DART: Diversify-Aggregate-Repeat Training Improves Generalization of Neural Networks

Samyak Jain* Sravanti Addepalli* Pawan Sahu Priyam Dey R. Venkatesh Babu

Vision and AI Lab, Department of Computational and Data Sciences
Indian Institute of Science, Bangalore

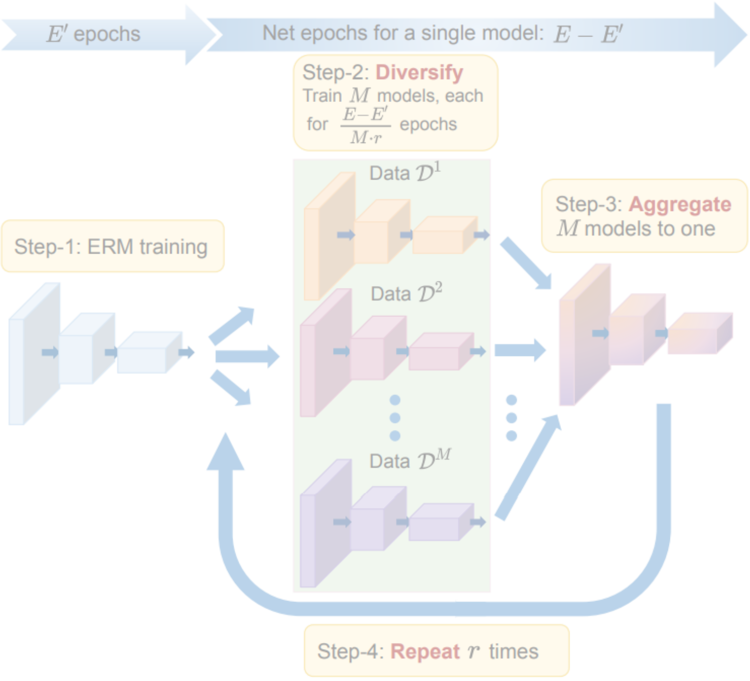
Paper Tag:
WED-PM-352

INDIAN INSTITUTE OF SCIENCE
भारतीय विज्ञान संस्थान

Vision & AI Lab

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

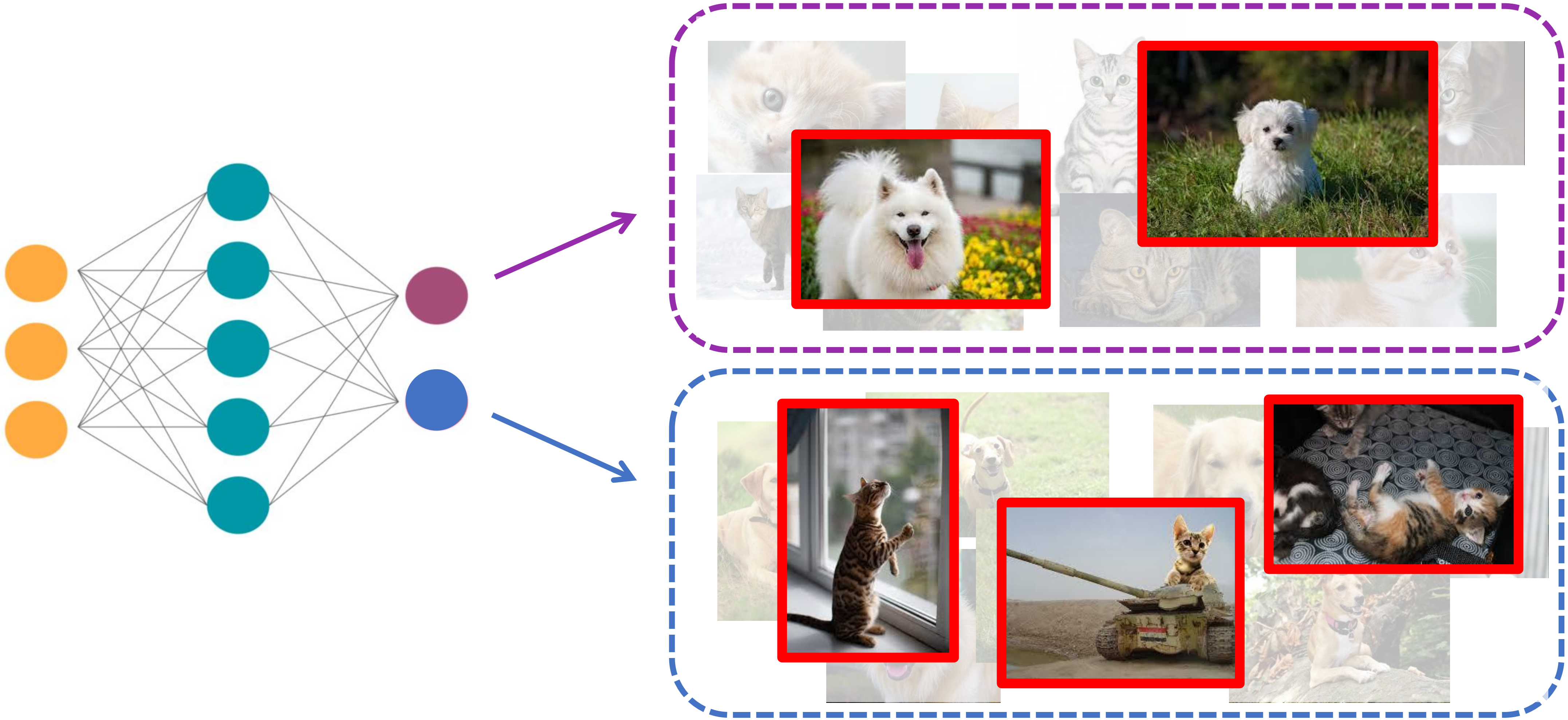
Overview



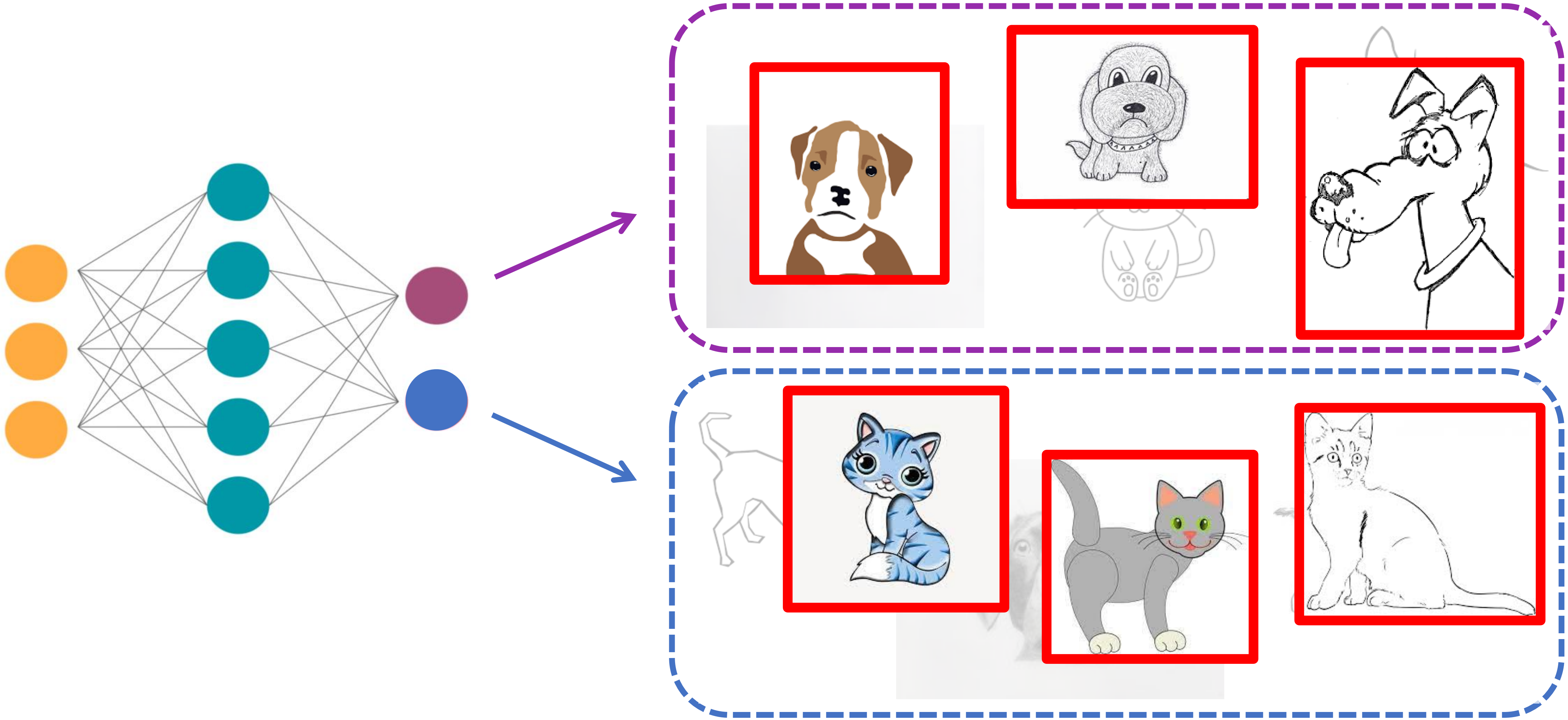
Method	CIFAR-10	CIFAR-100
ERM+EMA (Pad+Crop+HFlip)	96.41	81.67
ERM+EMA (AutoAugment)	97.50	84.20
ERM+EMA (Cutout)	97.43	82.33
ERM+EMA (Cutmix)	97.11	84.05
Learning Subspaces	97.46	83.91
ERM+EMA (Mixed Training-MT)	97.69 $\pm$ 0.19	85.57 $\pm$ 0.13
DART (Ours)	**97.96** $\pm$ 0.06	**86.46** $\pm$ 0.12

Algorithm	VLCS	PACS	OfficeHome	TerraInc	DomainNet	Avg
ERM	77.5 $\pm$ 0.4	85.5 $\pm$ 0.2	66.5 $\pm$ 0.3	46.1 $\pm$ 1.8	40.9 $\pm$ 0.1	63.3
+ DART (Ours)	78.5 $\pm$ 0.7	87.3 $\pm$ 0.5	70.1 $\pm$ 0.2	48.7 $\pm$ 0.8	45.8 $\pm$ 0.0	66.1
SWAD	79.1 $\pm$ 0.1	88.1 $\pm$ 0.1	70.6 $\pm$ 0.2	50.0 $\pm$ 0.3	46.5 $\pm$ 0.1	66.9
+ DART (Ours)	**80.3** $\pm$ 0.2	**88.9** $\pm$ 0.1	**71.9** $\pm$ 0.1	**51.3** $\pm$ 0.2	**47.1** $\pm$ 0.0	**67.9**

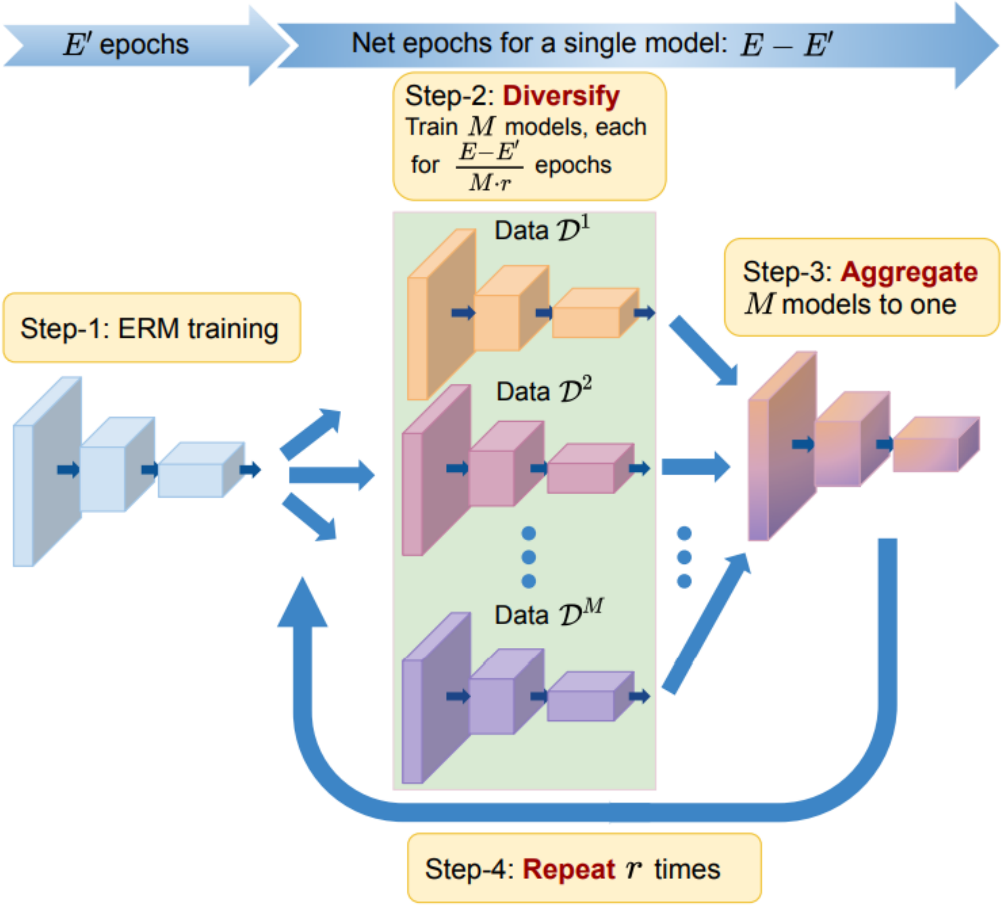
Generalization of Deep Neural Networks



Generalization of Deep Neural Networks



DART- Diversify Aggregate Repeat Training



- **Traversing to the basin of optimal solutions:** Mixed Training (MT) of a single model is done for E' epochs
- **Diversify - Exploring the basin:** Individual experts are trained using different augmentations/ domains to improve diversity across models
- **Aggregate - Combining diverse experts:** Weights of all experts are averaged to obtain a single model
- **Repeat:** Each expert is reinitialized with the interpolated model, and this process is repeated until convergence

Empirical Results

- Improved performance in the In-Domain setting

Method	CIFAR-10	CIFAR-100
ERM+EMA (Pad+Crop+HFlip)	96.41	81.67
ERM+EMA (AutoAugment)	97.50	84.20
ERM+EMA (Cutout)	97.43	82.33
ERM+EMA (Cutmix)	97.11	84.05
Learning Subspaces	97.46	83.91
ERM+EMA (Mixed Training-MT)	97.69 $\pm$ 0.19	85.57 $\pm$ 0.13
DART (Ours)	**97.96** $\pm$ 0.06	**86.46** $\pm$ 0.12

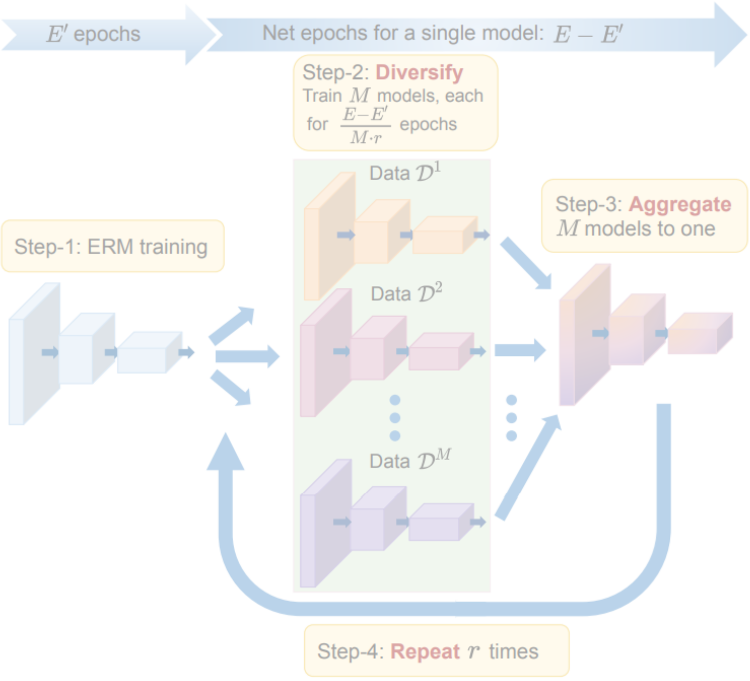
ERM+EMA	ERM+SWA	DART	SAM+EMA	DART+SAM+EMA
85.57 $\pm$ 0.13	85.44 $\pm$ 0.09	86.46 $\pm$ 0.12	87.05 $\pm$ 0.15	**87.26** $\pm$ 0.02

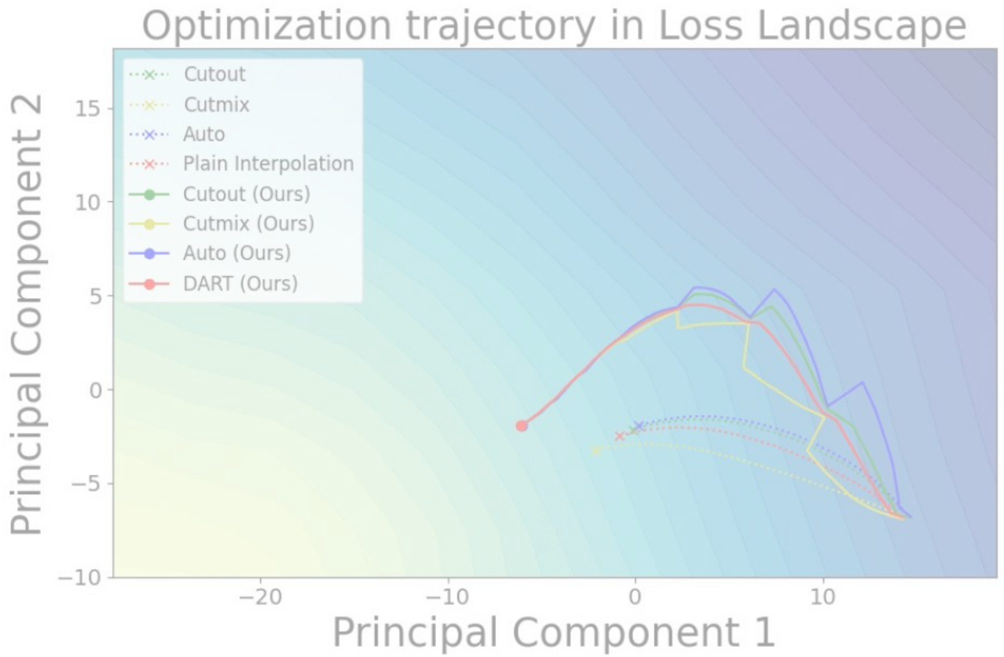
	Stanford-CARS		CUB-200		Imagenet-1K	
	ERM + EMA	DART	ERM + EMA	DART	ERM + EMA	DART
SA	88.11	**90.42**	78.55	**79.75**	78.55	**78.96**
MA	90.88	**91.95**	81.72	**82.83**	79.06	**79.20**

- SOTA performance in the Domain Generalization (DG) setting

Algorithm	VLCS	PACS	OfficeHome	TerraInc	DomainNet	Avg
ERM	77.5 $\pm$ 0.4	85.5 $\pm$ 0.2	66.5 $\pm$ 0.3	46.1 $\pm$ 1.8	40.9 $\pm$ 0.1	63.3
+ DART (Ours)	78.5 $\pm$ 0.7	87.3 $\pm$ 0.5	70.1 $\pm$ 0.2	48.7 $\pm$ 0.8	45.8 $\pm$ 0.0	66.1
SWAD	79.1 $\pm$ 0.1	88.1 $\pm$ 0.1	70.6 $\pm$ 0.2	50.0 $\pm$ 0.3	46.5 $\pm$ 0.1	66.9
+ DART (Ours)	**80.3** $\pm$ 0.2	**88.9** $\pm$ 0.1	**71.9** $\pm$ 0.1	**51.3** $\pm$ 0.2	**47.1** $\pm$ 0.0	**67.9**

Proposed Approach: DART





Diverse Training for Improved Generalization

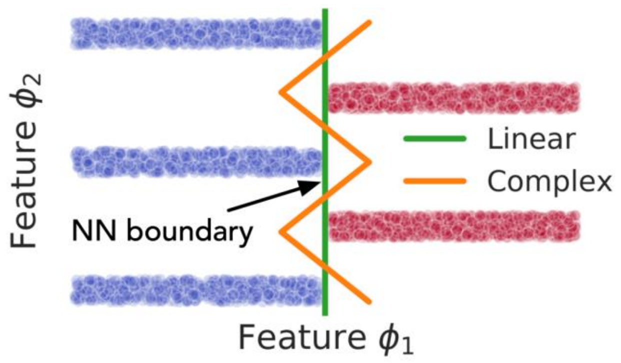


Image source: Shah et al., The Pitfalls of Simplicity Bias in Neural Networks

- Simplicity Bias – tendency to rely on simpler features that are often spurious correlations to the labels, when compared to the harder robust features
- Simplicity Bias is one of the reasons for the sensitivity of Deep Networks to distribution shifts

Data Augmentations

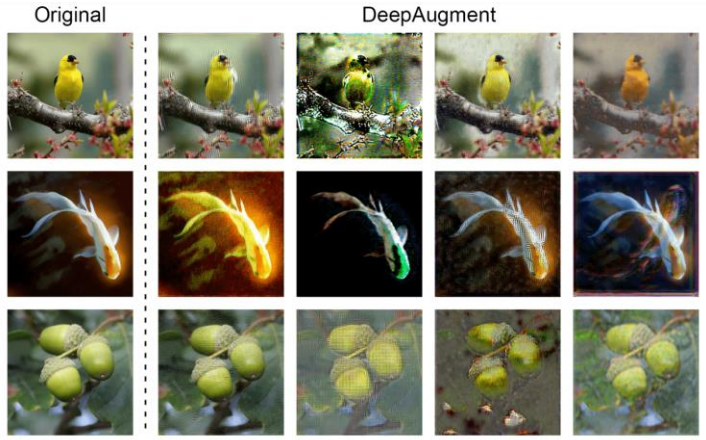


Image source: Hendrycks et al., The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization

Training on data from diverse domains

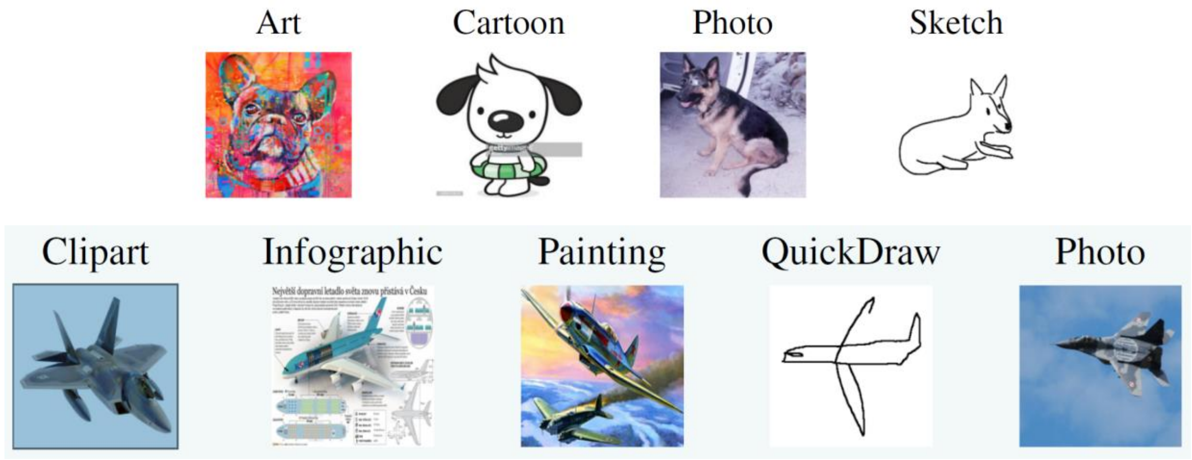


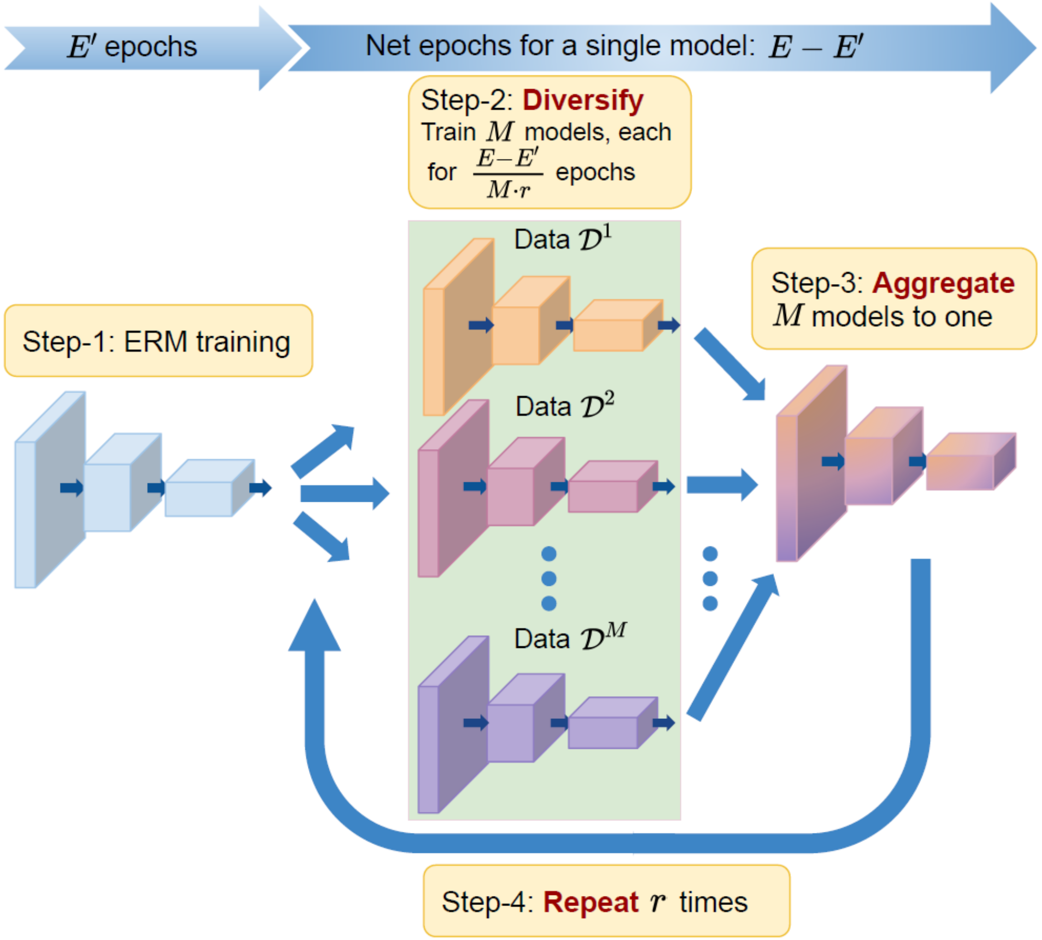
Image source: Gulrajani et al., In Search of Lost Domain Generalization

Ensemble of diverse experts

- Training a model using augmented data **specializes** the model to the same distribution
- **Mixed Training:** Generalization of the model improves when diverse augmentations are used in a single training minibatch, but performance is limited by the capacity of the model
- Using an ensemble of diverse models trained on different augmentations results in improved **generalization**, but with **higher inference time**

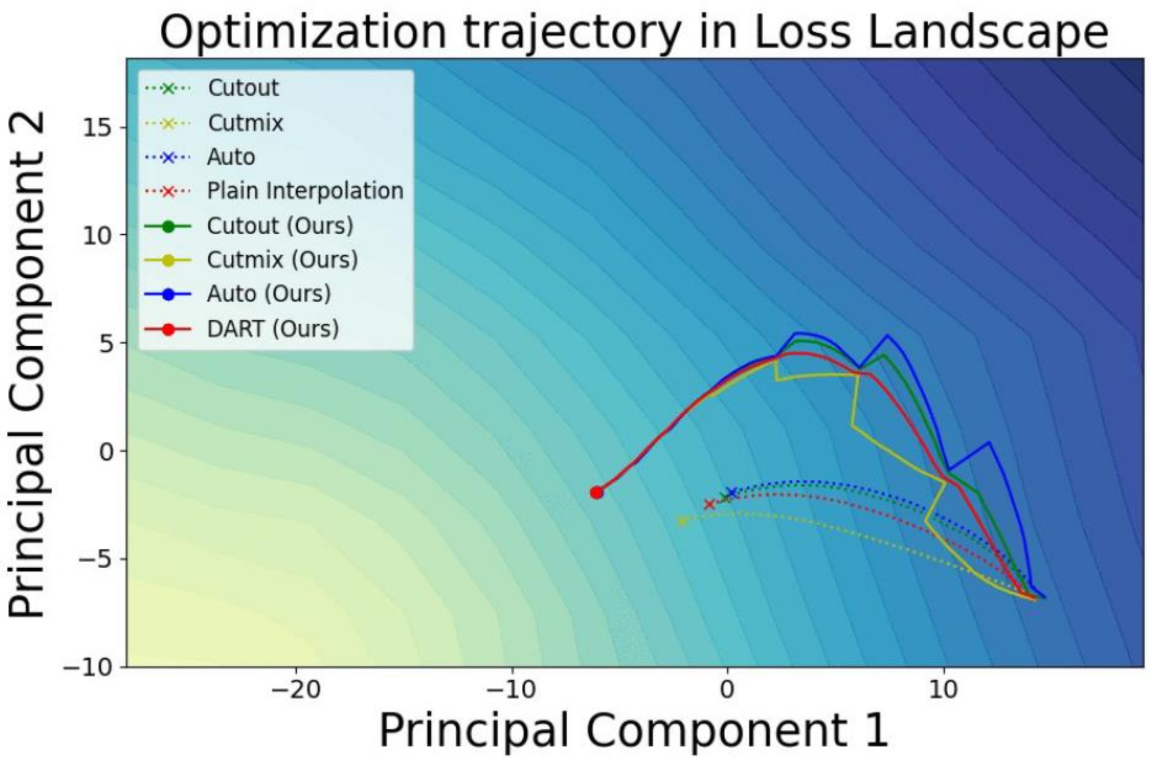
	Test Augmentation			
Train Augmentation	No Aug.	Cutout	Cutmix	AutoAugment
Pad+Crop+HFlip (PC)	78.51	67.04	56.52	58.33
Cutout (CO)	77.99	74.58	56.12	58.47
Cutmix (CM)	80.54	74.05	**77.35**	61.23
AutoAugment (AA)	79.18	71.26	60.97	73.91
Mixed-Training (MT)	81.43	77.31	73.20	**74.73**
Ensemble (CM+CO+AA)	**83.61**	**79.19**	73.19	73.90

DART- Diversify Aggregate Repeat Training



- **Traversing to the basin of optimal solutions:** Mixed Training (MT) of a single model is done for E' epochs
- **Diversify - Exploring the basin:** Individual experts are trained using different augmentations/ domains to improve diversity across models
- **Aggregate - Combining diverse experts:** Weights of all experts are averaged to obtain a single model
- **Repeat:** Each expert is reinitialized with the interpolated model, and this process is repeated until convergence

Optimization trajectory



- The models explore more in the initial phase of training, and lesser thereafter due to the cosine learning rate schedule
- Initial exploration increases the diversity of models **improving robustness to spurious features**
- Repeated aggregation ensures that the models remain close to each other
- Smaller steps towards the end help in **retaining the flatter optima** obtained after Aggregation

Theoretical and Empirical results

Proposition 1 *The convergence time for learning any feature patch v_i $\forall i \in [1, K]$ in at least one channel $c \in C$ of the weight averaged model f_θ using the augmentations defined in Eq.5, is given by* $O\left(\frac{K}{\sigma_0^{q-2}}\right)$, *if* $\frac{\sigma^q}{\sqrt{d}} \ll \frac{1}{K}$, $m = K$.

Method	CIFAR-10	CIFAR-100
ERM+EMA (Pad+Crop+HFlip)	96.41	81.67
ERM+EMA (AutoAugment)	97.50	84.20
ERM+EMA (Cutout)	97.43	82.33
ERM+EMA (Cutmix)	97.11	84.05
Learning Subspaces	97.46	83.91
ERM+EMA (Mixed Training-MT)	97.69 $\pm$ 0.19	85.57 $\pm$ 0.13
DART (Ours)	**97.96** $\pm$ 0.06	**86.46** $\pm$ 0.12

Method	Pad+Crop+HFlip	AutoAug.	Cutout	Cutmix	Mixed-Train.
ERM	81.48	83.93	82.01	83.02	85.54
ERM + EMA	81.67	84.20	82.33	84.05	85.57
DART (Ours)	**82.31**	**85.02**	**84.15**	**84.72**	**86.13**

Impact of Weight Averaging

- Weights learned by a single layer neural network are a combination of the feature and noisy patches present in images

$$w = \sum_{k=1}^{K_{cut}} v_k + \sum_{k > K_{cut}} y^{(k)} \epsilon^{(k)}$$

- Weight averaging can improve learning and achieve robustness to spurious features as:
  - Use of diverse augmentations helps in **learning less frequent (hard) feature** patches
  - Weight averaging helps in **reducing the variance** of noise

$$w = \frac{1}{m} \sum_{j=1}^{m} \sum_{k=1}^{K_{cut_j}} v_{k_j} + \frac{1}{m} \sum_{j=1}^{m} \sum_{k > K_{cut_j}} y_j^{(k)} \epsilon_j^{(k)}$$

Theoretical setup from: Ruoqi Shen, Sebastian Bubeck and Suriya Gunasekar, Data Augmentations as Feature Manipulation (ICML-2022)

Theoretical Results

Proposition 1 *The convergence time for learning any feature patch v_i $\forall i \in [1, K]$ in at least one channel $c \in C$ of the weight averaged model f_θ using the augmentations defined in Eq.5, is given by $O\left(\frac{K}{\sigma_0^{q-2}}\right)$, if $\frac{\sigma^q}{\sqrt{d}} \ll \frac{1}{K}$, $m = K$.*

On weight averaging, the **convergence time for learning less frequent feature patches decreases**. Thus, the learning of hard features becomes easier.

Proposition 2 *If the noise patches learned by each f_θ^k are i.i.d. Gaussian random variables $\sim \mathcal{N}(0, \frac{\sigma^2}{d} I_d)$ then with high probability, convergence time of learning a noisy patch $\epsilon^{(j)}$ in at least one channels $c \in [1, C]$ of the weight averaged model f_θ is given by $O\left(\frac{nm}{\sigma_0^{q-2}\sigma^q}\right)$, if $d \gg n^2$.*

On weight averaging, the **convergence time for learning noisy/ spurious features increases by O(m),** where m is the number of expert models.

Theoretical Results

Proposition 3 *If the noise learned by each* f_θ^k *are i.i.d. Gaussian random variables* $\sim \mathcal{N}\left(0, \frac{\sigma^2}{d} I_d\right)$, *and model weight averaging is performed at epoch* T, *the convergence time of learning a noisy patch* $\epsilon^{(j)}$ *in at least one channels* $c \in [1, C]$ *of the weight averaged model* f_θ *is given by* $T + O\left(\frac{nm^{(q-2)} d^{(q-2)/2}}{\sigma^{(2q-2)}}\right)$ *if* $d \gg n^2$.

Intermediate interpolations **increase the convergence time for learning spurious features** when compared to weight averaging only during inference

Empirical Results: In Domain Generalization

Method	CIFAR-10	CIFAR-100
ERM+EMA (Pad+Crop+HFlip)	96.41	81.67
ERM+EMA (AutoAugment)	97.50	84.20
ERM+EMA (Cutout)	97.43	82.33
ERM+EMA (Cutmix)	97.11	84.05
Learning Subspaces	97.46	83.91
ERM+EMA (Mixed Training-MT)	97.69 $\pm$ 0.19	85.57 $\pm$ 0.13
DART (Ours)	**97.96** $\pm$ 0.06	**86.46** $\pm$ 0.12

Model Architecture: WideResNet-28-10

Using same augmentations across models

Method	Pad+Crop+HFlip	AutoAug.	Cutout	Cutmix	Mixed-Train.
ERM	81.48	83.93	82.01	83.02	85.54
ERM + EMA	81.67	84.20	82.33	84.05	85.57
DART (Ours)	**82.31**	**85.02**	**84.15**	**84.72**	**86.13**

Empirical Results: In Domain Generalization

- DART can also be combined with SAM to obtain better results (CIFAR-100, WRN-34-10)

ERM+EMA	ERM+SWA	DART	SAM+EMA	DART+SAM+EMA
85.57 $\pm$ 0.13	85.44 $\pm$ 0.09	86.46 $\pm$ 0.12	87.05 $\pm$ 0.15	**87.26** $\pm$ 0.02

- DART obtains improved results on ImageNet and fine-grained datasets

	Stanford-CARS		**CUB-200**		**Imagenet-1K**	
	ERM + EMA	**DART**	**ERM + EMA**	**DART**	**ERM + EMA**	**DART**
SA	88.11	**90.42**	78.55	**79.75**	78.55	**78.96**
MA	90.88	**91.95**	81.72	**82.83**	79.06	**79.20**

SA: Single Augmentation, MA: Mixed Augmentations

Empirical Results: Domain Generalization

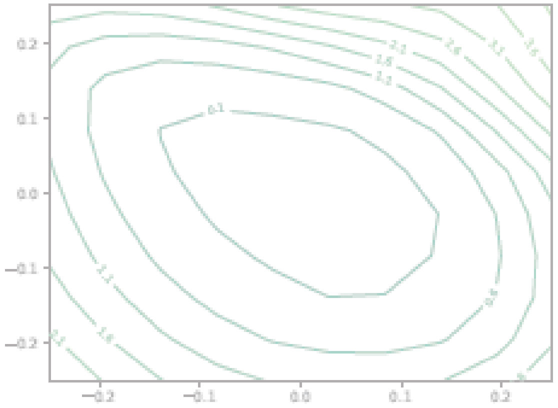
Algorithm	VLCS	PACS	OfficeHome	TerraIncognita	DomainNet	Avg
ERM	77.5 ± 0.4	85.5 ± 0.2	66.5 ± 0.3	46.1 ± 1.8	40.9 ± 0.1	63.3
IRM	78.5 ± 0.5	83.5 ± 0.8	64.3 ± 2.2	47.6 ± 0.8	33.9 ± 2.8	61.6
GroupDRO	76.7 ± 0.6	84.4 ± 0.8	66.0 ± 0.7	43.2 ± 1.1	33.3 ± 0.2	60.7
Mixup	77.4 ± 0.6	84.6 ± 0.6	68.1 ± 0.3	47.9 ± 0.8	39.2 ± 0.1	63.4
MLDG	77.2 ± 0.4	84.9 ± 1.0	66.8 ± 0.6	47.7 ± 0.9	41.2 ± 0.1	63.6
CORAL	78.8 ± 0.6	86.2 ± 0.3	68.7 ± 0.3	47.6 ± 1.0	41.5 ± 0.1	64.5
MMD	77.5 ± 0.9	84.6 ± 0.5	66.3 ± 0.1	42.2 ± 1.6	23.4 ± 9.5	58.8
DANN	78.6 ± 0.4	83.6 ± 0.4	65.9 ± 0.6	46.7 ± 0.5	38.3 ± 0.1	62.6
CDANN	77.5 ± 0.1	82.6 ± 0.9	65.8 ± 1.3	45.8 ± 1.6	38.3 ± 0.3	62.0
MTL	77.2 ± 0.4	84.6 ± 0.5	66.4 ± 0.5	45.6 ± 1.2	40.6 ± 0.1	62.9
SagNet	77.8 ± 0.5	86.3 ± 0.2	68.1 ± 0.1	48.6 ± 1.0	40.3 ± 0.1	64.2
ARM	77.6 ± 0.3	85.1 ± 0.4	64.8 ± 0.3	45.5 ± 0.3	35.5 ± 0.2	61.7
VREx	78.3 ± 0.2	84.9 ± 0.6	66.4 ± 0.6	46.4 ± 0.6	33.6 ± 2.9	61.9
RSC	77.1 ± 0.5	85.2 ± 0.9	65.5 ± 0.9	46.6 ± 1.0	38.9 ± 0.5	62.7
SWAD	79.1 ± 0.1	88.1 ± 0.1	70.6 ± 0.2	50.0 ± 0.3	46.5 ± 0.1	66.9
DART w/o SWAD	78.5 ± 0.7	87.3 ± 0.5	70.1 ± 0.2	48.7 ± 0.8	45.8	66.1
DART w/ SWAD	**80.3** ± 0.2	**88.9** ± 0.1	**71.9** ± 0.1	**51.3** ± 0.2	**47.2**	**67.9**

Combining DART with other DG methods

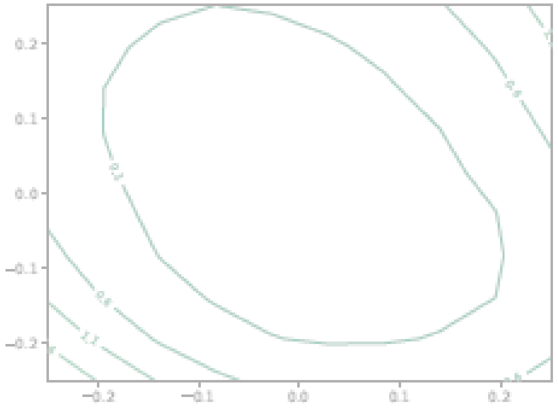
- DART can be integrated with several base domain generalization approaches – both with and without SWAD, to obtain substantial gains across the respective baselines.

Algorithm	**Vanilla**	**DART** (w/o SWAD)	**SWAD**	**DART** (+ SWAD)
ERM	66.5	70.31	70.60	**72.28**
ARM	64.8	69.24	69.75	**71.31**
SAM	67.4	70.39	70.26	**71.55**
Cutmix	67.3	70.07	71.08	**71.49**
Mixup	68.1	71.14	71.15	**72.38**
DANN	65.9	70.32	69.46	**70.85**
CDANN	65.8	70.75	69.70	**71.69**
SagNet	68.1	70.19	70.84	**71.96**
MIRO	70.5	72.54	72.40	**72.71**
MIRO (CLIP)	83.3	86.14	84.80	**87.37**

Conclusion



(a) ERM+EMA (PCH)



(b) DART (Ours, PCH)

Conclusion

- We propose the Mixed Training (MT) benchmark which uses a combination of diverse augmentations during training in a single minibatch, and obtains improved results in an in-domain generalization setting
- We propose DART – Diversify Aggregate Repeat Training, an algorithm to improve generalization of models by firstly training diverse models, and further aggerating their weights throughout training.
- Theoretical results:
  - Lower convergence time for learning hard features (learning of diverse features)
  - Higher convergence time for learning noisy features by incorporating intermediate weight averaging (robustness to spurious features)
- Empirical results:
  - Improved performance in the In-Domain setting
  - SOTA performance in the Domain Generalization (DG) setting
  - Adaptable with different Domain Generalization methods

Thank You!

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

Paper Tag: WED-PM-352

https://github.com/val-iisc/dart

INDIAN INSTITUTE OF SCIENCE

भारतीय विज्ञान संस्थान

Vision & AI Lab

Acknowledgements: This work was supported by the research grant CRG/2021/005925 from SERB, DST, Govt. of India. Sravanti Addepalli is supported by Google PhD Fellowship.