Summary
We take a closer look at Model Agnostic Meta-Learning (MAML) and show that it requires depth — shallow models fail because they lack parameters to shape the gradients during fast adaptation.

• Surprisingly, MAML fails to adapt on very simple tasks even with a model expressive enough to solve them perfectly; but, an over-parameterized model succeeds.
• Our analysis shows that this is because upper layers meta-learn update functions for the bottom layers.
• We propose three solutions to combat this issue:
  1. Using deeper non-linear models,
  2. Adding extra linear (collapsable) layers at the end of the model,
  3. Training with KFO (Kronecker-Factored Optimizer), a new meta-optimizer which scales to large deep networks.

MAML: Model Agnostic Meta-Learning
The MAML [1] objective is simply expressed as:
\[
\min_\theta \mathbb{E}_r \left[ \mathcal{L}_r \left( \theta - \alpha \nabla_{\theta} \mathcal{L}_r \left( \theta \right) \right) \right]
\]
where:
- \( \theta \) are the parameters to be learned,
- \( r \) is a task index,
- \( \mathcal{L}_r \) is the loss associated with a task.

Intuition: MAML tries to meta-learn parameters that can be quickly adapt to any task from your training distribution.

References

Failure Mode
MAML fails to meta-learn with shallow models, even though they have sufficient capacity to solve all tasks. However, meta-learning succeeds when over-parameterizing the models (with linear layers) without changing their original capacity.

Insights
• Theoretical analysis on 1D shallow and deep models shows that:
  - deep models are required for meta-learning, because
  - the upper layers of the model facilitate (meta-)optimization.
• We can interpret those upper layers as “meta-optimizers that work from the inside” as they learn to modify the adaptation gradient of lower layers.
• We empirically verify this theory on linear & logistic regression, and with deep network architectures.

Solutions
• Use larger deeper models: current go-to solution, undesirable in compute-limited environments.
• Add extra linear layers on top of the model: simple, universal, works decently but incurs small performance penalty.
• Move optimization parameters to a KFO meta-optimizer: best performance, lightweight post-adaptation, but expensive during meta-training.

Empirical Results
Extra linear layers improve shallow meta-learning.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>MAML</th>
<th>MAML w/ LinNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Omniglot</td>
<td>66.6</td>
<td>74.07 94.63 92.27</td>
</tr>
<tr>
<td>CIFAR-FS</td>
<td>62.2</td>
<td>68.92 88.37 66.42</td>
</tr>
<tr>
<td>mini-ImageNet</td>
<td>52.6</td>
<td>59.90 58.95 58.47</td>
</tr>
</tbody>
</table>

Meta-optimizers outperform MAML on 2-layer CNNs.

Meta-optimizers are most effective with shallower models.

See our paper for more details, including:
• Theoretical analysis of 1D linear and logistic regression.
• Combining ANIL [2] with Meta-Optimizers.
• Why collapsing extra linear networks fails.