Meta-Learning without Memorization

Mingzhang Yin[*†], George Tucker[†], Mingyuan Zhou[*],
Sergey Levine[★†], Chelsea Finn[‡†]

[*]UT Austin, [†]Google Research, Brain Team, [★]UC Berkeley, [‡]Stanford

Contact: mzyin@utexas.edu

How does meta-learning work?

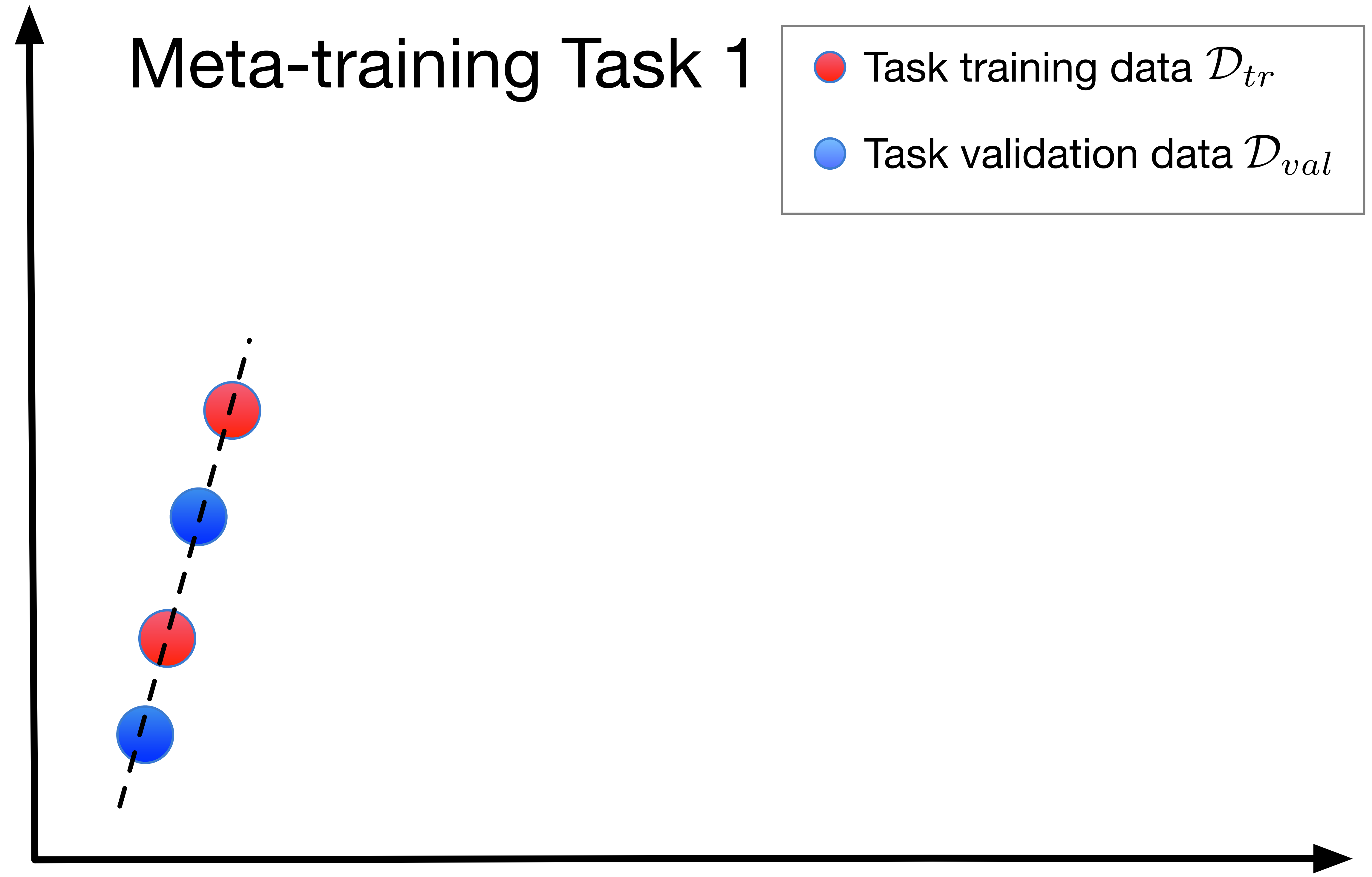
- There are multiple tasks $\mathcal{T}_j \sim P(\mathcal{T})$
- Each task has training data $\mathcal{D}_{tr}$ and validation data $\mathcal{D}^*_{val} = (X^*, Y^*)$
- Meta-learning can solve an unseen task by

  - leveraging past experience from previous tasks

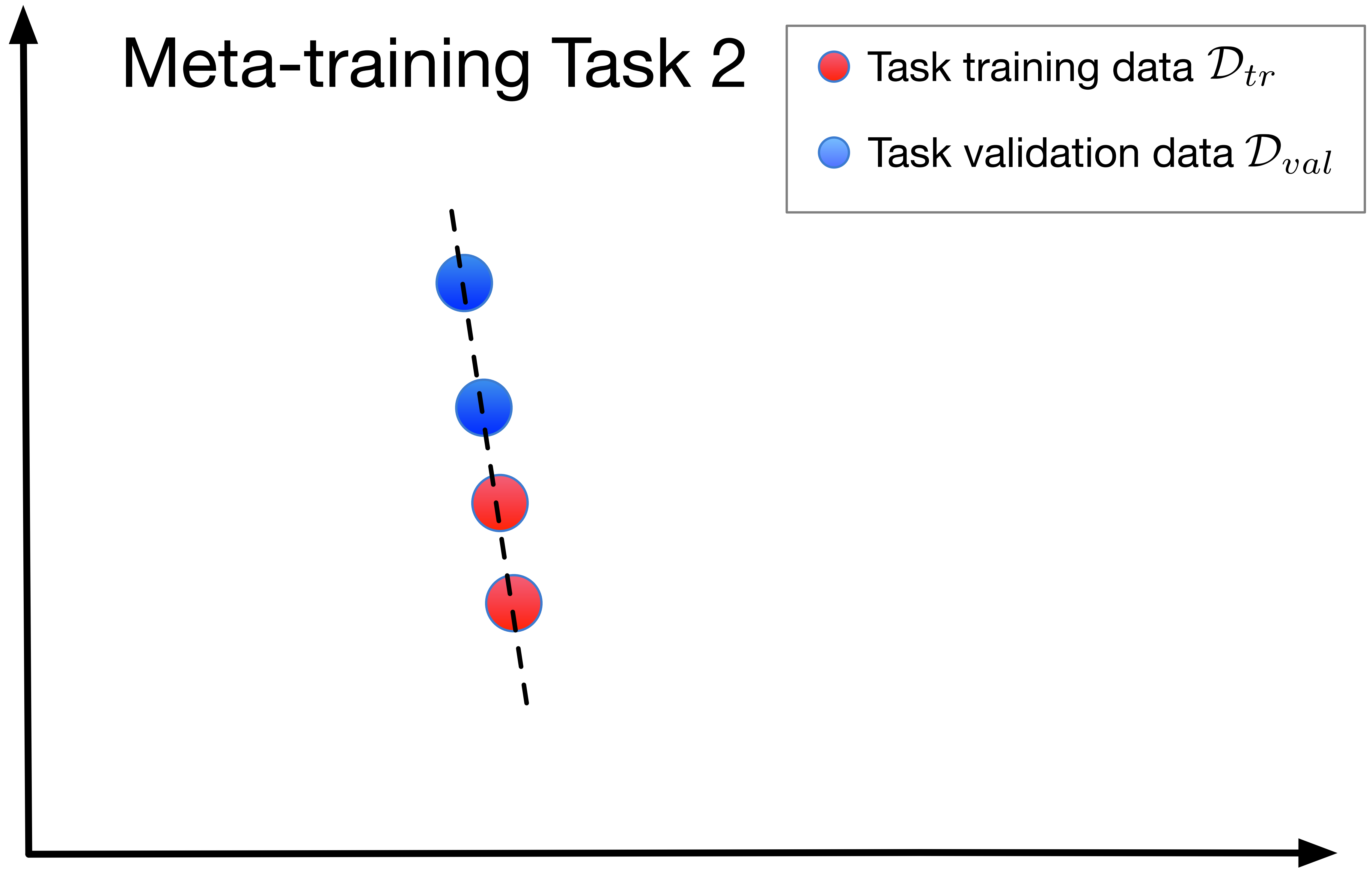
  - adapting to new task training data

Both are necessary!

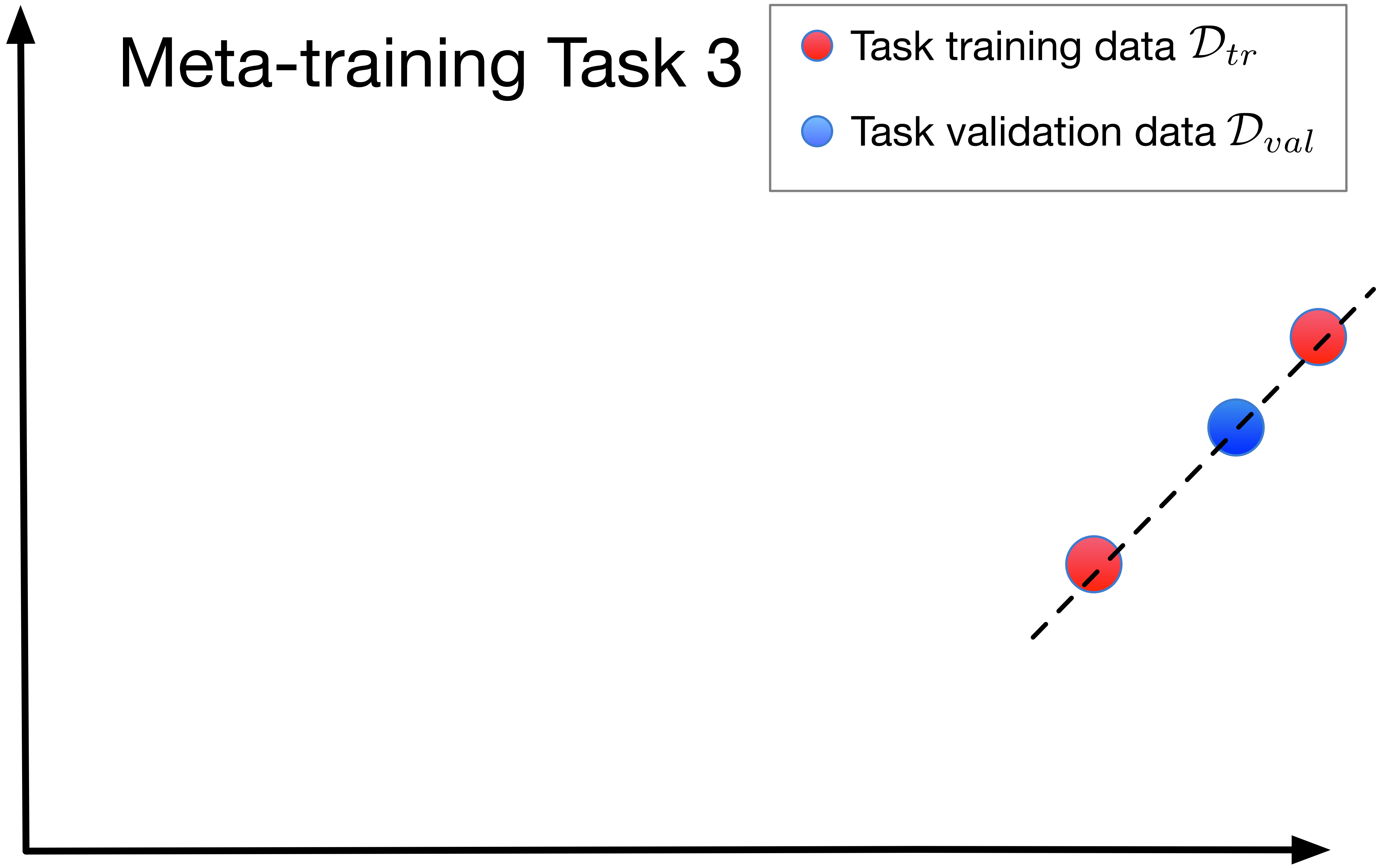
Example: regression on linearly related data



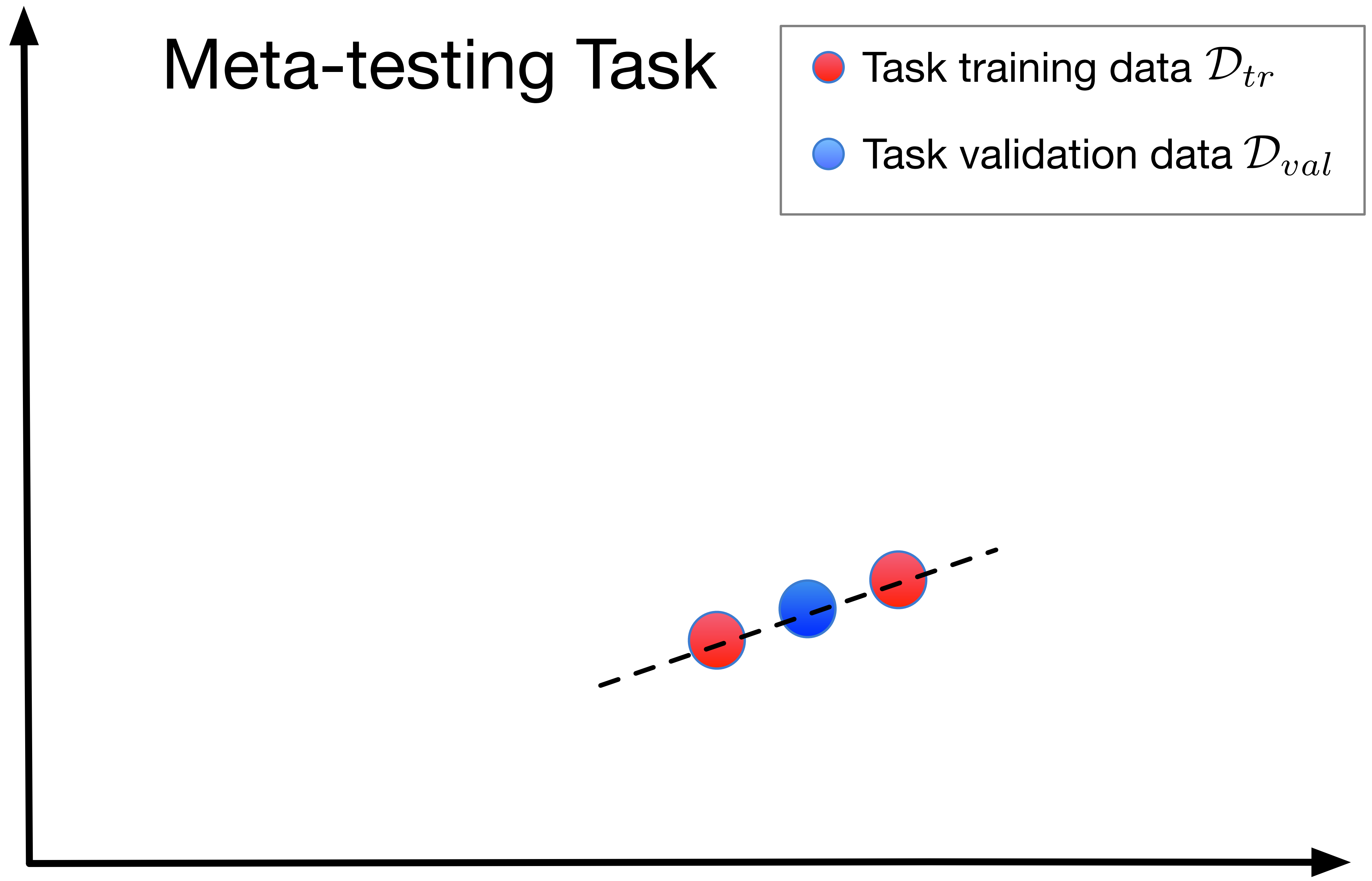
Example: regression on linearly related data



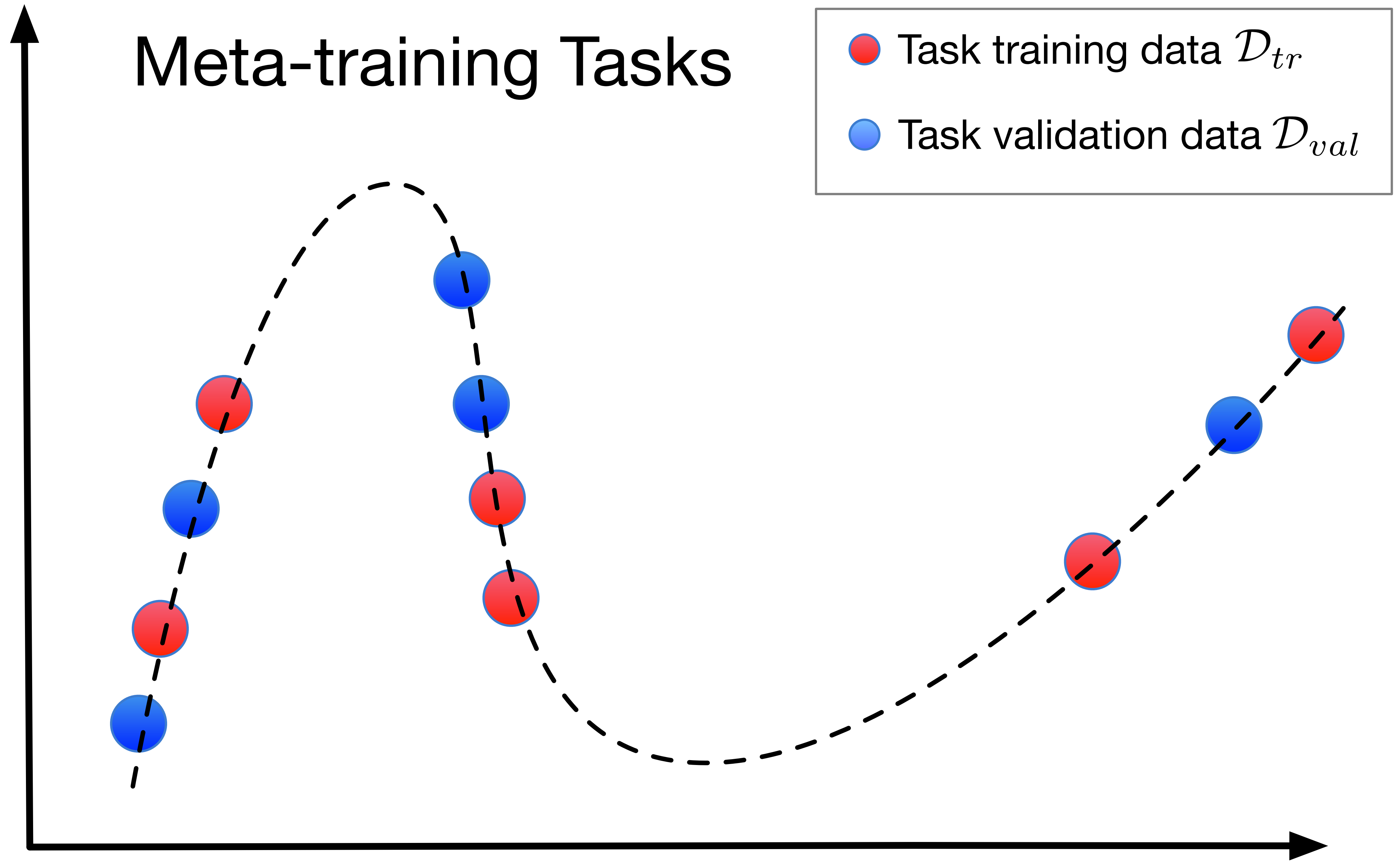
Example: regression on linearly related data



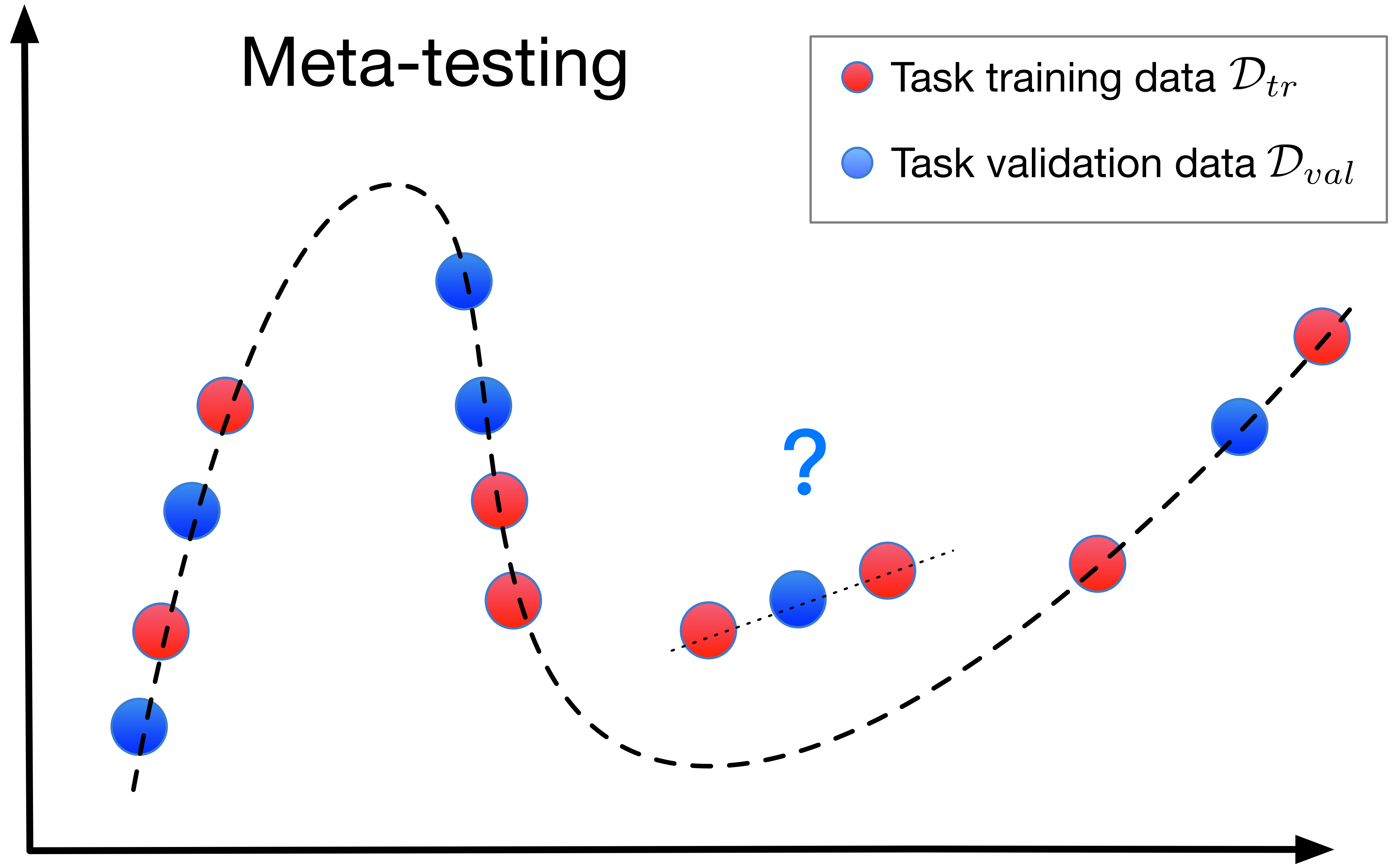
Example: regression on linearly related data



What if all of the meta-training tasks can be solved by a single model?



A single model can solve all of the training tasks zero-shot

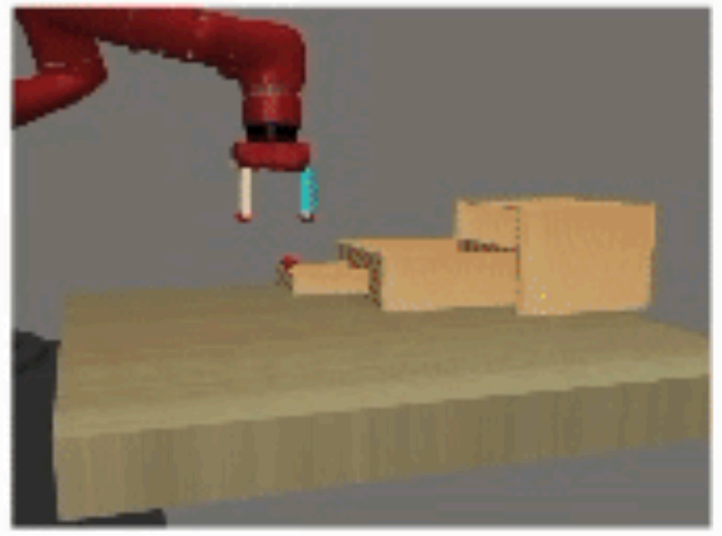


However, such solution cannot solve meta-testing tasks without using the task training data

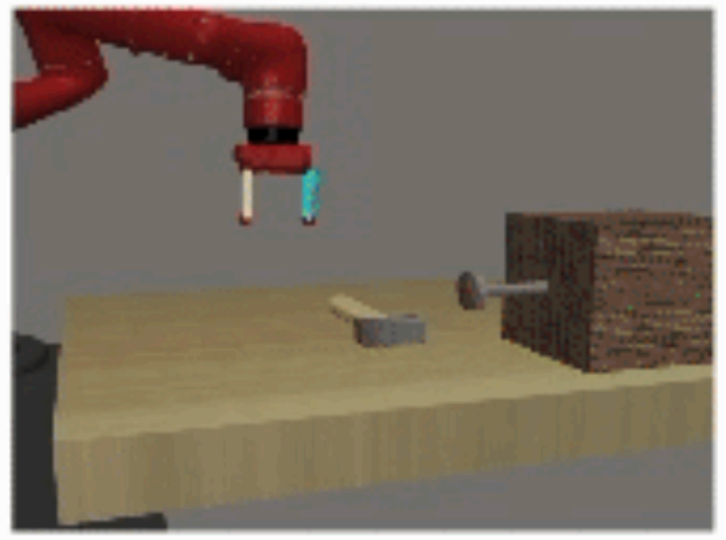
Another example

meta-training

"close drawer"

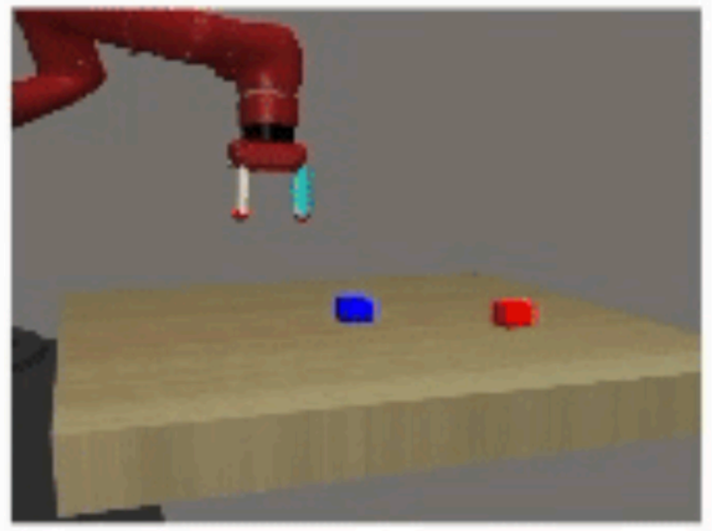
$\mathcal{T}_1$ 

"hammer"

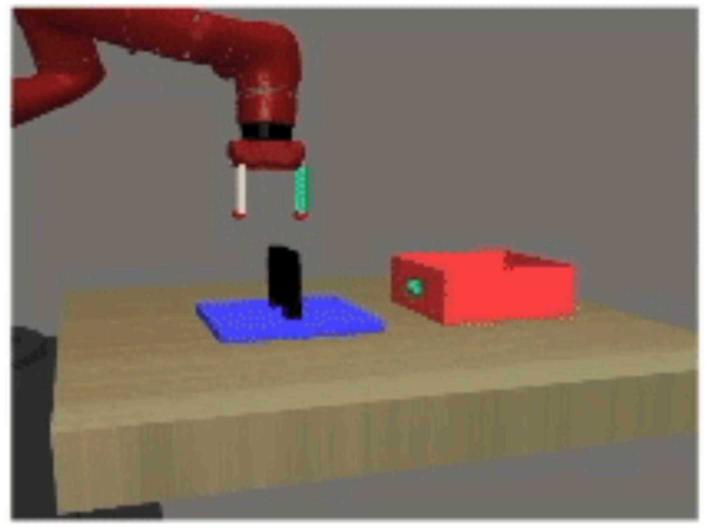
$\mathcal{T}_2$ 

$\dots$

"stack"

$\mathcal{T}_{50}$ 

"close box"

$\mathcal{T}_{\text{test}}$ 

If you tell the robot the task goal, the robot can **ignore** the trials.

T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. Meta-World. CoRL '19

- We formally define it as the (complete) memorization problem:

$$I(\hat{y}^*_{val}; \mathcal{D}_{tr} | x^*_{val}, \theta) = 0, \text{ or equivalently } \hat{y}^*_{val} \perp \mathcal{D}_{tr} | x^*_{val}, \theta$$

- We identify that memorization is a general problem in many meta-learning algorithms, e.g. MAML, CNP

Can we do something about it?

- For mutually exclusive tasks (single function cannot solve all tasks):

  —> Not a problem!

  e.g. Few-shot classification: randomly shuffle the class labels across tasks

- For non-mutually exclusive tasks (single function can solve all tasks):

  —> multiple local optimums in the meta-learning objective

An entire spectrum of local optimums are based on how **information** flows.

Suggests a potential approach: control information flow.

Meta-regularization (MR)

minimize meta-training loss + information in θ

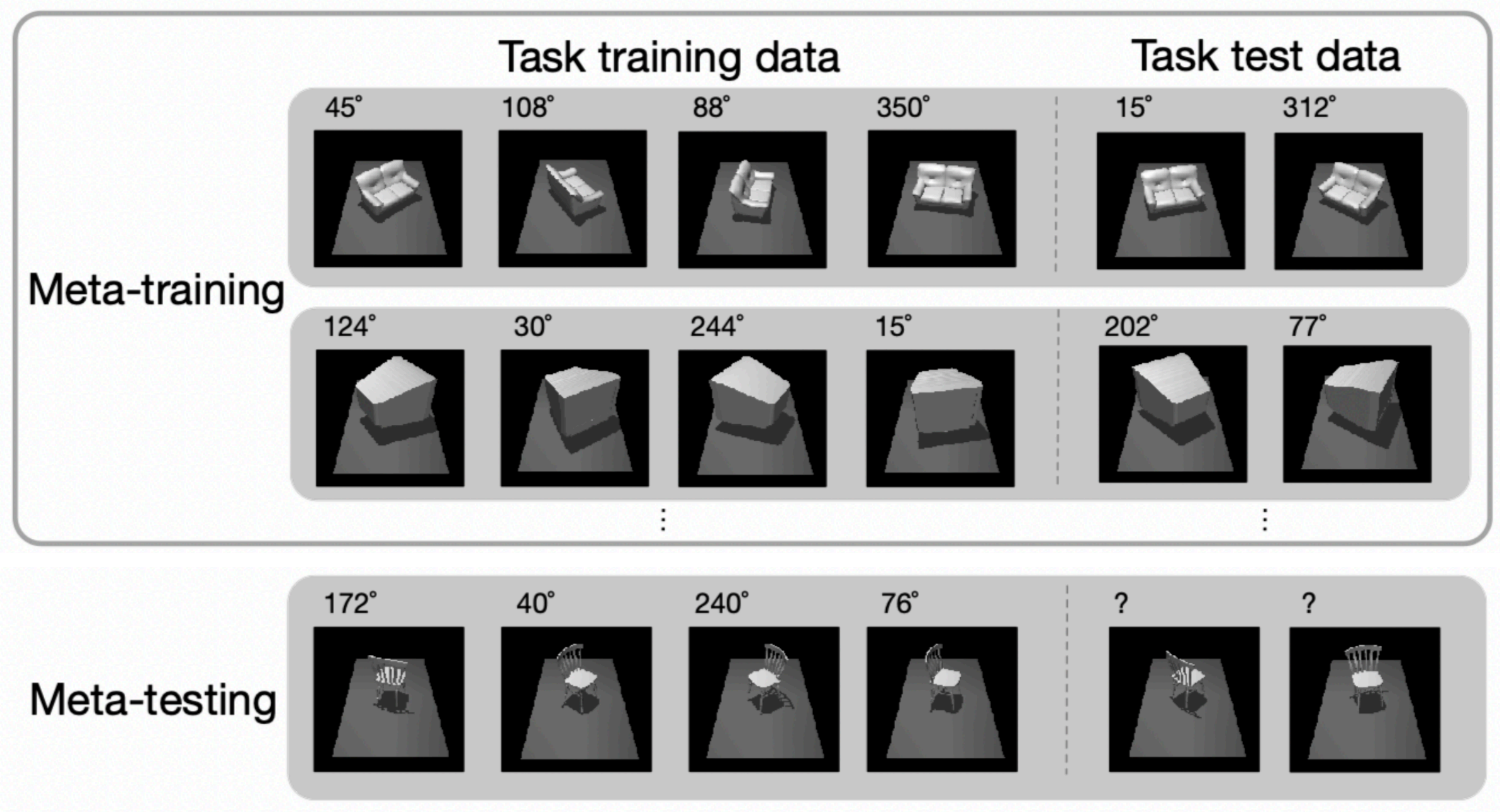
$$\mathscr{L}(\theta, \mathscr{D}_{meta-train}) + \beta D_{KL}(q(\theta; \theta_\mu, \theta_\sigma) \| p(\theta))$$

- Regularizes parameters that don't control the adaptation
- Can be derived from PAC-Bayes theory
- Can combine with many meta-learning algorithms, eg.
  MR-MAML, MR-CNP

Omniglot without label shuffling: "non-mutually-exclusive" Omniglot

NME Omniglot	20-way 1-shot	20-way 5-shot
MAML	7.8 (0.2)%	50.7 (22.9)%
TAML	9.6 (2.3)%	67.9 (2.3)%
MR-MAML (W) (ours)	**83.3 (0.8)**%	**94.1 (0.1)**%

On **pose prediction** task:



Method	MAML	MR-MAML(W) (ours)	CNP	MR-CNP(W) (ours)
MSE	5.39 (1.31)	**2.26 (0.09)**	8.48 (0.12)	2.89 (0.18)

(and it's not just as simple as standard regularization)

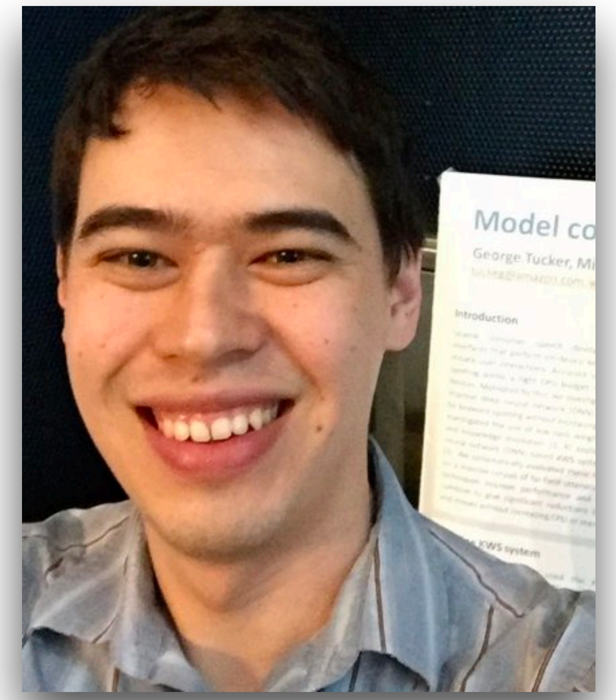
CNP	CNP + Weight Decay	CNP + BbB	MR-CNP (W) (ours)
8.48 (0.12)	6.86 (0.27)	7.73 (0.82)	**2.89 (0.18)**

TAML: Jamal & Qi. **Task-Agnostic Meta-Learning for Few-Shot Learning.** CVPR '19

Takeaways

- Memorization is a prevalent problem for many meta-learning tasks and algorithms
- Whether the algorithm converges to the memorization solution is related to the information flow
- Meta-regularization places precedence on using information from $\mathscr{D}_{\mathrm{tr}}$ over storing info in θ.

Collaborators




Code link: https://github.com/google-research/google-research/tree/master/meta_learning_without_memorization
Poster, slides & video link: https://mingzhang-yin.github.io/publications