradients (PCGrad)

Want to Learn More?

CS330: Deep Multi-Task & Meta-Learning
Lecture videos online!

Working on Meta-RL?



Try out the Meta-World benchmark

Collaborators


IRIS - RAIL retreat in Sonoma, CA

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. '19
T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. **Meta-World**. CoRL '19
T Yu, S Kumar, A Gupta, S Levine, K Hausman, C Finn. **Gradient Surgery for Multi-Task Learning**. '19