Meta-learning:
Basics and Recent Advancements

Mertcan Yetkin
Department of Industrial and Systems Engineering
Lehigh University

OptML meetings
23 September 2020
1 Meta-learning basics
   - Definition
   - Formal definition
   - Mathematical definition

2 Meta-learning landscape

3 Common approaches

4 Application areas

5 Challenges
Meta-learning

"Being aware of and taking control of one's own learning" [1]

Learning to learn

Understanding how automatic learning can become flexible in solving learning problems
  - Improve the existing learning algorithms
  - Learn the learning algorithm itself
Motivation

training data

Braque

Cezanne

test datapoint

By Braque or Cezanne?
The meta-data is already accumulating

Figure: The ultimate goal
Formal definition [1]

1. A metalearning system must include a learning subsystem, which adapts with experience.
2. Experience is gained by exploiting metaknowledge extracted
   - in a previous learning episode on a single dataset, and/or
   - from different domains or problems.
Examples?
Neural architecture search

Figure: Neural architecture search
Hyperparameter optimization

**Figure**: Hyperparameter optimization
The big picture

- Meta-knowledge
- Gain experience
- Data analysis
- Dynamic bias selection
- Algorithm recommendation
- Inductive transfer
- Examining different domains
- Meta-analysis
- Ensemble methods
- Machine learning
An example ensemble method

- Bagging:

**Figure:** Bagging
Another example ensemble method

- Boosting:

**Figure: Boosting**
What is meta-data?

Some examples include:

- Algorithm runtime on a specific task
- Accuracy of a classification task
- Used step sizes in an iterative method
- ... any information obtained from a learning task (from a new experience)
Bias notion in meta-learning

- set of assumptions influencing the choice of hypotheses for explaining the data
  - declarative bias, i.e. representing hypotheses using neural networks only
  - procedural bias, i.e. preferring hypotheses with smaller runtime
- bias in base-learning is fixed
- meta-learning tries to find the right bias
Mathematical definition

- conventional machine learning:
  - given training dataset \( D = \{x_1, y_1, \ldots, x_N, y_N\} \)
  - We can train a predictive model \( \hat{y} = f_\theta(x) \), parametrized by \( \theta \)
  - by solving \( \theta^* = \text{arg min}_\theta L(D; \theta, w) \)
- \( w \) is included to represent factors such as
  - choice of optimizer for \( \theta \)
  - function class \( f \)
  - initial point for \( \theta \)
  - further settings affecting bias
meta-learning:
- define task $T = \{D, L\}$, given a distribution of tasks $p(T)$
- learning how to learn becomes $\min_w = \mathbb{E}_{T \sim p(T)} L(D; w)$
- by solving $\theta^* = \arg \min_{\theta} L(D; \theta, w)$
- where $L(D; w)$ measures the performance of a model trained using $w$ on dataset $D$

in practice, we sample from tasks for meta-training stage
$D_{\text{source}} = \{(D_{\text{source}}^{\text{train}}, D_{\text{source}}^{\text{val}})\}$
- meta-training step: $w^* = \arg \max_w \log p(w | D_{\text{source}})$
- meta-testing on new task $i$: $\theta^{*i} = \arg \max_{\theta} \log p(\theta | w^*, D_{\text{target}}^{\text{train}, i})$
- evaluate the accuracy of meta-learner $\theta^{*i}$ on $D_{\text{target}}^{\text{test}}$
Figure: Data structure in few-shot classification
An example benchmarking dataset

Figure: A glance at the Omniglot dataset
Figure: Some example tasks within Omniglot dataset
Corresponds to learning a bias $w$ that constrains the hypothesis space of $\theta$ too tightly around solutions to the source tasks.
Bilevel optimization view

To solve the meta-training step

\[ w^* = \arg \min_w \sum_{i=1}^{M} L_{\text{meta}}(\theta^*_i(w), w, D_{\text{val},i}) \]

s.t. \[ \theta^*_i(w) = \arg \min_\theta L_{\text{task}}(\theta, w, D_{\text{train},i}) \]
Meta-learning landscape [2]
Main categories of meta-learning [3]

- learning from model evaluations
  - task-independent recommendations (ranking of configurations)
  - configuration space design (i.e. hyperparameters)
  - configuration transfer for a new task (i.e. surrogate models)
Main categories ctd..

- learning from task properties
  - meta-features
  - warm-starting optimization from similar tasks
- to tune or not to tune?

<table>
<thead>
<tr>
<th>Name</th>
<th>Formula</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nr instances</td>
<td>$n$</td>
<td>Speed, Scalability (Michie et al., 1994)</td>
</tr>
<tr>
<td>Nr features</td>
<td>$p$</td>
<td>Curse of dimensionality (Michie et al., 1994)</td>
</tr>
<tr>
<td>Nr classes</td>
<td>$c$</td>
<td>Complexity, imbalance (Michie et al., 1994)</td>
</tr>
<tr>
<td>Nr missing values</td>
<td>$m$</td>
<td>Imputation effects (Kalousis, 2002)</td>
</tr>
<tr>
<td>Nr outliers</td>
<td>$o$</td>
<td>Data noisiness (Rousseeuw and Hubert, 2011)</td>
</tr>
<tr>
<td>Skewness</td>
<td>$\frac{E(X - \mu_X)^3}{\sigma_X^3}$</td>
<td>Feature normality (Michie et al., 1994)</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>$\frac{E(X - \mu_X)^4}{\sigma_X^4}$</td>
<td>Feature normality (Michie et al., 1994)</td>
</tr>
<tr>
<td>Correlation</td>
<td>$\rho_{X_1X_2}$</td>
<td>Feature interdependence (Michie et al., 1994)</td>
</tr>
<tr>
<td>Covariance</td>
<td>$\text{cov}_{X_1X_2}$</td>
<td>Feature interdependence (Michie et al., 1994)</td>
</tr>
<tr>
<td>Concentration</td>
<td>$\tau_{X_1X_2}$</td>
<td>Feature interdependence (Kalousis and Hilario, 2001)</td>
</tr>
<tr>
<td>Sparsity</td>
<td>$\text{sparsity}(X)$</td>
<td>Degree of discreteness (Salama et al., 2013)</td>
</tr>
<tr>
<td>Gravity</td>
<td>$\text{gravity}(X)$</td>
<td>Inter-class dispersion (Ali and Smith-Miles, 2006a)</td>
</tr>
<tr>
<td>ANOVA p-value</td>
<td>$p_{\text{ANOVA}}$</td>
<td>Feature redundancy (Kalousis, 2002)</td>
</tr>
<tr>
<td>Coeff. of variation</td>
<td>$\frac{\mu_X}{\sigma_X}$</td>
<td>Variation in target (Soares et al., 2004)</td>
</tr>
<tr>
<td>PCA $\rho_k$</td>
<td>$\frac{\mu_k}{\lambda_k}$</td>
<td>Variance in first PC (Michie et al., 1994)</td>
</tr>
<tr>
<td>PCA skewness</td>
<td>$\text{skewness}(X)$</td>
<td>Skewness of first PC (Feuer et al., 2014)</td>
</tr>
<tr>
<td>PCA 95%</td>
<td>$\text{dim}_{95%\text{lux}}$</td>
<td>Intrinsic dimensionality (Bardenet et al., 2013)</td>
</tr>
<tr>
<td>Class probability</td>
<td>$P(C)$</td>
<td>Class distribution (Michie et al., 1994)</td>
</tr>
<tr>
<td>Class entropy</td>
<td>$H(C)$</td>
<td>Class imbalance (Michie et al., 1994)</td>
</tr>
<tr>
<td>Norm. entropy</td>
<td>$\frac{\mu_{\text{norm}}}{\text{log}_2(n)}$</td>
<td>Feature informativeness (Cassiello et al., 2005)</td>
</tr>
<tr>
<td>Mutual inform.</td>
<td>$MI(C,X)$</td>
<td>Feature importance (Michie et al., 1994)</td>
</tr>
<tr>
<td>Uncertainty coeff.</td>
<td>$\frac{H(U)}{H(C)}$</td>
<td>Feature importance (Agresti, 2002)</td>
</tr>
<tr>
<td>Equiv. nr. feats</td>
<td>$\frac{M(C,X)}{H(C) - H(C</td>
<td>X)}$</td>
</tr>
<tr>
<td>Noise-signal ratio</td>
<td>$\frac{H(X,T) - MI(C, X)}{MI(C, X)}$</td>
<td>Noisiness of data (Michie et al., 1994)</td>
</tr>
</tbody>
</table>
Main categories ctd..

- learning from prior models
  - transfer learning
  - few-shot learning (model-agnostic meta learning)
  - other unsupervised learning
Key idea: Train a neural network to represent $p(\theta^i|D_{\text{source}}^{\text{train}}, w)$

Use deterministic $\theta^i = f_w(D_{\text{source}}^{\text{train}})$

Some common forms for $f_w$
- LSTM
- Neural turing machine (NTM)
- Self-attention
- 1D convolutions
Optimization-based approach

- Consider the problem of initializing good weights $\theta$ for different tasks.
- Meta-learning: $\min_w \sum_{i=1}^M L^{meta}(w - \alpha \nabla_w L^{task}(w, D_{train,i}^{source}, D_{val,i}^{source}))$
- Meta-testing: $\theta \leftarrow w - \alpha \nabla_w L^{task}(w, D_{train,i}^{target})$

Figure: MAML idea
Optimization vs black-box approaches

Generalization of learning procedures to extrapolated tasks:
Other non-parametric approaches

- Obtain task similarity
  - Siamese networks to predict 2 images belong to same class
  - K-nearest neighbours
Application areas

- computer vision: few-shot learning methods
  - classification
  - object detection
  - object segmentation
  - density estimation
- meta reinforcement learning & robotics
  - exploration
  - optimization
  - knowledge-transfer? (opening jar vs opening door)
- environment learning (simulator)
- continual learning (learning new things, without forgetting, DNN)
- language and speech
  - language modelling (filling in missing words)
  - speech recognition (different accents)
- systems
  - learning with label noise
  - adversarial attacks and defenses
Challenges

- meta-generalization (new tasks, conflicting gradients)
- one solution for all (distribution over tasks)
- computation cost (many inner loops) → few-shot learning
- cross-modal transfer (visual imitation learning)
Meta-learning with differentiable convex optimization [4]
Bilevel programming for hyperparameter optimization and meta-learning [5]
Model-agnostic meta-learning (MAML) [6]
Probabilistic model-agnostic meta-learning [7]
Convergence theory for gradient-based MAML algorithms [8]


Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto.
Meta-learning with differentiable convex optimization.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil.
Bilevel programming for hyperparameter optimization and meta-learning.

Chelsea Finn, Pieter Abbeel, and Sergey Levine.
Model-agnostic meta-learning for fast adaptation of deep networks.