2019 CS420 Machine Learning, Lecture 16

Meta Learning

Weinan Zhang

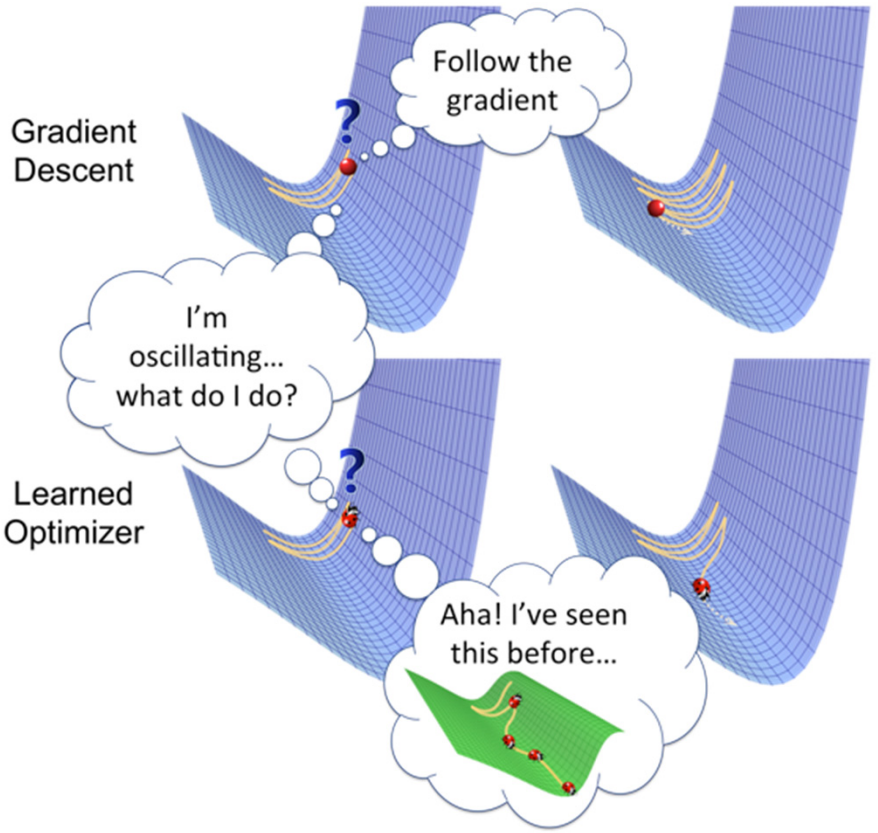
Shanghai Jiao Tong University

http://wnzhang.net

http://wnzhang.net/teaching/cs420/index.html

What is Meta-Learning?

- If you have learned 100 tasks, given a new task, can you figure out how to learn more efficiently?
  - Now have multiple tasks is a huge advantage!
- Meta-learning is very close to multi-task learning
- Meta-learning = learning to learn



Slide credit: Sergey Levine

https://bair.berkeley.edu/blog/2017/09/12/learning-to-optimize-with-rl/

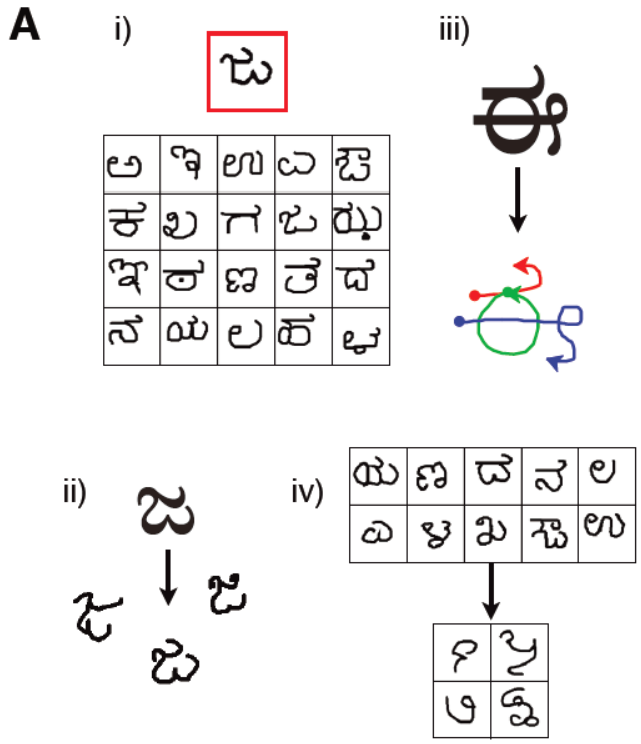
Early Approaches to Meta-Learning

- Jürgen Schmidhuber
  - Genetic Programming. PhD thesis. 1987
  - Learning to control fast-weight memories: An alternative to dynamic recurrent networks. Neural Computation 1992
  - A neural network that embeds its own meta-levels. ICNN 1993.

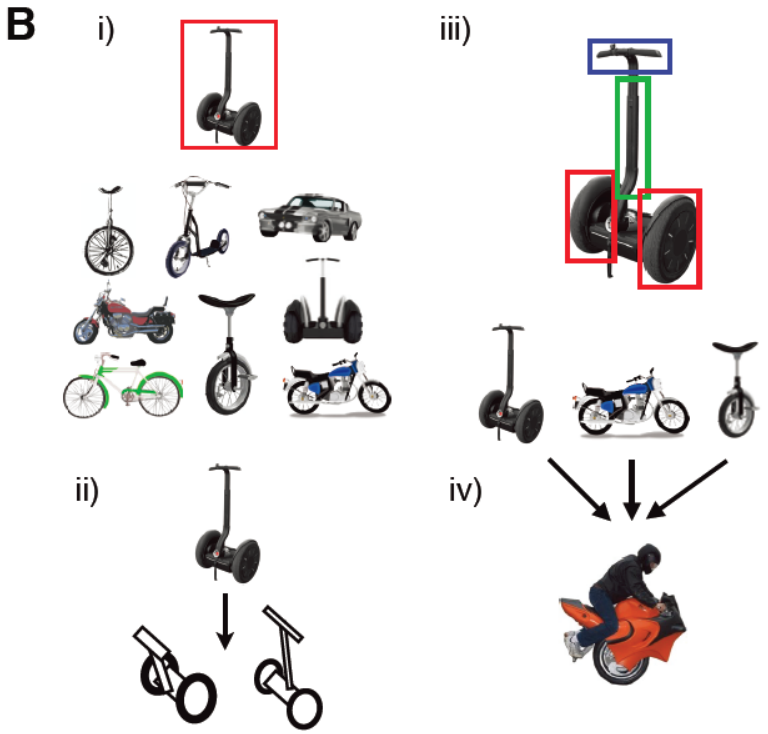
- Yoshua Bengio
  - Learning a synaptic learning rule. Univ. Montreal. 1990.
  - On the search for new learning rules for ANN. Neural Processing Letters 1992
    - Learning update rules with SGD

Meta-Learning is Beyond Traditional ML

- Lake et al. have argued forcefully for its importance as a building block in artificial intelligence



human-level learning of a novel handwritten characters

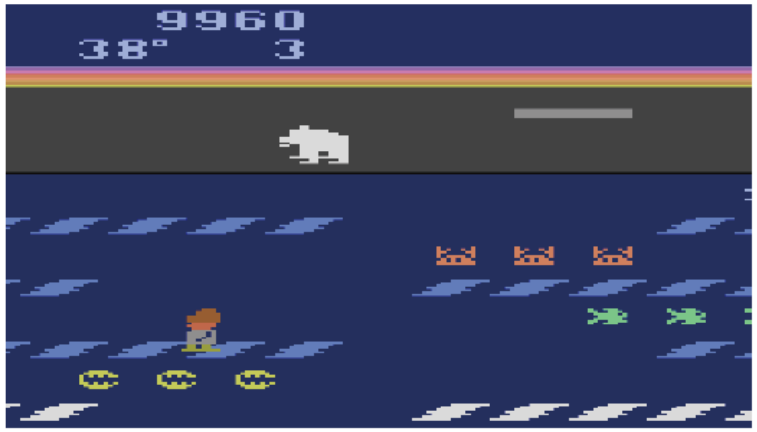


with the same abilities also illustrated for a novel two-wheeled vehicle

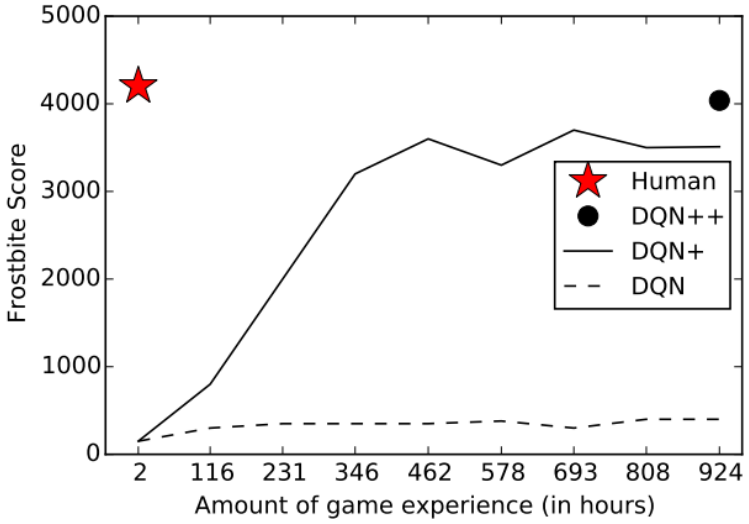
Lake, Brenden M., et al. "Building machines that learn and think like people." Behavioral and Brain Sciences 40 (2017).

Meta-Learning is Beyond Traditional ML

- Lake et al. have argued forcefully for its importance as a building block in artificial intelligence



Atari 2600 game Frostbite

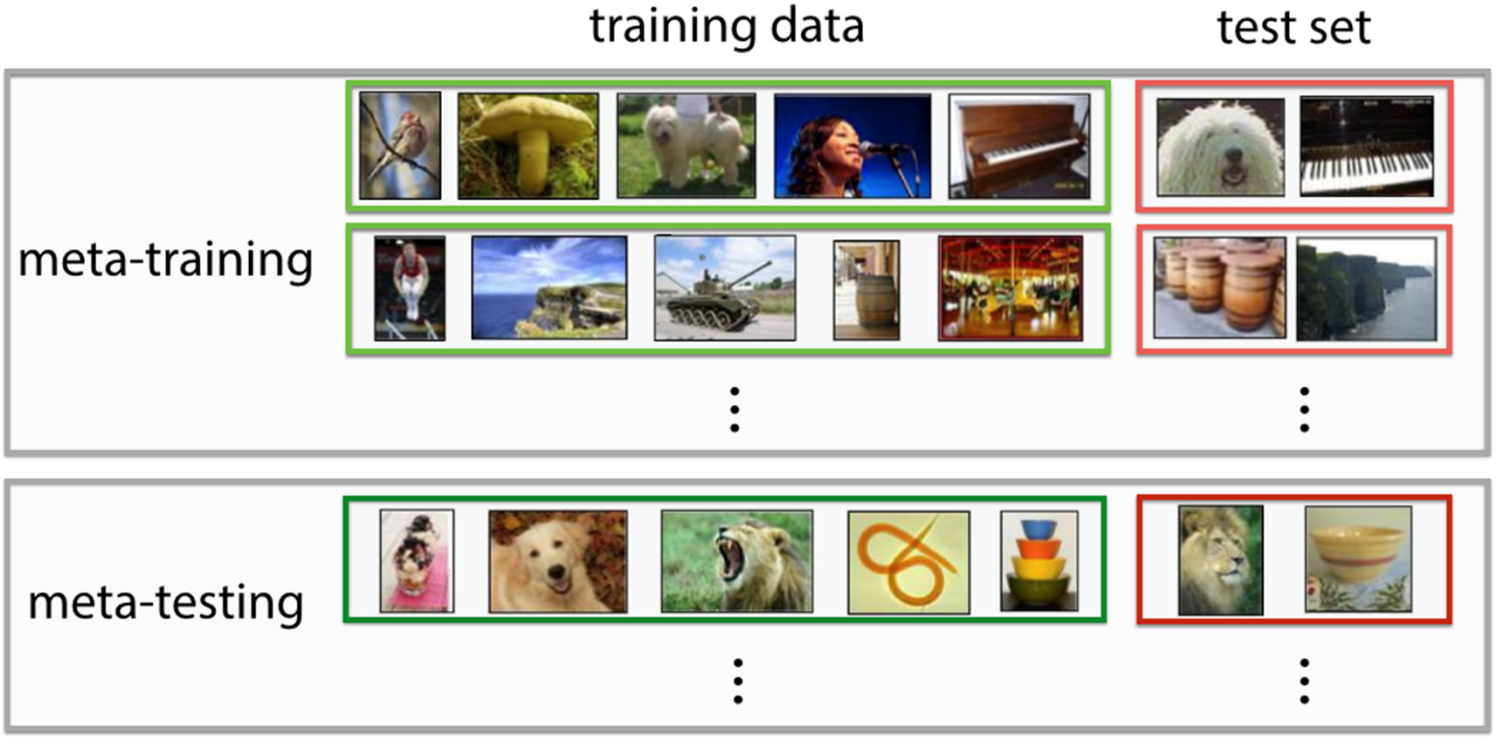


Test performance

- Human knows how to learn but RL methods fail to do so

Lake, Brenden M., et al. "Building machines that learn and think like people." Behavioral and Brain Sciences 40 (2017).

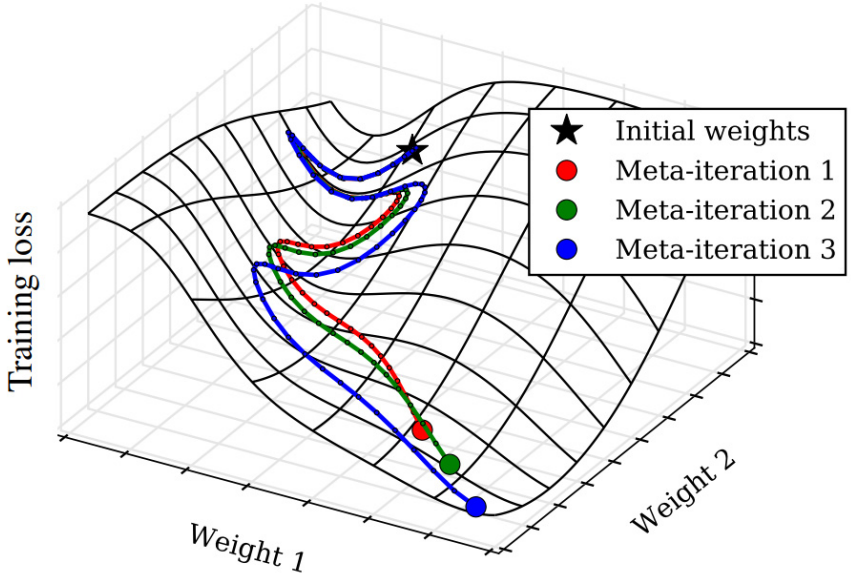
General Paradigm of Meta-Learning

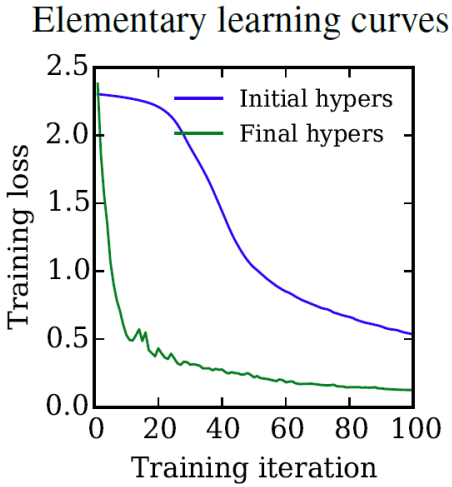


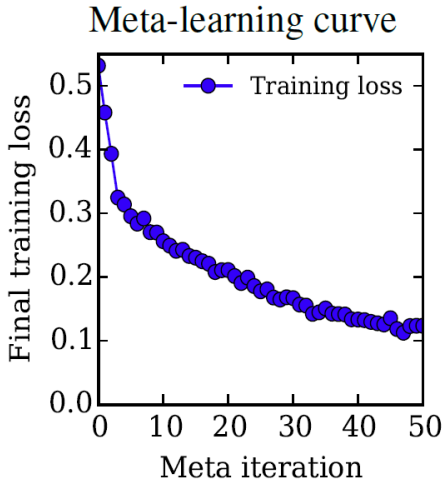
Example meta-learning set-up for few-shot image classification

- During meta-learning, the model is trained to learn tasks in the meta-training set. Two optimizations:
  - The learner, which learns new tasks
  - The meta-learner, which trains the learner

Various Meta-Learning Tasks

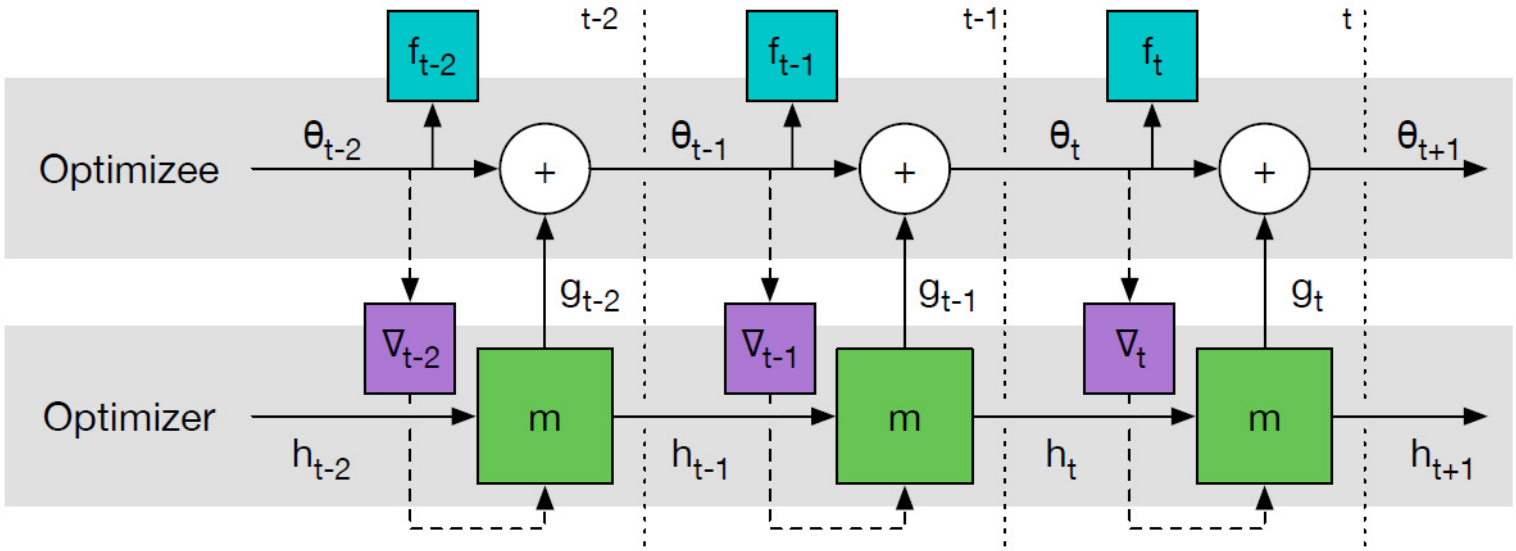






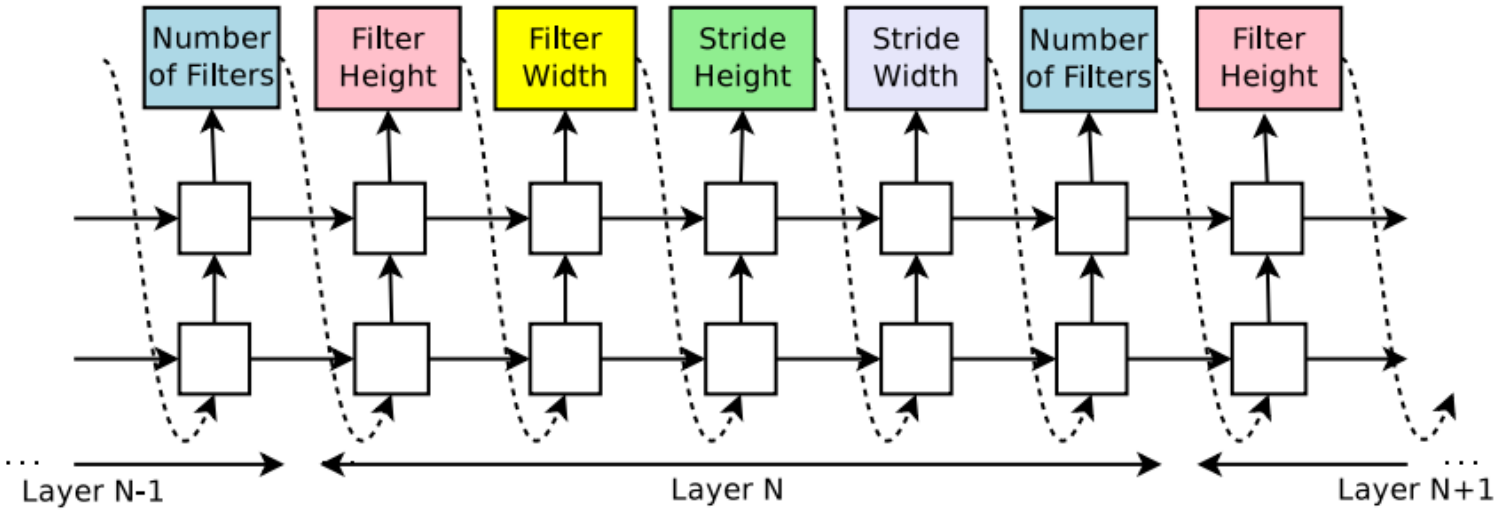
- Hyperparameter optimization
  - Compute exact gradients of cross-validation performance with respect to all hyperparameters by chaining derivatives backwards through the entire training procedure
  - Maclaurin et al. Gradient-based Hyperparameter Optimization through Reversible Learning. ICML 2015.

Various Meta-Learning Tasks



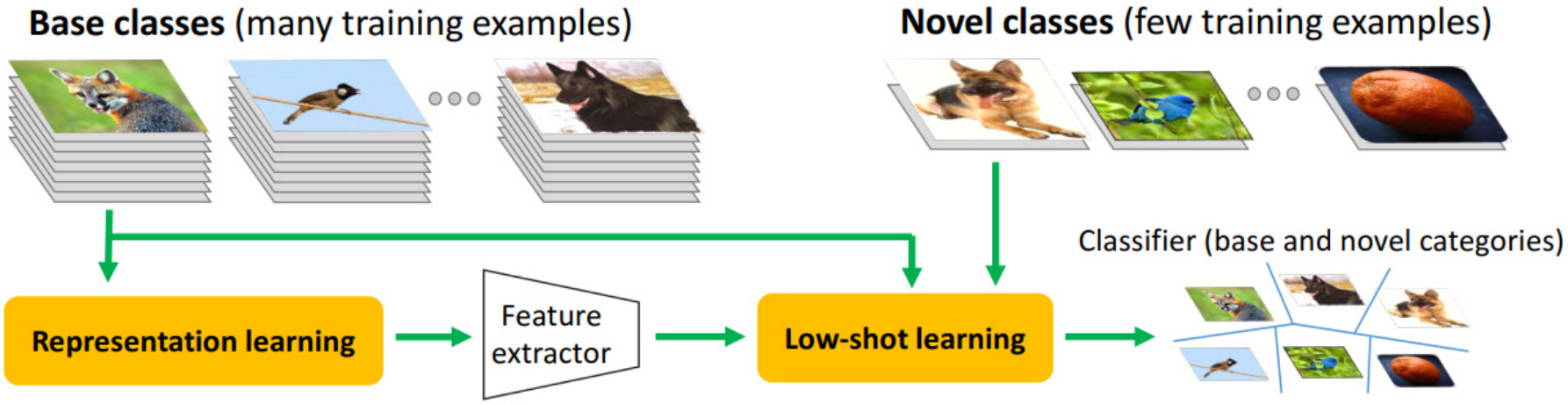
- Learn to produce good gradient
  - An RNN with hidden memory units takes in new raw gradient and outputs a tuned gradient so as to better train the model
  - Andrychowicz et al. Learning to learn by gradient descent by gradient descent. NIPS 2016.

Various Meta-Learning Tasks



- Automatically search good network architectures
  - RL takes action of creating a network architecture and obtains reward by training & evaluating the network on a dataset
  - Barret Zoph and Quoc V. Le. Neural architecture search by reinforcement learning. ICLR 2017.

Various Meta-Learning Tasks



- Few-shot learning
  - Learn a model from a large dataset that can be easily adapted to new classes with few instances
  - Hariharan and Girshick. Low-shot Visual Recognition by Shrinking and Hallucinating Features. ICCV 2017.

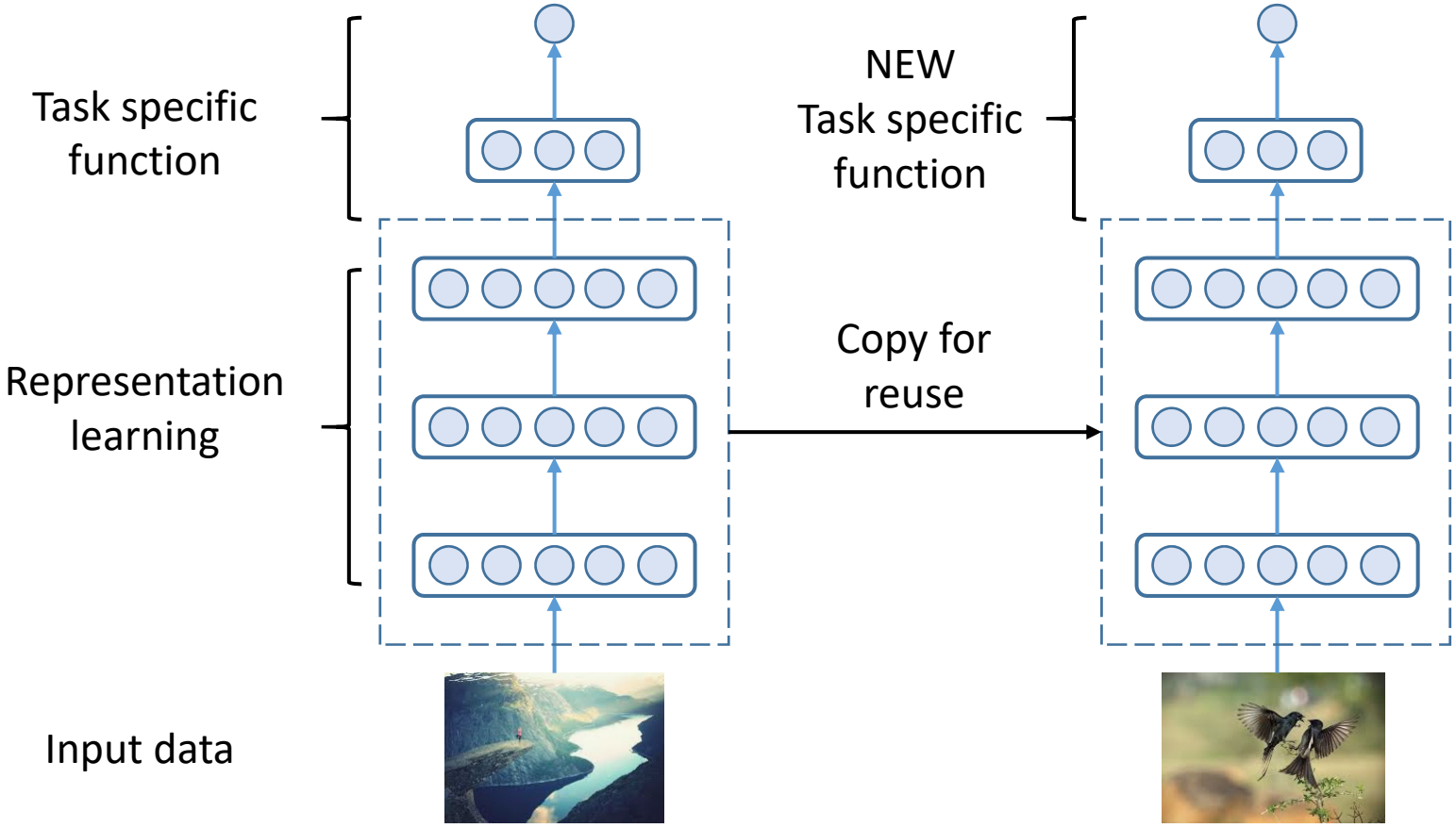
Meta-learning Methods

- Initialization based methods
  - Learning how to initialize the model for the new task
- Recurrent neural network methods
  - Learning how to produce good gradient in an auto-regressive manner
- Reinforcement learning methods
  - Learning how to produce good gradient in a reinforcement learning manner

REVIEW

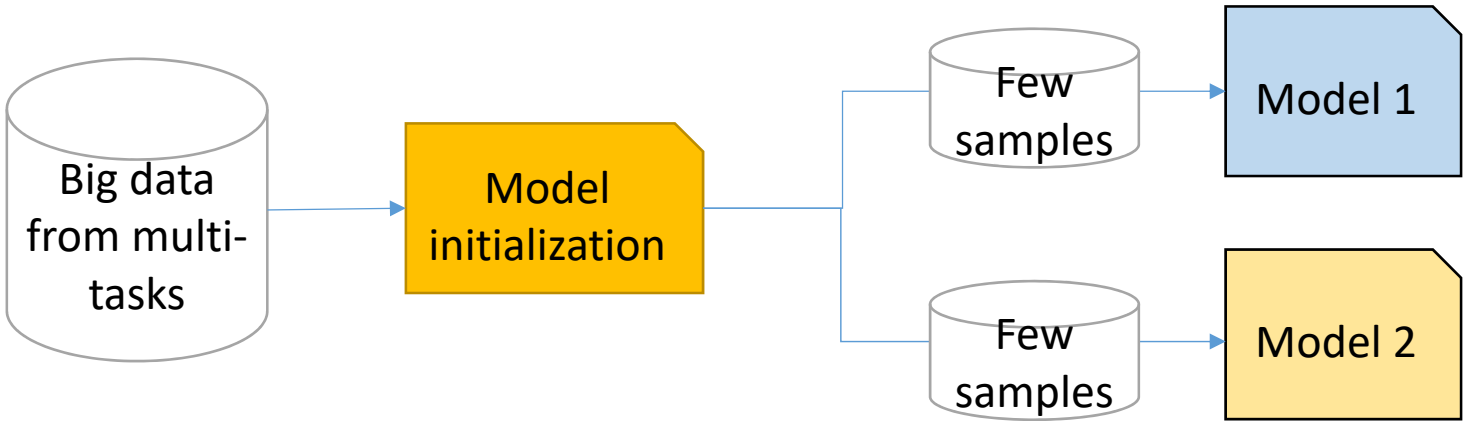
Network Parameter Reuse

- Treat lower layers as representation learning module and reuse them as good feature extractors



Model-Agnostic Meta Learning

- Goal: train a model that can be fast adapted to different tasks via few shots
- MAML idea: directly optimize for an initial representation that can be effectively fine-tuned from a small number of examples



Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. ICML 2017.

Model-Agnostic Meta Learning

- Goal: train a model that can be fast adapted to different tasks via few shots

Traditional multi-task learning SGD

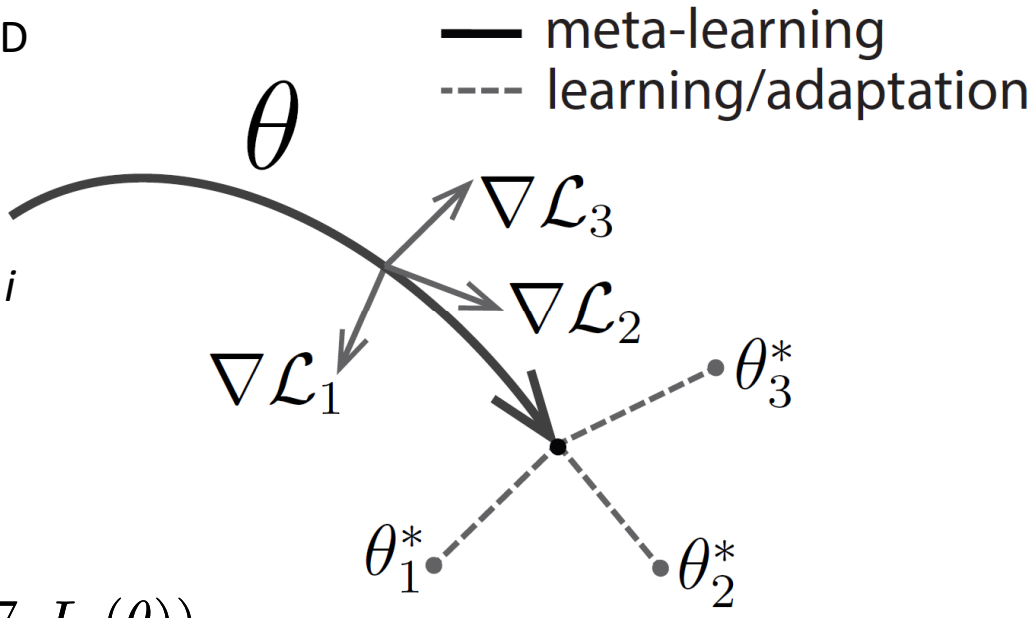
$$\theta \leftarrow \theta - \eta \sum_i \nabla_\theta L(\theta)$$

Imagined good parameter for task *i*

$$\theta^i \leftarrow \theta - \eta \nabla_\theta L_i(\theta)$$

MAML SGD

$$\theta \leftarrow \theta - \eta \sum_i \nabla_\theta L_i(\theta - \eta \nabla_\theta L_i(\theta))$$



Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. ICML 2017.

Model-Agnostic Meta Learning

- The MAML meta-gradient update involves a gradient through a gradient

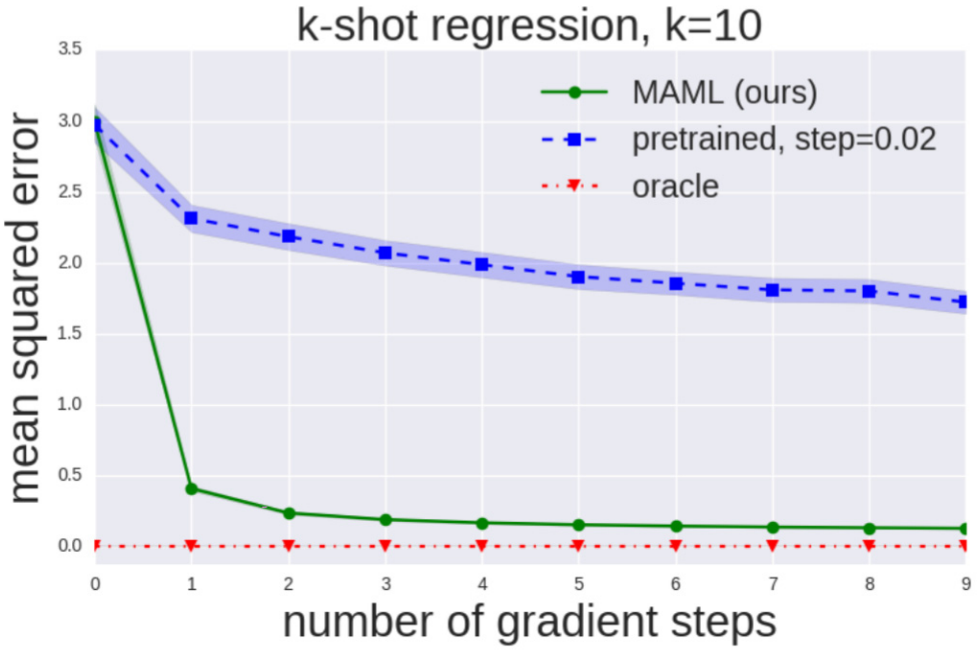
$$\theta \leftarrow \theta - \eta \sum_i \nabla_\theta L_i(\theta - \eta \nabla_\theta L_i(\theta))$$

  - this requires an additional backward pass through *f* to compute Hessian-vector
  - supported by standard deep learning libraries such as TensorFlow

Finn et al. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. ICML 2017.

MAML for Few-Shot Learning

- Pre-trained: use a pre-trained network as the initialization and perform traditional SGD
- MAML could be better than RNN meta-learners (talk later)



MiniImagenet (Ravi & Larochelle, 2017)	5-way Accuracy 1-shot	5-shot
fine-tuning baseline	$28.86 \pm 0.54\%$	$49.79 \pm 0.79\%$
nearest neighbor baseline	$41.08 \pm 0.70\%$	$51.04 \pm 0.65\%$
matching nets (Vinyals et al., 2016)	$43.56 \pm 0.84\%$	$55.31 \pm 0.73\%$
meta-learner LSTM (Ravi & Larochelle, 2017)	$43.44 \pm 0.77\%$	$60.60 \pm 0.71\%$
MAML, first order approx. (ours)	$\mathbf{48.07 \pm 1.75\%}$	$\mathbf{63.15 \pm 0.91\%}$
MAML (ours)	$\mathbf{48.70 \pm 1.84\%}$	$\mathbf{63.11 \pm 0.92\%}$

Meta-learning Methods

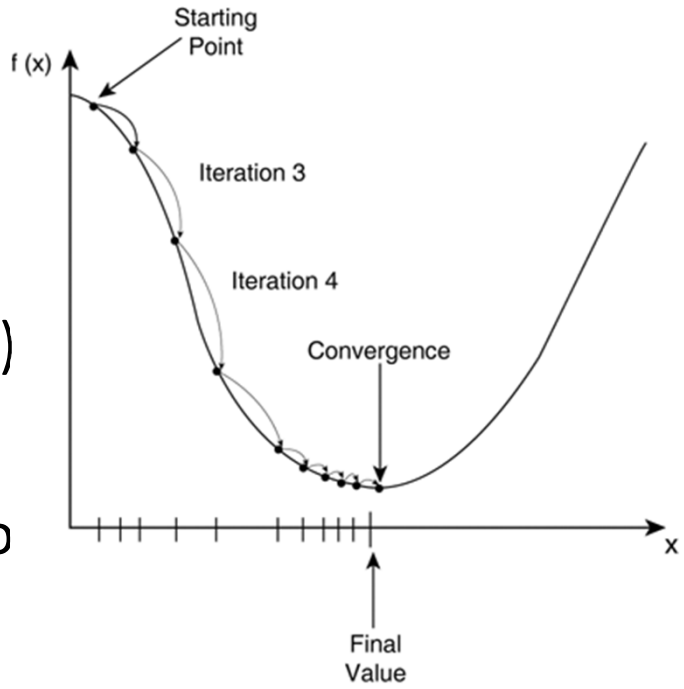
- Initialization based methods
  - Learning how to initialize the model for the new task
- Recurrent neural network methods
  - Learning how to produce good gradient in an auto-regressive manner
- Reinforcement learning methods
  - Learning how to produce good gradient in a reinforcement learning manner

Rethink About the Gradient Learning

- The traditional gradient in machine learning

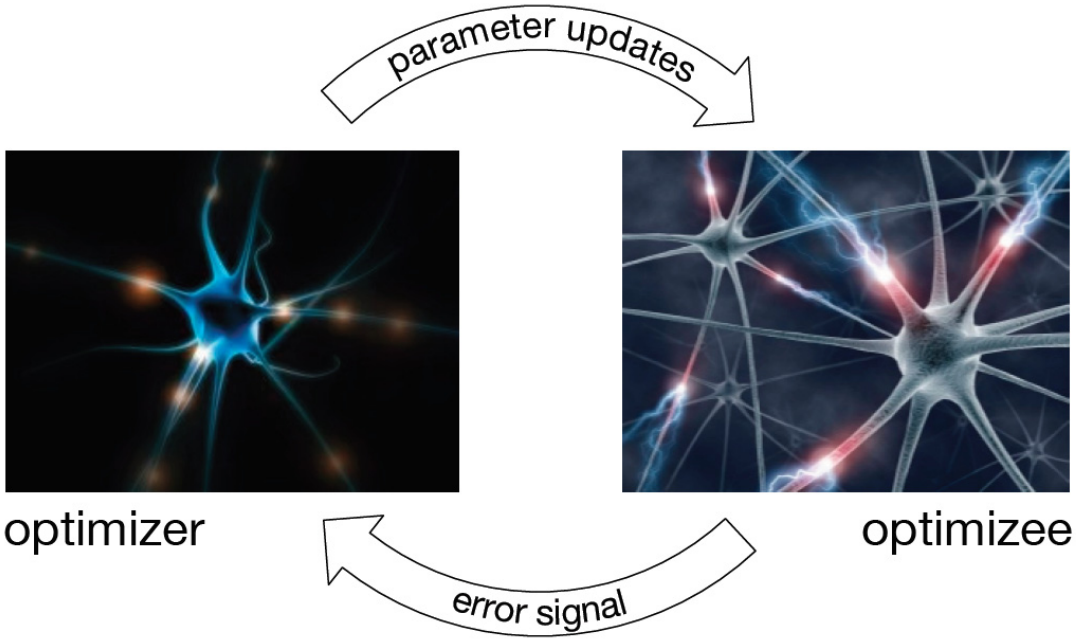
$$\theta_{t+1} = \theta_t - \eta_t \nabla_{\theta_t} L(\theta_t)$$

- Problems of it
  - Learning rate is fixed or changes with a heuristic rule
  - No consideration of second-order information (or even higher-order)
- Feasible idea
  - Memorize the historic gradients to better decide the next gradient



Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

An Optimizer Module to Decide How to Optimize



- Two components: optimizer and optimize
- The optimizer (left) is provided with performance of the optimizee (right) and proposes updates to increase the optimizee's performance

Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

Recurrent Network for Meta-Learning

- The optimizer decides the gradient in an auto-regressive manner, with RNNs as implementation

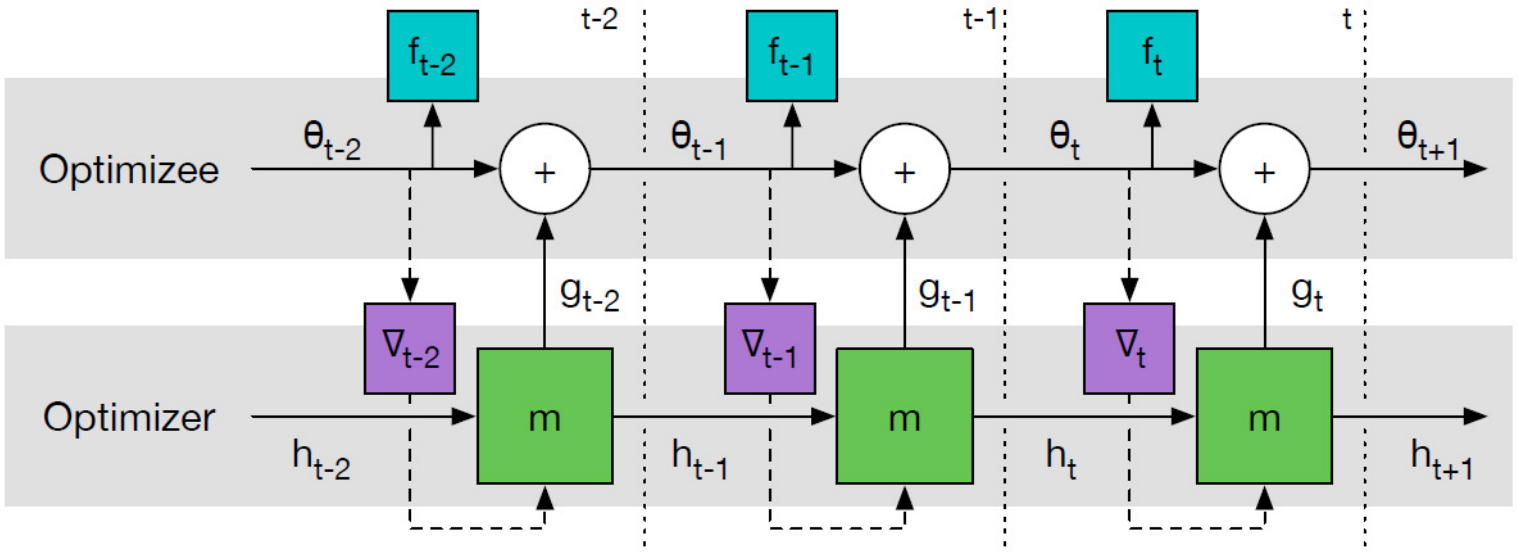
$$\theta_{t+1} = \theta_t - \eta_t \nabla_{\theta_t} L(\theta_t)$$

$$\theta_{t+1} = \theta_t - g_t(\nabla_{\theta_t} L(\theta_t), \phi)$$

g_t can be implemented with an RNN

Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

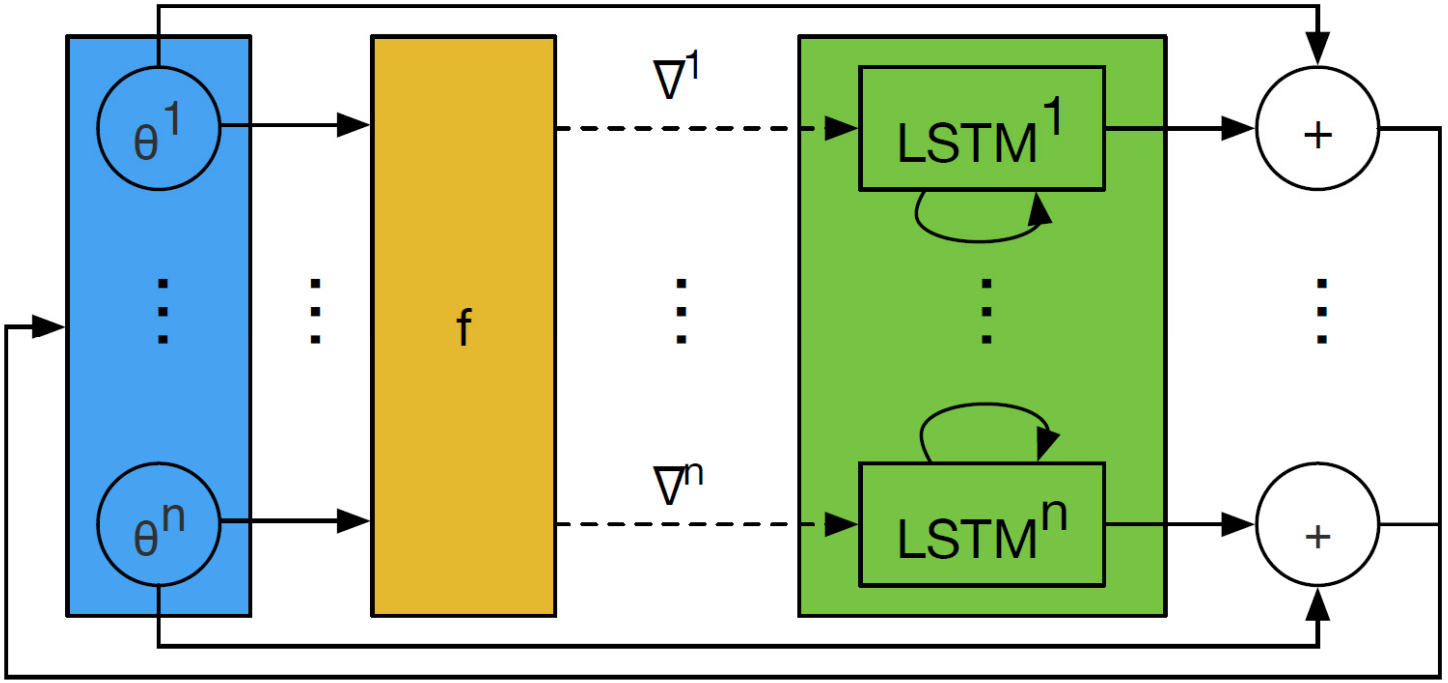
Recurrent Network for Meta-Learning



- With an RNN, the optimizer memorize the historic gradient information within its hidden layer
- The RNN can be directly updated with back-prop algorithms from the loss function

Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

Coordinatewise LSTM Optimizer



- Normally the parameter number *n* is large, thus a fully connected RNN is not feasible to train
- Above presents a coordinated LSRM, i.e., an LSTM^i for each individual parameter ϑ^i with shared LSTM parameters

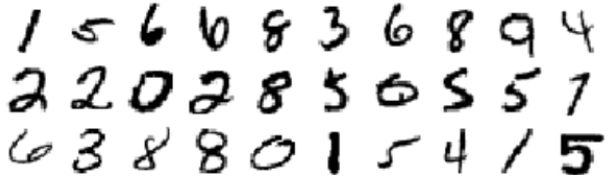
Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

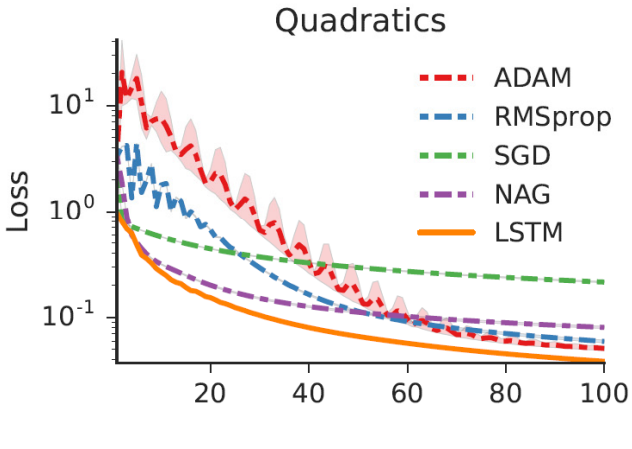
Meta-Learning Experiments

- Quadratic function

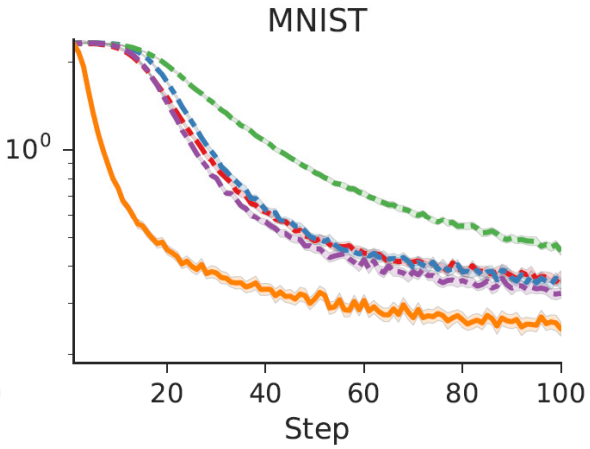
$$f(\theta) = \|W\theta - y\|^2$$

- MNIST

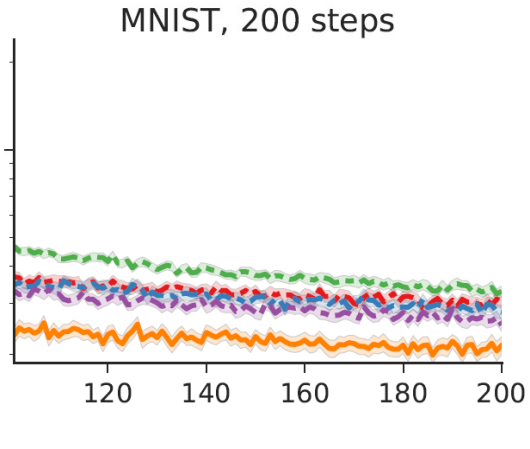




Performance of different optimizers on randomly sampled 10-dimensional quadratic functions.



Performance on MNIST. LSTM outperforms all other algorithms



Learning curves for steps 100-200 by an optimizer trained to optimize for 100 steps (continuation of center plot)

Andrychowicz, Marcin, et al. "Learning to learn by gradient descent by gradient descent." *NIPS* 2016.

Meta-learning Methods

- Initialization based methods
  - Learning how to initialize the model for the new task
- Recurrent neural network methods
  - Learning how to produce good gradient in an auto-regressive manner
- Reinforcement learning methods
  - Learning how to produce good gradient in a reinforcement learning manner

Review of Meta-Learning Methods

Algorithm 1 General structure of optimization algorithms

Require: Objective function f
  $x^{(0)} \leftarrow$ random point in the domain of f
  for $i = 1, 2, \ldots$ **do**
    $\Delta x \leftarrow \phi(\{x^{(j)}, f(x^{(j)}), \nabla f(x^{(j)})\}_{j=0}^{i-1})$
    if stopping condition is met **then**
      return $x^{(i-1)}$
    end if
    $x^{(i)} \leftarrow x^{(i-1)} + \Delta x$
  end for

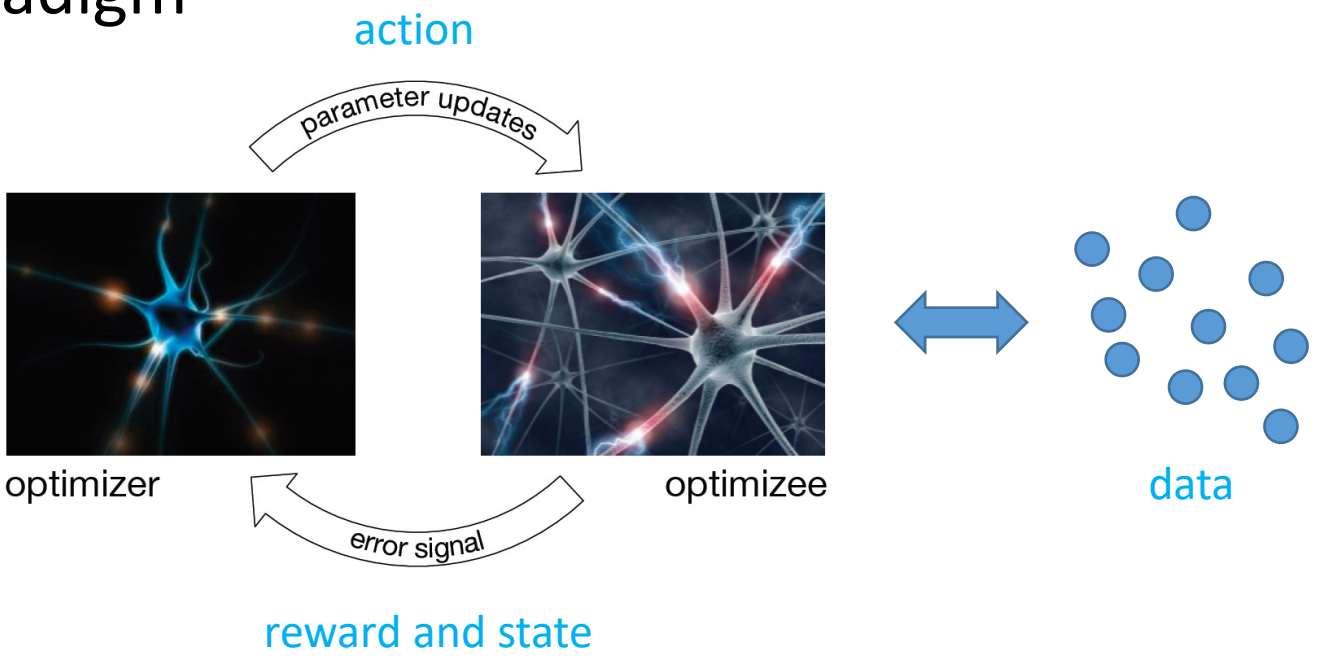
Gradient descent	$\phi(\cdot) = -\eta \nabla f(x^{(i-1)})$
Momentum	$\phi(\cdot) = -\eta \Big(\sum_{j=0}^{i-1} \eta^{i-1-j} \nabla f(x^{(j)}) \Big)$
Learned Algo.	$\phi(\cdot) =$ Neural Net

- The key of meta-learning (or learning to learn) is to design a good function that
  - takes previous observations and learning behaviors
  - outputs appropriate gradient for the ML model to update

Li, Ke, and Jitendra Malik. "Learning to optimize." *arXiv preprint arXiv:1606.01885* (2016).

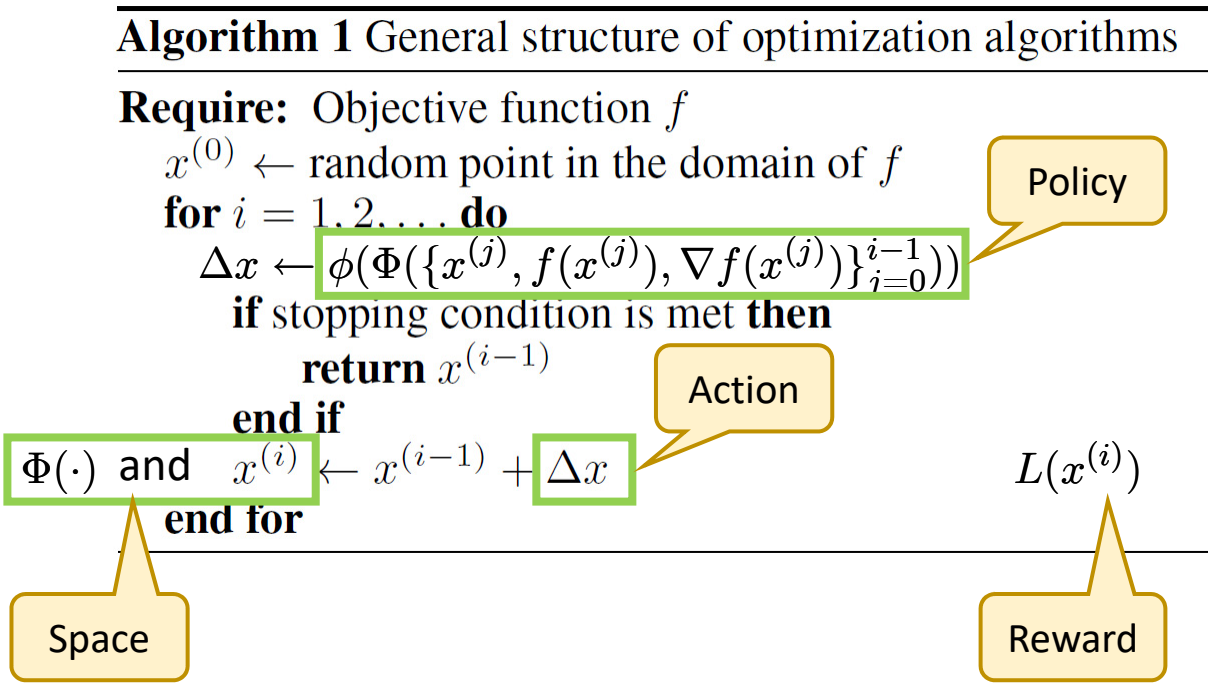
Reinforcement Learning for Meta-Learning

- Idea: at each timestep, the meta-learner learns to deliver an optimization action to the learner and then observe the performance of the learner, which is very similar with reinforcement learning paradigm



Li, Ke, and Jitendra Malik. "Learning to optimize." *arXiv preprint arXiv:1606.01885* (2016).

Formulation as an RL Problem



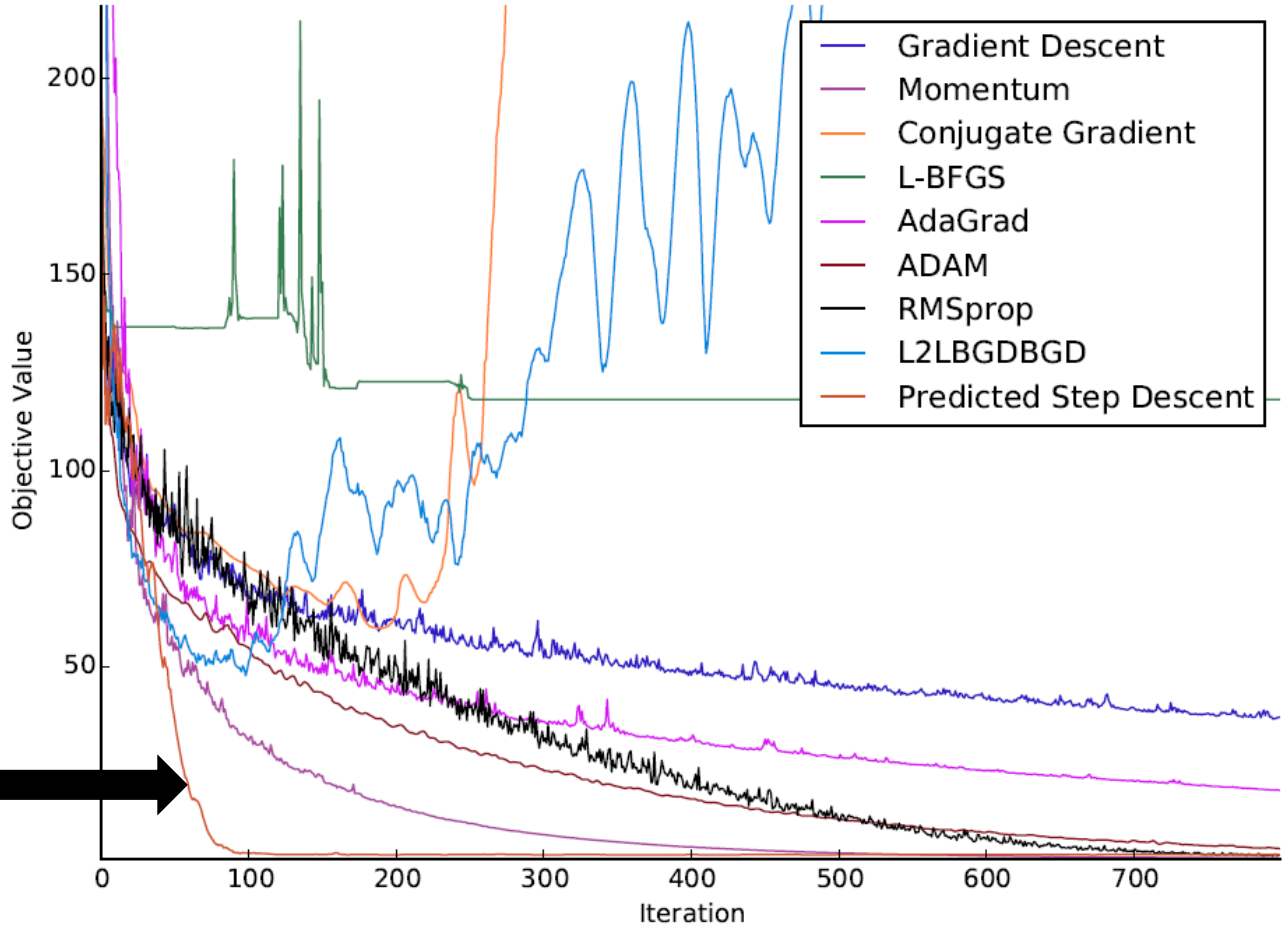
- State representation is generated from a function $\Phi(\cdot)$ mapping the observed data and learning behavior to a latent representation
- The policy outputs the gradient which is the action
- The reward is from the loss function w.r.t. the current model parameters

Li, Ke, and Jitendra Malik. "Learning to optimize." *arXiv preprint arXiv:1606.01885* (2016).

Learning to Learn Experiments

The light red curve is an optimizer trained using reinforcement learning.

Unlike the optimizer trained using supervised learning, it does not diverge in later iterations.



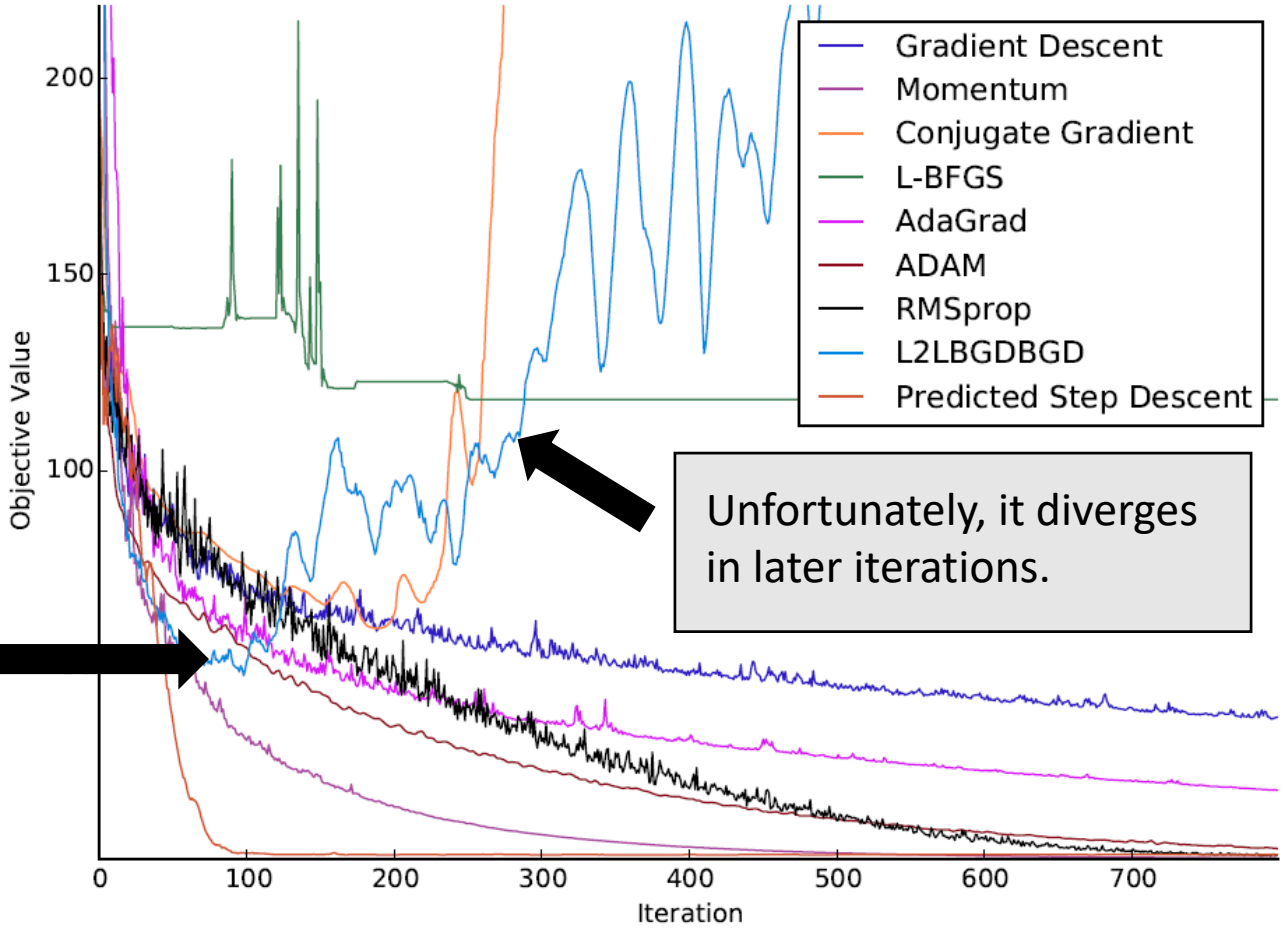
Data: randomly projected and normalized version of the MNIST training set with dimensionality 48 and unit variance in each dimension.

https://bair.berkeley.edu/blog/2017/09/12/learning-to-optimize-with-rl/

Learning to Learn Experiments

The light blue curve is an optimizer trained using supervised learning.

Here, it is used to train a neural net on a new task. Initially, it does reasonably well.



Unfortunately, it diverges in later iterations.

Data: randomly projected and normalized version of the MNIST training set with dimensionality 48 and unit variance in each dimension.

https://bair.berkeley.edu/blog/2017/09/12/learning-to-optimize-with-rl/

References of Meta-Learning

- Good blogs
  - https://medium.com/huggingface/from-zero-to-research-an-introduction-to-meta-learning-8e16e677f78a
  - https://bair.berkeley.edu/blog/2017/09/12/learning-to-optimize-with-rl/
  - https://bair.berkeley.edu/blog/2017/07/18/learning-to-learn/
- Sergey Levin's RL course
  - http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_16_meta_learning.pdf
- Two interesting papers
  - Learning to learn by gradient descent by gradient descent
  - Learning to Learn without Gradient Descent by Gradient Descent