Learning to generalize: meta-learning for domain generalization

Da Li, Yongxin Yang, Yi-Zhe Song, Timothy M. Hospedales

Published in: *Association for the Advancement of Artificial Intelligence (AAAI) 2018*

Centre for Intelligent Sensing
Queen Mary University of London

CIS centre for intelligent sensing

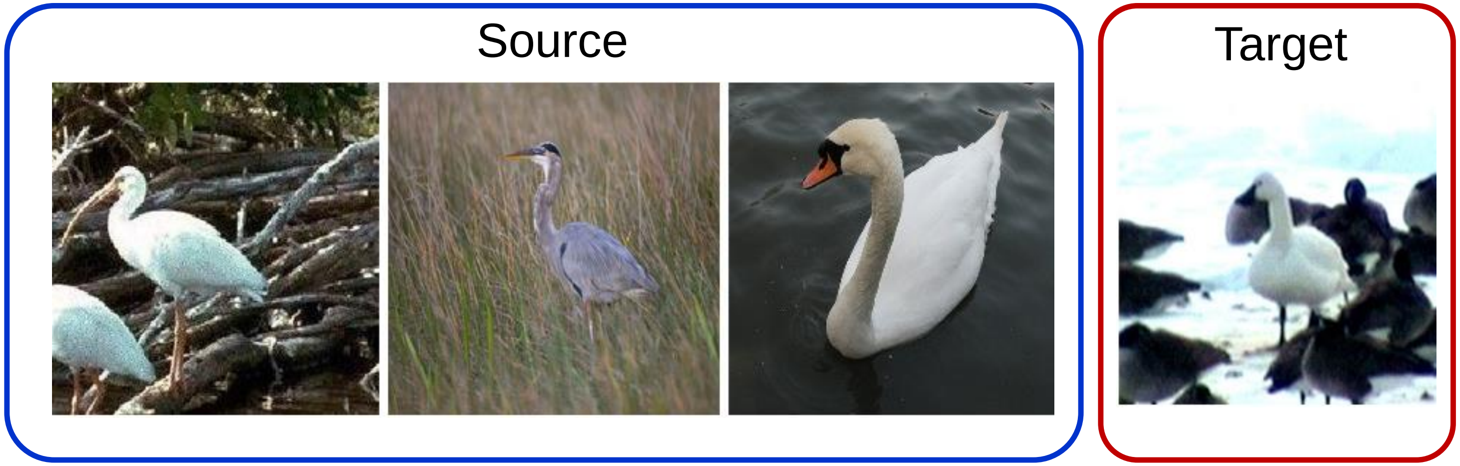
Queen Mary University of London

Domain Generalization

- *What* is domain generalization?
  - Domain generalization (DG) assumes a model is trained from multiple observed (source) domains while it is expected to perform well on any unseen (target) domains.

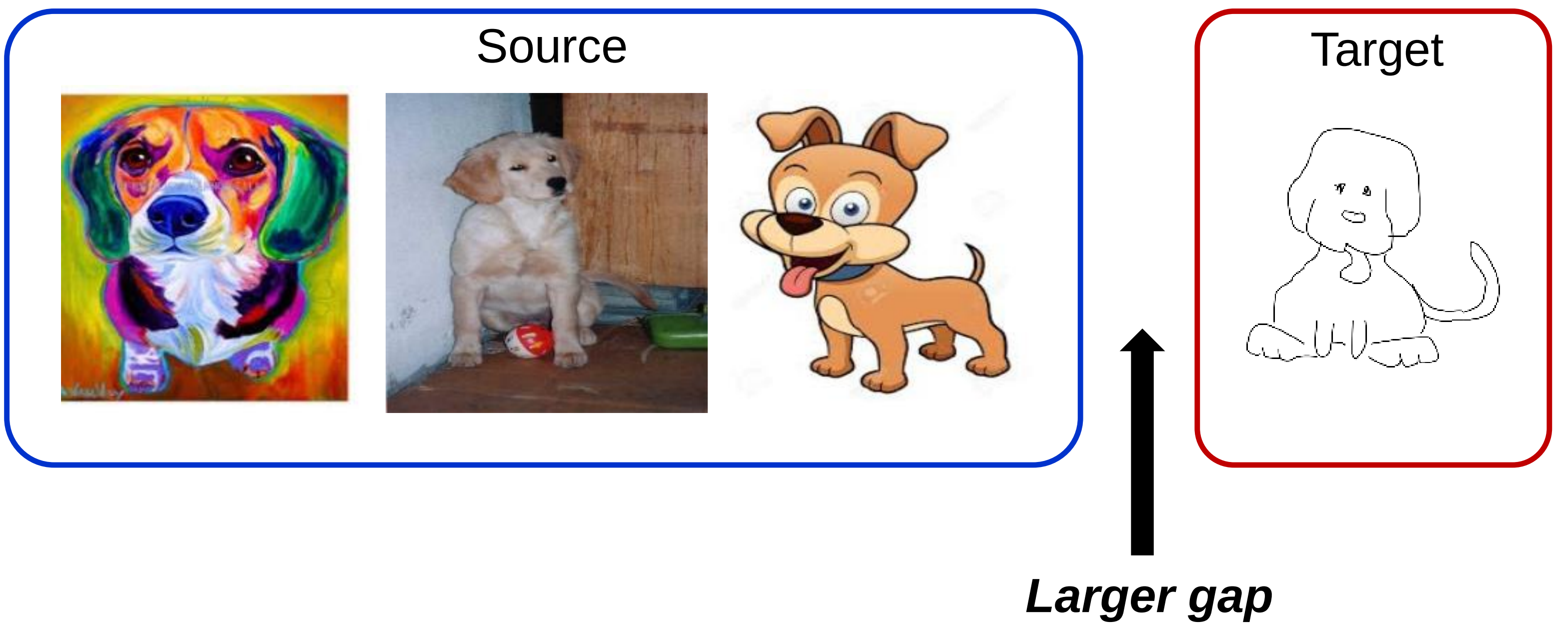
Domain Generalization

- *What* is domain generalization?
  - Domain generalization (DG) assumes a model is trained from multiple observed (source) domains while it is expected to perform well on any unseen (target) domains.



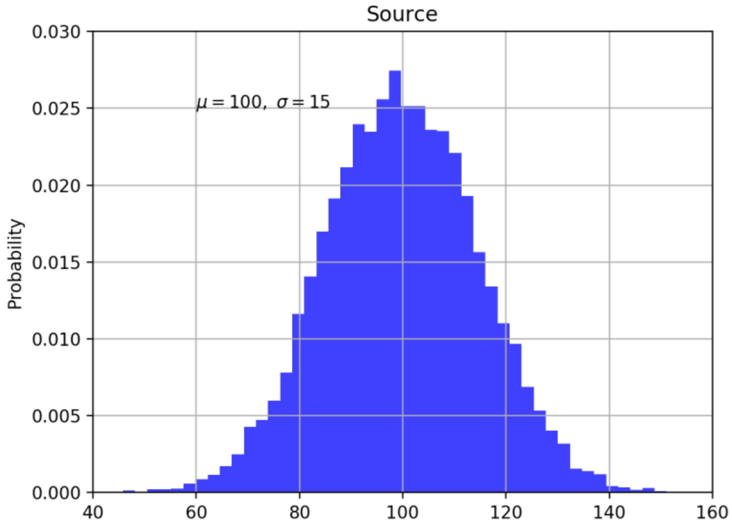
Domain Generalization

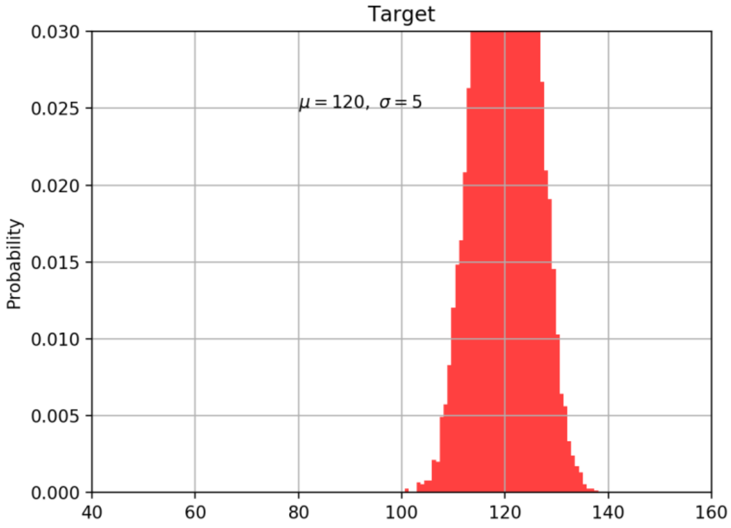
- *What* is domain generalization?
  - Domain generalization (DG) assumes a model is trained from multiple observed (source) domains while it is expected to perform well on any unseen (target) domains.



Domain Generalization

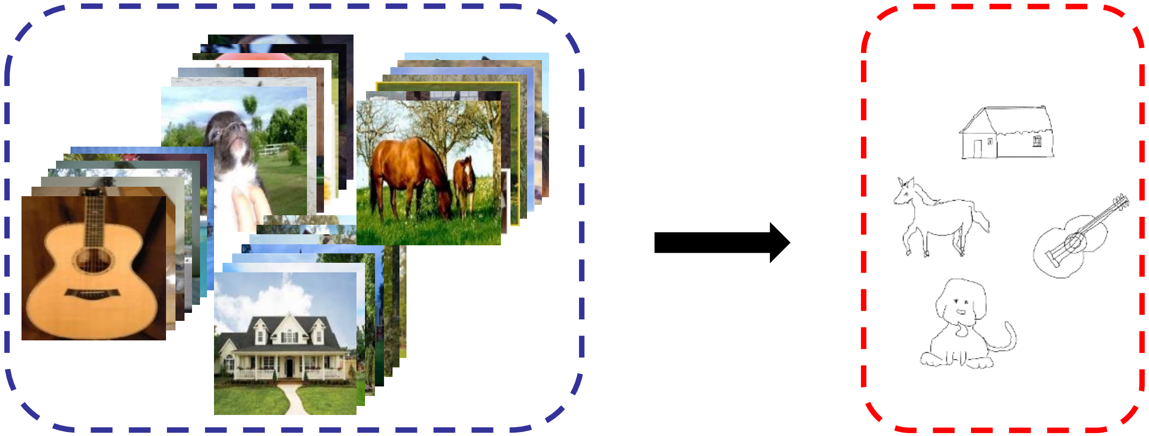
- *Why* DG is useful?
  - Data distribution shift could not be avoided, DG helps to solve this problem.





Domain Generalization

- *Why* DG is useful?
  - Data distribution shift could not be avoided, DG helps to solve this problem.
  - For the data (domain) of interest, but with too sparse data to train a good model, eg, Sketch. DG helps to train the model from the domain with large scale data, eg, Photo.



Domain Generalization

- *How* to reach DG*?*
  - Domain invariant feature learning.
    - [Muandet et al, ICML’13] ~*Kernel*

Domain Generalization

- *How* to reach DG*?*
  - Domain invariant feature learning.
    - [Muandet et al, ICML’13] ~*Kernel*
    - [Ghifary et al, ICCV’15; Li et al, CVPR’18] ~*Auto encoder*

Domain Generalization

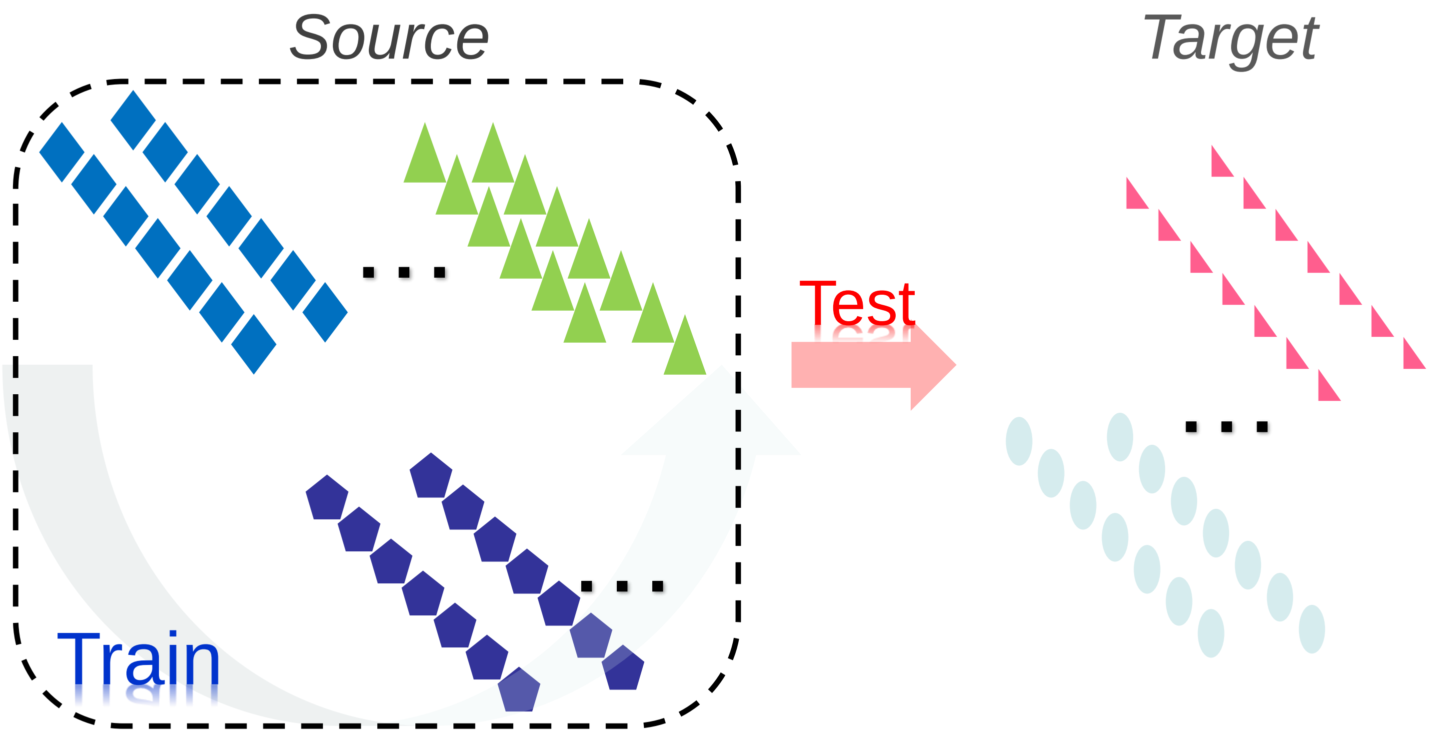
- *How* to reach DG*?*
  - Domain invariant feature learning.
    - [Muandet et al, ICML'13] ~*Kernel*
    - [Ghifary et al, ICCV'15; Li et al, CVPR'18] ~*Auto encoder*
  - Agnostic domain learning. [Li et al, ICCV'17] ~*CNN*

Domain Generalization

- *How* to reach DG?
  - Domain invariant feature learning.
    - [Muandet et al, ICML'13] ~*Kernel*
    - [Ghifary et al, ICCV'15; Li et al, CVPR'18] ~*Auto encoder*
  - Agnostic domain learning. [Li et al, ICCV'17] ~*CNN*
  - Data augmentation. [Shankar et al, ICLR'18] ~*Bayesian Network*

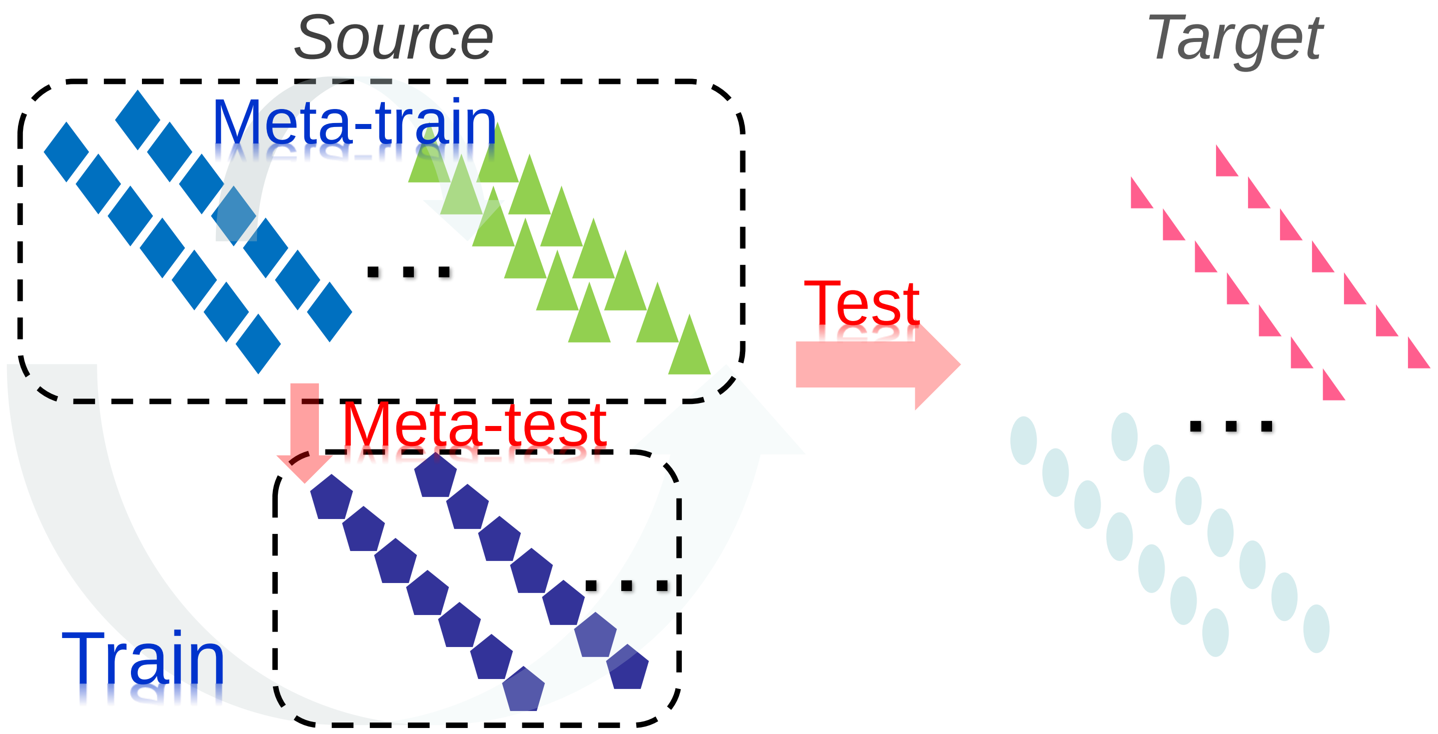
Learning to Generalize

- Conventional methods.



Learning to Generalize

- MLDG method.



Supervised Learning

- Meta-learning DG in supervised learning scenario.

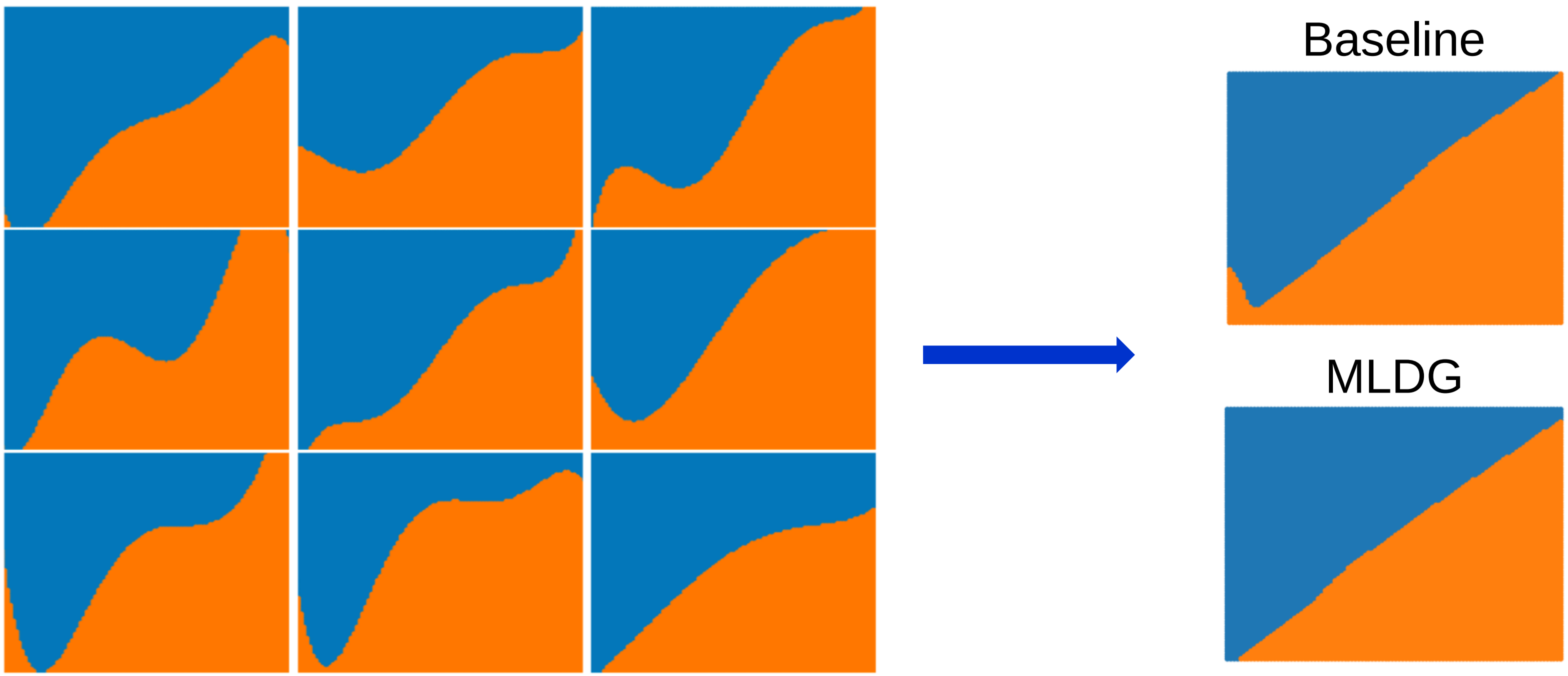
Algorithm 1 Meta-Learning Domain Generalization

```
procedure MLDG
    Input: Domains S
    Init: Model parameters Θ. Hyperparameters α, β, γ.
    for ite in iterations do
        Split: S̄ and S̆ ← S
        Meta-train: Gradients ∇_Θ = F'_Θ(S̄; Θ)
        Updated parameters Θ' = Θ − α∇_Θ
        Meta-test: Loss is G(S̆; Θ').
        Meta-optimization: Update Θ
```

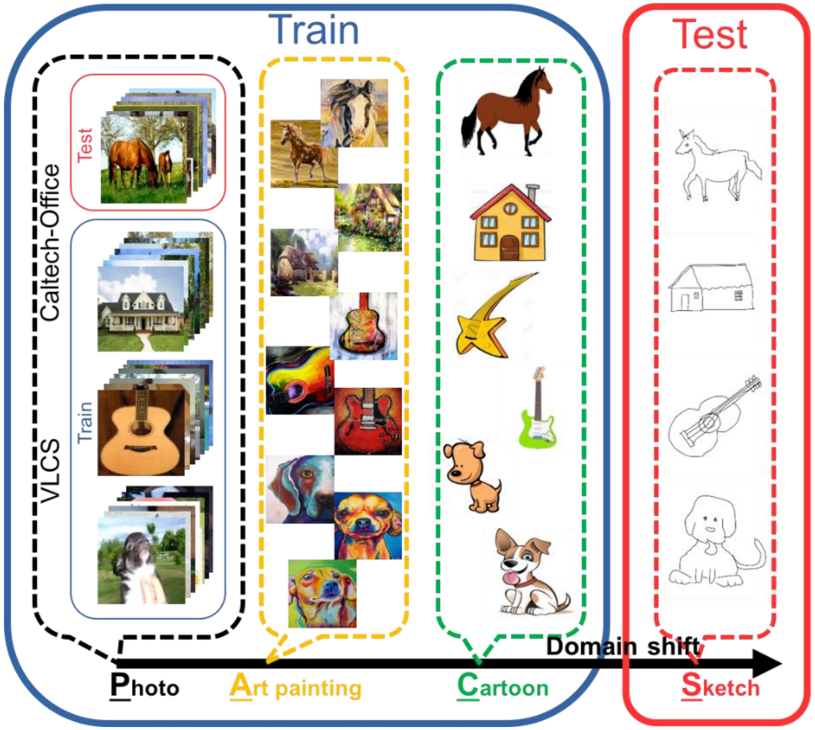
$$\Theta = \Theta - \gamma \frac{\partial(\mathcal{F}(\bar{\mathcal{S}};\Theta) + \beta\mathcal{G}(\breve{\mathcal{S}};\Theta - \alpha\nabla_\Theta))}{\partial\Theta}$$

```
    end for
end procedure
```

Supervised Learning – Synthetic experiment



Supervised Learning – PACS



	D-MTAE (Ghifary et al. 2015)	Deep-all	DSN (Bousmalis et al. 2016)	AlexNet+TF (Li et al. 2017)	MLDG (CNN)
art_painting	60.27	64.91	61.13	62.86	**66.23**
cartoon	58.65	64.28	66.54	**66.97**	66.88
photo	**91.12**	86.67	83.25	89.50	88.00
sketch	47.86	53.08	58.58	57.51	**58.96**
Ave.	64.48	67.24	67.37	69.21	**70.01**

https://domaingeneralization.github.io

Taylor Series (first order)

meta-train loss meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta \mathcal{G}(\Theta - \alpha \mathcal{F}'(\Theta))$$

Taylor Series (first order)

meta-train loss $\qquad$ meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

Taylor Series (first order)

meta-train loss meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \ \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

$$\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta)) = \mathcal{G}(\Theta) + \mathcal{G}'(\Theta) \cdot (-\alpha\mathcal{F}'(\Theta))$$

Taylor Series (first order)

meta-train loss   meta-test loss

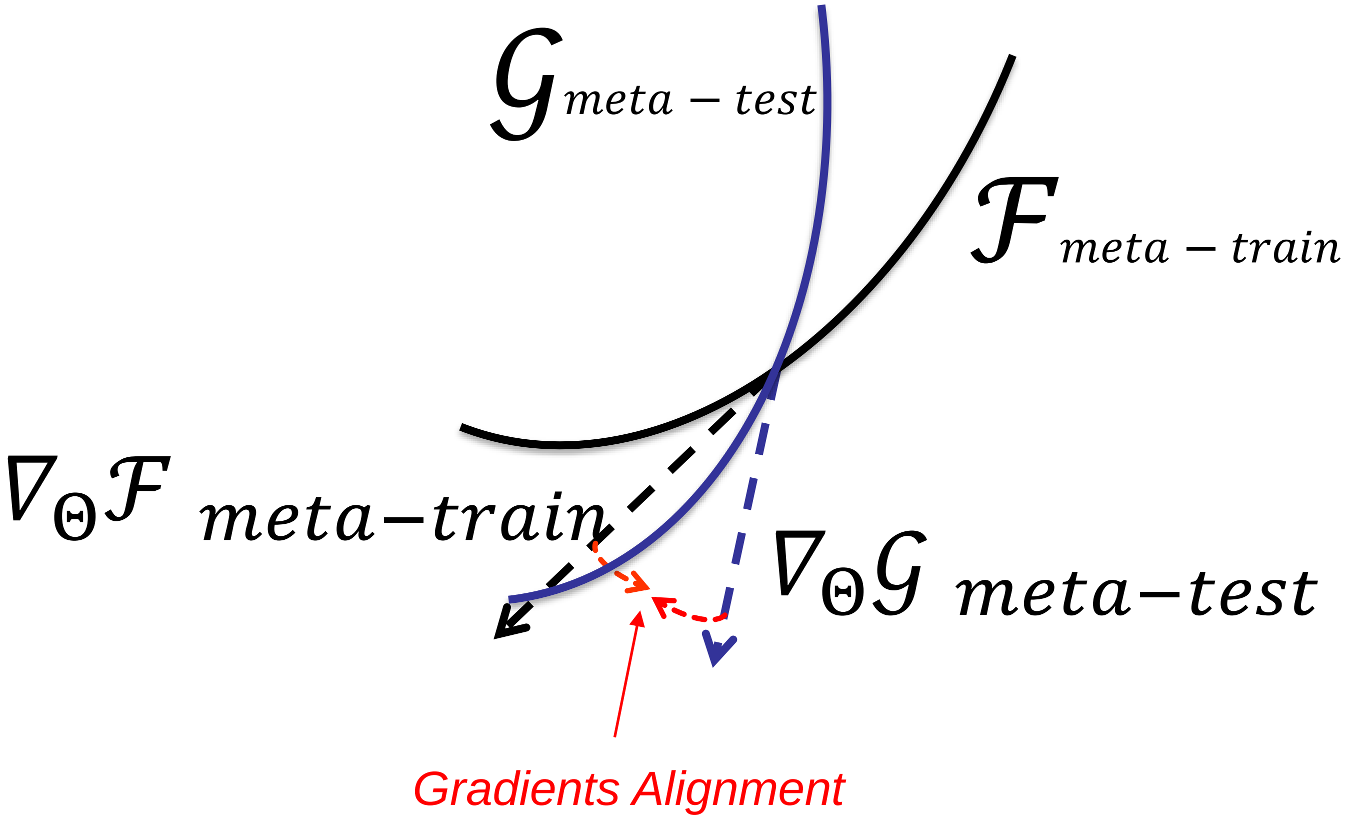
$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

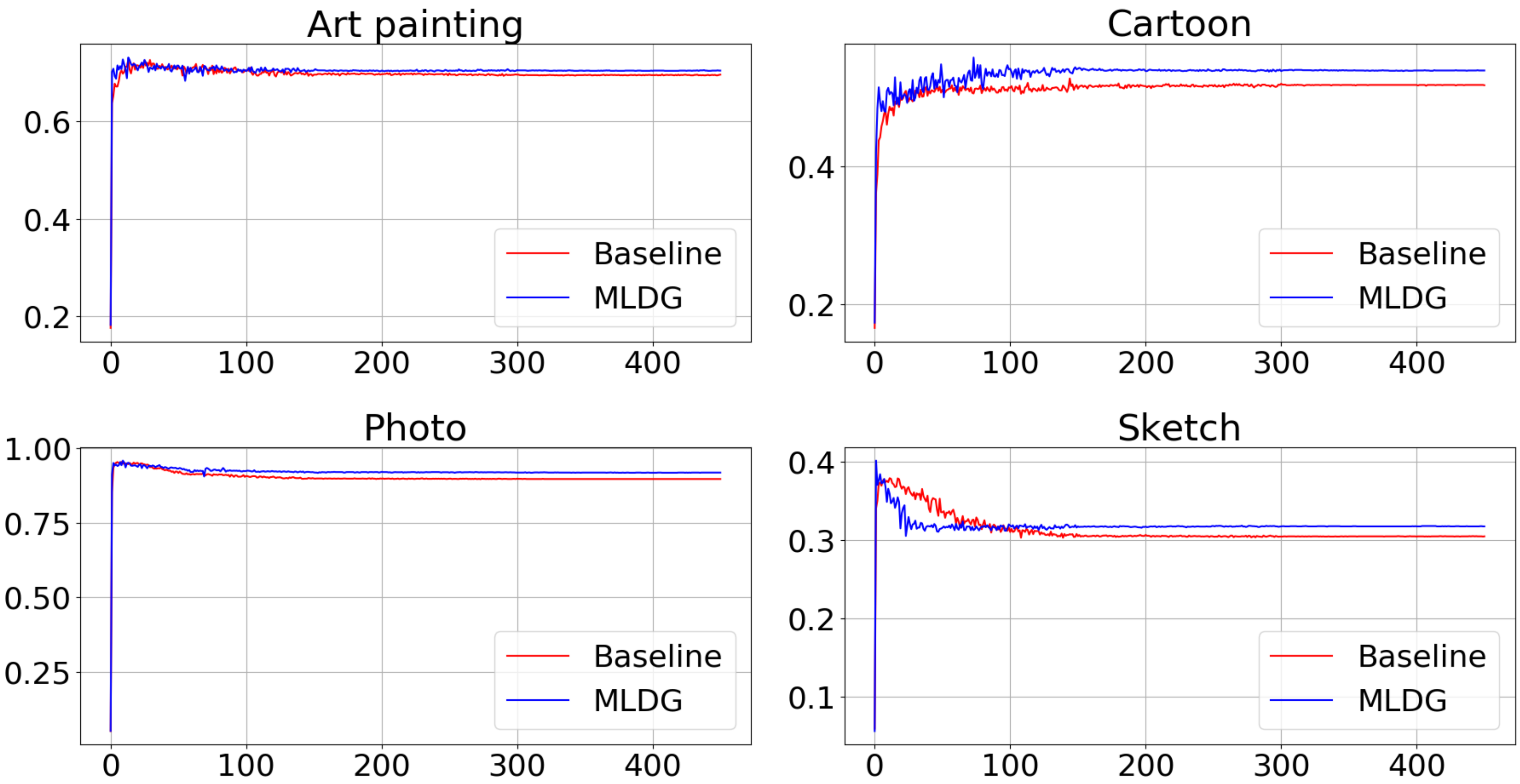
$$\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta)) = \mathcal{G}(\Theta) + \mathcal{G}'(\Theta) \cdot (-\alpha\mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha(\mathcal{G}'(\Theta) \cdot \mathcal{F}'(\Theta))$$

Gradients Alignment



Test performance curve



CIS centre for intelligent sensing

Queen Mary University of London

Taylor Series (first order)

meta-train loss meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta \mathcal{G}(\Theta - \alpha \mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

$$\mathcal{G}(\Theta - \alpha \mathcal{F}'(\Theta)) = \mathcal{G}(\Theta) + \mathcal{G}'(\Theta) \cdot (-\alpha \mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta \mathcal{G}(\Theta) - \beta\alpha(\mathcal{G}'(\Theta) \cdot \mathcal{F}'(\Theta))$$

Taylor Series (first order)

meta-train loss meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \ \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

$$\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta)) = \mathcal{G}(\Theta) + \mathcal{G}'(\Theta) \cdot (-\alpha\mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \ \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha(\mathcal{G}'(\Theta) \cdot \mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \ \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha \frac{\mathcal{F}'(\Theta) \cdot \mathcal{G}'(\Theta)}{\|\mathcal{F}'(\Theta)\|_2 \|\mathcal{G}'(\Theta)\|_2}$$

Taylor Series (first order)

meta-train loss   meta-test loss

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta))$$

$$\mathcal{G}(x) = \mathcal{G}(\dot{x}) + \mathcal{G}'(\dot{x}) \times (x - \dot{x})$$

$$\mathcal{G}(\Theta - \alpha\mathcal{F}'(\Theta)) = \mathcal{G}(\Theta) + \mathcal{G}'(\Theta) \cdot (-\alpha\mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha(\mathcal{G}'(\Theta) \cdot \mathcal{F}'(\Theta))$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha\frac{\mathcal{F}'(\Theta) \cdot \mathcal{G}'(\Theta)}{\|\mathcal{F}'(\Theta)\|_2\|\mathcal{G}'(\Theta)\|_2}$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\|\mathcal{G}'(\Theta - \alpha\mathcal{F}'(\Theta))\|_2^2$$

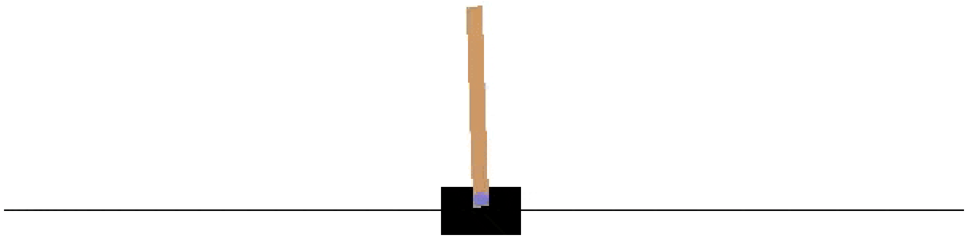
Variants

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\mathcal{G}(\Theta) - \beta\alpha \frac{\mathcal{F}'(\Theta) \cdot \mathcal{G}'(\Theta)}{\|\mathcal{F}'(\Theta)\|_2 \|\mathcal{G}'(\Theta)\|_2}$$

$$\underset{\Theta}{\operatorname{argmin}} \; \mathcal{F}(\Theta) + \beta\|\mathcal{G}'(\Theta - \alpha\mathcal{F}'(\Theta))\|_2^2$$

	Deep-All	MLDG-GC (Eq. 8)	MLDG-GN (Eq. 9)
art_painting	64.91	64.71	63.64
cartoon	64.28	65.30	63.47
photo	86.67	86.79	87.88
sketch	53.08	56.92	54.94
Ave.	67.24	68.43	67.48

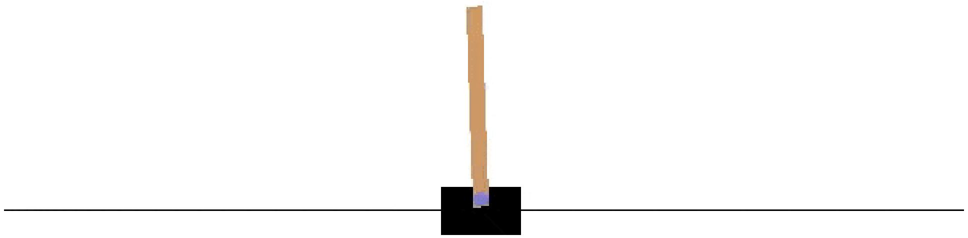
Reinforcement Learning – Cart pole



Method	RL-Random-Source	RL-All	RL-Undobias
Return	133.74 ± 6.79	97.39 ± 73.49	113.52 ± 11.65
Method	RL-MLDG	RL-MLDG-GC	RL-MLDG-GN
Return	165.34 ± 3.38	129.56 ± 2.51	175.25 ± 3.16

Across pole length. Return: reward score (upper bound=200)

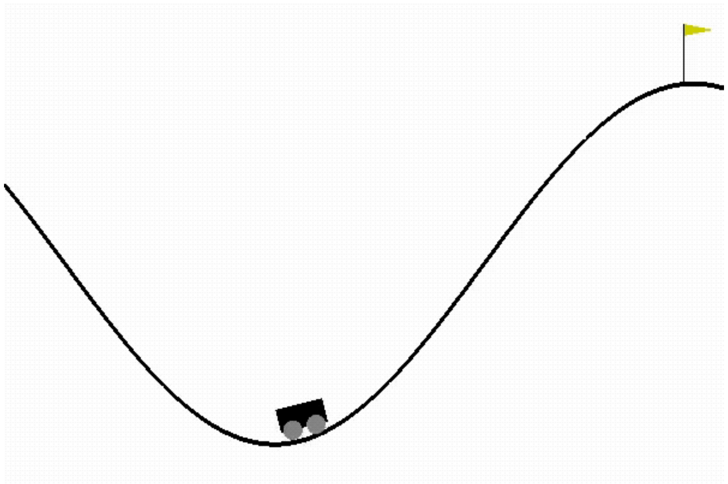
Reinforcement Learning – Cart pole



Method	RL-Random-Source	RL-All	RL-Undobias
Return	98.22 ± 20.35	144.21 ± 9.23	150.46 ± 17.59
Method	RL-MLDG	RL-MLDG-GC	RL-MLDG-GN
Return	170.81 ± 9.90	147.76 ± 4.41	164.97 ± 8.45

Across pole length and cart mass. Return: reward score (upper bound=200)

Reinforcement Learning – Mountain car



Mountain Car	RL-Random-Source	RL-All	RL-Undobias
Avg. F Rate	0.55 ± 0.07	0.05 ± 0.02	0.08 ± 0.04
Avg. Return	-191.07 ± 3.01	-141.35 ± 2.64	-124.48 ± 3.22
Mountain Car	**RL-MLDG**	**RL-MLDG-GC**	**RL-MLDG-GN**
Avg. F Rate	0.05 ± 0.02	0.0 ± 0.0	1.0 ± 0.0
Avg. Return	-125.73 ± 2.76	-311.80 ± 3.92	-

Across mountain height. Return: negative reward score

Thank you!