Supervised Meta-Learning

- Assume task \( T_i \sim p(T) \); Each task consists of task training data \( D_i = (x_i, y_i) \) and validation data \( D_i^* = (x_i^*, y_i^*) \);
- \( x_i = (x_{i1}, \ldots, x_{iK}) \), \( y_i = (y_{i1}, \ldots, y_{iK}) \) \( \sim p(x, y | T_i) \) and similarly for \( D_i^* \).
- Entire meta-training set is \( M = \{ D_i, D_i^* \} \) \( \forall i \).
- The objective is

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) \right] ,
\]

where \( q(\theta, M) \) summarizes meta-training data, \( q(\phi, D, \theta) \) summarizes the per-task training set and \( q(y_i | x_i, \phi, \theta) \) is the predictive distribution.

The Memorization Problem

Definition 1 (Complete Meta-Learning Memorization). Complete memorization in meta-learning is when the learned model ignores the task training data such that

\[
I(\hat{y}_i^* | D_i, \theta) = 0 \quad (i.e., \ q(\hat{y}_i^* | x_i^*, \phi, \theta, D_i) = q(\hat{y}_i^* | x_i^*, \phi, \theta) = \mathbb{E}_{D_i^*}[\{q(y_i | x_i^*, \phi, \theta)\}]).
\]

Meta-Learning without Memorization

- Related to a novel PAC Bayes bound for meta-learning (see paper for details).
- The objective is

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(\theta; \theta_o, \beta) || r(\theta)) \right]
\]

Meta-regularization on activations

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(\theta; \theta_o, \beta) || r(\theta)) \right]
\]

Meta-regularization on weights

- Sources of information in the predictive distribution \( q(y_i | x_i, \phi, \theta) \) come from input, meta-parameters, and data.
- Encourage using task training data \( D_i \) by restricting information flow from other sources (\( x_i^* \) and \( \theta \)) to \( y_i^* \).

Meta-Regularization on Activations

- Introduce an intermediate stochastic bottleneck variable \( z_i \) such that \( q(\hat{y}_i^* | z_i, \phi, \theta) = \int q(\hat{y}_i^* | z_i, \phi, \theta) q(z_i | x_i, \theta) dz_i \).
- Optimize with the regularized training objective

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(z_i | x_i, \theta) || r(z_i)) \right]
\]

Meta-Regularization on Weights

- Limit the information about the training tasks stored in the meta-parameters \( \theta \) by penalizing \( I(y_i^* | D_i, \theta; x_i^*, y_i^*) \).

Why does it happen in Meta-Learning?

- Assume task \( T \sim p(T) \); Each task consists of task training data \( D = (x, y) \) and validation data \( D^* = (x^*, y^*) \);
- \( x = (x_1, \ldots, x_K) \), \( y = (y_1, \ldots, y_K) \) \( \sim p(x, y | T) \) and similarly for \( D^* \)
- Entire meta-training set is \( M = \{ D, D^* \} \) \( \forall T \).
- The objective is

\[
\frac{1}{N} \sum_{T_i} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(\theta; \theta_o, \beta) || r(\theta)) \right]
\]

Meta-Learning without Memorization

- Related to a novel PAC Bayes bound for meta-learning (see paper for details).
- The objective is

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(\theta; \theta_o, \beta) || r(\theta)) \right]
\]

Meta-regularization on activations

- Sources of information in the predictive distribution \( q(y_i | x_i, \phi, \theta) \) come from input, meta-parameters, and data.
- Encourage using task training data \( D_i \) by restricting information flow from other sources (\( x_i^* \) and \( \theta \)) to \( y_i^* \).

Meta-Regularization on Activations

- Introduce an intermediate stochastic bottleneck variable \( z_i \) such that \( q(\hat{y}_i^* | z_i, \phi, \theta) = \int q(\hat{y}_i^* | z_i, \phi, \theta) q(z_i | x_i, \theta) dz_i \).
- Optimize with the regularized training objective

\[
\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{(x_i, y_i) \in M} \left[ \frac{1}{K} \sum_{(x_{i1}, y_{i1}) \in D_i} \log p(y_i = y_{i1} | x_{i1}, \phi, \theta) + \beta D_{KL}(q(z_i | x_i, \theta) || r(z_i)) \right]
\]

Meta-Regularization on Weights

- Limit the information about the training tasks stored in the meta-parameters \( \theta \) by penalizing \( I(y_i^* | D_i, \theta; x_i^*, y_i^*) \).