What’s Wrong with Meta-Learning
(and how we might fix it)

Sergey Levine
UC Berkeley
Google Brain
Visual Distractors

real time

autonomous execution
Training Phase
Four robots collectively train a single door opening policy. 1x speed
Kalashnikov, Irpan, Pastor, Ibarz, Herzong, Jang, Quillen, Holly, Kalakrishnan, Vanhoucke, Levine. QT-Opt: Scalable Deep Reinforcement Learning of Vision-Based Robotic Manipulation Skills
can we transfer past experience in order to learn how to learn?

people can learn new skills extremely quickly

how?

we never learn from scratch!

can we transfer past experience in order to learn how to learn?
The meta-learning/few-shot learning problem

A simpler, model-agnostic, meta-learning method

Unsupervised meta-learning
The meta-learning/few-shot learning problem

A simpler, *model-agnostic*, meta-learning method

*Unsupervised* meta-learning
Few-shot learning: problem formulation in pictures

image credit: Ravi & Larochelle ‘17
Few-shot learning: problem formulation in equations

- How to read in training set?
  - Many options, RNNs can work

- Many examples of training set:
  - Image 1, Label 1
  - Image 2, Label 2
  - Image 3, Label 3

Supervised learning: $f(x) \rightarrow y$
- Input (e.g., image)
- Output (e.g., label)

Supervised meta-learning: $f(D_{\text{train}}, x) \rightarrow y$
- Training set

Test input:
- $x_{\text{test}}$

Test label:
- $y_{\text{test}}$
Some examples of representations

Santoro et al. “Meta-Learning with Memory-Augmented Neural Networks.”

Vinyals et al. “Matching Networks for One-Shot Learning”

Snell et al. “Prototyping Networks for Few-Shot Learning”

...and many many many others!
What kind of algorithm is learned?

RNN-based meta-learning

\[
(x_1, y_1) \rightarrow (x_2, y_2) \rightarrow (x_3, y_3) \rightarrow x_{\text{test}} \rightarrow y_{\text{test}}
\]

test input

test label

this implements the “learned learning algorithm”

- Does it converge?
  - Kind of?
- What does it converge to?
  - Who knows...
- What to do if it’s not good enough?
  - Nothing...
The meta-learning/few-shot learning problem

A simpler, *model-agnostic*, meta-learning method

*Unsupervised* meta-learning
Let’s step back a bit...

is pretraining a *type* of meta-learning?
better features = faster learning of new task!
Model-agnostic meta-learning

a general recipe:

\[
\theta \leftarrow \theta - \beta \sum_i \nabla_{\theta} L(\theta - \alpha \nabla_{\theta} L(\theta, D_{\text{train}}^i), D_{\text{test}}^i)
\]

“meta-loss” for task \(i\)

* in general, can take more than one gradient step here
** we often use 4 – 10 steps

Finn et al., “Model-Agnostic Meta-Learning”
What did we just do?

supervised learning: \( f(x) \to y \)

supervised meta-learning: \( f(\mathcal{D}_{\text{train}}, x) \to y \)

model-agnostic meta-learning: \( f_{\text{MAML}}(\mathcal{D}_{\text{train}}, x) \to y \)

\[ f_{\text{MAML}}(\mathcal{D}_{\text{train}}, x) = f_{\theta'}(x) \]

\[ \theta' = \theta - \alpha \sum_{(x,y) \in \mathcal{D}_{\text{train}}} \nabla_{\theta} \mathcal{L}(f_\theta(x), y) \]

Just another computation graph...
Can implement with any autodiff package (e.g., TensorFlow)
Why does it work?

RNN-based meta-learning

- Does it converge?
  - Kind of?
- What does it converge to?
  - Who knows...
- What to do if it’s not good enough?
  - Nothing...

MAML

- Does it converge?
  - Yes (it’s gradient descent...)
- What does it converge to?
  - A local optimum (it’s gradient descent...)
- What to do if it’s not good enough?
  - Keep taking gradient steps (it’s gradient descent...)
Universality

Did we lose anything?

**Universality**: meta-learning can learn any “algorithm”
more precisely, can represent any function $f(D_{\text{train}}, x)$

---

Finn & Levine. “Meta-Learning and Universality”
Model-agnostic meta-learning: forward/backward locomotion after MAML training

- after 1 gradient step (forward reward)
- after 1 gradient step (backward reward)
Related work

Maclaurin et al. “Gradient-based hyperparameter optimization”

Li & Malik. “Learning to optimize”

Andrychowicz et al. “Learning to learn by gradient descent by gradient descent.”

Ravi & Larochelle. “Optimization as a model for few-shot learning”

...and many many many others!
Follow-up work

Program Synthesis

Question: How many CFL teams are from York College?

SQL: SELECT COUNT CFL Team FROM CFL where College = "York"

Result: 8

Huang, Wang, Singh, Yih, He NAACL ’18

Learning to Learn Distributions

Reed, Chen, Paine, van den Oord, Eslami, Rezende, Vinyals, de Freitas ICLR ’18

Chen, Dong, Li, He arXiv ’18

Finn et al. ’17: 63.11%
Li et al. ’17: 64.03%
Kim et al. ’18 (AutoMeta): 76.29%

Federated Learning

Multi-Agent Competitions

Al-Shedivat, Bansal, Burda, Sutskever Mordatch, Abbeel ICLR ’18

Learning the learning rate

Li, Zhou, Chen, Li arXiv ’17

Masked Transformations

Lee & Choi arXiv ’18

Domain Generalization

Li, Yang, Song, Hospedales AAAI ’18

Semi-Supervised Few-Shot Learning

Boney & Ilin ICLR workshop track ’18

MinilImagenet few-shot benchmark: 5-shot 5-way

Finn et al. ’17: 63.11%
Li et al. ’17: 64.03%
Kim et al. ’18 (AutoMeta): 76.29%

...and the results keep getting better
The meta-learning/few-shot learning problem

A simpler, *model-agnostic*, meta-learning method

*Unsupervised* meta-learning
Let’s Talk about Meta-Overfitting

• Meta learning requires task distributions
• When there are too few meta-training tasks, we can meta-overfit
• Specifying task distributions is hard, especially for meta-RL!
• Can we propose tasks automatically?
A General Recipe for Unsupervised Meta-RL

Random Task Proposals

- Use randomly initialize discriminators for reward functions

\[ R(s, z) = \log p_D(z|s) \]

D \rightarrow \text{randomly initialized network}

- Important: Random functions over state space, \textbf{not} random policies
Diversity-Driven Proposals

Task Reward for UML: \( R(s, z) = \log p_D(z|s) \)

Eysenbach, Gupta, Ibarz, Levine. Diversity is All You Need.
Examples of Acquired Tasks

Cheetah

Ant

Eisenbach, Gupta, Ibarz, Levine. Diversity is All You Need.
Does it work?


Meta-test performance with rewards
What about supervised learning?
Can we meta-train on only *unlabeled* images?

But... does it outperform unsupervised learning?

Hsu, Levine, Finn. Unsupervised Learning via Meta-Learning.
Results: unsupervised meta-learning

- unsupervised learning
- task proposals
- meta-learning

A few choices:
- BiGAN – Donahue et al. ‘17
- DeepCluster – Caron et al. ‘18

Clustering to Automatically Construct Tasks for Unsupervised Meta-Learning (CACTUs)

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BiGAN KNN</td>
<td>31.10%</td>
</tr>
<tr>
<td>BiGAN logistic</td>
<td>33.91%</td>
</tr>
<tr>
<td>BiGAN MLP + dropout</td>
<td>29.06%</td>
</tr>
<tr>
<td>BiGAN cluster matching</td>
<td>29.49%</td>
</tr>
<tr>
<td>CACTUs</td>
<td>51.28%</td>
</tr>
<tr>
<td>DeepCluster</td>
<td>53.97%</td>
</tr>
</tbody>
</table>

Same story across:
- 3 different embedding methods
- 4 datasets (Omniglot, miniImageNet, CelebA, MNIST)

Hsu, Levine, Finn. Unsupervised Learning via Meta-Learning.
The meta-learning/few-shot learning problem

A simpler, *model-agnostic*, meta-learning method

*Unsupervised* meta-learning
What’s next?

Probabilistic meta-learning: learn to sample *multiple hypotheses*


Meta-learning online learning & continual learning


Instruction: Move blue triangle to green goal.

Correction 1: Enter the blue room.
Correction 2: Enter the red room.

Meta-learning to interpret weak supervision and natural language

