Model-Agnostic Meta-Learning

Universality, Inductive Bias, and Weak Supervision

Chelsea Finn

BAIR
BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Why Learn to Learn?

- effectively **reuse data** on other tasks
- **replace manual engineering** of architecture, hyperparameters, etc.
- learn to **quickly adapt to unexpected scenarios** (inevitable failures, long tail)
- learn how to learn **with weak supervision**

Problem Domains:
- few-shot classification & generation
- hyperparameter optimization
- architecture search
- faster reinforcement learning
- domain generalization
- learning structure
- …

Approaches:
- recurrent networks
- learning optimizers or update rules
- learning initial parameters & architecture
- acquiring metric spaces
- Bayesian models
- …

What is the meta-learning problem statement?

The Meta-Learning Problem

Supervised Learning:

Inputs: $\mathbf{x}$ Outputs: $\mathbf{y}$ Data: $\{(\mathbf{x}, \mathbf{y})_i\}$

$$\mathbf{y} = f(\mathbf{x}; \theta)$$

Meta-Supervised Learning:

Inputs: $\underbrace{\mathcal{D}_{\text{train}}}_{\{(\mathbf{x}, \mathbf{y})_{1:K}\}}$ $\mathbf{x}_{\text{test}}$ Outputs: $\mathbf{y}_{\text{test}}$ Data: $\{\mathcal{D}_i\}$

$$\mathcal{D}_i : \{(\mathbf{x}, \mathbf{y})_j\}$$

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

Why is this view useful?

Reduces the problem to the design & optimization of *f*.

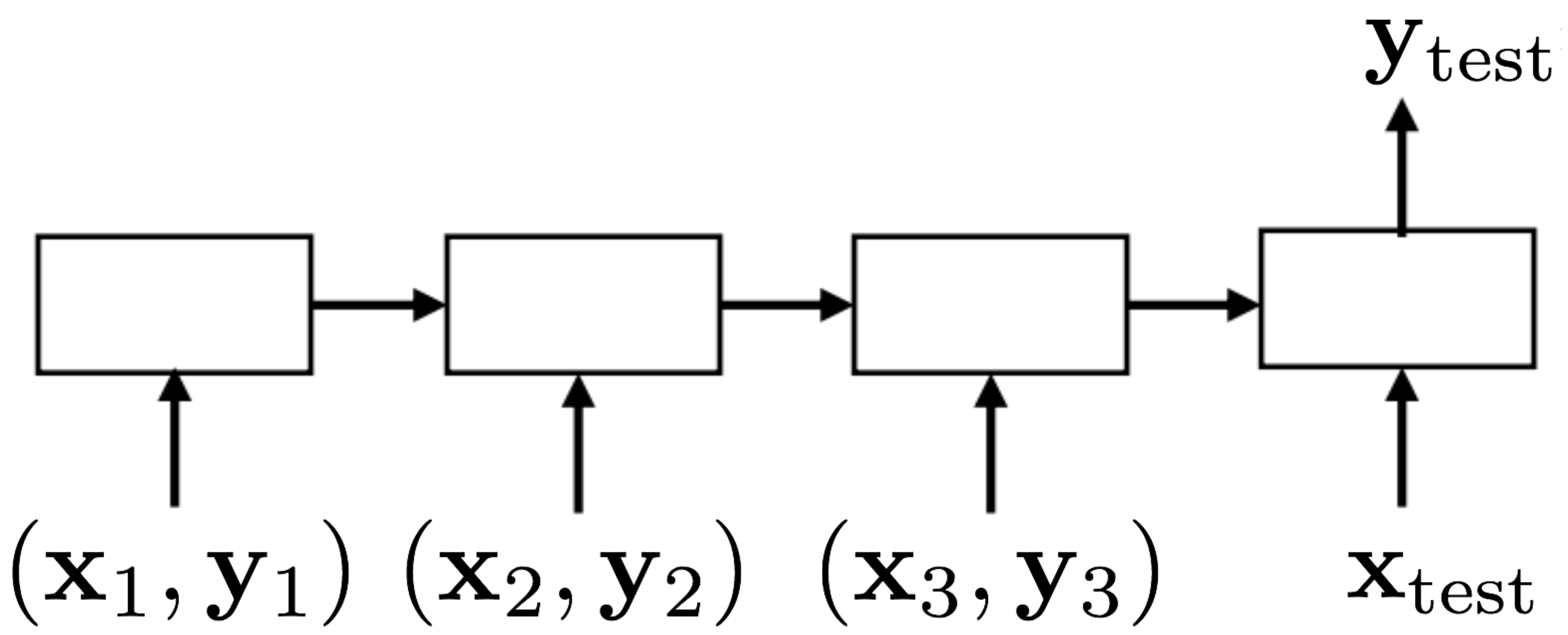
Design of f?

$\mathcal{D}_{\text{train}} \; \mathbf{x}_{\text{test}} \longrightarrow \mathbf{y}_{\text{test}}$

Recurrent network
(LSTM, NTM, Conv)

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

Santoro et al. '16, Duan et al. '17, Wang et al. '17, Munkhdalai & Yu '17, Mishra et al. '17, …



Design of f ?

$$\mathcal{D}_{\text{train}}\ \mathbf{x}_{\text{test}} \longrightarrow \mathbf{y}_{\text{test}}$$

Recurrent network
(LSTM, NTM, Conv)

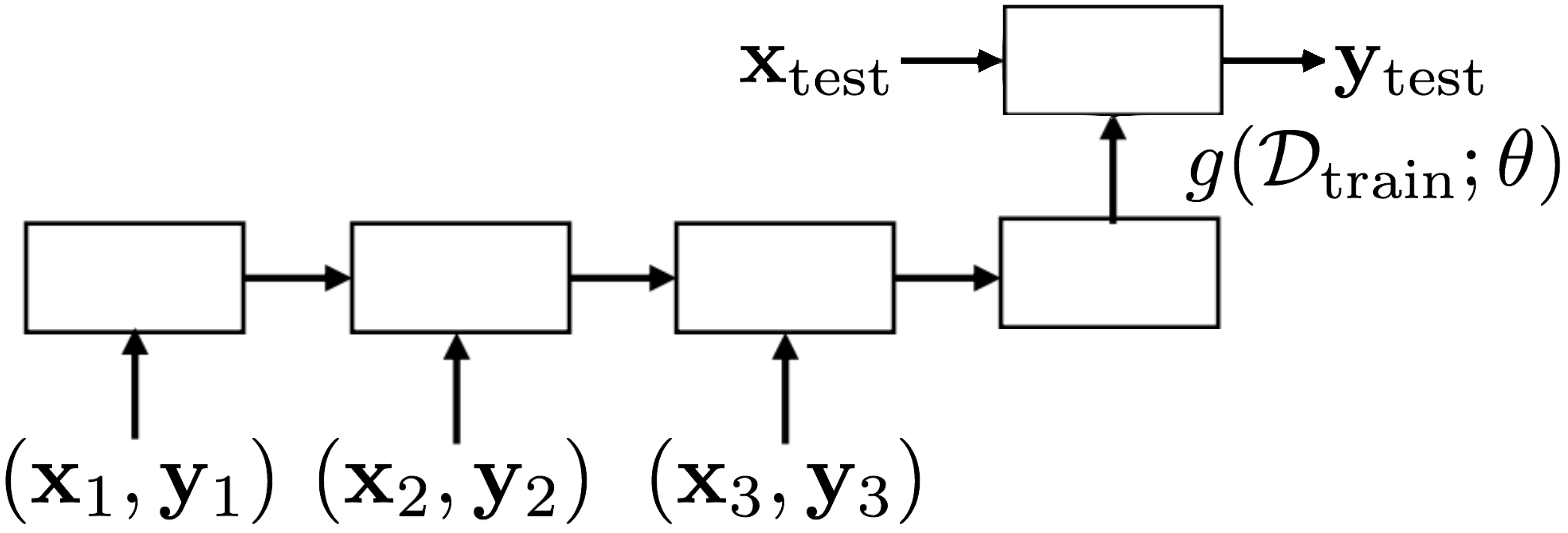
$$\mathbf{y}_{\text{test}} = f(\mathcal{D}_{\text{train}}, \mathbf{x}_{\text{test}}; \theta)$$

Santoro et al. '16, Duan et al. '17, Wang et al. '17, Munkhdalai & Yu '17, Mishra et al. '17, …

Learned optimizer
(often uses recurrence)

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; g(\mathcal{D}_{\text{train}}; \theta))$$

Schmidhuber et al. '87, Bengio et al. '90, Hochreiter et al. '01, Li & Malik '16, Andrychowicz et al. '16, Ha et al. '17, Ravi & Larochelle '17, …



Design of f?

$$\mathcal{D}_{\text{train}} \; \mathbf{x}_{\text{test}} \longrightarrow \mathbf{y}_{\text{test}}$$

Recurrent network
(LSTM, NTM, Conv)

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

Santoro et al. '16, Duan et al. '17, Wang et al. '17, Munkhdalai & Yu '17, Mishra et al. '17, …

Learned optimizer
(often uses recurrence)

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; g(\mathcal{D}_{\text{train}}; \theta))$$

Schmidhuber et al. '87, Bengio et al. '90, Hochreiter et al. '01, Li & Malik '16, Andrychowicz et al. '16, Ha et al. '17, Ravi & Larochelle '17, …

These approaches are general and quite powerful.

What happens when the task is very different? Or very little meta-training?

Impose Structure

Bergstra et al. '11, Snoek et al. '12, Koch '15, Maclaurin et al. '15, Vinyals et al. '16, Zoph & Le '17, Snell et al. '17, …

Can we build a general meta-learning algorithm that interpolates between **learning from scratch** and **few-shot** learning?

pretrained parameters

fine-tuning: $$\theta' \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta)$$

[test-time]

test task

Model-**A**gnostic **M**eta-**L**earning: **(MAML)**

$$\min_\theta \sum_{\text{tasks}} \mathcal{L}_{\text{v}}(\theta - \alpha \nabla_\theta \mathcal{L}_{\text{tr}}(\theta))$$

Key idea: Train over many tasks, to learn parameter vector θ that transfers

In-distribution task: k-shot learning

Base case: learning from scratch

Related but out-of-distribution task: somewhere in between

Finn, Abbeel, Levine ICML '17

Design of f ?

$$\mathcal{D}_{\text{train}} \; \mathbf{x}_{\text{test}} \longrightarrow \mathbf{y}_{\text{test}}$$

Recurrent network
(LSTM, NTM, Conv)

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

Santoro et al. '16, Duan et al. '17, Wang et al. '17, Munkhdalai & Yu '17, Mishra et al. '17, …

Learned optimizer
(often uses recurrence)

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; g(\mathcal{D}_{\text{train}}; \theta))$$

Schmidhuber et al. '87, Bengio et al. '90, Hochreiter et al. '01, Li & Malik '16, Andrychowicz et al. '16, Ha et al. '17, Ravi & Larochelle '17, …

Impose Structure

Bergstra et al. '11, Snoek et al. '12, Koch '15, Maclaurin et al. '15, Vinyals et al. '16, Zoph & Le '17, Snell et al. '17, …

MAML
(learned initialization)

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; \theta - \alpha \nabla_{\theta} \mathcal{L}(\mathcal{D}_{\text{train}}))$$

Finn et al. '17, Grant et al. '17, Reed et al. '17, Li et al. '17, …

Theoretical & Empirical Questions

1. What happens when MAML faces **out-of-distribution tasks**?
2. How **expressive** are deep representations + gradient descent?
3. Can we interpret MAML in a **probabilistic framework**?
4. Can we use MAML to learn from **weak supervision**?

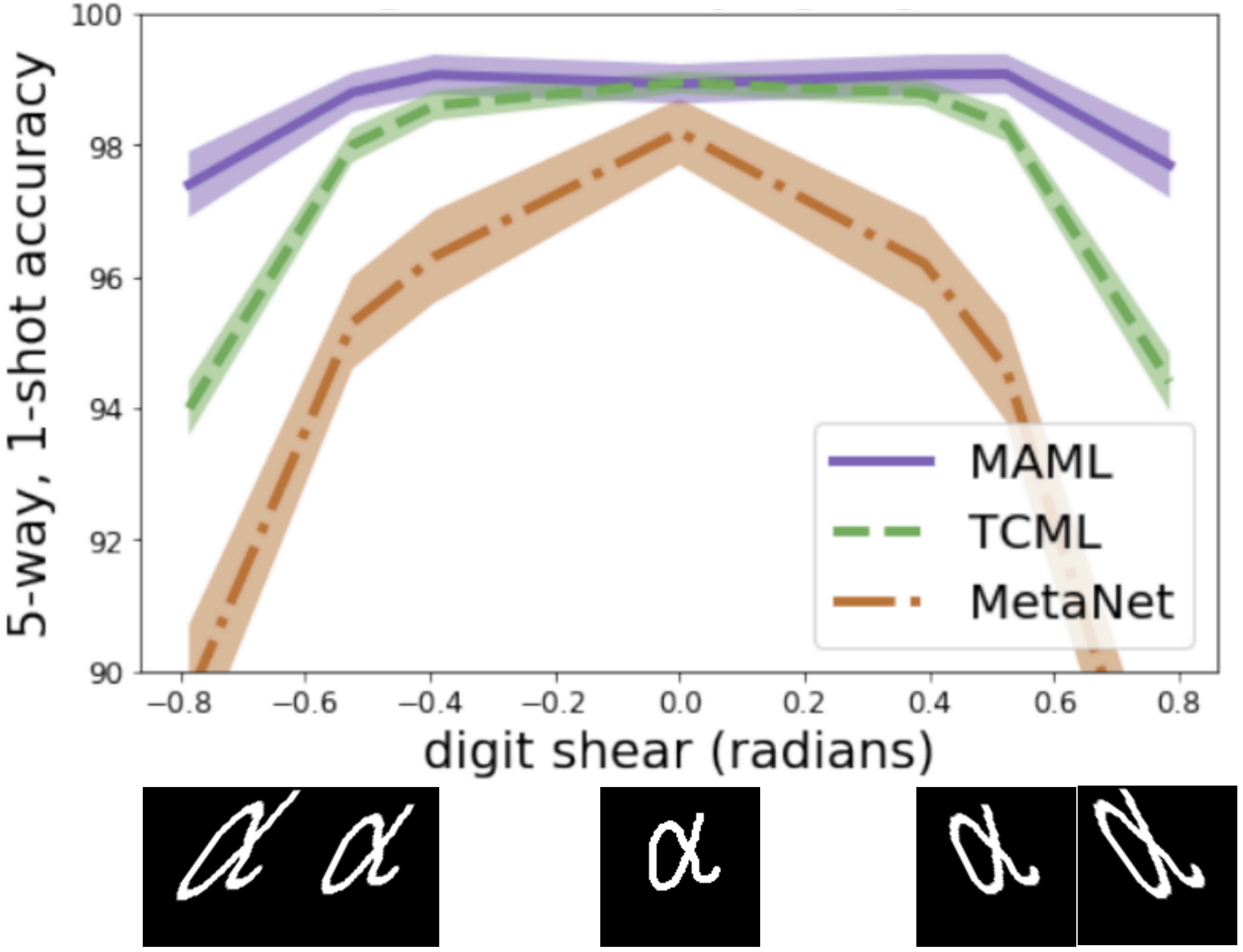
How well can methods generalize to similar, but extrapolated tasks?

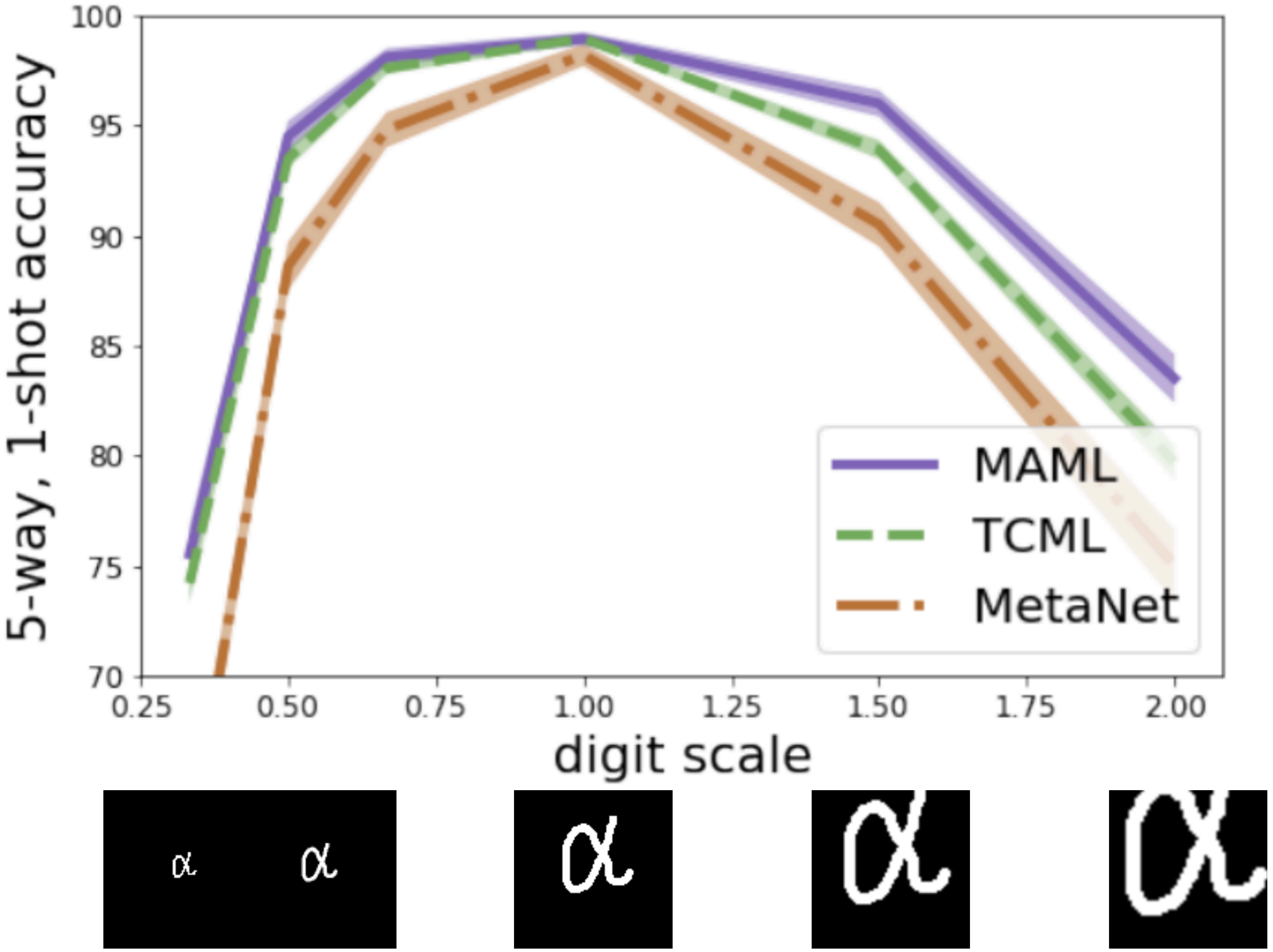
The world is non-stationary.

MAML **TCML, MetaNetworks**

Omniglot image classification

performance





task variability

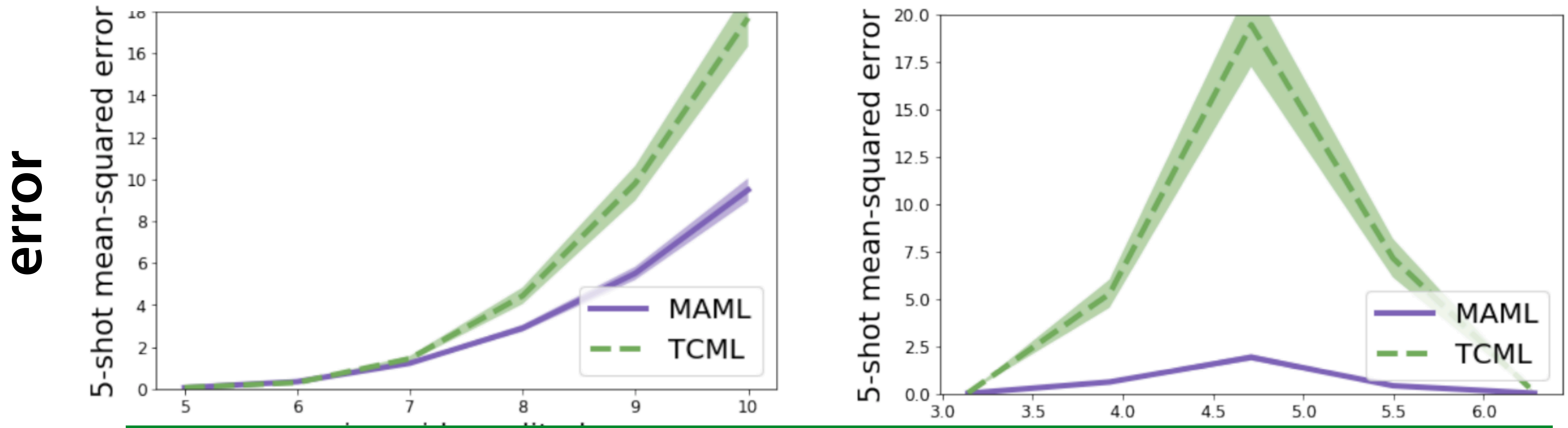
Finn & Levine ’17 (under review)

How well can methods generalize to similar, but extrapolated tasks?

The world is non-stationary.

MAML TCML

Sinusoid curve regression



Takeaway: Strategies learned with MAML consistently generalize better to out-of-distribution tasks

Finn & Levine '17 (under review)

Theoretical & Empirical Questions

1. What happens when MAML faces **out-of-distribution tasks**?
2. How **expressive** are deep representations + gradient descent?
3. Can we interpret MAML in a **probabilistic framework**?
4. Can we use MAML to learn from **weak supervision**?

Universal Function Approximation Theorem

Hornik et al. '89, Cybenko '89, Funahashi '89

A neural network with one hidden layer of finite width can approximate any continuous function.

$$\mathbf{y} = f(\mathbf{x}; \theta)$$

"universal function approximator"

How can we define a notion of universality / expressive power for meta-learning?

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

"universal learning procedure approximator"

Recurrent network

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

Learned optimizer

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; g(\mathcal{D}_{\text{train}}; \theta))$$

With sufficient depth, both are universal learning procedure approximators.

Are we losing expressive power when using MAML?

Finn & Levine '17 (under review)

How expressive is MAML?

$$\mathbf{y}_{\text{test}} = f(\mathbf{x}_{\text{test}}; \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{D}_{\text{train}}))$$

Assumptions:
- cross entropy or mean-squared error loss
- datapoints $\mathbf{x}_i$ in training dataset are unique

Result: For a sufficiently deep f_θ, $f(\mathbf{x}_{\text{test}}; \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{D}_{\text{train}}))$ is a universal learning procedure approximator.

[It can approximate any function of $\mathcal{D}_{\text{train}}$ $\mathbf{x}_{\text{test}}$]

Why is this interesting?

MAML has both benefits of inductive bias and expressive power.

Finn & Levine ’17 (under review)

Theoretical & Empirical Questions

1. What happens when MAML faces **out-of-distribution tasks**?
2. How **expressive** is deep representation + gradient descent?
3. Can we interpret MAML in a **probabilistic framework**?
4. Can we use MAML to learn from **weak supervision**?

Can we interpret MAML in a probabilistic framework?

meta-learning ≈ learning a prior

Bayesian concept learning

[Tenenbaum '99, Fei-Fei et al. '03, Lawrence & Platt '04, …]

formulate few-shot learning as probabilistic inference problem

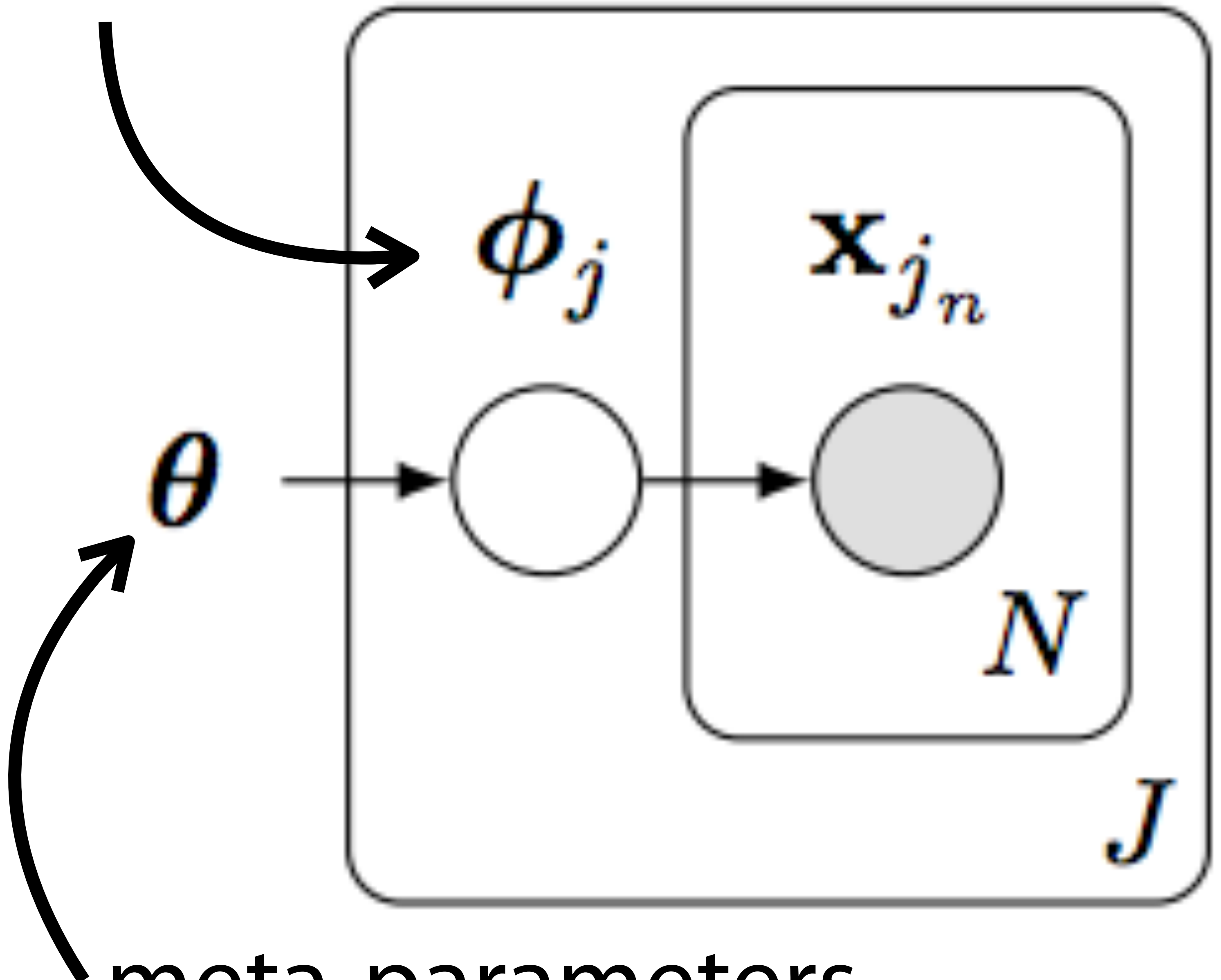
+ can effectively generalize from limited evidence

- hard to scale to complex models

Can we interpret MAML in a probabilistic framework?

Bayesian meta-learning approach

task-specific parameters



meta-parameters

$$\max_{\theta} \prod_j p(\mathcal{D}_{\text{train}}^{(j)} | \theta)$$

$$= \prod_j \int p(\mathcal{D}_{\text{train}}^{(j)} | \phi_j) p(\phi_j | \theta) d\phi_j$$

(empirical Bayes)

$$\approx \prod_j p(\mathcal{D}_{\text{train}}^{(j)} | \hat{\phi}_j) p(\hat{\phi}_j | \theta)$$

MAP estimate

How to compute MAP estimate?

Gradient descent with early stopping = MAP inference under Gaussian prior with mean at initial parameters [Santos '96]

(exact in linear case, approximate in nonlinear case)

MAML approximates hierarchical Bayesian inference. [Grant et al. '17]

Erin Grant

Theoretical & Empirical Questions

1. What happens when MAML faces **out-of-distribution tasks**?
2. How **expressive** is deep representation + gradient descent?
3. Can we interpret MAML in a **probabilistic framework**?
4. Can we use MAML to learn from **weak supervision**?

Learning to Learn from Weak Supervision

Meta-Supervised Learning:

Inputs: $\boxed{\mathcal{D}_{\text{train}}}$ $\boxed{\mathbf{x}_{\text{test}}}$ Outputs: $\boxed{\mathbf{y}_{\text{test}}}$ Data: $\{\mathcal{D}_i\}$

weakly supervised

fully supervised

During meta-training: access full supervision for each task

During meta-testing: only use weakly-supervised datapoints

With MAML: $\min_\theta \sum \mathcal{L}_{\text{v}}(\theta - \alpha \nabla_\theta \mathcal{L}_{\text{tr}}(\theta))$

Key insight: **inner loss** can be different than **outer loss**

Weak Supervision Results

- **Learning from positive examples**
  Grant, Finn, Peterson, Abbott, Levine, Darrell, Griffiths, NIPS '17 CIAI workshop
- **One-shot Imitation from human video**
  (in preparation, with Yu, Abbeel, Levine)

Typical Objective of Few-Shot Learning

Image recognition

Given 1 example of 5 classes:

Classify new examples

Human Concept Learning

Given *1 positive example*:

Classify new examples:

Beyond how humans learn, this setting is also more interesting.

Grant et al. ’17 (NIPS CIAI workshop)

Human Concept Learning

Given *1 positive example*:

Classify new examples:

$$\min_{\theta} \sum \mathcal{L}_{\mathrm{v}}(\theta - \alpha \nabla_{\theta} \mathcal{L}_{\mathrm{tr}}(\theta))$$

only positive examples

both positive & negatives

Why does this make sense?

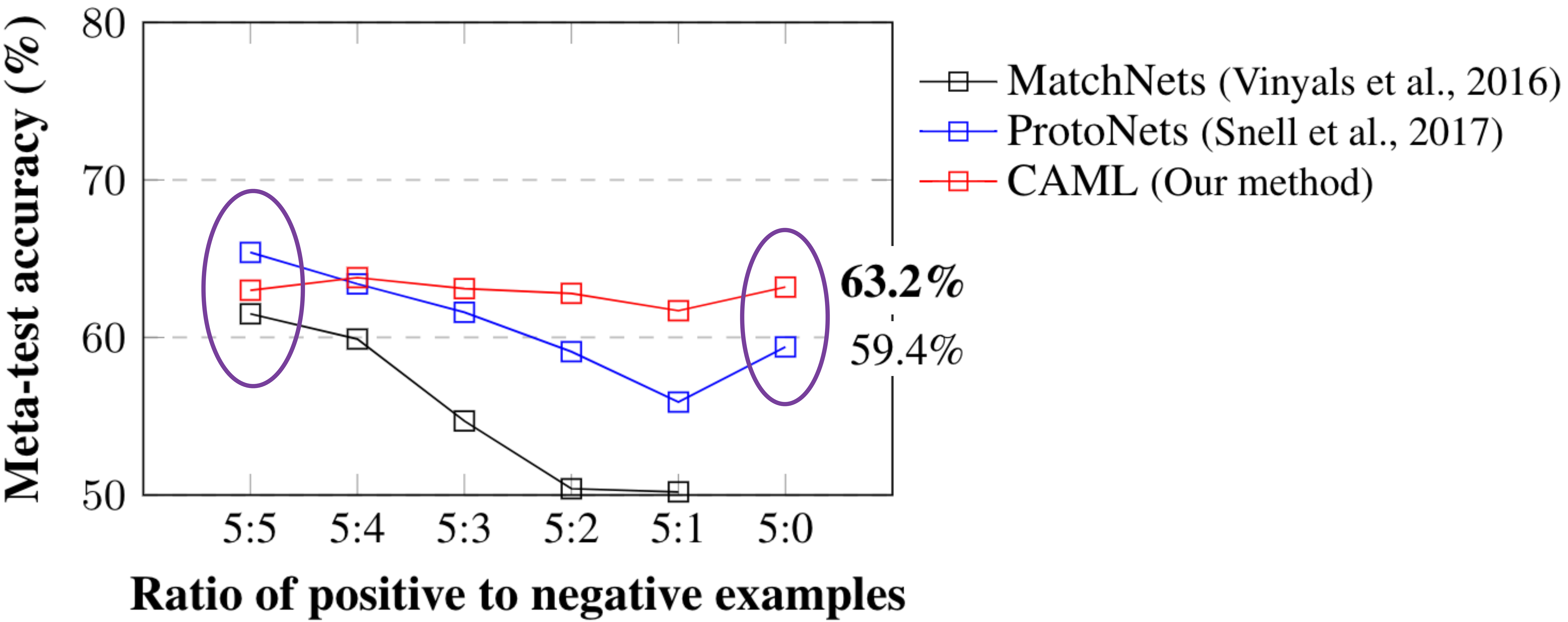
MAML approximates hierarchical Bayesian inference

Concept **A**cquisition through **M**eta-**L**earning (CAML)

Grant et al. ’17 (NIPS CIAI workshop)

Few-Shot Image Classification from Positive Examples

MiniImagenet dataset



Grant et al. '17 (NIPS CIAI workshop)

One-Shot **Visual** Imitation Learning

Goal: Given one visual demonstration of a new task, learn a policy

Visual imitation is expensive.

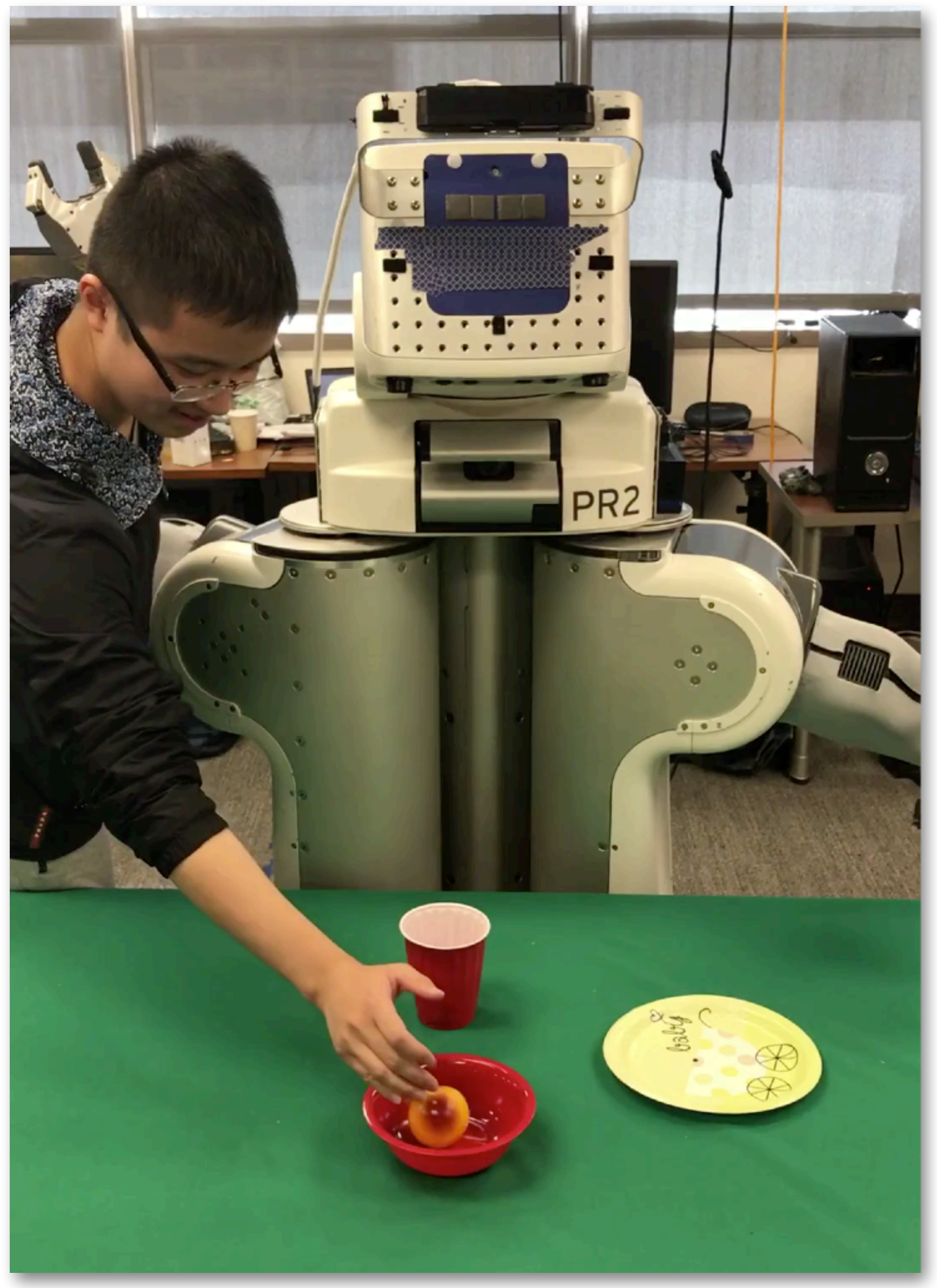
behavior cloning / supervised learning

Rahmanizadeh et al. '17   Zhang et al. '17

learns from raw pixels,

but requires many demonstrations

No direct supervision signal
in video of human.



Through meta-learning: reuse data from other tasks/objects/envionrments

Yu*, Finn*, et al. (in prep.)

One-Shot Visual Imitation from Humans

imitation loss

$$\mathcal{L} = \sum_t \|\pi_\theta(o_t) - a_t^*\|^2$$

meta-training time $$\min_\theta \sum_{\text{tasks}} \mathcal{L}_{\text{v}}(\theta - \alpha \nabla_\theta \mathcal{L}_{\text{tr}}(\theta))$$

meta-training tasks

val demo (robot demo)

training demo (video of human)

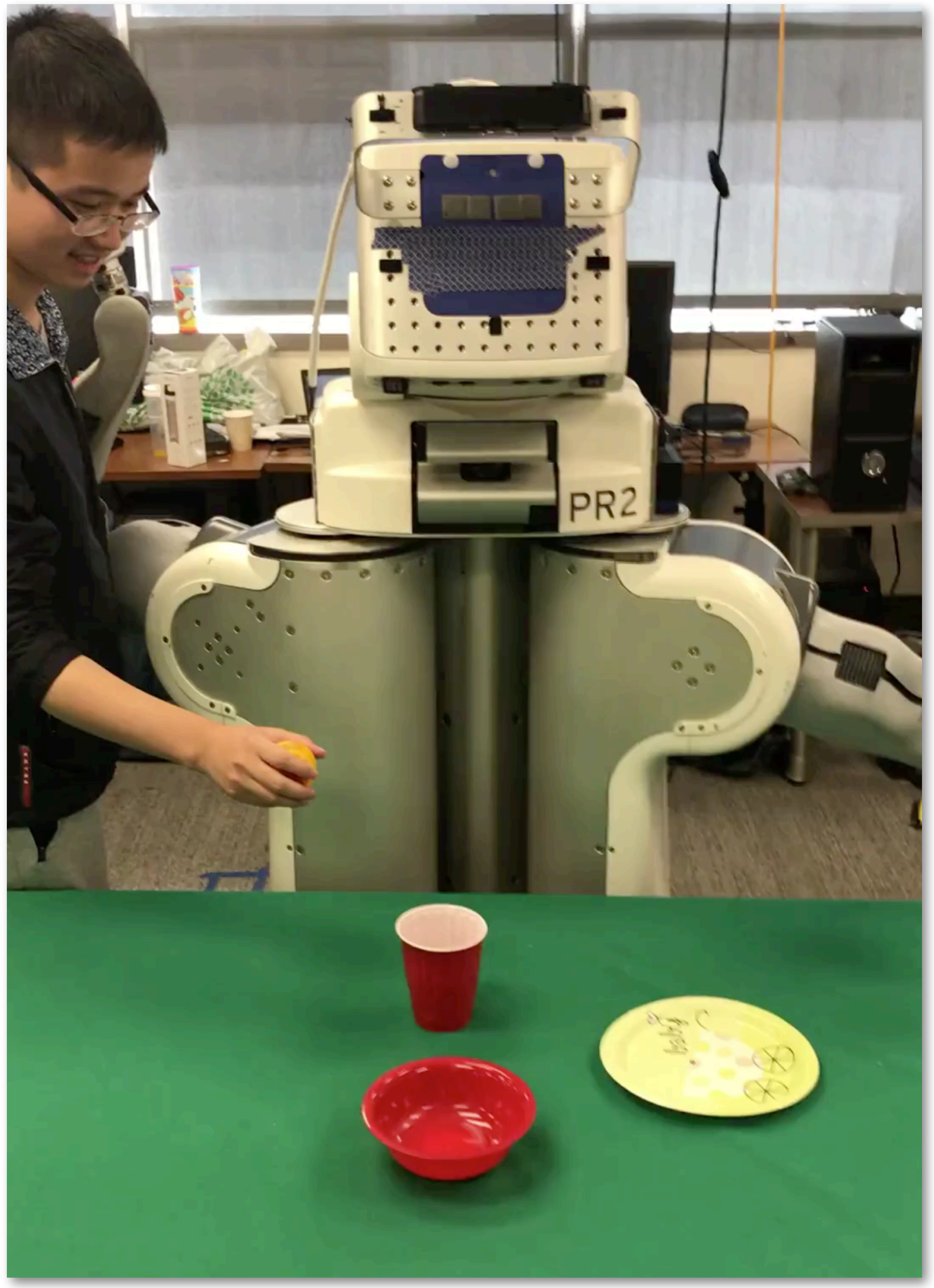
meta-test time $$\theta' \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta)$$

demo of meta-test task (video of human)

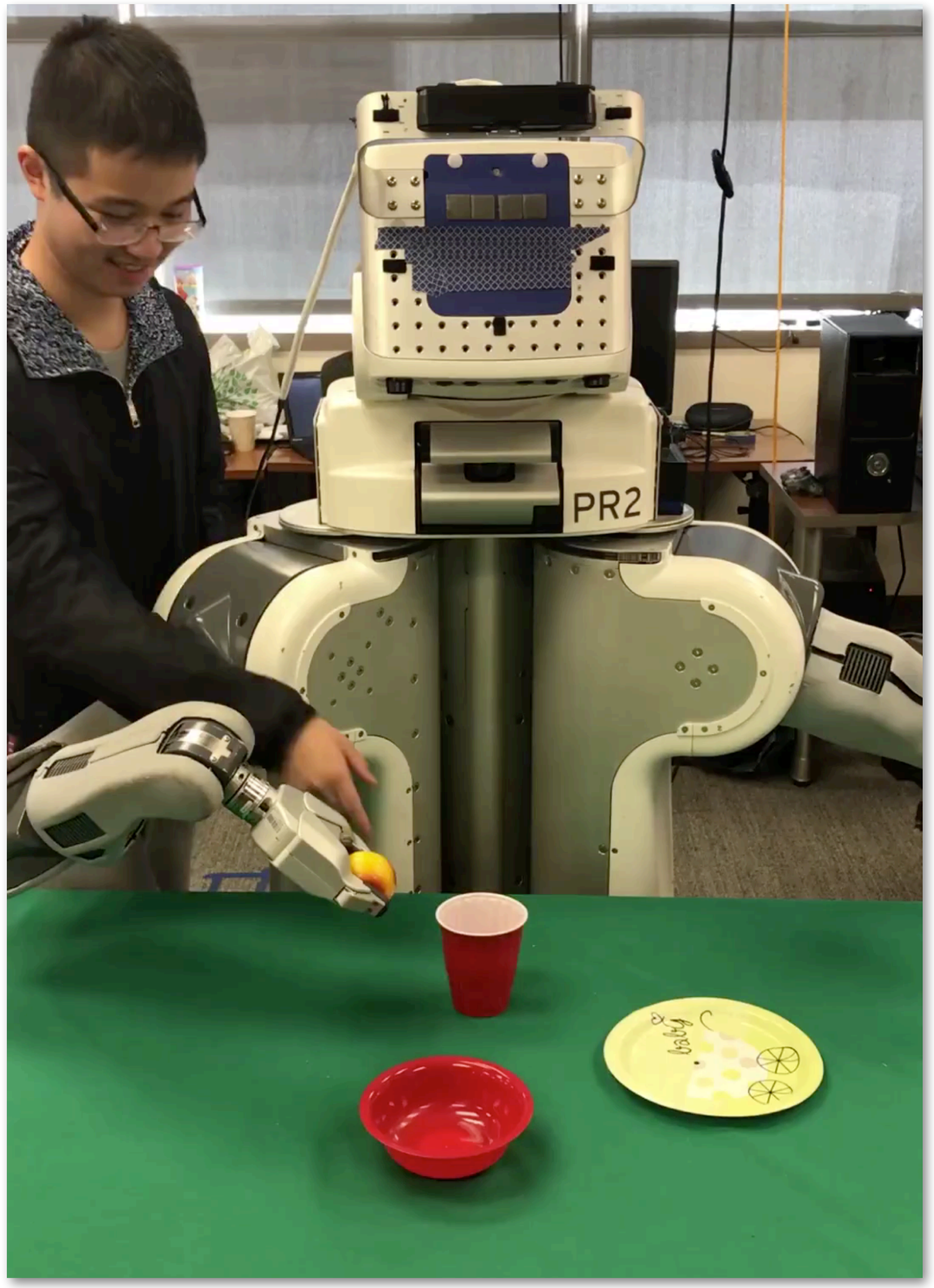
Yu*, Finn*, et al. (in prep.)

On-going work: One-shot imitation from human video

input human demo



resulting policy



Yu*, Finn*, et al. (in prep.)

Takeaways

- Meta-learning can be seen as learning a function

$$\mathcal{D}_{\text{train}} \ \mathbf{x}_{\text{test}} \longrightarrow \mathbf{y}_{\text{test}}$$

- Embedding gradient descent provides beneficial **inductive bias** while maintaining **universality**
- MAML is equivalent to **empirical Bayes**
- Can learn how to learn from "weak" supervision

From *1 positive example*:

From a *video of a human*:

Collaborators

Sergey Levine

Pieter Abbeel

Tianhe Yu

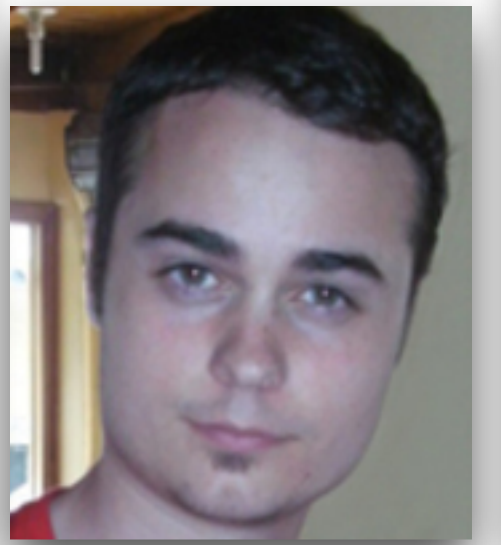
Tianhao Zhang

Erin Grant

Josh Abbott

Tom Griffiths

Josh Peterson



Trevor Darrell

Blog post, code, and papers: eecs.berkeley.edu/~cbfinn

Questions?