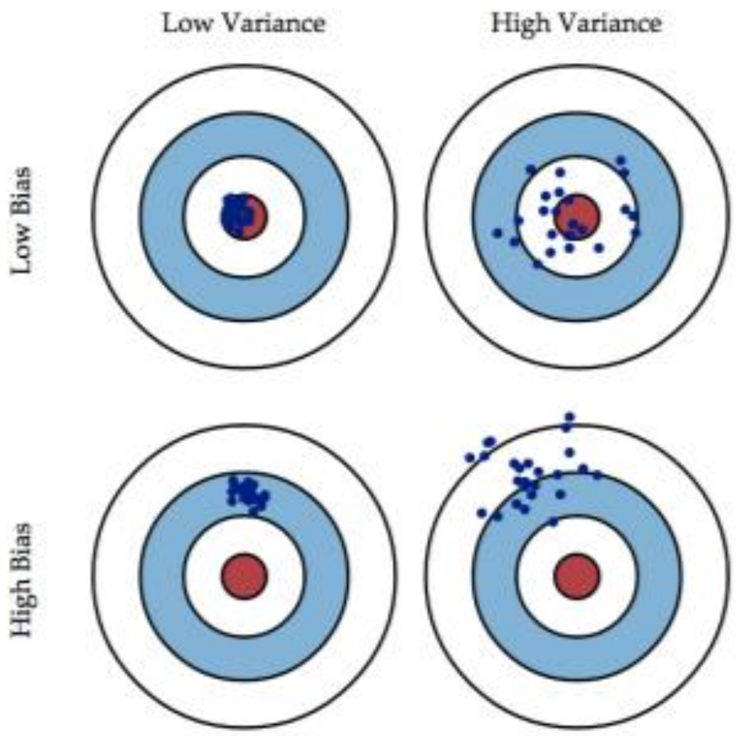
Meta Dropout
Learning to Perturb Features for Generalization

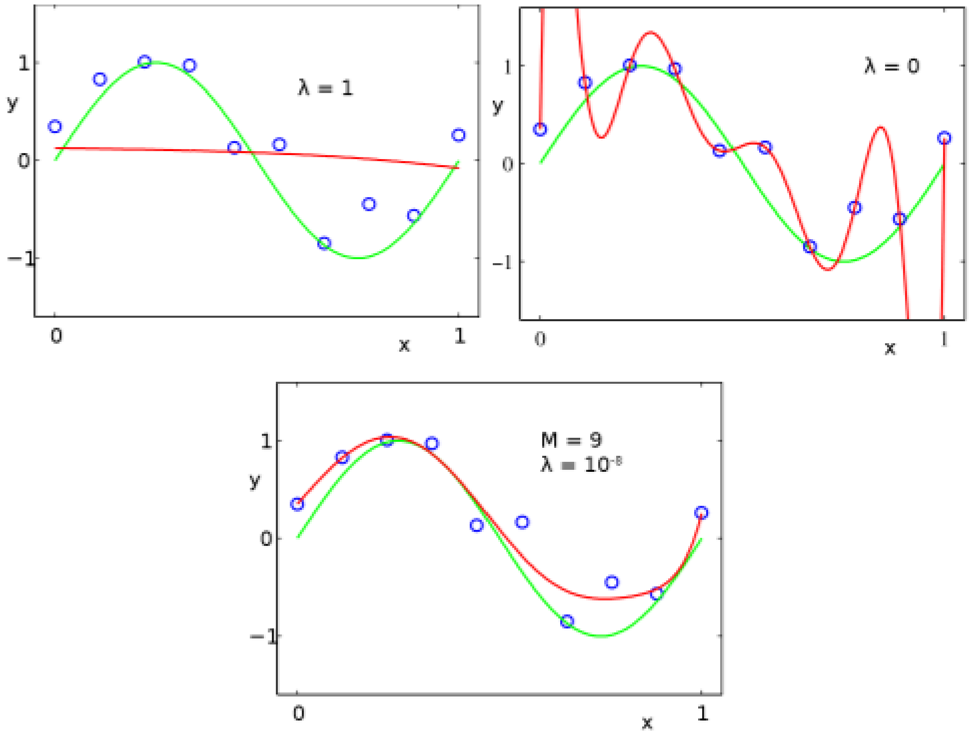
Hae Beom Lee, Taewook Nam, Eunho Yang, Sung Ju Hwang

Tobias Schmidt

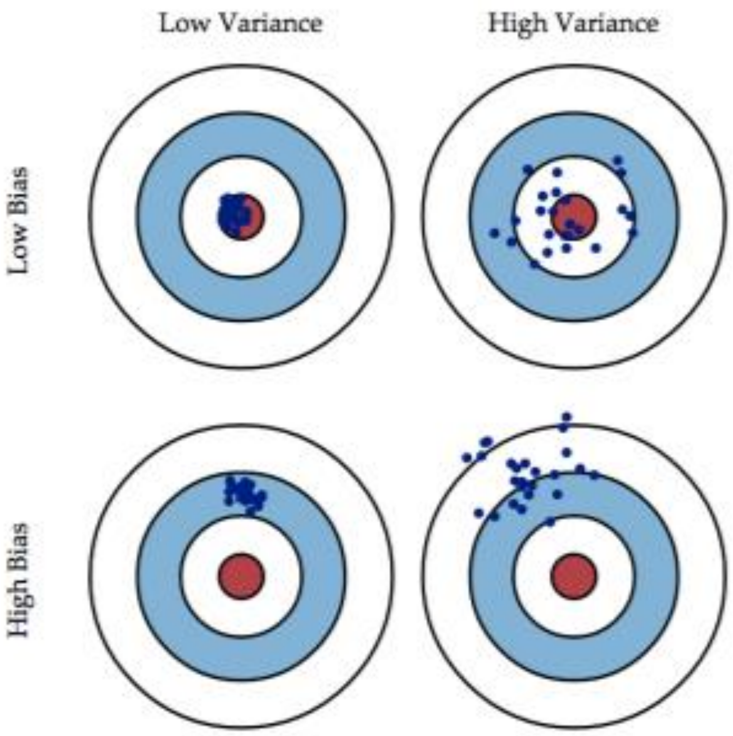
Munich, 7th July 2021

Motivation – Generalization





Motivation – Generalization



Common Solutions - Bias/Variance Trade-Off

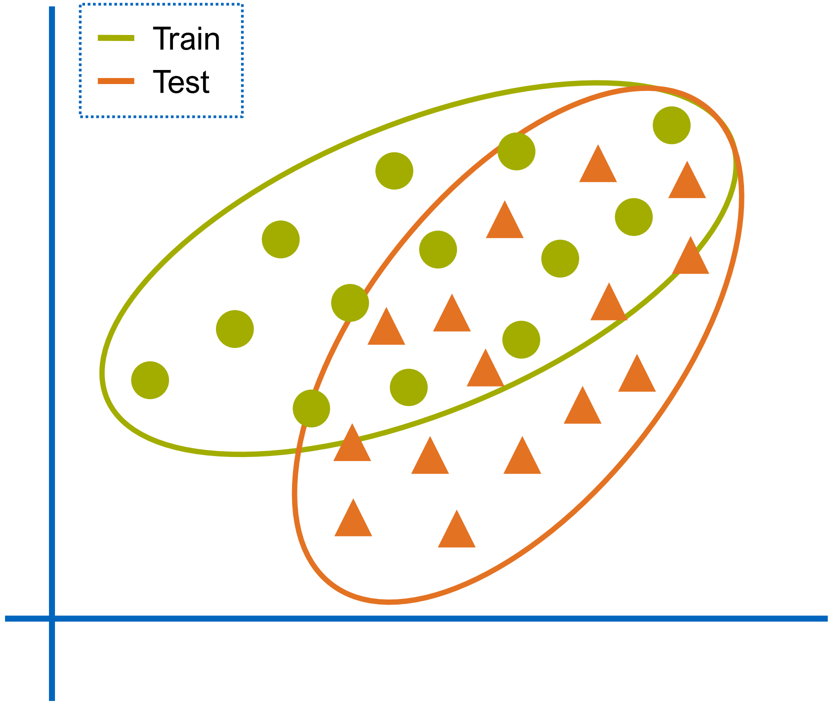
Appropriate Priors

- Image Convolutions (Shift-Invariance)
- Graph Convolutions (Permutation-Invariance)
- …

Regularization

- Reducing model capacity
- Reducing information from inputs
- Smoothing loss surface
- Multi-task training
- Meta-Learning

Motivation – Train/Test Distribution Mismatch



Common Solutions - Bias/Variance Trade-Off

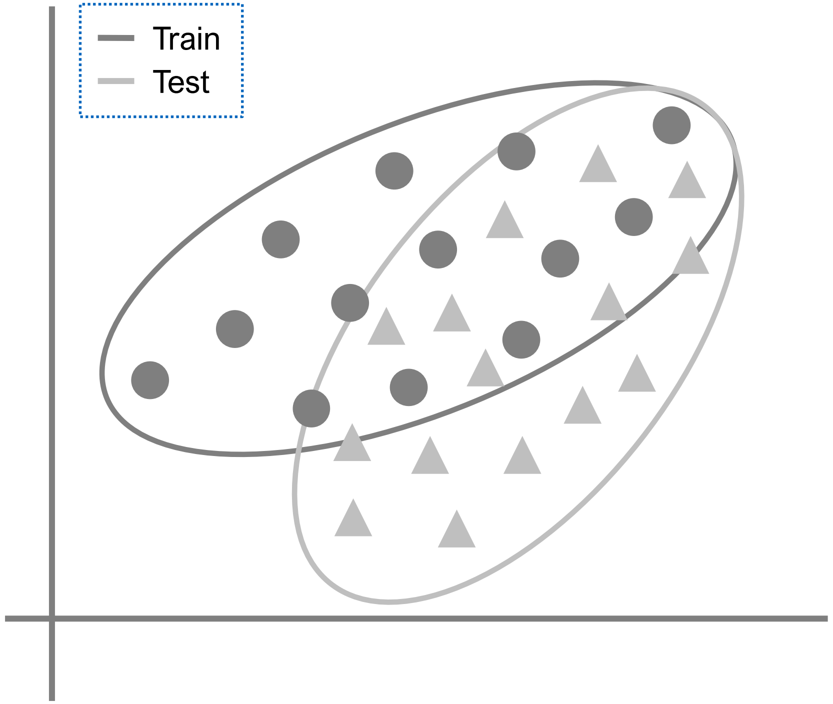
Appropriate Priors

- Image Convolutions (Shift-Invariance)
- Graph Convolutions (Permutation-Invariance)
- …

Regularization

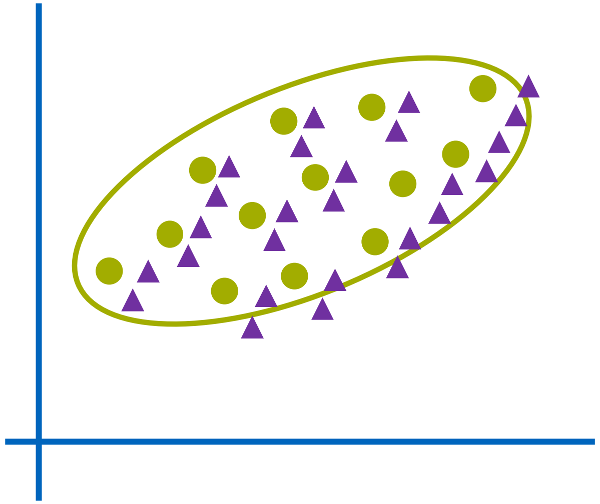
- Reducing model capacity
- Reducing information from inputs
- Smoothing loss surface
- Multi-task training
- Meta-Learning

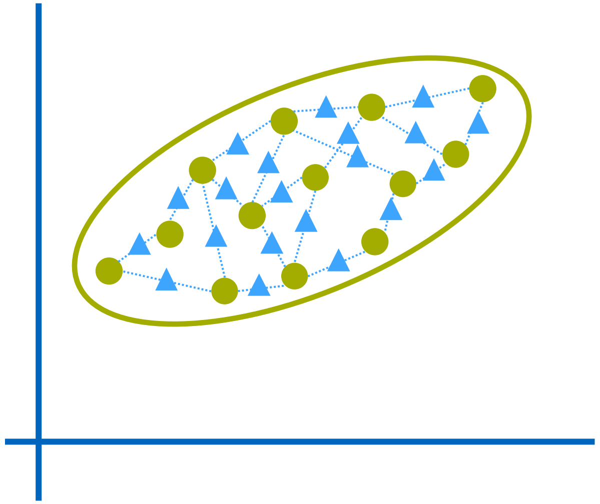
Motivation – Train/Test Distribution Mismatch



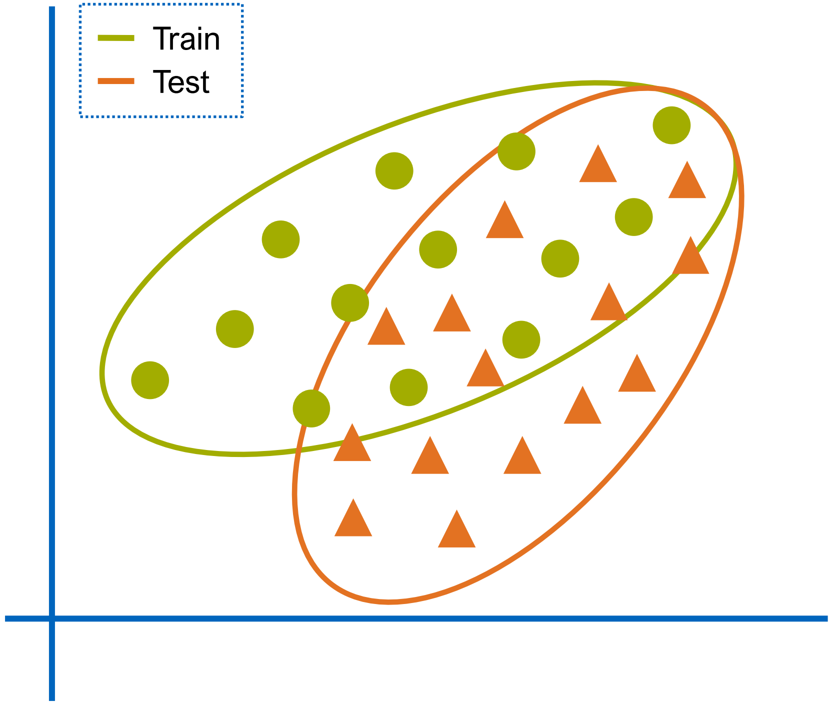
Alternative Solutions - Simulate Test Samples

Data augmentations	Mixup



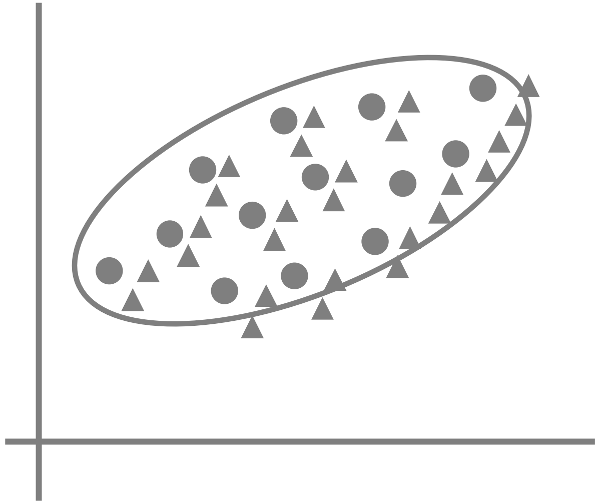


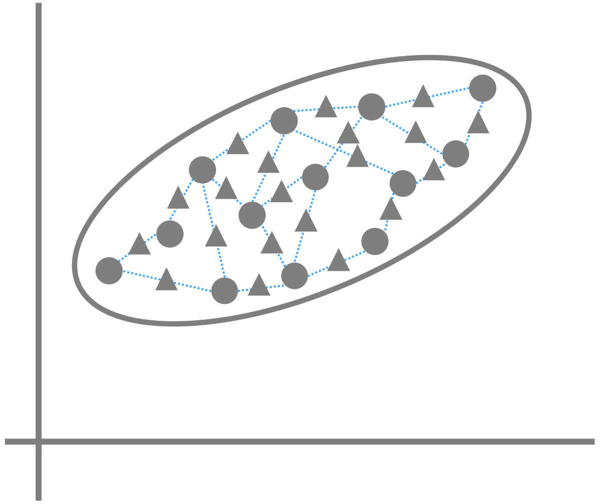
Motivation – Train/Test Distribution Mismatch



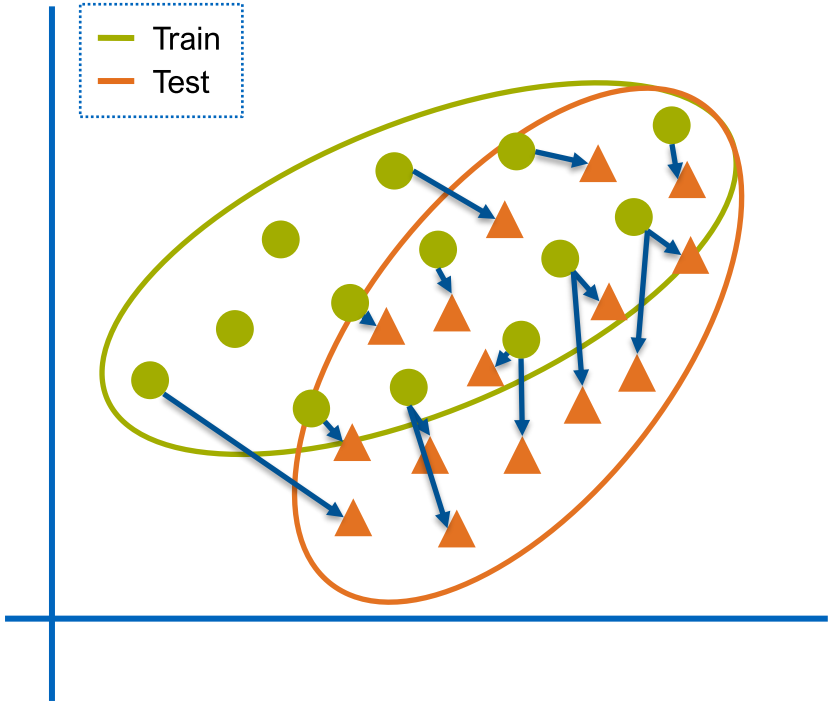
Alternative Solutions - Simulate Test Samples

Data augmentations	Mixup



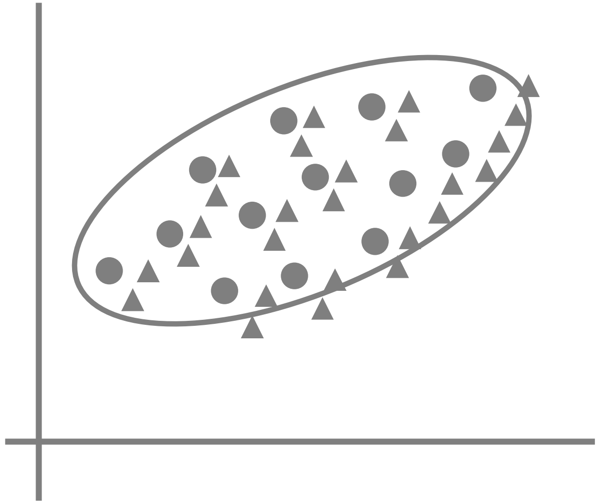


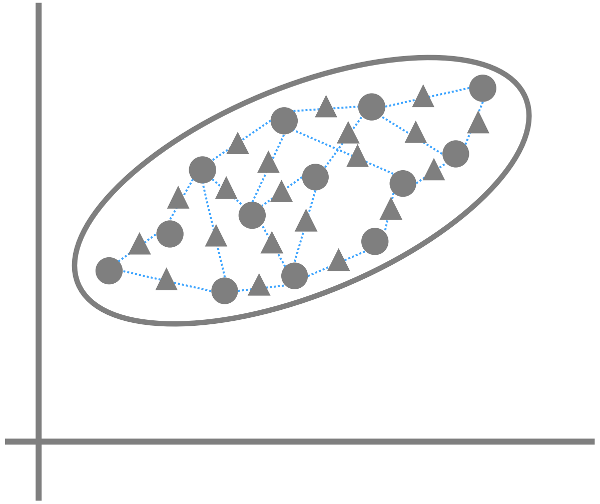
Motivation – Train/Test Distribution Mismatch



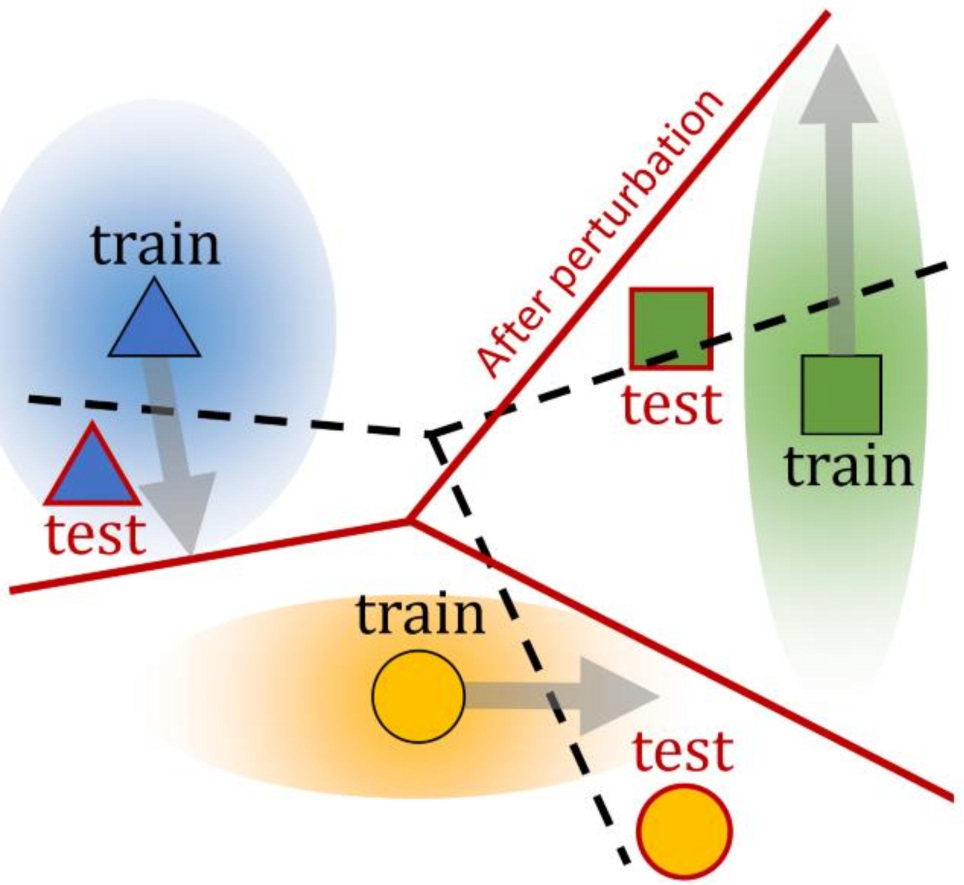
Alternative Solutions - Simulate Test Samples

Data augmentations	Mixup





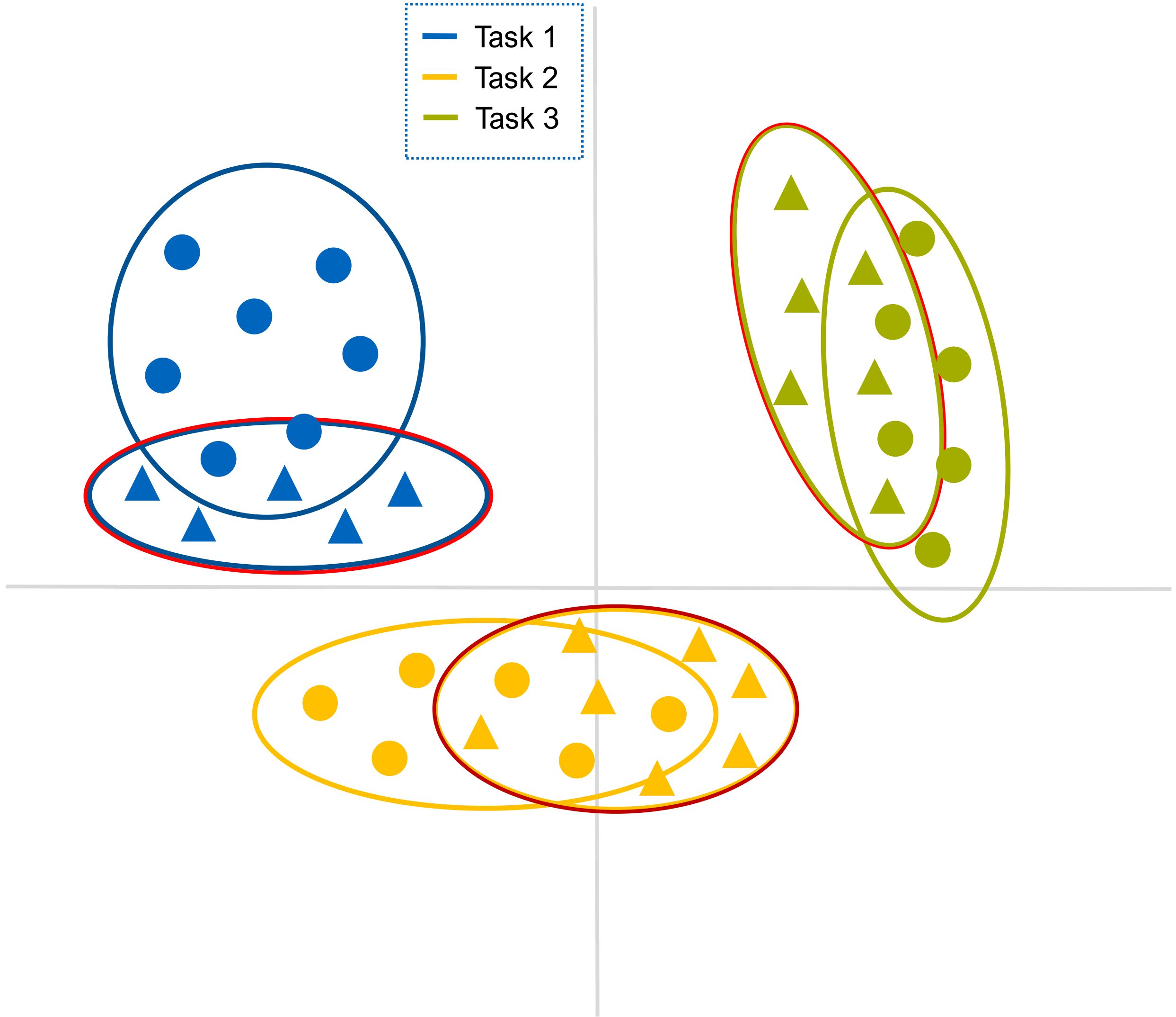
Idea: Learn to perturb the data for better generalization



Challenges

A training instance may need to cover multiple test instances

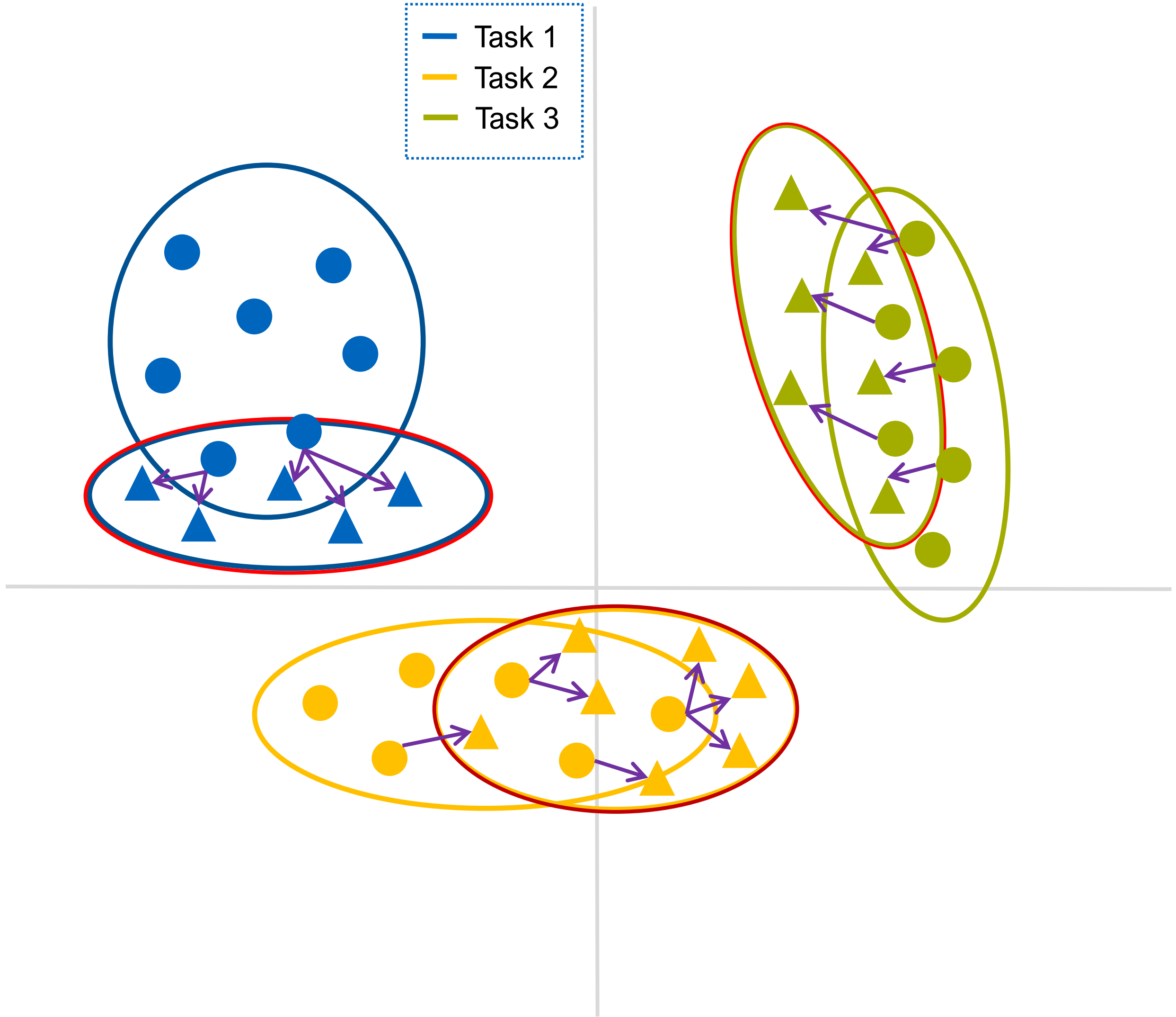
Meaningful directions differ from one task to another



Challenges

A training instance may need to cover multiple test instances

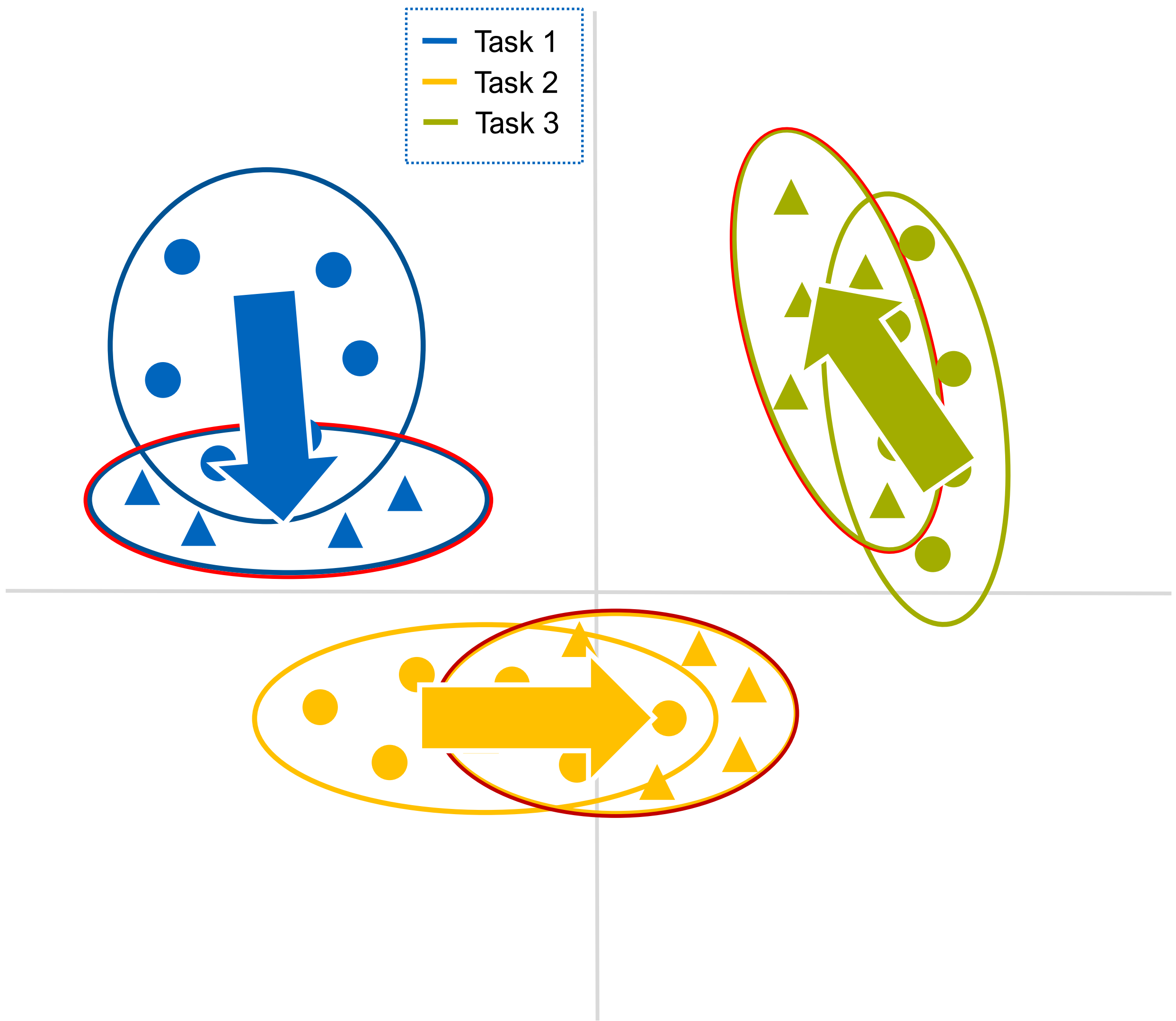
Meaningful directions differ from one task to another



Challenges

A training instance may need to cover multiple test instances

Meaningful directions differ from one task to another



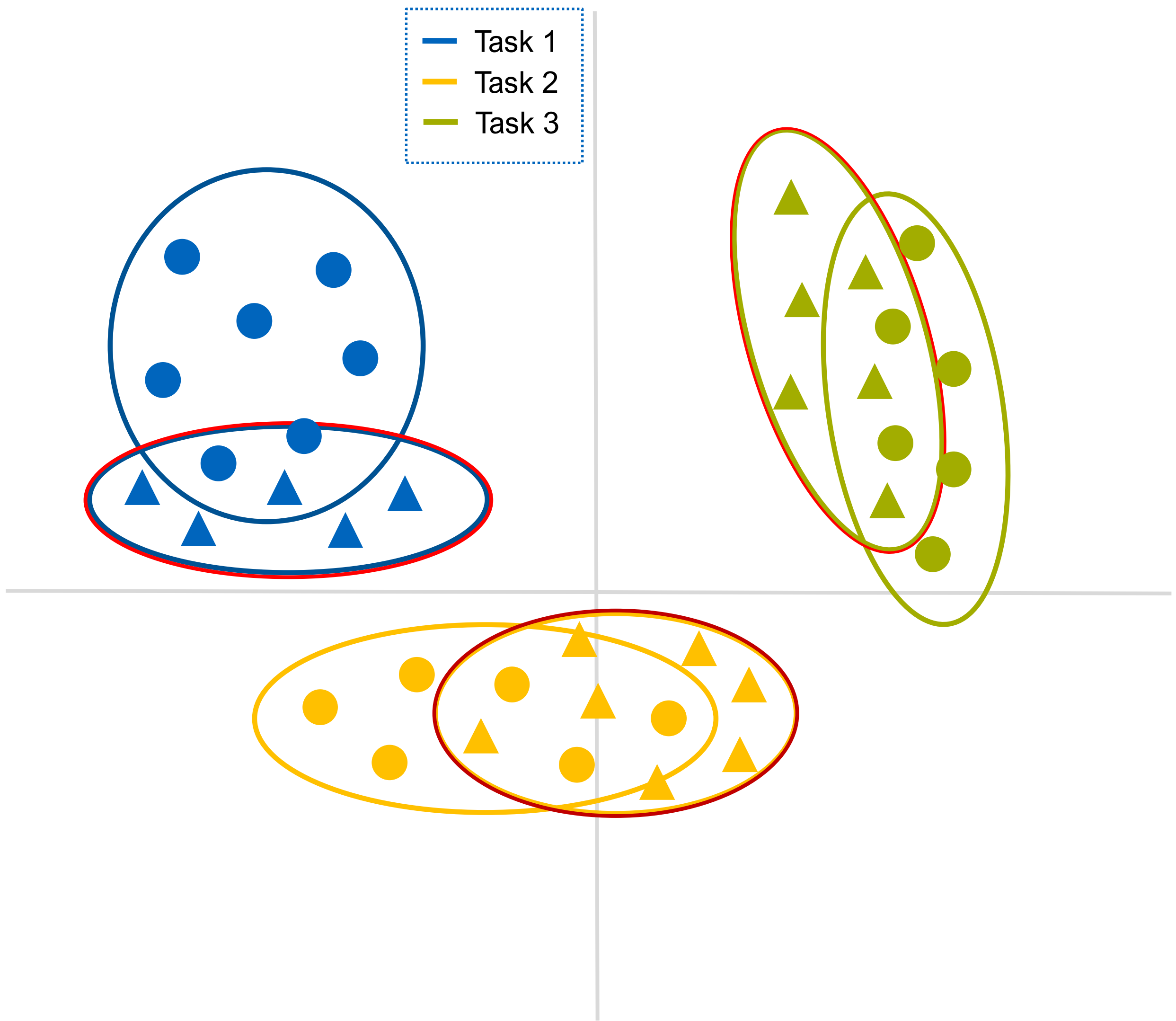
Challenges

A training instance may need to cover multiple test instances

→ Noise Distribution

Meaningful directions differ from one task to another

→ Input-Dependent Noise



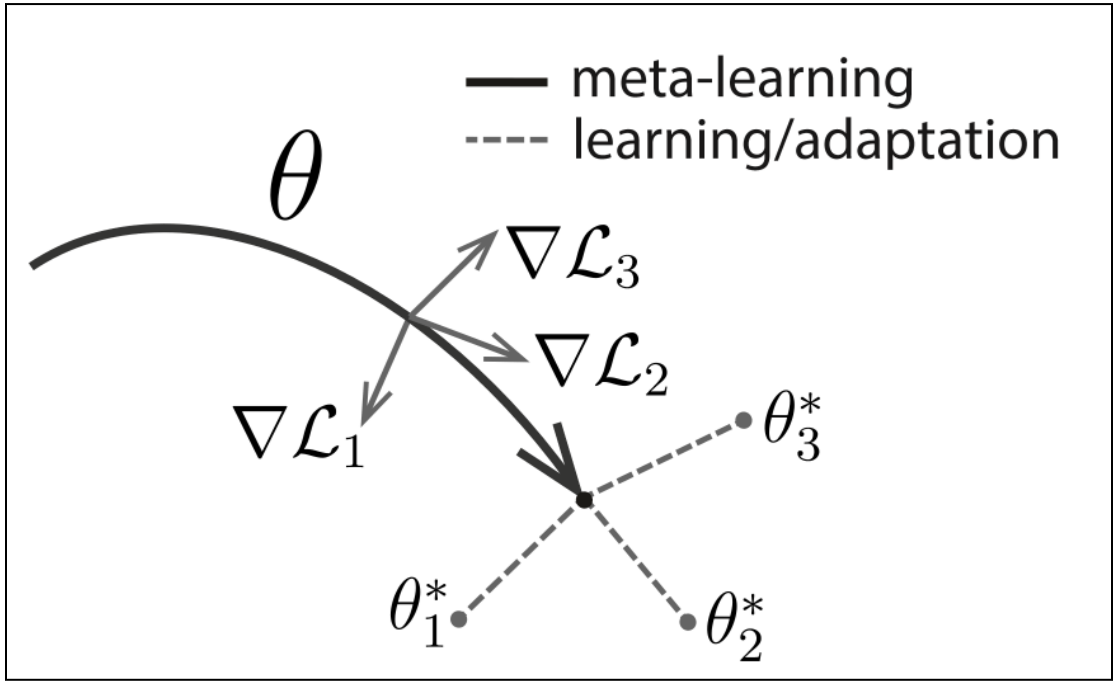
Meta Learning Framework - MAML

Properties / Limitations of MAML

knowledge transfer via learned parameter $\boldsymbol{\theta}$

parameter $\boldsymbol{\theta}$ only implicitly captures test distributions

→ Misses out on important knowledge about task distribution



Recap: MAML

Update over task distribution:

- **initial parameter θ**

Outer Loop

Algorithm 1 Model-Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters

randomly initialize θ
while not done **do**
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ with respect to K examples
    Compute adapted parameters with gradient descent: $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
  end for
  Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i'})$
end while

Inner Loop

Perform Few-Shot Learning for each task

Meta Dropout

Update over task distribution:

- **initial parameter** θ
- **parameters ϕ of noise $p(z)$**

Outer Loop

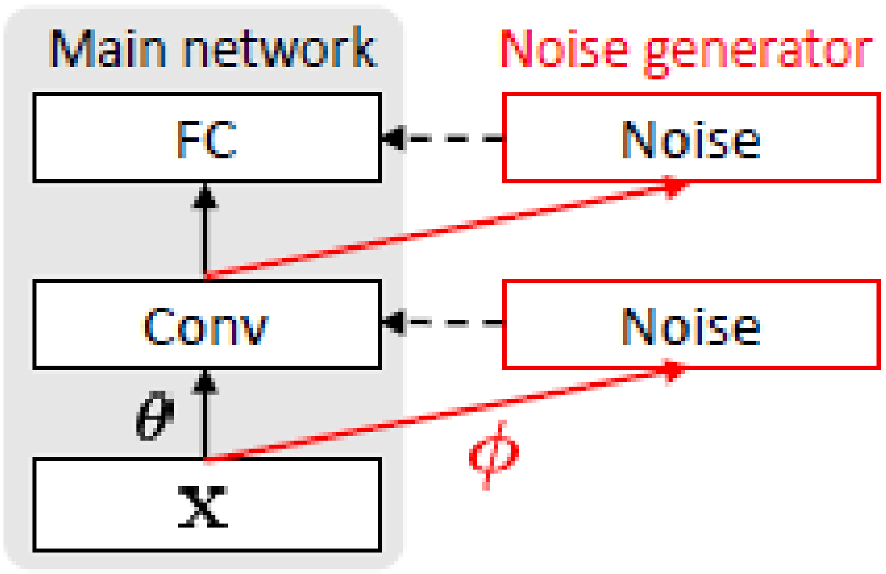
Algorithm 1 Meta-training

```
Input: Task distribution p(T), Number of inner steps K,
while not converged do
    Sample (D^tr, D^te) ~ p(T)
    θ_0 ← θ
    for k = 0 to K − 1 do
        Sample z̃_i ~ p(z_i | x_i^tr; φ, θ_k)   for  i = 1, ..., N
        θ_{k+1} ← θ_k + α ∇_{θ_k} (1/N) Σ_{i=1}^{N} log p(y_i^tr | x_i^tr, z̃_i; θ_k)
    end for
    θ* ← θ_K
    θ ← θ + β (1/M) Σ_{j=1}^{M} ∇_θ log p(y_j^te | x_j^te, z_j = z̄_j; θ*)
    φ ← φ + β (1/M) Σ_{j=1}^{M} ∇_φ log p(y_j^te | x_j^te, z_j = z̄_j; θ*)
end while
```

Inner Loop

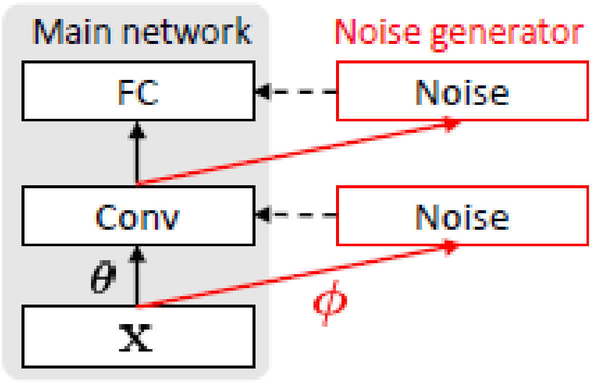
Perform Few-Shot Learning for each task **perturbing the input x_i^{tr} with multiplicative noise $\tilde{z}_i$**

Meta-Dropout: Model Architecture



Meta-Dropout: Implementation Detail

Model Architecture



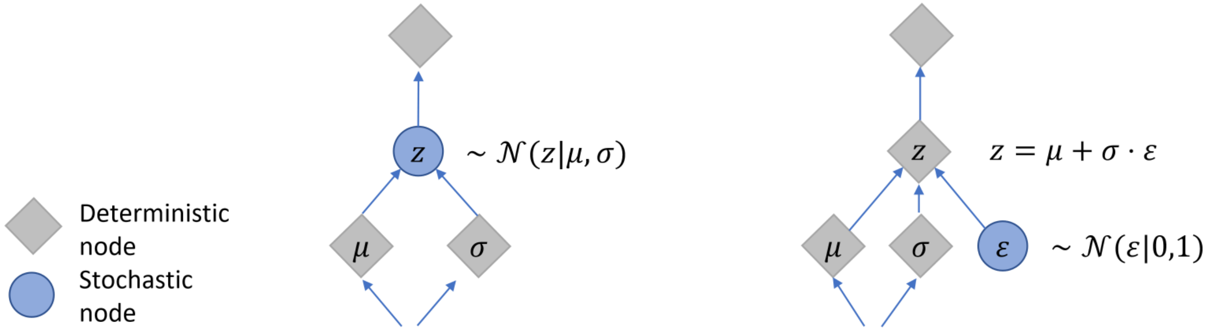
Reparameterization Trick

$$\log p(Y_i^{tr} \mid X_i^{tr}, \theta, \phi) \geq \sum_{i=1}^{N} \mathbb{E}_{z_i \sim p(z_i \mid x_i^{tr}, \theta, \phi)}[\log p(y_i^{tr} \mid x_i^{tr}, \theta, \phi)]$$

$$\approx \sum_{i=1}^{N} \boxed{\sum_{s=1}^{S}} \log p(y_i^{tr} \mid x_i^{tr}, z_i^{(s)}, \theta) \;\; with \;\; z_i^{(s)} \sim p(z_i \mid x_i^{tr}, \boxed{\theta, \phi})$$

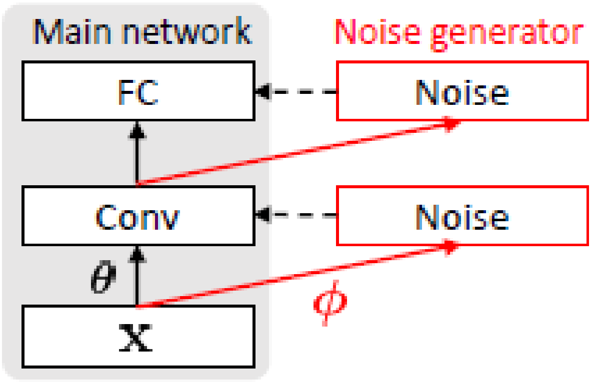
Approximation via Monte-Carlo

Impossible to calculate gradient!



Meta-Dropout: Implementation Detail

Model Architecture



Form of the Noise: $p\left(\boldsymbol{y}_i^{tr} \mid \boldsymbol{x}_i^{tr}, \boldsymbol{z}_i^{(s)}, \boldsymbol{\theta}\right)$ $with\ \boldsymbol{z}_i^{(s)} \sim p(\boldsymbol{z}_i|\boldsymbol{x}_i^{tr}, \boldsymbol{\theta}, \boldsymbol{\phi})$

Additive Noise

$$\boldsymbol{h}^{(0)} = \boldsymbol{x}_i^{tr}$$
$$\boldsymbol{h}^{(l)} = ReLU\left(f^{(l)}\left(\boldsymbol{h}^{(l-1)}\right) + \boldsymbol{z}^{(l)}\right)$$

$$\boldsymbol{z}_i^{(s)} \sim N\left(\boldsymbol{z}^{(l)} | 0, \lambda^2 diag\left(\sigma^2\right)\right)$$

Multiplicative Noise

$$\boldsymbol{h}^{(0)} = \boldsymbol{x}_i^{tr}$$
$$\boldsymbol{h}^{(l)} = ReLU\left(f^{(l)}\left(\boldsymbol{h}^{(l-1)}\right) \circ \boldsymbol{z}^{(l)}\right)$$

$$\boldsymbol{z}^{(l)} = Softplus(\boldsymbol{a}^{(l)})$$
$$\boldsymbol{a}^{(l)} \sim N\left(\boldsymbol{a}^{(l)} \middle| \boldsymbol{\mu}^{(l)}, \mathrm{I}\right)$$

Experiments - Datasets

Omniglot

Handwritten character classification
→ 20 instances of ~1600 characters from 50 alphabets

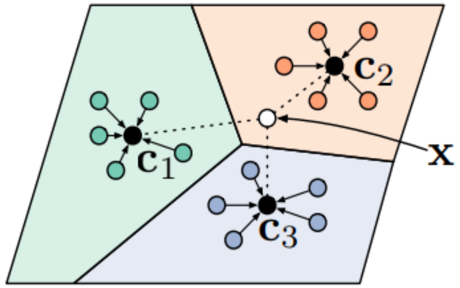
miniImageNET

Small version of ImageNET
→ 100 classes with 600 samples

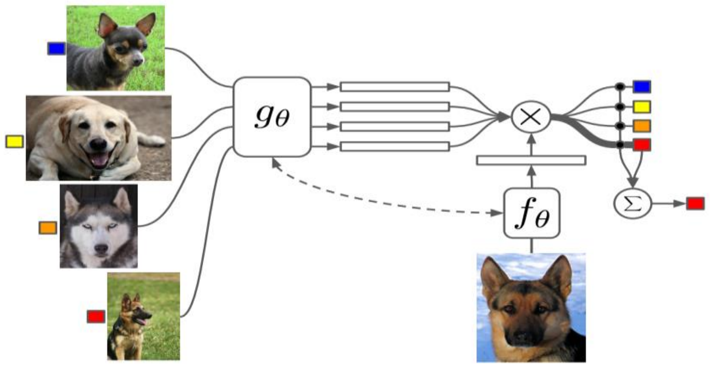
Experiments
Meta-Learning Frameworks

- MAML
- Meta-SGD
- Prototypical Networks
- Matching Networks

- Reptile
- Amortized Bayesian ML
- Probabilitsitic MAML
- MT-NET
- CAVIA



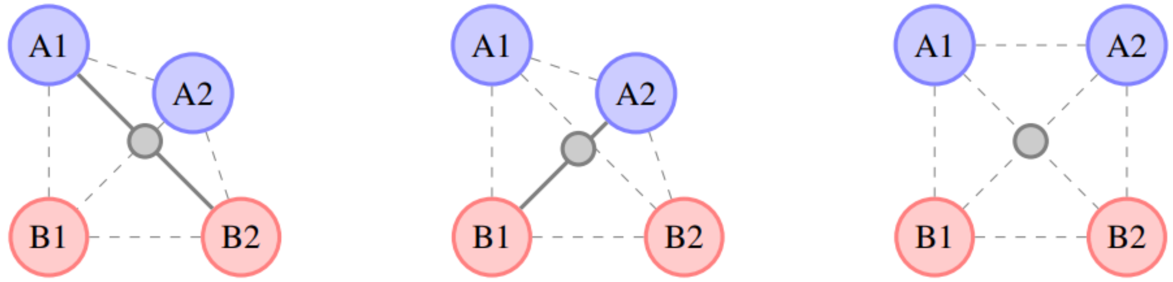
Prototypical Networks



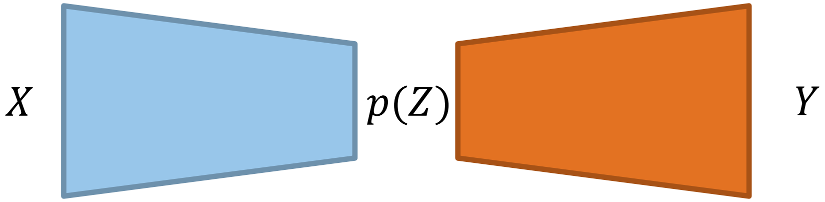
Matching Networks

Experiments
Perturbation Based Methods:

- **Input & Manifold Mixup**
- **Variational Information Bottleneck**
- **Information Dropout**
- **Adversarial learning/Training**

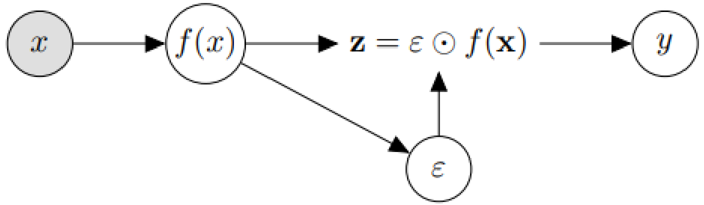


Input & Manifold Mixup



Encode maximal information about Target Y in latent stochastic encoding Z measured by mutual information $I(Z,Y) < I_c$ where I_c is information constraint

Variational Information Bottleneck



Information Dropout

Few-shot classification performance

Models	Omniglot 20-way 1-shot	Omniglot 20-way 5-shot	miniImageNet 5-way 1-shot	miniImageNet 5-way 5-shot
Meta-Learning LSTM (Ravi & Larochelle, 2017)	-	-	43.44±0.77	60.60±0.71
Matching Networks (Vinyals et al., 2016)	93.8	98.7	43.56±0.84	55.31±0.73
Prototypical Networks (Snell et al., 2017)	95.4	98.7	46.14±0.77	65.77±0.70
Prototypical Networks (Snell et al., 2017) (Higher way)	96.0	98.9	49.42±0.78	**68.20±0.66**
MAML (our reproduction)	95.23±0.17	98.38±0.07	49.58±0.65	64.55±0.52
Meta-SGD (our reproduction)	96.16±0.14	98.54±0.07	48.30±0.64	65.55±0.56
Reptile (Nichol et al., 2018)	89.43±0.14	97.12±0.32	49.97±0.32	65.99±0.58
Amortized Bayesian ML (Ravi & Beatson, 2019)	-	-	45.00±0.60	-
Probabilistic MAML (Finn et al., 2018)	-	-	50.13±1.86	-
MT-Net (Lee & Choi, 2018)	96.2±0.4	-	51.70±1.84	-
CAVIA (512) (Zintgraf et al., 2019)	-	-	51.82±0.65	65.85±0.55
MAML + Meta-dropout	96.63±0.13	98.73±0.06	**51.93±0.67**	**67.42±0.52**
Meta-SGD + Meta-dropout	**97.02±0.13**	**99.05±0.05**	50.87±0.63	65.55±0.57

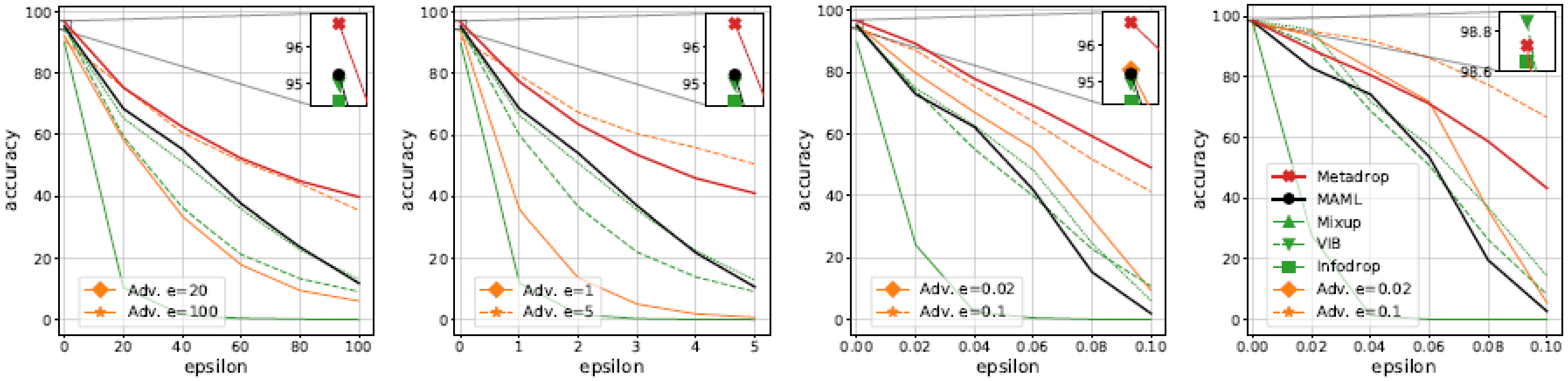
Few-shot classification performance

Models (MAML +)	Noise Type	Hyper-parameter	Omniglot 20-way 1-shot	Omniglot 20-way 5-shot	miniImageNet 5-way 1-shot	miniImageNet 5-way 5-shot
No perturbation		None	$95.23_{\pm 0.17}$	$98.38_{\pm 0.07}$	$49.58_{\pm 0.65}$	$64.55_{\pm 0.52}$
Input & Manifold Mixup	Pairwise	$\gamma = 0.2$	$89.78_{\pm 0.25}$	$97.86_{\pm 0.08}$	$48.62_{\pm 0.66}$	$63.86_{\pm 0.53}$
(Zhang et al., 2017)		$\gamma = 1$	$87.00_{\pm 0.28}$	$97.27_{\pm 0.10}$	$48.24_{\pm 0.62}$	$62.32_{\pm 0.54}$
(Verma et al., 2019)		$\gamma = 2$	$87.26_{\pm 0.28}$	$97.14_{\pm 0.17}$	$48.42_{\pm 0.64}$	$62.56_{\pm 0.55}$
Variational	Add.	$\beta = 10^{-5}$	$92.09_{\pm 0.22}$	$98.85_{\pm 0.07}$	$48.12_{\pm 0.65}$	$64.78_{\pm 0.54}$
Information Bottleneck		$\beta = 10^{-4}$	$93.01_{\pm 0.20}$	$98.80_{\pm 0.07}$	$46.75_{\pm 0.63}$	$64.07_{\pm 0.54}$
(Alemi et al., 2017)		$\beta = 10^{-3}$	$94.98_{\pm 0.16}$	$98.75_{\pm 0.07}$	$47.59_{\pm 0.60}$	$63.30_{\pm 0.53}$
Information Dropout	Mult.	$\beta = 10^{-5}$	$94.49_{\pm 0.17}$	$98.50_{\pm 0.07}$	$50.36_{\pm 0.68}$	$65.91_{\pm 0.55}$
(ReLU ver.)		$\beta = 10^{-4}$	$94.36_{\pm 0.17}$	$98.53_{\pm 0.07}$	$49.14_{\pm 0.63}$	$64.96_{\pm 0.54}$
(Achille & Soatto, 2018)		$\beta = 10^{-3}$	$94.28_{\pm 0.17}$	$98.65_{\pm 0.07}$	$43.78_{\pm 0.61}$	$63.36_{\pm 0.56}$
Meta-dropout	Add.	0.1	$96.55_{\pm 0.14}$	**$99.04_{\pm 0.05}$**	$50.25_{\pm 0.66}$	$66.78_{\pm 0.53}$
(See Appendix B for Add.)	**Mult.**	None	**$96.63_{\pm 0.13}$**	$98.73_{\pm 0.06}$	**$51.93_{\pm 0.67}$**	**$67.42_{\pm 0.52}$**

Ablation study on the noise type

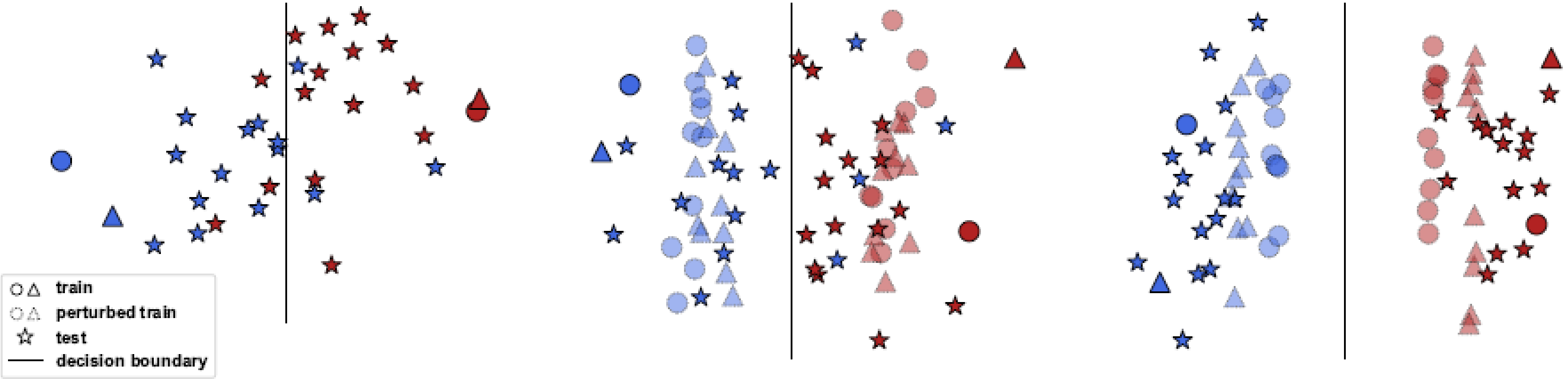
Models (MAML+)	Rand. samp.	Learn. mult.	Input dep.	Omniglot 20-way 1-shot	Omniglot 20-way 5-shot	miniImageNet 5-way 1-shot	miniImageNet 5-way 5-shot
None	X	X	X	$95.23_{\pm 0.17}$	$98.38_{\pm 0.07}$	$49.58_{\pm 0.65}$	$64.55_{\pm 0.52}$
Fixed Gaussian (✓)	O	X	X	$95.44_{\pm 0.17}$	**$98.99_{\pm 0.06}$**	$49.39_{\pm 0.63}$	$66.84_{\pm 0.54}$
Weight Gaussian	O	X	X	$94.32_{\pm 0.18}$	$98.35_{\pm 0.07}$	$49.37_{\pm 0.64}$	$64.78_{\pm 0.54}$
Independent Gaussian	O	O	X	$94.36_{\pm 0.18}$	$98.26_{\pm 0.08}$	$50.31_{\pm 0.64}$	$66.97_{\pm 0.54}$
MAML + More param	X	O	O	$95.83_{\pm 0.15}$	$97.85_{\pm 0.09}$	$50.63_{\pm 0.64}$	$65.20_{\pm 0.51}$
Determ. Meta-drop. (✓)	X	O	O	$95.99_{\pm 0.14}$	$97.78_{\pm 0.09}$	$50.75_{\pm 0.63}$	$65.62_{\pm 0.53}$
Meta-drop. w/ learned var.	O	O	O	$95.98_{\pm 0.15}$	$98.87_{\pm 0.06}$	$50.93_{\pm 0.68}$	$66.15_{\pm 0.56}$
Meta-dropout	O	O	O	**$96.63_{\pm 0.13}$**	$98.73_{\pm 0.06}$	**$51.93_{\pm 0.67}$**	**$67.42_{\pm 0.52}$**

Adversarial Robustness



(a) Omniglot 1-shot (ℓ_1) (b) Omniglot 1-shot (ℓ_2) (c) Omniglot 1-shot (ℓ_∞) (d) Omniglot 5-shot (ℓ_∞)

Qualitative study on generalization capability



(a) MAML (miniImageNet) (b) Meta-dropout (miniImageNet) (c) Meta-dropout (Omniglot)

Conclusion

Main Claim:

"Using Meta-Dropout to perturb the latent features of training examples in a Meta-Learning Framework improves generalization capabilities"

+

Improves:

- *Decision boundary*
- *Adversarial robustness*
- *Few-Shot learning performance*

- *Hypothesis supported by experiments across large variate of baseline models*
- *Code available*

−

- *Evaluated on only two datasets*
- *More Shots / More Ways*
- *Relatively small performance increase*
- *Discrepancy in the results*
- *Generalization across datasets domains not discussed*
- *No Computational Cost Reported*
- *Comparison noise \phi for every layer or shared for all*

References

[1] Hae Beom Lee, Taewook Nam, Eunho Yang, and Sung Ju Hwang. Meta dropout: Learning to perturb latent features for generalization. In ICLR, 2020

[2] Alessandro Achille and Stefano Soatto. Information Dropout: Learning Optimal Representations Through Noisy Computation. In PAMI, 2018.

[3] Alex Alemi, Ian Fischer, Josh Dillon, and Kevin Murphy. Deep variational information bottleneck. In ICLR, 2017.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In ICML, 2017.

[5] D. P. Kingma, T. Salimans, and M. Welling. Variational Dropout and the Local Reparameterization Trick. In NIPS, 2015

[6] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few shot learning. arXiv preprint arXiv:1707.09835, 2017.

[7] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In NIPS, 2017.

[8] Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In NIPS, 2016.