Understanding Hessian Alignment for Domain Generalization

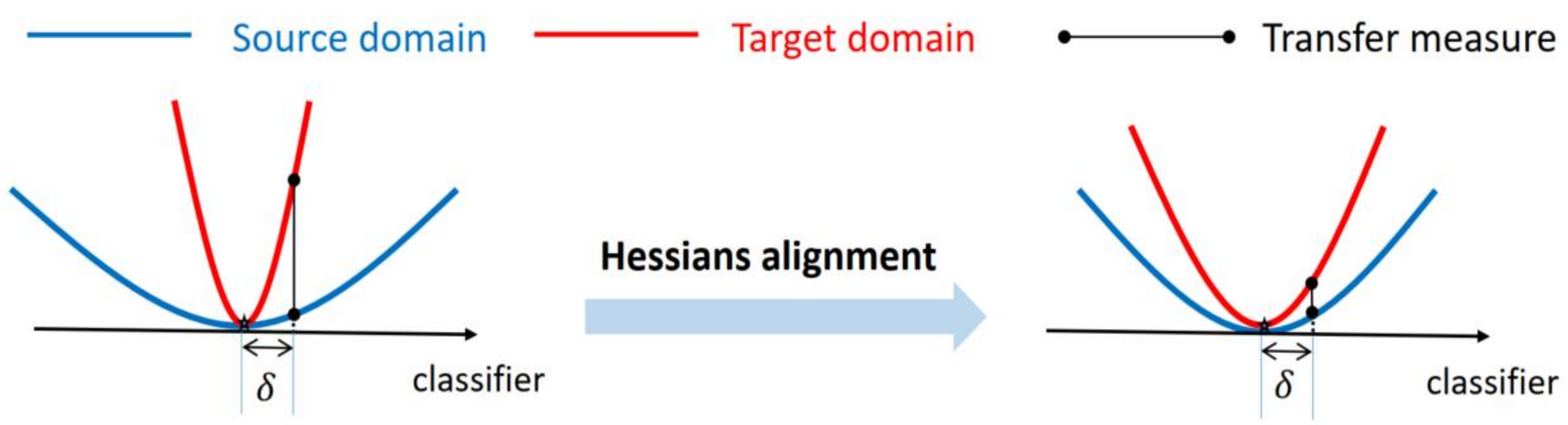
Sobhan Hemati* Guojun Zhang* Amir Estiri Xi Chen *Equal Contribution

Huawei Noah's Ark Lab

ICCV23 PARIS

HUAWEI

Main results



- Domain Generalization aims to learn invariant mechanisms from multiple source domains, so as to generalize to unseen target domains.
- ***What is the role of Hessian Alignment in DG?***
- **Summary 1**: The distance between the classifier's head Hessians is an upper bound of the transfer measure that quantifies the domain shift
- **Summary 2:** Hessians and gradient alignment simultaneously encourage invariant representation learning at different levels.
- **Summary 3:** To align Hessians efficiently, we propose two simple yet effective Hessian alignment methods, based on different estimations

Preliminaries

Transferable (Zhang et al. 2021): Every near-optimal source classifier is also near-optimal on the target

$$\operatorname{argmin}(L_{\mathcal{D}}, \delta_{\mathcal{D}}) := \{h_\theta \in \mathcal{H}, L_{\mathcal{D}}(\theta) \le \inf_{h_\theta \in \mathcal{H}} L_{\mathcal{D}}(\theta) + \delta_{\mathcal{D}}\}$$

Definition: $\mathcal{S}$ is $(\delta_{\mathcal{S}}, \delta_{\mathcal{T}})$-transferable to $\mathcal{T}$ if

$$\operatorname{argmin}(L_{\mathcal{S}}, \delta_{\mathcal{S}}) \subseteq \operatorname{argmin}(L_{\mathcal{T}}, \delta_{\mathcal{T}})$$

Use Transfer Measure to quantify transferability

$$T_\Gamma(\mathcal{S}||\mathcal{T}) = \sup_{h_\theta \in \Gamma} [L_{\mathcal{T}}(\theta) - \inf_{h_\theta \in \mathcal{H}} L_{\mathcal{T}}(\theta) - (L_{\mathcal{S}}(\theta) - \inf_{h_\theta \in \mathcal{H}} L_{\mathcal{S}}(\theta))]$$

Transferable ≡ Small transfer measure, if $\Gamma = \operatorname{argmin}(L_{\mathcal{S}}, \delta_{\mathcal{S}})$.

Theoretical results

Theorem. Under mild assumptions, the spectral norm of Hessian Differences between source and target domains is an upper bound for Transfer Measure:

$$T_\Gamma(\mathcal{S}||\mathcal{T}) \le \frac{1}{2}\delta^2 \|H_{\mathcal{S}} - H_{\mathcal{T}}\|_2 + o(\delta^2)$$

Proposition (Feature matching) Let $\widehat{y_p}$ and y_p be the network prediction and true target with the p-th class, z_i be the i-th feature before the classifier. Matching the gradients and Hessians w.r.t. the classifier head across domains aligns:

$$\frac{\partial \ell}{\partial b_p} = (\widehat{y_p} - y_p), \ (Error)$$

$$\frac{\partial \ell}{\partial w_{p,q}} = (\widehat{y_p} - y_p) z_q, \ (Error \times Feature)$$

$$\frac{\partial^2 \ell}{\partial b_u \partial b_v} = \widehat{y_u}(\delta_{u,v} - \widehat{y_v}), \ (Logit)$$

$$\frac{\partial^2 \ell}{\partial w_{p,q} \partial b_u} = z_q \widehat{y_p}(\delta_{p,u} - \widehat{y_u}), \ (Logit \times Feature)$$

$$\frac{\partial^2 \ell}{\partial w_{p,q} \partial w_{u,v}} = \widehat{y_p} z_q z_v(\delta_{p,u} - \widehat{y_u}), \ (Logit \times Covariance)$$

Hessian and gradient alignment can be seen a generalization of other invariant representation learning methods

Alignment attribute	Loss	Feature	Covariance	Error	Error × Feature	Logit	Logit × Feature	Logit × Covariance
V-Rex	✓[1]	✗	✗	✗	✗	✗	✗	✗
CORAL	✗	✓	✓	✗	✗	✗	✗	✗
IGA	✗	✗	✗	✓	✓	✗	✗	✗
Fish	✗	✗	✗	✓	✓	✗	✗	✗
Fishr	✓	✗	✗	✗	✓[2]	✗	✗	✗
Hessian Alignment	✗	✗	✗	✓	✓	✓	✓	✓

Algorithms

How to align Hessians and gradient across training domains efficiently?

Hessian-Gradient Product

$$L_{HGP} = \frac{1}{n}\sum_{e=1}^{n} L_{S_e} + \alpha \left|\left| H_{S_e}\nabla_\theta L_{S_e} - \overline{H_S \nabla_\theta L_S} \right|\right|_2^2 + \beta \left|\left|\nabla_\theta L_{S_e} - \overline{\nabla_\theta L_S}\right|\right|_2^2$$

Hutchinson's diagonal estimator

$$L_{Hutchinson} = \frac{1}{n}\sum_{e=1}^{n} L_{S_e} + \alpha \left|\left| D_{S_e} - \overline{D_S} \right|\right|_2^2 + \beta \left|\left|\nabla_\theta L_{S_e} - \overline{\nabla_\theta L_S}\right|\right|_2^2$$

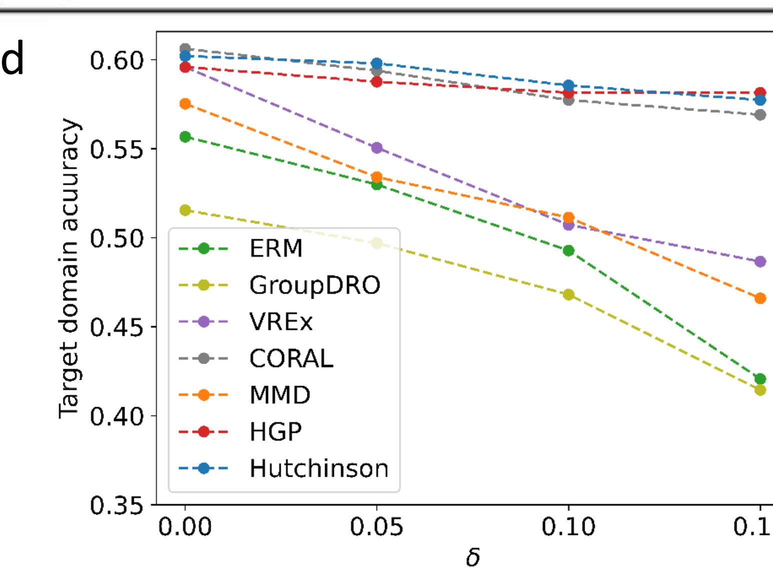
where the bar notation means average over all training environments

Algorithm	VLCS	PACS	OfficeHome	DomainNet	Avg
ERM (Vapnik, 1999)	77.2	83.0	65.7	40.6	66.6
IRM (Arjovsky et al., 2019)	76.3	81.5	64.3	33.5	63.9
GroupDRO (Sagawa et al., 2020)	77.9	83.5	65.2	33.0	64.9
Mixup (Wang et al., 2020)	77.7	83.2	67.0	38.5	66.6
MLDG (Li et al., 2018a)	77.2	82.9	66.1	41.0	66.8
CORAL (Sun and Saenko, 2016)	78.7	82.6	**68.5**	41.1	67.7
MMD (Li et al., 2018b)	77.3	83.2	60.2	23.4	61.0
DANN (Ganin et al., 2016)	76.9	81.0	64.9	38.2	65.2
CDANN (Zhou et al., 2021)	77.5	78.8	64.3	38.0	64.6
MTL (Blanchard et al., 2021)	76.6	83.7	65.7	40.6	66.7
SagNet (Nam et al., 2020)	77.5	82.3	67.6	40.2	66.9
ARM (Zhang et al., 2020)	76.6	81.7	64.4	35.2	64.5
VREx (Krueger et al., 2021)	76.7	81.3	64.9	33.4	64.1
RSC (Huang et al., 2020)	77.5	82.6	65.8	38.9	66.2
Fishr (Rame et al., 2022)	78.2	**85.4**	67.8	-	-
HGP	76.7	82.2	67.5	41.1	66.9
Hutchinson	**79.3**	84.8	**68.5**	**41.4**	**68.5**

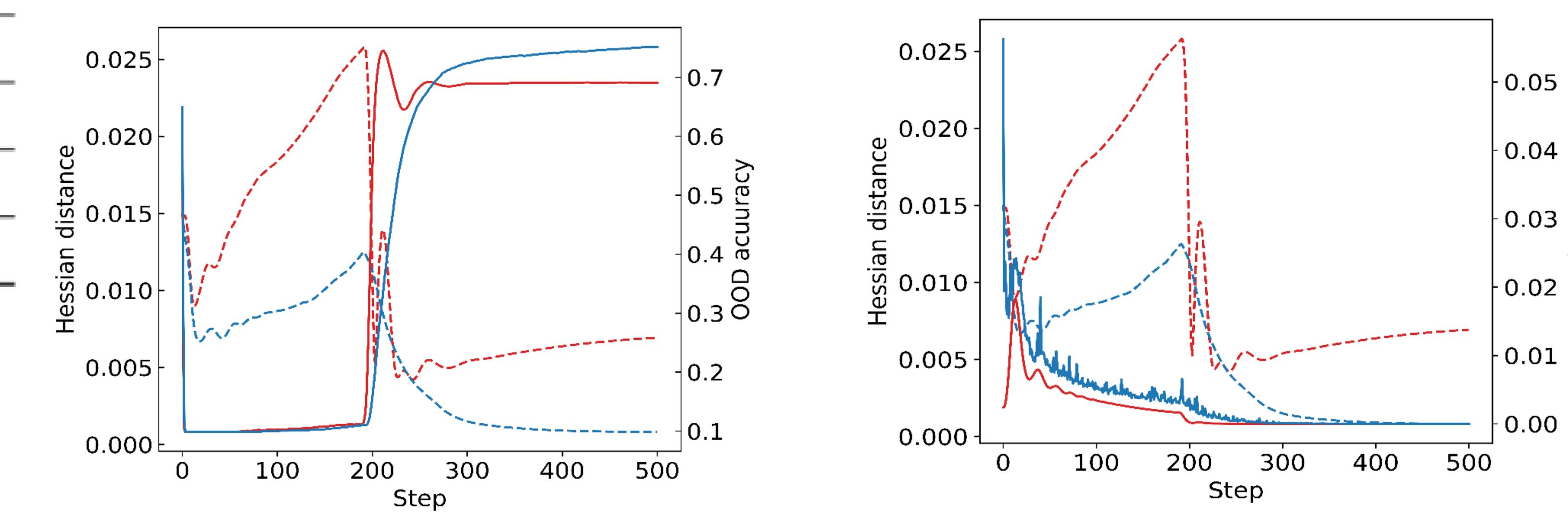
DomainBed

Method	Train acc.	Test acc.
ERM	86.4 ± 0.2	14.0 ± 0.7
IRM	71.0 ± 0.5	65.6 ± 1.8
V-REx	71.7 ± 1.5	67.2 ± 1.5
Fishr	71.0 ± 0.9	69.5 ± 1.0
HGP	71.0 ± 1.5	69.4 ± 1.3
Hutchinson	61.7 ± 1.9	**74.0 ± 1.2**

Comparison of HGP and Hutchinson with other baselines for CMNIST



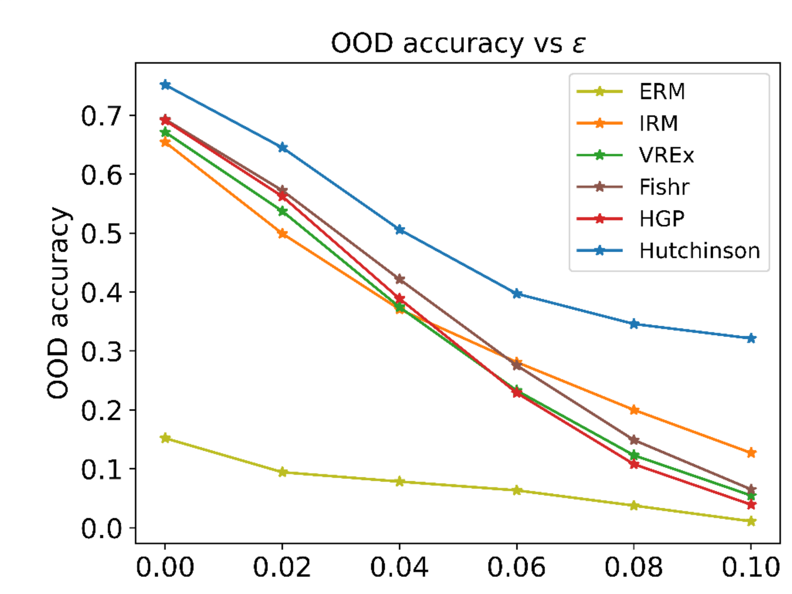
Transferability experiment on OfficeHome



Correlation between Hessian distances and OOD accuracies/losses for HGP and Hutchinson regularization during the training for Colored MNIST

Method	Test acc.
Hessian & gradient	81.4
Gradient only	77.0
Hessian only	79.4

Ablation study of the Hutchinson method on PACS when the test domain is Sketch



Adversarial robustness under perturbation