Leveraging sparse and shared feature activations for disentangled representation learning

Anonymous Authors[1]

Abstract

Research on recovering the latent factors of variation of high dimensional data has so far focused on simple synthetic settings. Mostly building on unsupervised and weakly-supervised objectives, prior work missed out on the positive implications for representation learning on real world data. In this work, we propose to leverage knowledge extracted from a diversified set of supervised tasks to learn a common disentangled representation. Assuming that each supervised task only depends on an unknown subset of the factors of variation, we disentangle the feature space of a supervised multi-task model, with features activating sparsely across different tasks and information being shared as appropriate. Importantly, we never directly observe the factors of variations, but establish that access to multiple tasks is sufficient for identifiability under sufficiency and minimality assumptions. We validate our approach on six real world distribution shift benchmarks, and different data modalities (images, text), demonstrating how disentangled representations can be transferred to real settings.

1. Introduction

A fundamental question in deep learning is how to learn meaningful and reusable representation from high dimensional data observations (Bengio et al., 2013; Salakhutdinov, 2014; Schölkopf et al., 2021; Schmidhuber, 1992). A core area of research pursuing is centered on disentangled representation learning (DRL) (Locatello et al., 2019; Bengio et al., 2013; Higgins et al., 2017) where the aim is to learn a representation which recovers the factors of variations (FoVs) underlying the data distribution. Disentangled representations are expected to contain all the information present in the data in a compact and interpretable structure (Kulkarni et al., 2015; Chen et al., 2016) and to enable robust downstream predictions, which was partially validated in synthetic settings (Dittadi et al., 2021; Locatello et al., 2020b). Unfortunately, these benefits did not materialize in real world representations learning problems, largely limited by a lack of scalability of existing approaches.

In this work we focus on leveraging knowledge from different task objectives to learn better representations, exploring the link with disentanglement and out-of-distribution (OOD) generalization on real data distributions. Representations learned from a large diversity of tasks are indeed expected to be richer and generalize better to new, possibly OOD, tasks. However, this is not always the case, as different tasks can compete with each other (Marx et al., 2005; Wang et al., 2019; Standley et al., 2020) leading to noisy features, increase of the sensitivity to spurious correlations (Hu et al., 2022; Geirhos et al., 2020; Beery et al., 2018)and weaker models. Instead, assuming that each task only depends on an unknown subset of FoVs, we build on two following inductive biases, showing that disentanglement naturally emerges from them:

- *Sparse sufficiency*: Features should activate sparsely with respect to tasks. The representation is *sparsely sufficient* in the sense that any given task can be solved using few features.
- *Minimality*: Features are maximally shared across tasks whenever possible. The representation is *minimal* in the sense that features are encouraged to be reused, i.e., duplicated or split features are avoided.

We demonstrate how these intuitive properties are desirable in order to obtain features that (i) are disentangled w.r.t. to the factors of variations underlying the task data distribution (which we also theoretically argue in Proposition B.1), (ii) generalize better in settings where test data undergo distribution shifts with respect to the training distributions, and (iii) suffer less from problems related to negative transfer phenomena. To learn such representations in practice, we implement a meta learning approach, enforcing feature sufficiency and sharing with a *sparsity* regularizer and a entropy based *feature sharing* regularizer, respectively, incorporated in the base learner. Experimentally, we show that our model learns meaningful disentangled representations that enable strong generalization on real world data sets.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the SCIS workshop.

2. Method

Given a distribution of tasks $t \sim \mathcal{T}$ and data $(\mathbf{x_t}, y_t) \sim \mathcal{P}_t$ $\forall t$, we aim to learn a disentangled representation $g(\mathbf{x}) = \hat{\mathbf{z}} \in \hat{\mathcal{Z}} \subseteq \mathbb{R}^M$, which generalizes well to unseen tasks. We learn g by imposing the sparse sufficiency and minimality.

Learning sparse and shared features Our architecture (see Figure 4) is composed of a backbone module g_θ that is shared across all tasks and a separate linear classification head f_{ϕ_t}, which is specific to each task t. The backbone is responsible to compute and learn a general feature representation for all classification tasks. The linear head solves a specific classification problem for the task-specific data $(\mathbf{x_t}, y_t) \sim \mathcal{P}_t$ in the feature space $\hat{\mathcal{Z}}$ while enforcing the feature sufficiency and minimality principles. Adopting the typical meta-learning setting (Hospedales et al., 2020), the backbone module g_θ can be viewed as the *meta learner* while the task-specific classification heads f_{ϕ_t} can be viewed as the *base learners*. In the meta-learning setting we assume to have access to samples for a new task give by a *support set* U, with elements $(\mathbf{x}^U, y^U) \in U$. These samples are used to fit the linear head f_{ϕ^*} leading to the optimal feature weights for the given task. For a *query* $\mathbf{x}^Q \in Q$, predictions are obtained by computing$\hat{y} = f_{\phi^*}(g_\theta(\mathbf{x}^Q))$.

Enforcing feature minimality and sufficiency. To solve a task in the feature space $\hat{\mathcal{Z}}$ of the backbone module we impose the following regularizer $Reg(\phi)$ on the classification heads f_ϕ with parameter $\phi \in \mathbb{R}^{T \times M \times C}$, where T is the number of tasks, M the number of features, and C the number of classes. The regularizer is responsible for enforcing the feature minimality and sufficiency properties. It is composed of the weighted sum of a sparsity penalty Reg_{L1} and an entropy-based feature sharing penalty $Reg_{sharing}$:

$$Reg(\phi) = \alpha Reg_{L_1}(\phi) + \beta Reg_{sharing}(\phi), \tag{1}$$

with scalar weights α and β. The penalty terms are:

$$Reg_{L_1}(\phi) = \frac{1}{TC} \sum_{t,c,m} |\phi_{t,m,c}| \tag{2}$$

$$Reg_{sharing}(\phi) = H(\tilde{\phi}_m) = -\sum_m \tilde{\phi}_m log(\tilde{\phi}_m) \tag{3}$$

where $\tilde{\phi}_m = \frac{1}{TC} \frac{\sum_{t,c} |\phi_{t,c,m}|)}{\sum_{t,c,m} |\phi_{t,c,m}|}$ are the normalized classifier parameters. Sufficiency is enforced by a sparsity regularizer given by the L_1-norm, which constrains classification head to use only a sparse subset of the features. Minimality is enforced by the feature sharing term: minimizing the entropy of the distribution of feature importances (i.e. normalized $|\phi_t|$) averaged across a mini batch of T tasks, leads to a more peaked distribution of activations across tasks. This forces features to cluster across tasks and therefore be reused by different tasks, when useful.

Training method We train the model in meta-learning fashion by minimizing the test error over the expectation of the task distribution $t \sim \mathcal{T}$. This can be formalized as a *bi-level optimization problem*. The optimal backbone model g_{θ^*} is given by the *outer optimization problem*:

$$\min_\theta \mathbb{E}_t[\mathcal{L}_{outer}(f_{\phi^*}(g_\theta(\mathbf{x}_t^Q), y_t^Q))], \tag{4}$$

where f_{ϕ^*} are the optimal classifiers obtained from solving the *inner optimization problem*, and $(\mathbf{x}_t^Q, y_t^Q) \in Q_t$ are the test (or query) datum from the query set Q_t for task t. Let U_t be the support set with samples $(\mathbf{x}_t^U, y_t^U) \in U$ for task t, where typically the support set is distinct from the query set, i.e., $U \cap Q = \emptyset$. The optimal classifiers f_{ϕ^*} are given by the *inner optimization problem*:

$$\min_\phi \frac{1}{T} \sum_t \mathcal{L}_{inner}(\hat{y}_t^U, y_t^U) + Reg(\phi), \tag{5}$$

where $\hat{y}_t^U = f_\phi(g_\theta(\mathbf{x}_t^U)$. For both the inner loss $\mathcal{L}_{inner}$ and outer loss $\mathcal{L}_{outer}$ we use the cross entropy loss. In practice we solve the bi-level optimization problem (4) and (5) as described in the algorithm in section D.1 of the Appendix.

3. Experiments

We start by highlighting here the experimental setup of this paper along with its motivation. Experimental details are fully described in Appendix E.

Synthetic experiments. We first evaluate our method on benchmarks from the disentanglement literature (Matthey et al., 2017; Burgess & Kim, 2018; Reed et al., 2015; LeCun et al., 2004) where we have access to the FoVs and we can assess quantitatively how well we can learn disentangled representations. We show how minimality is correlated with disentanglement measures (Section 3.1) and how our representations, learned from a limited set of tasks, can generalize their composition. The purpose of these experiments is to validate our theoretical statement, showing that if the assumptions of Proposition B.1 hold, our method quantitatively recovers the FoVs.

Domain shifts and transferability. On real data sets, we can neither quantitatively measure disentanglement nor are we guaranteed identifiability (as assumptions may be violated). Ultimately, the goal of disentangled representations is to learn features easily and robustly transferrable to downstream tasks. Therefore, we first evaluate the usefulness of our representations with respect to downstream tasks subject to distribution shifts, where isolating spurious features was found to improve generalization in synthetic settings (Dittadi et al., 2021; Locatello et al., 2020b) We evaluate our method on domain generalization and domain shift tasks on six different benchmarks (Section 3.2). Lastly, we test the OOD adaptability of our method in a few-shot transfer learning setting in Appendix F.5.

3.1. Synthetic experiments

We start by demonstrating that our approach is able to recover the FoVs underlying a synthetic data distribution like (Matthey et al., 2017). For these experiments, we assume

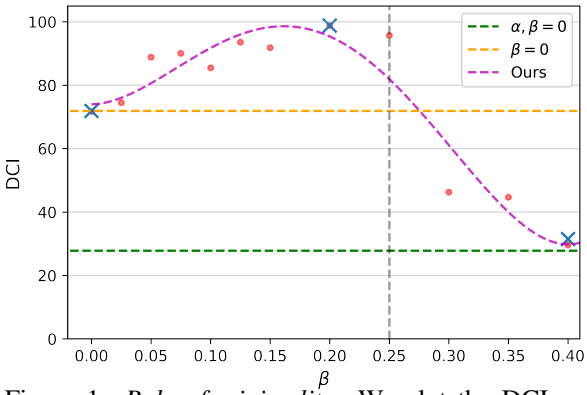


Figure 1: *Role of minimality*: We plot the DCI metric of a set of models (*red dots*) trained on fixed tasks from `DSprites`: Training without regularizers leads to no disentanglement (*green*). Enforcing sparsity alone (*yellow*, akin to (Lachapelle et al., 2022a)) achieves good disentanglement ($DCI = 71.9$), but some features may be split or duplicated. Enforcing both minimality and sparse sufficiency (*magenta*) attains the best DCI (98.8). When β is too high (> 0.25) activated features collapses into few clusters with respect to tasks. For exact values and qualitative evidence see Table 7 and Figure 5 in Appendix.

to have partial information on a subset of FoVs Z, and we aim to learn a representation $\hat{\mathbf{z}}$ that aligns with them while ignoring any spurious factors. We sample random tasks from a distribution $\mathcal{T}$ (see Appendix E.3 for details) and focus on binary tasks, with $Y = \{0, 1\}$. For the `DSprites` dataset an example of valid task is *"There is a big object on the left of the image"*. In this case, the partially observed factors (quantized to only two values) are the *x position* and *size*. In Table 1, we show how the sparse sufficiency and minimality properties enable disentanglement in the learned representations. We train two identical models on a random distribution of sparse tasks defined on FoVs, showing that, for different datasets (Matthey et al., 2017; Burgess & Kim, 2018; LeCun et al., 2004; Reed et al., 2015), the same model without regularizers achieves a similar in-distribution (ID) accuracy, but a much lower disentanglement.

We then randomly draw and fix 2 groups of tasks with supports S_1, S_2 (18 in total), which all have support on two FoVs, $|S_1| = |S_2| = 2$. The groups share one factor of variation and differ in the other one, i.e. $S_1 \cap S_2 = \{i\}$ for some $\{i\} \in Z$. We start from an overestimate of the dimension of $\tilde{\mathbf{z}}$ of 6, trying to recover $\mathbf{z}$ of size 3. We train our network to solve these tasks, enforcing sufficiency and minimality on the representation with different regularization degrees. In Figure 1, we show how the alignment of the learned features with the ground truth factors of variations depend on the choice of α, β, going from no disentanglement ($DCI = 27.8$) to good alignment ($DCI = 98.8$) as we enforce sufficiency and minimality.

Disentanglement and minimality are correlated. For 15 models trained on `Dsprites` increasing β from 0 to 0.2 linearly, we observe a correlation coefficient with the DCI metric of 94.7, showing that the feature sharing property strongly encourages disentanglement. This confirms again that sufficiency alone (i.e. enforcing sparsity) is not enough to attain good disentanglement.

Table 1: *Enforcing disentanglement*: DCI (Eastwood & Williams, 2018) score and ID accuracy on test samples for a model trained enforcing sufficency and minimality (bottom row), and a model without (top row). While attaining comparable accuracy, the regularized model always shows higher disentanglement.

	Dsprites	3Dshapes	SmallNorb	Cars
No reg (DCI,Acc)	(16.6,94.4)	(44.4,96.2)	(16.5,96.1)	(60.5,99.8)
α, β (DCI,Acc)	(**69.9**,95.8)	(**87.7**, 95.8)	(**55.8**,95.6)	(**92.3**,99.8)

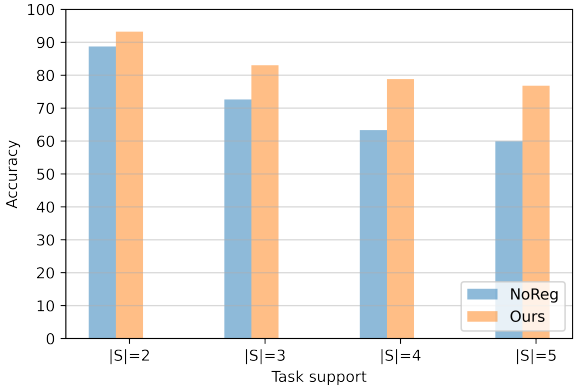


Figure 2: *Task compositional generalization*: Mean accuracy over 100 random test tasks reported for group of tasks of growing support (*second, third, fourth column*) for a model trained without inductive biases (*blue*, attaining $DCI = 29.4$) and enforcing them (*orange*, $DCI = 59.4$). The latter show better compositional generalization resulting from the properties enforced on the representation. Exact values are reported in Table 8 in Appendix.

Task compositional generalization. Finally, we evaluate the generalization capabilities of our method by testing our model on a set of unseen tasks obtained by combining tasks seen during training. To do this, we first train two models on the `AbstractDSprites` dataset using a random distribution of tasks, where we limit the support of each task to be within 2 (i.e. $|S| = 2$). The models differ in activating/deactivating the regularizers on the linear heads. In Figure 2, we test on 100 tasks drawn from a distribution with increasing support on the factors of variation ($|S| = 3, |S| = 4, |S| = 5$), which correspond to composition of tasks in the training distribution.

3.2. Domain Shift

In this section we evaluate our method on benchmarks coming from the domain generalization field (Gulrajani

Table 2: *Results on CivilComments*: we report the accuracy on test averaged across all demographic groups (*left*), and the worst group accuracy (*right*). We show that our method performs similarly in terms of average accuracy and outperforms in terms of worst group accuracy, without using any knowledge on the group composition in the training data.

	avg acc	worst group acc
ERM	**92.2**	56.5
DRO	90.2	69
Ours	91.2 ± 0.2	**75.45**± 0.1

& Lopez-Paz, 2021; Wenzel et al., 2022; Qiu et al., 2022) and subpopulation shifts (Sagawa et al., 2019; Koh et al., 2021), to show that a feature space learned with our inductive biases performs under real world data distribution shift.

Subpopulation shifts. Subpopulation shifts occur when the distribution of minority groups changes across domains. Our claim is that a feature space that satisfies sparse sufficiency and minimality is more robust to spurious correlations which may affect minority groups, and should transfer better to new distributions. To validate this, we test on two benchmarks `Waterbirds` (Sagawa et al., 2019), and `CivilComments` (Koh et al., 2021). In Table 4, last row, we report the results on the test set of `Waterbirds` for the different groups in the dataset, comparing with ERM. For `CivilComments` we report the average and worst accuracy in Table 9, where we compare with ERM and groupDRO (Sagawa et al., 2019). While performing almost on par w.r.t. ERM, our method is more robust to spurious correlation in the dataset, showing the higher worst group accuracy. Importantly, we outperform GroupDRO, which uses information on the subdomain statistics, while we do not assume any prior knowledge about them. Results per group are reported in the Appendix (Table 10).

DomainBed & Camelyon17 We evaluate the domain generalization performance on the `PACS`, `VLCS` and `OfficeHome` datasets from the DomainBed (Gulrajani & Lopez-Paz, 2021) test suite (see Appendix E.1 for more details). On these datasets, we train on $N-1$ and leave one out for testing. Regularization parameters α and β are tuned according to validation sets of `PACS`, and used accordingly on the other dataset. Results are reported in Table 4, showing how enforcing sparse sufficiency and minimality leads consistently to better OOD performance. Comparisons with 13 additional baselines is in Appendix F.4. On Camelyon17 the model is trained according to the original splits in the dataset. In Table 3 we report the accuracy of our model on in-distribution and OOD splits, compared with different baselines (Sun et al., 2017; Arjovsky et al., 2019). Our method shows the best performance on the OOD test domains. The intuition is that, due to minimality, we retain features shared across the three training domains, giving less weight to the domain-specific ones which are spuriously correlated with the hospital environment. This can be further enforced at test time, as shown in Appendix F.10, trading off in distribution performance for OOD accuracy.

Table 3: Quantitative evaluation on Camelyon17: we report accuracy both on ID and OOD splits. Our approach achieves significantly higher validation and test OOD accuracy.

	Validation(ID)	Validation (OOD)	Test (OOD)
ERM	93.2	84	70.3
CORAL	**95.4**	86.2	59.5
IRM	91.6	86.2	64.2
Ours	93.2 ±0.3	**89.9**±0.6	**74.1**±0.2

Table 4: Results for domain generalization on DomainBed. Our approach achieves consistently higher average OOD generalization, outperforming ERM in all cases except one.

Dataset/Algorithm	**OOD accuracy (by domain)**				
PACS	S	A	P	C	Average
ERM	77.9 ± 0.4	**88.1** ± 0.1	97.8 ± 0.0	79.1 ± 0.9	85.7
Ours	**83.1** ± 0.1	86.7± 0.8	**97.8** ± 0.1	**83.5** ± 0.1	**87.5**
VLCS	C	L	V	S	Average
ERM	97.6± 1.0	63.3 ± 0.9	76.4 ± 1.5	72.2 ± 0.5	77.4
Ours	**98.1**± 0.2	**63.4**± 0.5	**78.2** ± 0.7	**73.9**± 0.8	**78.4**
OfficeHome	C	A	P	R	Average
ERM	53.4± 0.6	62.7 ± 1.1	76.5 ± 0.4	77.3 ± 0.	67.5
Ours	**56.3**± 0.1	**66.7** ± 0.7	**79.2**± 0.5	**81.3** ± 0.4	**70.9**
Waterbirds	LL	LW	WL	WW	Average
ERM	98.6 ± 0.3	52.05 ± 3	68.5 ± 3	93 ± 0.3	81.3
Ours	**99.5** ± 0.1	**73.0** ± 2.5	**85.0** ± 2	**95.5** ± 0.4	**90.5**

Additional results In Appendix F we report a large collection of additional results, including results on few-shot transfer learning (F.5), a comparison with 14 baseline methods on the domain shift benchmarks (F.4), a qualitative and quantitative analysis on the minimality and sparse sufficiency properties in the real setting (F.2), an additional comparison on meta learning benchmarks with 6 baselines(F.9), an ablation study on the effect of clustering features at test time (F.10), and a demonstration on the possibility to obtain a task similarity measure as a consequence of our approach (F.8).

4. Conclusions and limitations

In this paper, we demonstrated how to learn disentangled representations from a distribution of tasks by enforcing feature sparsity and sharing. We validated identifiability of our approach experimentally in a controlled settings, and showed that these representations are beneficial for generalizing OOD in real-world scenarios, isolating spurious and domain-specific factors that should not be used under distribution shift. The main limitation of our work is the global assumption on the strength of the sparsity and feature sharing regularizers α and β across all tasks. In real settings these properties of the representations might need to change for different tasks while excessive regularization might hurt performance (e.g. $\beta > 0.25$ in Figure 1). Future work may exploit some level of knowledge on the task distribution (e.g. some measure of distance on tasks) in order to tune α, β adaptively during training.

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I. J., Hardt, M., and Kim, B. Sanity checks for saliency maps. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 9525–9536, 2018.

Ahuja, K., Shanmugam, K., Varshney, K. R., and Dhurandhar, A. Invariant risk minimization games. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 145–155. PMLR, 2020.

Albuquerque, I., Monteiro, J., Darvishi, M., Falk, T. H., and Mitliagkas, I. Generalizing to unseen domains via distribution matching. *ArXiv preprint*, abs/1911.00804, 2019.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *ArXiv preprint*, abs/1907.02893, 2019.

Bai, J., Men, R., Yang, H., Ren, X., Dang, K., Zhang, Y., Zhou, X., Wang, P., Tan, S., Yang, A., et al. Ofasys: A multi-modal multi-task learning system for building generalist models. *ArXiv preprint*, abs/2212.04408, 2022.

Bandi, P. Camelyon17 dataset.

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

Bertinetto, L., Henriques, J. F., Torr, P. H. S., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Blanchard, G., Deshmukh, A. A., Dogan, Ü., Lee, G., and Scott, C. Domain generalization by marginal transfer learning. *The Journal of Machine Learning Research*, 22 (1):46–100, 2021.

Blondel, M., Berthet, Q., Cuturi, M., Frostig, R., Hoyer, S., Llinares-López, F., Pedregosa, F., and Vert, J.-P. Efficient and modular implicit differentiation. *ArXiv preprint*, abs/2105.15183, 2021.

Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. Nuanced metrics for measuring unintended bias with real data for text classification. *ArXiv preprint*, abs/1903.04561, 2019.

Brehmer, J., De Haan, P., Lippe, P., and Cohen, T. Weakly supervised causal representation learning. *ArXiv preprint*, abs/2203.16437, 2022.

Burgess, C. and Kim, H. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

Caruana, R. Multitask learning. *Machine learning*, 28(1): 41–75, 1997.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 2172–2180, 2016.

Dhillon, G. S., Chaudhari, P., Ravichandran, A., and Soatto, S. A baseline for few-shot image classification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Dittadi, A., Träuble, F., Locatello, F., Wuthrich, M., Agrawal, V., Winther, O., Bauer, S., and Schölkopf, B. On the transfer of disentangled representations in realistic settings. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Dixon, L., Li, J., Sorensen, J., Thain, N., and Vasserman, L. Measuring and mitigating unintended bias in text classification. 2018.

Eastwood, C. and Williams, C. K. I. A framework for the quantitative evaluation of disentangled representations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

Fei-Fei, L., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

Fumero, M., Cosmo, L., Melzi, S., and Rodolà, E. Learning disentangled representations via product manifold projection. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3530–3540. PMLR, 2021.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Geng, Z., Zhang, X., Bai, S., Wang, Y., and Lin, Z. On training implicit models. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 24247–24260, 2021.

Goyal, A., Lamb, A., Hoffmann, J., Sodhani, S., Levine, S., Bengio, Y., and Schölkopf, B. Recurrent independent mechanisms. In *International Conference on Learning Representations*, 2020.

Griewank, A. and Walther, A. *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM, 2008.

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439*, 2020.

Hu, Z., Zhao, Z., Yi, X., Yao, T., Hong, L., Sun, Y., and Chi, E. H. Improving multi-task generalization via regularizing spurious correlation. *ArXiv preprint*, abs/2205.09797, 2022.

Huang, Z., Wang, H., Xing, E. P., and Huang, D. Self-challenging improves cross-domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 124–140. Springer, 2020.

Hyvärinen, A., Sasaki, H., and Turner, R. E. Nonlinear ICA using auxiliary variables and generalized contrastive learning. In Chaudhuri, K. and Sugiyama, M. (eds.), *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pp. 859–868. PMLR, 2019.

Jiang, Y. and Veitch, V. Invariant and transportable representations for anti-causal domain shifts, 2022.

Khemakhem, I., Kingma, D. P., Monti, R. P., and Hyvärinen, A. Variational autoencoders and nonlinear ICA: A unifying framework. In Chiappa, S. and Calandra, R. (eds.), *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2207–2217. PMLR, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Kirichenko, P., Izmailov, P., and Wilson, A. G. Last layer re-training is sufficient for robustness to spurious correlations. *ArXiv preprint*, abs/2204.02937, 2022.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B., Haque, I. S., Beery, S. M., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. WILDS: A benchmark of in-the-wild distribution shifts. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 2021.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Le Priol, R., and Courville, A. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

Kulkarni, T. D., Whitney, W. F., Kohli, P., and Tenenbaum, J. B. Deep convolutional inverse graphics network. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M.,

and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2539–2547, 2015.

Lachapelle, S., Deleu, T., Mahajan, D., Mitliagkas, I., Bengio, Y., Lacoste-Julien, S., and Bertrand, Q. Synergies between disentanglement and sparsity: a multi-task learning perspective. *ArXiv preprint*, abs/2211.14666, 2022a.

Lachapelle, S., Rodriguez, P., Sharma, Y., Everett, K. E., Le Priol, R., Lacoste, A., and Lacoste-Julien, S. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. In *Conference on Causal Learning and Reasoning*, pp. 428–484. PMLR, 2022b.

LeCun, Y., Huang, F. J., and Bottou, L. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pp. II–104. IEEE, 2004.

Lee, K., Maji, S., Ravichandran, A., and Soatto, S. Meta-learning with differentiable convex optimization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 10657–10665. Computer Vision Foundation / IEEE, 2019.

Li, D., Yang, Y., Song, Y., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 5543–5551. IEEE Computer Society, 2017.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018a.

Li, H., Pan, S. J., Wang, S., and Kot, A. C. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5400–5409, 2018b.

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 624–639, 2018c.

Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, S. CITRIS: causal identifiability from temporal intervened sequences. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13557–13603. PMLR, 2022.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124. PMLR, 2019.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. A sober look at the unsupervised learning of disentangled representations and their evaluation. *J. Mach. Learn. Res.*, 21:209:1–209:62, 2020a.

Locatello, F., Poole, B., Rätsch, G., Schölkopf, B., Bachem, O., and Tschannen, M. Weakly-supervised disentanglement without compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6348–6359. PMLR, 2020b.

Lu, C., Wu, Y., Hernández-Lobato, J. M., and Schölkopf, B. Invariant causal representation learning for out-of-distribution generalization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=-e4EXDWXnSn.

Marx, Z., Rosenstein, M. T., Kaelbling, L. P., and Dietterich, T. G. Transfer learning with an ensemble of background tasks. *Inductive Transfer*, 10, 2005.

Matthey, L., Higgins, I., Hassabis, D., and Lerchner, A. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

Miller, J., Taori, R., Raghunathan, A., Sagawa, S., Koh, P. W., Shankar, V., Liang, P., Carmon, Y., and Schmidt, L. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7721–7735. PMLR, 2021.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 10–18. JMLR.org, 2013.

Nam, H., Lee, H., Park, J., Yoon, W., and Yoo, D. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8690–8699, 2021.

Oreshkin, B. N., López, P. R., and Lacoste, A. TADAM: task dependent adaptive metric for improved few-shot learning. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 719–729, 2018.

Parascandolo, G., Kilbertus, N., Rojas-Carulla, M., and Schölkopf, B. Learning independent causal mechanisms. In *Proceedings of the 35th International Conference on Machine Learning, PMLR 80:4036-4044*, 2018.

Park, J. H., Shin, J., and Fung, P. Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2799–2804, Brussels, Belgium, 2018. Association for Computational Linguistics.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

Qiu, J., Zhu, Y., Shi, X., Wenzel, F., Tang, Z., Zhao, D., Li, B., and Li, M. Are multimodal models robust to image and text perturbations? *ArXiv preprint*, abs/2212.08044, 2022.

Reed, S. E., Zhang, Y., Zhang, Y., and Lee, H. Deep visual analogy-making. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 1252–1260, 2015.

Russell, B. C., Torralba, A., Murphy, K. P., and Freeman, W. T. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77 (1):157–173, 2008.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *ArXiv preprint*, abs/1911.08731, 2019.

Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8346–8356. PMLR, 2020.

Salakhutdinov, R. Deep learning. In Macskassy, S. A., Perlich, C., Leskovec, J., Wang, W., and Ghani, R. (eds.), *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pp. 1973. ACM, 2014.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv preprint*, abs/1910.01108, 2019.

Schmidhuber, J. Learning factorial codes by predictability minimization. *Neural computation*, 4(6):863–879, 1992.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5): 612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Seigal, A., Squires, C., and Uhler, C. Linear causal disentanglement via interventions. *ArXiv preprint*, abs/2211.16467, 2022.

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. FLAVA: A foundational language and vision alignment model. *ArXiv preprint*, abs/2112.04482, 2021.

Snell, J., Swersky, K., and Zemel, R. S. Prototypical networks for few-shot learning. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4077–4087, 2017.

Sorrenson, P., Rother, C., and Köthe, U. Disentanglement by nonlinear ICA with general incompressible-flow networks (GIN). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Standley, T., Zamir, A. R., Chen, D., Guibas, L. J., Malik, J., and Savarese, S. Which tasks should be learned together in multi-task learning? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9120–9132. PMLR, 2020.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pp. 443–450. Springer, 2016.

Sun, B., Feng, J., and Saenko, K. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*, pp. 153–171. Springer, 2017.

Veitch, V., D'Amour, A., Yadlowsky, S., and Eisenstein, J. Counterfactual invariance to spurious correlations: Why and how to pass stress tests, 2021.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 5385–5394. IEEE Computer Society, 2017.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., and Wierstra, D. Matching networks for one shot learning. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3630–3638, 2016.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Wang, Z. and Veitch, V. A unified causal view of domain invariant representation learning. *ArXiv preprint*, abs/2208.06987, 2022.

Wang, Z., Dai, Z., Póczos, B., and Carbonell, J. G. Characterizing and avoiding negative transfer. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 11293–11302. Computer Vision Foundation / IEEE, 2019.

Wattenberg, M., Viégas, F., and Johnson, I. How to use t-sne effectively. *Distill*, 1(10):e2, 2016.

Wenzel, F., Dittadi, A., Gehler, P. V., Simon-Gabriel, C.-J., Horn, M., Zietlow, D., Kernert, D., Russell, C., Brox, T., Schiele, B., Schölkopf, B., and Locatello, F. Assaying out-of-distribution generalization in transfer learning. In *Neural Information Processing Systems*, 2022.

Wiles, O., Gowal, S., Stimberg, F., Rebuffi, S., Ktena, I., Dvijotham, K., and Cemgil, A. T. A fine-grained analysis on distribution shift. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Willetts, M. and Paige, B. I don't need u: Identifiable non-linear ica without side information. *ArXiv preprint*, abs/2106.05238, 2021.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv preprint*, abs/1910.03771, 2019.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Lopes, R. G., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998. PMLR, 2022.

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. SUN database: Large-scale scene recognition from abbey to zoo. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pp. 3485–3492. IEEE Computer Society, 2010.

Yan, S., Song, H., Li, N., Zou, L., and Ren, L. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv:2001.00677*, 2020.

Yao, W., Sun, Y., Ho, A., Sun, C., and Zhang, K. Learning temporally causal latent processes from general temporal data. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Yuan, L., Chen, D., Chen, Y., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., Liu, C., Liu, M., Liu, Z., Lu, Y., Shi, Y., Wang, L., Wang, J., Xiao, B., Xiao, Z., Yang, J., Zeng, M., Zhou, L., and Zhang, P. Florence: A new foundation model for computer vision. *ArXiv preprint*, abs/2111.11432, 2021.

Zhang, M., Marklund, H., Dhawan, N., Gupta, A., Levine, S., and Finn, C. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in Neural Information Processing Systems*, 34:23664–23678, 2021.

Zhang, Y. and Yang, Q. An overview of multi-task learning. *National Science Review*, 5(1):30–43, 2018.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

Zhou, K., Liu, Z., Qiao, Y., Xiang, T., and Loy, C. C. Domain generalization: A survey. 2021.

Zhu, J., Zhu, X., Wang, W., Wang, X., Li, H., Wang, X., and Dai, J. Uni-perceiver-moe: Learning sparse generalist models with conditional moes. *ArXiv preprint*, abs/2206.04674, 2022.

A. Related work

Learning from multiple tasks and domains. Our method addresses the problem of learning a general representation across multiple and possibly unseen tasks (Caruana, 1997; Zhang & Yang, 2018) and environments (Zhou et al., 2021; Gulrajani & Lopez-Paz, 2021; Koh et al., 2021; Wortsman et al., 2022; Miller et al., 2021; Wiles et al., 2022; Muandet et al., 2013) that may be competing with each other during training (Marx et al., 2005; Wang et al., 2019; Standley et al., 2020). Prior research tackled task competition by introducing task specific modules that do not interact during training (Parascandolo et al., 2018; Yuan et al., 2021; Singh et al., 2021). While successfully learning specialized modules, these approaches can not leverage synergistic information between tasks, when present. On the other hand, our approach is closer to multi-task methods that aim at learning a generalist model, leveraging multi-task interactions (Zhu et al., 2022; Bai et al., 2022). Other approaches that leverage a meta-learning objective for multi-task learning have been formulated (Dhillon et al., 2020; Snell et al., 2017; Lee et al., 2019; Bertinetto et al., 2019). In particular, (Lee et al., 2019) proposes to learn a generalist model in a few-shot learning setting without explicitly favoring feature sharing, nor sparsity. Instead, we rephrase the multi-task objective function encoding both feature sharing and sparsity to avoid task competition.

Similar to prior work in domain generalization, we assume the existence of stable features for a given task (Muandet et al., 2013; Arjovsky et al., 2019; Veitch et al., 2021; Jiang & Veitch, 2022; Wang & Veitch, 2022) and amortize the learning over the multiple environments. Differently than prior work, we do not aim to learn an invariant representation a priori. Instead, we learn sufficient and minimal features for each task, which are selected at test time fitting the linear head on them. In light of (Gulrajani & Lopez-Paz, 2021), one can interpret our approach as learning the final classifier using empirical risk minimization but over features learned with information from the multiple domains.

Disentangled representations. Disentanglement representation learning (Bengio et al., 2013; Higgins et al., 2017) aims at recovering the factors of variations underlying a given data distribution. (Locatello et al., 2019) proved that without any form of supervision (whether direct or indirect) on the Factors of Variation (FOV) is not possible to recover them. Much work has then focused on identifiable settings (Locatello et al., 2020b; Fumero et al., 2021) from non-i.i.d. data, even allowing for latent causal relations between the factors. Different approaches can be largely grouped in two categories. First, data may be non-independently sampled, for example assuming sparse interventions or a sparse latent dynamics (Goyal et al., 2020; Lippe et al., 2022; Brehmer et al., 2022; Yao et al., 2022; Ahuja et al., 2020; Seigal et al., 2022; Lachapelle et al., 2022b). Second, data may be non-identically distributed, for example being clustered in annotated groups (Hyvärinen et al., 2019; Khemakhem et al., 2020; Sorrenson et al., 2020; Willetts & Paige, 2021; Lu et al., 2022). Our method follows the latter, but we do not make assumptions on the factor distribution across tasks (only their relevance in terms of sufficiency and minimality). This is also reflected in our method, as we train for supervised classification as opposed to contrastive or unsupervised learning as common in the disentanglement literature. The only exception is the work of (Lachapelle et al., 2022a) discussed in Section B.

B. Theoretical analysis

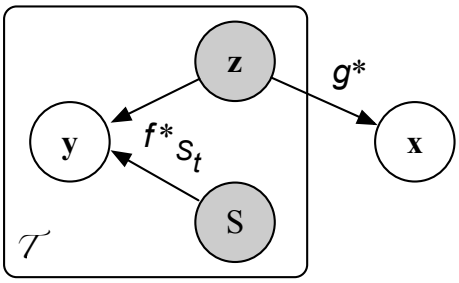


Figure 3: Assumed causal generative model: the gray variables are unobserved. Observations $\mathbf{x}$ are generated by some unknown mixing of a set of factors of variations $\mathbf{z}$. Additionally, we observe a distribution of supervised tasks, only depending on a subset of factors of variations indexed by S.

We analyze the implications of the proposed minimality and sparse sufficiency principles and show in a controlled setting that they indeed lead to identifiability. As outlined in Figure 3, we assume that there exists a set of independent latent factors $\mathbf{z} \sim \prod_{i=1}^{d} p(z_i)$ that generate the observations via an unknown mixing function $\mathbf{x} = g^*(\mathbf{z})$. Additionally, we assume that

the labels for a task t only depend on a subset of the factors indexed by $S_t \sim P(S)$, where S is an index set on $\mathbf{z} \in \mathcal{Z}$, via some unknown mixing function $y = f^*_{S_t}(\mathbf{z})$ (potentially different for different tasks). We formalize the two principles that are imposed on f^* by:

1. *sufficiency*: $f^*_t = f^*_t|_{S_t}$ for $S_t \sim p(\mathcal{S})$
2. *minimality*: $\nexists S' \neq S_t \subset \mathcal{S}$ s.t. $f^*_t|_{S'} = f^*_t$,

where $f|_{S_t}$ denotes that the input to a function f is restricted to the index set given by S_t (all remaining entries are set to zero). (1) states that f^*_t only uses a subset of features, and (2) states that there are not be duplicate features.

Proposition B.1. *Assume that g^* is a diffeomorphism (smooth with smooth inverse), f^* satisfies the sufficiency and minimality properties stated above, and $p(S)$ satisfies: $p(S \cap S' = \{i\}) > 0$ or $p(\{i\} \in (S \cup S') - (S' \cap S)) > 0$. Observing unlimited data from $p(X, Y)$, it is possible to recover a representation $\hat{\mathbf{z}}$ that is an axis aligned, component wise transformation of $\mathbf{z}$.*

Remarks: Overall, we see this proposition as validation that in an idealized setting our inductive biases are sufficient to recover the factors of variation. Note that the proof is non-constructive and does not entail a specific method. In practice, we rely on the same constraints as inductive biases that lead to this theoretical identifiability and experimentally show that disentangled representations emerge in controlled synthetic settings. On real data, (1) we cannot directly measure disentanglement, (2) a notion of global ground-truth factors may even be ill-posed, and (3) the assumptions of Proposition B.1 are likely violated. Still, sparse sufficiency and minimality yield some meaningful factorization of the representation for the considered tasks.

Relation to (Lachapelle et al., 2022a) and (Locatello et al., 2020b): Our theoretical result can be reconnected with concurrent work (Lachapelle et al., 2022a) and can be seen as a corollary with a different proof technique and slightly relaxed assumptions. The main difference is that our feature minimality allows us to also cover the case where the number of factors of variations is unknown, which we found critical in real world data sets (the main focus of our paper). Instead, they only assume sparse sufficiency, which is enough for identifiability if the ground-truth number of factors is known, but is not enough to recover high disentaglement when this is not the case (see Figure 1) and does not translate well to real data, see Table 16 with the empirical comparison in Appendix F.9. Interestingly, their analysis also hints at the fact that our approach also benefits in terms of sample complexity on transfer learning downstream tasks. Our proof technique follows the general construction developed for multi-view data in (Locatello et al., 2020b), adapted to our different setting. Instead of observing multiple views with shared factors of variation, we observe a single task that only depend on a subset of the factors.

C. Proof of Proposition 1

To prove PropositionB.1 we rely on the same proof construction of (Locatello et al., 2020b), adapting it to our setting. The proof is sketched in three steps:

- First, we prove identifiability when the support S of a task is arbitrary but fixed, where we drop the subscript t for convenience.
- Second, we randomize on S, to extend the proof for S drawn at random.
- Third, we extend the proof to the case when the dimensionality of $\mathcal{Z}$ is unknown and we start on overestimate of it to recover it.

Identifiability with fixed task support We assume the existence of the generative model in Figure 3, which we report here for convenience:

$$p(\mathbf{z}) = \prod_i p(z_i) \qquad S \sim p(S) \tag{6}$$

$$\mathbf{x} = g^*(\mathbf{z}) \qquad y = f^*_S(\mathbf{z}) \tag{7}$$

together with the assumptions specified in theorem statement. We fix the support of the task S. We indicate with $g : Z \to X$ the invertible smooth, candidate function we are going to consider, whose inverse corresponds to $q(\mathbf{z}|\mathbf{x})$. We denote with $T \in S$ which indexes the coordinate subspace of image of g^{-1} corresponding to the unknown coordinate subspace S of factors of variation on which the fixed task depends on. Fixing T requires knowledge of $|S|$. The candidate function g^{-1}

must satisfy:

$$f|_T(g^{-1}(\mathbf{x})) = y \tag{8}$$

$$f|_{\bar{T}}(g^{-1}(\mathbf{x})) \neq y \tag{9}$$

where $\bar{T}$ denotes the indices in the complement of T. f denotes a predictor which satisfies the same assumptions on f^* on T . We parametrize g^{-1} with g^{*-1} and set:

$g^{-1} = h^{-1} \circ g^{*-1}$ where $h : [0, 1]^d \to Z$, mapping from the uniform distribution on $\mathbb{R}^d$ to Z. We can rewrite the two above constraints as:

$$f|_T(h^{-1}(z)) = y \tag{10}$$

$$f|_{\bar{T}}(h^{-1}(z)) \neq y \tag{11}$$

We claim that the only admissible functions h^{-1} maps each entry in $\mathbf{z}$ to unique coordinate in T. We observe that due to its smoothness and invertibility, h^{-1} maps Z to the submanifolds $\mathcal{M}_s, \mathcal{M}_{\bar{s}}$, which are disjoint. By contradiction:

- if $\mathcal{M}_{\bar{S}}$ does not lie in $\bar{T}$ then minimality is violated.
- if $\mathcal{M}_S$ does not lie in T then sufficiency is violated

h^{-1} maps each entry in $\mathbf{z}$ to unique coordinate in T. Therefore there exist a permutation π s.t.:

$$h_T^{-1}(\mathbf{z}) = \bar{h}_T(\mathbf{z}_{\pi(S)}) \tag{12}$$

$$h_{\bar{T}}^{-1}(\mathbf{z}) = \bar{h}_{\bar{T}}(\mathbf{z}_{\pi(\bar{S})}) \tag{13}$$

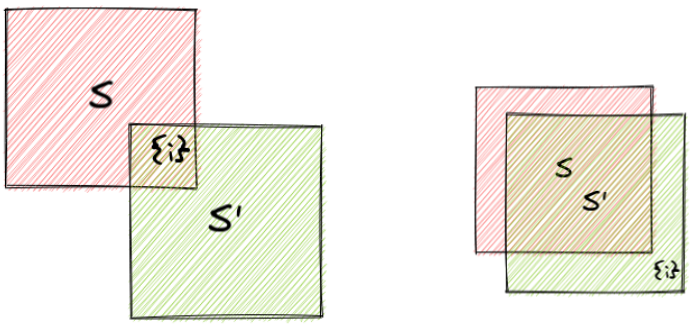
The Jacobian of h^{-1} is a blockwise matrix with block indexed by T. So we can identify the two blocks of factors in $S, \bar{S}$ but not necessarily the factors within, as they may be still entangled.

Randomization on S

we now consider S to be drawn at random, therefore we observe $p(\mathbf{x}, y|S)$ without never observing S directly. g^{-1} must now associate each $p(\mathbf{x}, y)$ with a unique T, as well as a unique predictor f , for each $S \sim p(S)$ Indeed suppose that $p(\mathbf{x}, y|S = S_1)$ and $p(\mathbf{x}, y|S = S_2)$ with $S_1, S_2 \sim p(S)$ and $S_1 \neq S_2$. Then if T would be the same for both tasks (as f), eq (6) could only be satisfied for a subset of size $|S_1 \cap S_2| < |S_1 \cup S_2|$, while T is required to be of size $|S_1 \cup S_2|$ This corresponds to say that each task has its own sparse support and its own predictor. Conversely all $p(\mathbf{x}, y) \in supp(p(\mathbf{x}, y|S))$ need to be associated to the T and the same predictor f, since they will all share the same subspace and cannot be associated to different T. Notice also that $|S_1 \cap S_2| = |T_1 \cap T_2|$ and $|S_1 \cup S_2| = |T_1 \cup T_2|$. We further assume:

$\forall z_i$ either $p(S \cap S' = \{i\}) > 0$ or $p(\{i\} \in (S \cup S') - (S' \cap S)) > 0$

We observe every factor as the intersection of the sets S, S' which will be reflected in T, T' or we observe single factors in the difference between the intersection and the union of S, S'. Examples of the two cases are illustrated below:



This together with (8) and (9) implies:

$$h_i^{-1}(\mathbf{z}) = \bar{h}_i(z_{\pi(i)}) \quad \forall i \in [d] \tag{14}$$

This further implies that the jacobian of $\bar{h}$ is diagonal. By the change of variable formula we have:

$$q(\hat{\mathbf{z}}) = p(\tilde{h}(\mathbf{z}_{\pi([d])})) \left| det \frac{\partial}{\partial \mathbf{z}_{\pi([d])}} \tilde{h} \right| = \prod_{i01}^{d} p(\tilde{h}_i(z_{\pi(i)})) \left| \frac{\partial}{\partial z_{\pi(i)}} \tilde{h}_i \right| \tag{15}$$

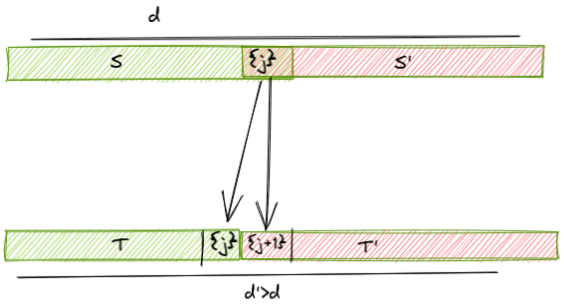
This holds for the jacobian being diagonal and invertibility of $\tilde{h}$. Therefore $q(\hat{\mathbf{z}})$ is a coordinate-wise reparametrization of $p(\mathbf{z})$ up to a permutation of the indices. A change in a coordinate of $\mathbf{z}$ implies a change in the unique corresponding coordinate of $\hat{\mathbf{z}}$, so g disentangles the factors of variation.

Dimensionality of the support S

Previously we assumed that the dimension of $\hat{\mathbf{z}}$ is the same as $\mathbf{z}$. We demonstrate that even when d is unknown starting from an overstimate of it, we can still recover the factors of variations. Specifically, we consider the case when $\hat{d} > d$. In this case our assumption about the invertibility of h is violated. We must instead ensure that h maps Z to a subspace of $\hat{Z}$ with dimension d. To substitute our assumption on inveribility on h, we will instead assume that $\mathbf{z}$ and $\hat{\mathbf{z}}$ have the same mutual information with respect to task labels Y, i.e.$I(Z, Y) = I(\hat{Z}, Y)$ Note that mutual information is invariant to invertible transformation, so this property was also valid in our previous assumption.

Now, consider two arbitrary tasks with $|S \cap S'| \neq \emptyset$ =k but $|T \cap T'| < k$, i.e. some features are duplicated/splitted. Hence f, f' while have different support , i.e.:

$$f|_T = f'|_{T'} = f^*$$



We observe that in this situation nor sufficiency, nor minimality are necessarily violated because:

- $f|_T = f'|_{T'} = f^*$ (sufficiency is not violated)
- $T \cap T' = \emptyset \implies T \not\subset T', T' \not\subset T$ (minimality is not violated)

In other words we must ensure that a single fov z_i is not mapped to different entries in $\hat{\mathbf{z}}$ (feature splitting or duplication). We fix two arbitrary tasks with $|S \cap S'| \neq \emptyset$ =k but $|T \cap T'| < k$, i.e. some features are duplicated. We know that $|S| = |T|$ and $|S'| = |T'|$ otherwise sufficency and minimaliy would be violated. Then if $|T \cap T'| < k$, then $|T \cup T'| > |S \cup S'| = d - k$ we have $p(|T \cup T'|)$=$p(supp(p(y|\hat{\mathbf{z}})) + supp(p'(y'|\hat{\mathbf{z}'}))) = p(\sum_i supp(f_i(.))$, and since

$$H[p(\sum_i supp(f_i(.))] > H[p(\sum_i supp(f_i(.))] \tag{16}$$

but we have assumed:

$$I(Z,Y) = I(\hat{Z},Y) \tag{17}$$

$$\cancel{H(Y)} - H(Y|\hat{Z}) = \cancel{H(Y)} - H(Y|Z) \tag{18}$$

$$H(Y|\hat{Z}) = H(Y|Z) \tag{19}$$

$$H[p(Y|\hat{Z}) > 0] = H[p(Y|Z) > 0] \tag{20}$$

$$2^{H[p(Y|\hat{Z})>0]} = 2^{H[p(Y|Z)>0]} \tag{21}$$

$$|supp(p(Y|\hat{Z}))| = |supp(p(Y|Z)| \tag{22}$$

this last passage is due to relation between cardinality and entropy: for uniform distributions the exponential of the entropy is equal to the cardinality of the support of the distribution.

$$|supp(f)| = |supp(f^*)| \tag{23}$$

We know that (12) must hold for every task, therefore: $\sum_i I(Z,Y_i) = \sum_i I(\hat{Z},Y_i)$ for each i then: $\sum_i |supp(\hat{f}_i)| = \sum_i |supp(f_i^*)|$ $|\bigcup_i T_i| = |\bigcup_i S_i|$ therefore (12) contradicts our assumption (13).

D. Implementation details

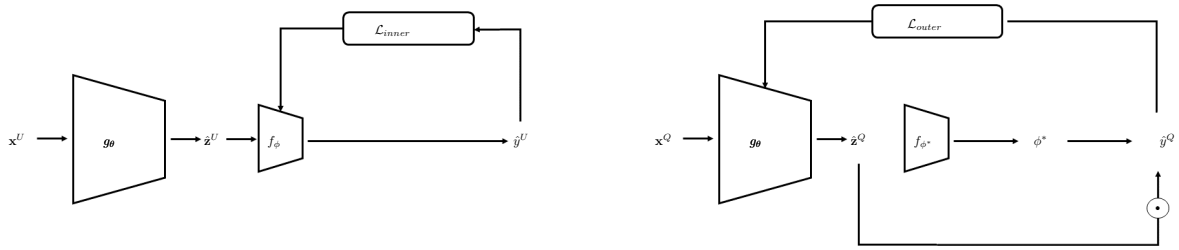


Figure 4: *Model scheme*: Illustrations of the (*Top*) the inner loop stage and outer loop following the steps of the algorithmic procedure described in Section D.1

D.1. Training algorithm

Real tasks generation . Our method can be applied in a standard supervised classification setting where we construct the tasks on the fly as follows. We define a task t as a C-way classification problem. We first select a random subset of C classes from a training domain D_{train} which contains K_{train} classes. For each class we consider the corresponding data points and select a random support set U_t with elements $(\mathbf{x}_t^U, y^U) \in U$ and a disjoint random query set Q_t with elements $(\mathbf{x}_t^Q, y^Q) \in Q_t$. In each iteration we sample a batch of T tasks with the associated support and query set as described above. First, we use the samples from the support set S_t to fit the linear heads f_ϕ by solving the inner optimization problem (5) using stochastic gradient descent for a fixed number of steps. Second, we use the samples from the query set Q_t to update the backbone g_θ by solving the outer optimization problem (4) using implicit differentiation (Geng et al., 2021; Blondel et al., 2021; Griewank & Walther, 2008).

Algorithm 1 Training algorithm

Input: A task distribution $\mathcal{T}$
while Not converged **do**
  Sample a batch B_T of T tasks $t \sim \mathcal{T}$
  Sample (U_t, Q_t) from each task in the batch
  # **Inner loop**
  for each t in B_T **do**
    Compute $\mathbf{z}_t^U = g_\theta(\mathbf{x}_t^U)$
  end for
  Solve $\phi^* = argmin_\phi \frac{1}{T} \sum_t \mathcal{L}_{inner}(f_\phi(\mathbf{z}_t^U), y_t^U) + Reg(\phi)$
  # **Outer loop**
  for each t **do**
    Compute $\mathbf{z}_t^Q = g_\theta(\mathbf{x}_t^Q)$
  end for
  Compute $\mathcal{L}_{outer}(f_{\phi^*}(g_\theta(\mathbf{x}_t^Q), y_t^Q))$
  Compute $\frac{\partial \mathcal{L}_{outer}(\theta)}{\partial \theta}$ as in (Geng et al., 2021)
  Update θ
end while

D.2. Implicit gradients

In the backward pass, denoting with $\mathcal{L}^*_{outer} = \mathcal{L}_{outer}(f^*_\phi(g_\theta(x^Q)), Y^Q)$ denoting the loss computed with respect to the optimal classifier f^*_ϕ on the query samples (x^Q, Y^Q), we have to compute the following gradient:

$$\frac{\partial \mathcal{L}^*_{outer}(\theta)}{\partial \theta} = \frac{\partial \mathcal{L}_{outer}(\theta, \phi^*)}{\partial \theta} + \frac{\mathcal{L}_{outer}(\theta, \phi^*)}{\partial \phi^*} \frac{\partial \phi^*}{\partial \theta} \tag{24}$$

where is the algorithm procedure to solve Eq1, i.e. SGD. While is just the gradient of the loss evaluated at the solution of the inner problem and can be computed efficiently with standard automatic backpropagation, requires further attention. Since the solution to C_{ϕ^*} is implemented via and iterative method (SGD), one strategy would be to compute this gradient would be to backpropagate trough the entire optimization trajectory in the inner loop. This strategy however is computational inefficient for many steps, and can suffer also from vanishing gradient problems.

E. Experimental details

All experiments were performed on a single gpu NVIDIA RTX 3080Ti and implemented with the Pytorch library (Paszke et al., 2019).

Experimental setting. To have a fair comparison with other methods in the literature, we adopt the standard experimental setting of prior work (Gulrajani & Lopez-Paz, 2021; Koh et al., 2021). Hyperparameters α and β are tuned performing model selection on validation set, unless specified otherwise. For comparison with baselines, we substitute our backbone with that of the baseline (e.g. for ERM models, we detach the classification head) and then fit a new linear head on the same data. The linear head module trained at test time on top of the features is the same both for our and compared methods. Despite its simplicity, we report the ERM baseline for comparison in our experiments in the main paper, since it has been shown to perform best in average on domain generalization benchmarks (Gulrajani & Lopez-Paz, 2021; Koh et al., 2021). We further compare with other consolidated approaches in the literature such as IRM (Arjovsky et al., 2019), CORAL (Sun & Saenko, 2016) and GroupDRO (Sagawa et al., 2019) and include a large and comprehensive comparison with (Yan et al., 2020; Blanchard et al., 2021; Li et al., 2018a;b; Ganin et al., 2016; Li et al., 2018c; Nam et al., 2021; Zhang et al., 2021; Huang et al., 2020; Krueger et al., 2021) in AppendixF.5. Experimental details are fully described below.

E.1. Datasets

We evaluate our method on a synthetic setting on the following benchmarks: `DSprites`, `AbstractDSprites`(Matthey et al., 2017), `3Dshapes` (Burgess & Kim, 2018),`SmallNorb` (LeCun et al., 2004), `Cars3D`(Reed et al., 2015) and the semi-synthetic `Waterbirds` (Sagawa et al., 2019).

For domain generalization and domain adaptation tasks, we evaluate our method on the (Gulrajani & Lopez-Paz, 2021) and (Koh et al., 2021) benchmarks, using the following datasets: `PACS`(Li et al., 2017), `VLCS`(Albuquerque et al., 2019), `OfficeHome`(Venkateswara et al., 2017) `Camelyon17`(Bandi), `CivilComments` (Borkan et al., 2019).

Dataset descriptions

The `Waterbirds` dataset (Sagawa et al., 2019) is a synthetic dataset where images are composed of cropping out birds from photos in the `Caltech-UCSD Birds-200-2011 (CUB)` dataset (Wah et al., 2011) and transferring them onto backgrounds from the Places dataset (Zhou et al., 2017). The dataset contains a large percentage of training samples ($\approx \%95$) which are spuriously correlated with the background information.

The `CivilComments` is a dataset of textual reviews annotated with demographics information for the task of detecting toxic comments. Prior work has shown that toxicity classifiers can pick up on biases in the training data and spuriously associate toxicity with the mention of certain demographics (Park et al., 2018; Dixon et al., 2018). These types of spurious correlations can significantly degrade model performance on particular subpopulations (Sagawa et al., 2020).

The `PACS` dataset (Li et al., 2017) is a collection of images coming from four different domains: *real images, art paintings, cartoon* and *sketch*. The `VLCS` dataset contains examples from 5 overlapping classes from the VOC2007 (Everingham et al.), LabelMe (Russell et al., 2008), Caltech-101 (Fei-Fei et al., 2004) , and SUN (Xiao et al., 2010) datasets. The `OfficeHome` dataset contains 4 domains (Art, ClipArt, Product, real-world) where each domain consists of 65 categories.

The `Camelyon17` dataset, is a collection of medical tissue patches scanned from different hospital environments. The task is to predict whether a patch contain a benign or tumoral tissue. The different hospitals represent the different domains in this problem, and the aim is to learn a predictor which is robust to changes in factors of variation across different hospitals.

E.2. Models

For synthetic datasets we use a CNN module for the backbone $g\theta$ following the architecture in Table 5. For real datasets that use images as modality we use a `ResNet50` architecure as backbone pretrained on the `Imagenet` dataset. For the experiments on the text modality we use `DistilBERT` model (Sanh et al., 2019) with pretrained weights downloaded from HuggingFace (Wolf et al., 2019).

E.3. Synthetic experiments

Table 5: Convolutional architecture used in synthetic experiments.

CNN backbone
Input : $64 \times 64 \times$ number of channels
4×4conv, 32 stride 2, padding 1, ReLU,BN
4×4conv, 32 stride 2, padding 1, ReLU,BN
4×4conv, 64 stride 2, padding 1, ReLU,BN
4×4conv, 64 stride 2, padding 1, ReLU,BN
FC, 256, Tanh
FC, d

Synthetic tasks generation For the synthetic experiments we have access to the ground truth factors of variations $\mathcal{Z}$ for each dataset. The task generation procedure relies on two hyperparameters: the first one is an index set $\mathbb{S}$ of possible factors of variations on which the distribution of tasks can depend on. The latter hyperparameter K, set the maximum number of factors of variations on which a single task can depend on. Then a task t is sampled drawing a number k_t from $\{1...K\}$, and then sampling randomly a subset S of size $|\mathbb{S}| - k_t$ from $\mathbb{S}$. The resulting set S will be the set indexing the factors of variation in Z on which the task t is defined. In this setting restrict ourselves to binary task: for each factors in S, we sample a random value v for it. The resulting set of values V, will determine uniquely the binary task.

Before selecting $v \in V$ we quantize the possible choices corresponding to factors of variations which may have more than six values to 2. We remark that this quantization affect only the task label definition. For examples for x axis factor, we consider the object to be on the left if its x coordinate is less than the medial axis of the image, on the right otherwise. The `DSprites` dataset has the following set of factors of variations $Z_{dsprites} = \{shape, size, angle, x_{pos}, y_{pos}\}$ and example

of task is *There is a big object on the right* where $k_t = 2$ the affected factors are $size, x_{pos}$. Another example is *There is a small heart on the top left* , where $k_t = 4$ the affected factors are $shape, size, x_{pos}, y_{pos}$. Obervations are labelled positively of negatively if their corresponding factors of variations matching in the values with the one specified by the current task.

We then samples random query Q and support U set of samples balanced with respect to postive and negative labels of task task t, using stratified sampling.

Real tasks generation . Our method can be applied in a standard supervised classification setting where we construct the tasks on the fly as follows. We define a task t as a C-way classification problem. We first select a random subset of C classes from a training domain D_{train} which contains K_{train} classes. For each class we consider the corresponding data points and select a random support set U_t with elements $(\mathbf{x}_t^U, y^U) \in U$ and a disjoint random query set Q_t with elements $(\mathbf{x}_t^Q, y^Q) \in Q_t$.

E.4. Experiments on domain shifts

In a domain generalization setting, we do not have access to samples coming from the testing domain, which is considered to be OOD w.r.t. to the training domains. However, in order to solve a new task, our method relies on a set labeled data at test time to fit the linear head on top of the feature space. Our strategy is to sample data points from the training distribution, balanced by class, assuming that the label set Y does not change in the testing domain, although its distribution may undergo subpopulation shifts. This sampling strategy is in line with what is highlighted in (Kirichenko et al., 2022), where it is shown that retraining the linear head of a deep classifier on a small set of balanced samples (w.r.t to minority groups in the training data) is sufficient to achieve robustness to spurious correlations in the test data. The main difference is that we typically don't assume to have labels on the minority groups in the training set and we just balance the sampling by the class label. To fit the linear head we sample 10 times with different samples sizes from the training domains and we report the mean score and standard deviation.

For the domain generalization and few-shot transfer learning experiments we put ourselves in the same settings of (Gulrajani & Lopez-Paz, 2021; Koh et al., 2021) to ensure a fair comparison. Namely, for each dataset we use the same augmentations, and same backbone models.

For solving the inner problem in Equation 5, we used Adam optimizer (Kingma & Ba, 2015), with a learning rate of $1e-2$, momentum 0.99, with the number of gradient steps varying from 50 to 100, from the synthetic setting to domain shifts experiments. For the latter, the task (or episode) sampling during training is done as follows: we sampled each task as a multiclass classification problem setting the number of classes $C = 5$ when the original number of classes K_{train} in the dataset was higher than five, i.e. $K_{train} > 5$, $C = K_{train}$ otherwise. During training, the sizes of the support set U and query sets Q where set to $|U| = 25, |Q| = 15$ similar to as done in prior meta-learning literature (Lee et al., 2019; Dhillon et al., 2020). Changing these parameters has similar effects from what has been observed in many meta learning approaches(e.g. (Lee et al., 2019; Dhillon et al., 2020)).

E.5. Selection of α and β

To find the best regularization parameters α, β weighting the sparsity and feature sharing regularizers in Equation 1 respectively, we perform model selection according to the highest accuracy on a validation set. We report in Table 6 the value selected for each experiment.

Table 6: Selected values for α and β for all experiments, applying model selection on validation set.

Experiment	α	β
Table 1	1e-2	0.15
Table 2	1e-2	5e-2
Table 3	2.5e-3	5e-2
Table 4	1.5e-3	1e-2
Table 5, 6	2.5e-3	1e-2
Table 7	2.5e-3	1e-2

Table 7: Quantitative results accompanying Figure 5

	$\alpha = 0, \beta = 0$	$\alpha = 1e-2, \beta = 0$	$\alpha = 1e-2, \beta = 0.2$	$\alpha = 1e-2, \beta = 0.4$
DCI	27.8	71.9	98.8	30.5

F. Additional results

F.1. Synthetic experiments

The role of minimality In Figure 5we show the qualitative results accompanying Figure1. The qualitative results in the Figure are produced visualizing matrices of feature importance (Locatello et al., 2020a) computed fitting Gradient Boosted Trees (GBT) on the learned representations w.r.t. task labels, and on the factors of variations w.r.t. task labels and compare the results. In each matrix the x axis represents the tasks and the y axis the features, and each entries the amount of feature importance (which goes from 0 to 1).

Task compositional generalization In Table 8 we show the quantitative results accompanying Figure 2.

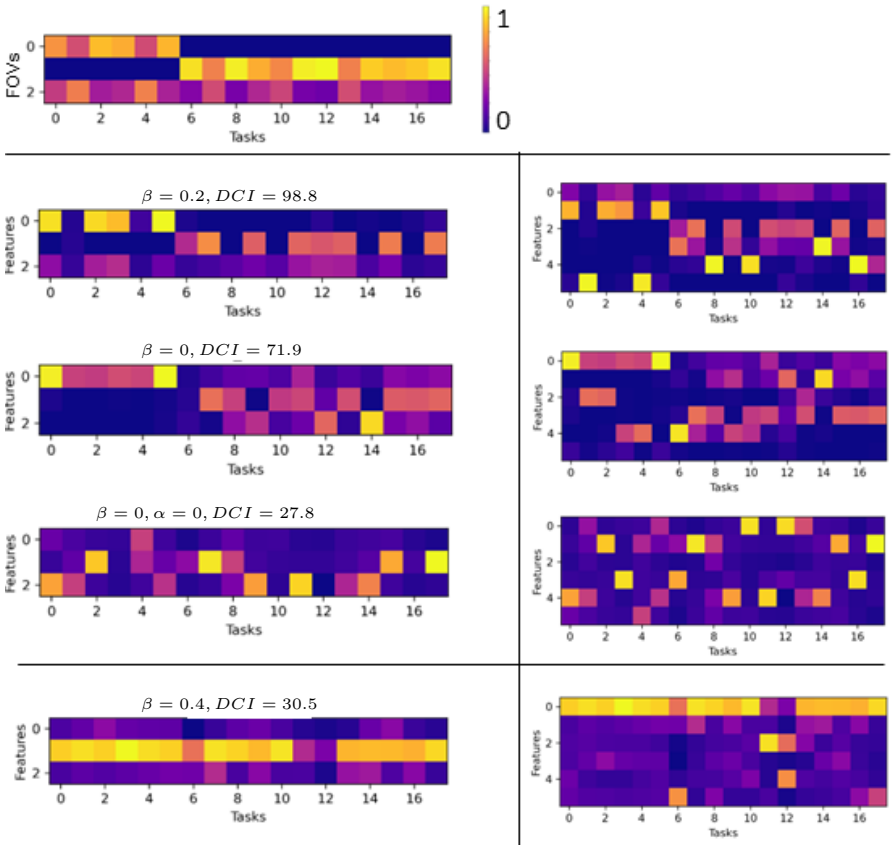


Figure 5: Qualitative dependency of disentanglement from the weight of our penalties ($\alpha = 0.01$ unless otherwise specified). The model that attains the best disentanglement ($DCI = 98.8$) uses both. *Left column, top*: ground-truth importance weights of each latent factor for each task. *Right column*: we train models with different β and visualize the weights assigned to each learned feature on each task. *Left column*: to determine whether the model recover the ground-truth latents, we select the 3 top features and compare their assigned weights on different tasks with the ground-truth weights. *Bottom row*: example of a failure case with high β.

Table 8: *Task compositional generalization*: Mean accuracy over 100 random tasks reported for group of tasks of growing support (*second, third, fourth column*) for a model trained without inductive biases (*top row*) and enforcing them (*bottom row*). The latter show better compositional generalization resulting from the properties enforced on the representation

	Acc ID	DCI	$\|S\| = 3$	$\|S\| = 4$	$\|S\| = 5$
No reg	88.7	22.8	72.6	63.3	59.9
α, β	**93.2**	**59.4**	**83.0**	**78.8**	**76.8**

F.2. Properties of the learned representations

Feature sufficiency. The sufficiency property is crucial for robustness to spurious correlations in the data. If the model can learn and select the relevant features for a task, while ignoring the spurious ones, sufficiency is satisfied, resulting in robust performance under subpopulation shifts, as shown in Tables 9 and 4. To get qualitative evidence of the sufficiency in the representations, in Figure 6 we show the saliency maps computed from the activations of our model and a corresponding model trained with ERM. Our model can learn features specific to the subject of the image, which are relevant for classification, while ignoring background information. This can be observed in both correctly classified (bottom row) and misclassified (top row) samples by ERM. In contrast, ERM activates features in the background and relies on them for prediction.

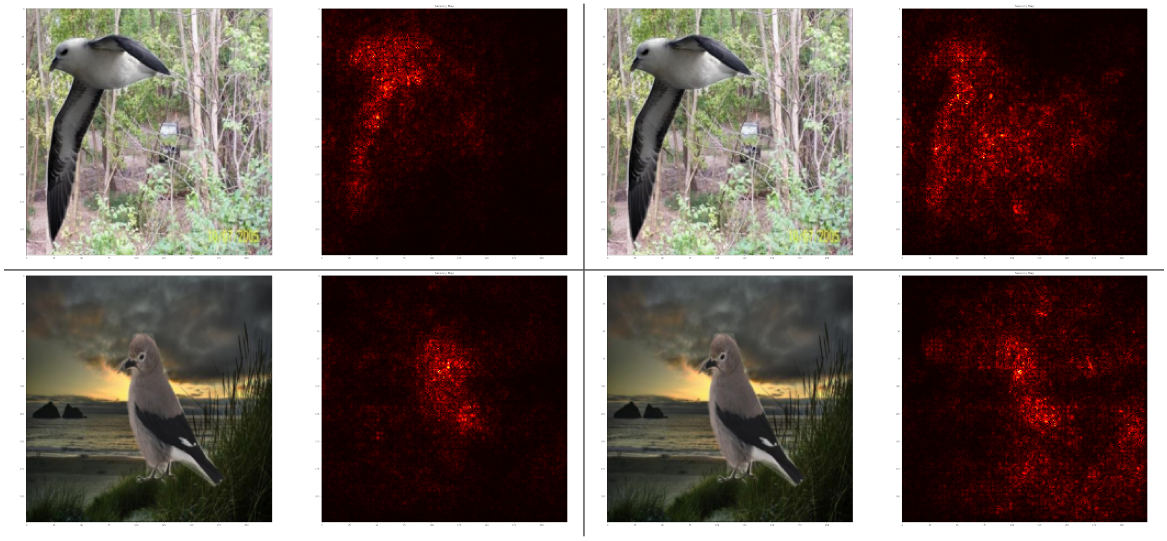


Figure 6: *Feature sufficiency: Left*, pairs of random samples and saliency maps computed on activations with our method. All samples are correctly classified. *Right*, corresponding saliency maps (Adebayo et al., 2018) an ERM based method: the first row is misclassifed by the network, the last is correctly classified. The ERM model depends on features from the background, resulting in a higher prediction error on mixed subdomains. Our model is robust to spurious correlations and satisfies the sufficiency assumptions.

Feature sharing. In this section, we study the minimality properties of the representations learned by our method. To achieve this, we conduct the following experiment. We randomly draw 14 tasks from the $\sum_{i=1}^{3} \binom{4}{i}$ possible combinations of the four domains in the `PACS` dataset. We use the data from these tasks to fit the linear head and test the model accuracy on the OOD domain (e.g. the *sketch* domain). In Figure 7, we show the performance on each task, ordered on the x axis according to OOD accuracy of a model trained with ERM (in yellow). We also report the fraction of activated features (in blue) shared between each task and the OOD task, and the same(red) for the ERM model. The fraction of activated features is computed by looking at the matrix of coefficients of the sparse linear head $\phi \in \mathbb{R}^{M \times C}$, where M is the number of features and C the number of classes, after fitting on each task. Specifically, is computed as $\frac{\sum_m [\tilde{\phi}_\epsilon \cap \tilde{\phi}_\epsilon^{OOD}]}{\sum_m [\tilde{\phi}_\epsilon \cup \tilde{\phi}_\epsilon^{OOD}]}$ where $\tilde{\phi}_\epsilon = \frac{1}{C} \sum_c |\phi_{m,c}| > \epsilon$ and ϕ^{OOD} is the matrix of coefficient of the OOD task. We set $\epsilon = 0.01$. From Figures 7 and 8 we draw the following conclusions: (i) When the accuracy of the ERM decreases (i.e., the current task is farther from the

OOD test task), our method is still able to retain a high and consistent accuracy, demonstrating that our features are more robust out-of-distribution. This is further supported by the higher number of shared features compared to ERM, as we move away from the testing domain. (ii) The correlation between the fraction of shared features and the accuracy OOD demonstrates that the method is able to learn general features that transfer well to unseen domains, thanks to the minimality constraint. Additionally, this measure serves as a reliable indicator of task distance, as discussed in the next section. (iii) Even though the same sparse linear head is used on top of the ERM and our features, our method is able to achieve better OOD performance with fewer features, further demonstrating our feature minimality.

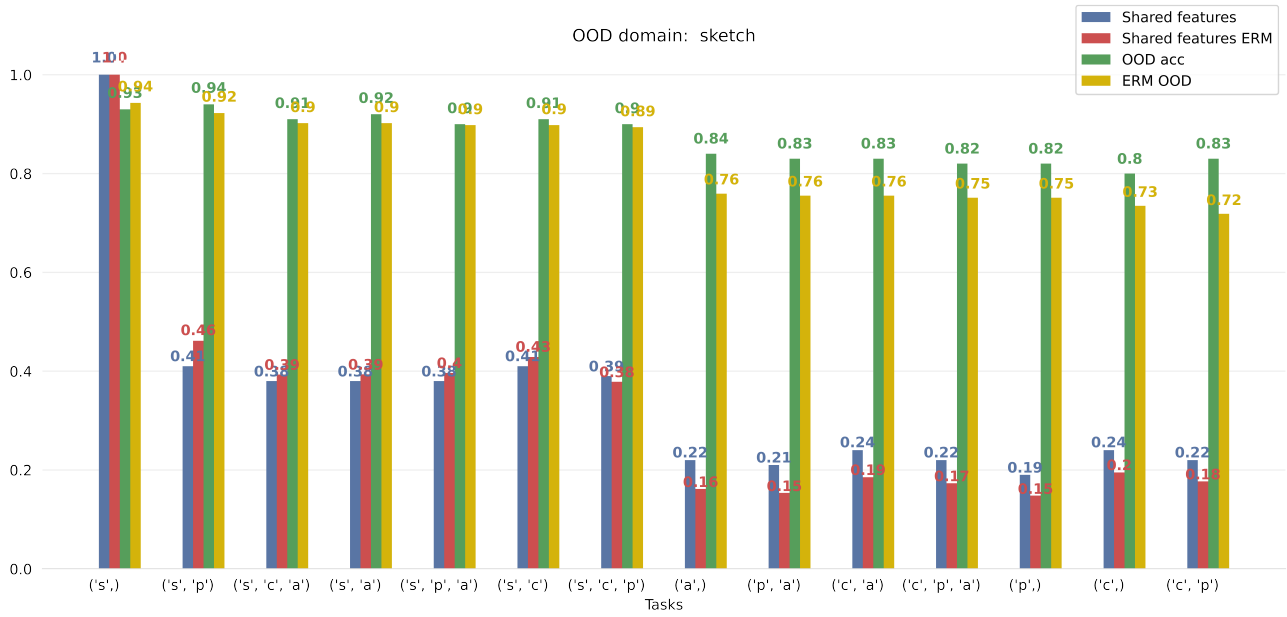


Figure 7: *Fraction of shared features VS accuracy*. Barplot of OOD accuracies on the *Sketch* domain for our model (green) and ERM (yellow) on the 14 tasks sampled from PACS, along with the fraction of shared features with the OOD domain for each task (blue for our model, red for ERM). Each task is sampled from a single domain or from the intersections of domains. Tasks are labelled according to the sampling domain on the x axis. The fraction of shared features and OOD accuracy have a correlation coefficient of 97.5.

F.3. CivilComments

See Table 9 for the quantitative results accompanying to Figure **??** in the paper and 10 for result on groups on the civil comments dataset.

Table 9: *Quantitative results on CivilComments*: we report the accuracy on test averaged across all demographic groups (*left*), and the worst group accuracy (*right*). We show that our method performs similarly in terms of average accuracy and outperforms in terms of worst group accuracy, without using any knowledge on the group composition in the training data. This Table accompanies Figure **??**

	avg acc	worst group acc
ERM	**92.2**	56.5
DRO	90.2	69
Ours	91.2 $\pm$ 0.2	**75.45** $\pm$ 0.1

F.4. Full results Domain generalization

We report here comparison with several methods in the domain generalization literature, namely (Yan et al., 2020; Blanchard et al., 2021; Li et al., 2018a;b; Ganin et al., 2016; Li et al., 2018c; Nam et al., 2021; Zhang et al., 2021; Huang et al., 2020;

Figure 8: Barplot of feature usage (number of activated features) for each task for our model (blue) and ERM model (green) referring to the experiment in Figure 7. Our method uses fewer features than ERM while also generalizing better.

Table 10: Civilcomments quantitative results pergroup.

	Male	Female	LGBTQ	Christian	Muslim	Other religion	Black	White
GroupDRO								
Toxic	75.1 ± 2.1	73.7 ± 1.5	73.7 ± 4	69.2 ± 2.0	72.1 ± 2.6	72.0 ± 2.5	79.6 ± 2.2	78.8 ± 1.7
Non Toxic	88.4 ± 0.7	90.0 ± 0.6	76.0 ± 3.6	92.6 ± 0.6	80.7 ± 1.9	87.4 ± 0.9	72.2 ± 2.3	73.4 ± 1.4
Ours								
Toxic	87.94 ± 0.07	89.17 ± 0.05	77.25 ± 0.16	92.25 ± 0.16	80.6 ± 0.29	87.79 ± 0.26	75.45 ± 0.17	78.35 ± 0.02
Non toxic	91.62 ± 0.11	91.52 ± 0.11	91.71 ± 0.16	91.11 ± 0.1	91.81 ± 0.12	91.32 ± 0.1	90.82 ± 0.12	92.04 ± 0.11

Krueger et al., 2021).

F.4.1. VLCS

Algorithm	**C**	**L**	**S**	**V**	**Avg**
ERM	97.7 ± 0.4	64.3 ± 0.9	73.4 ± 0.5	74.6 ± 1.3	77.5
IRM	98.6 ± 0.1	64.9 ± 0.9	73.4 ± 0.6	77.3 ± 0.9	78.5
GroupDRO	97.3 ± 0.3	63.4 ± 0.9	69.5 ± 0.8	76.7 ± 0.7	76.7
Mixup	98.3 ± 0.6	64.8 ± 1.0	72.1 ± 0.5	74.3 ± 0.8	77.4
MLDG	97.4 ± 0.2	65.2 ± 0.7	71.0 ± 1.4	75.3 ± 1.0	77.2
CORAL	98.3 ± 0.1	66.1 ± 1.2	73.4 ± 0.3	77.5 ± 1.2	78.8
MMD	97.7 ± 0.1	64.0 ± 1.1	72.8 ± 0.2	75.3 ± 3.3	77.5
DANN	99.0 ± 0.3	65.1 ± 1.4	73.1 ± 0.3	77.2 ± 0.6	78.6
CDANN	97.1 ± 0.3	65.1 ± 1.2	70.7 ± 0.8	77.1 ± 1.5	77.5
MTL	97.8 ± 0.4	64.3 ± 0.3	71.5 ± 0.7	75.3 ± 1.7	77.2
SagNet	97.9 ± 0.4	64.5 ± 0.5	71.4 ± 1.3	77.5 ± 0.5	77.8
ARM	98.7 ± 0.2	63.6 ± 0.7	71.3 ± 1.2	76.7 ± 0.6	77.6
VREx	98.4 ± 0.3	64.4 ± 1.4	74.1 ± 0.4	76.2 ± 1.3	78.3
RSC	97.9 ± 0.1	62.5 ± 0.7	72.3 ± 1.2	75.6 ± 0.8	77.1
Ours	98.1 ± 0.2	63.4 ± 0.5	78.2 ± 0.7	73.9 ± 0.8	78.4

F.4.2. PACS

Algorithm	A	C	P	S	Avg
ERM	84.7 ± 0.4	80.8 ± 0.6	97.2 ± 0.3	79.3 ± 1.0	85.5
IRM	84.8 ± 1.3	76.4 ± 1.1	96.7 ± 0.6	76.1 ± 1.0	83.5
GroupDRO	83.5 ± 0.9	79.1 ± 0.6	96.7 ± 0.3	78.3 ± 2.0	84.4
Mixup	86.1 ± 0.5	78.9 ± 0.8	97.6 ± 0.1	75.8 ± 1.8	84.6
MLDG	85.5 ± 1.4	80.1 ± 1.7	97.4 ± 0.3	76.6 ± 1.1	84.9
CORAL	88.3 ± 0.2	80.0 ± 0.5	97.5 ± 0.3	78.8 ± 1.3	86.2
MMD	86.1 ± 1.4	79.4 ± 0.9	96.6 ± 0.2	76.5 ± 0.5	84.6
DANN	86.4 ± 0.8	77.4 ± 0.8	97.3 ± 0.4	73.5 ± 2.3	83.6
CDANN	84.6 ± 1.8	75.5 ± 0.9	96.8 ± 0.3	73.5 ± 0.6	82.6
MTL	87.5 ± 0.8	77.1 ± 0.5	96.4 ± 0.8	77.3 ± 1.8	84.6
SagNet	87.4 ± 1.0	80.7 ± 0.6	97.1 ± 0.1	80.0 ± 0.4	86.3
ARM	86.8 ± 0.6	76.8 ± 0.5	97.4 ± 0.3	79.3 ± 1.2	85.1
VREx	86.0 ± 1.6	79.1 ± 0.6	96.9 ± 0.5	77.7 ± 1.7	84.9
RSC	85.4 ± 0.8	79.7 ± 1.8	97.6 ± 0.3	78.2 ± 1.2	85.2
Ours	83.1 ± 0.1	86.7± 0.8	97.8 ± 0.1	83.5 ± 0.1	87.5

F.4.3. OFFICEHOME

Algorithm	A	C	P	R	Avg
ERM	61.3 ± 0.7	52.4 ± 0.3	75.8 ± 0.1	76.6 ± 0.3	66.5
IRM	58.9 ± 2.3	52.2 ± 1.6	72.1 ± 2.9	74.0 ± 2.5	64.3
GroupDRO	60.4 ± 0.7	52.7 ± 1.0	75.0 ± 0.7	76.0 ± 0.7	66.0
Mixup	62.4 ± 0.8	54.8 ± 0.6	76.9 ± 0.3	78.3 ± 0.2	68.1
MLDG	61.5 ± 0.9	53.2 ± 0.6	75.0 ± 1.2	77.5 ± 0.4	66.8
CORAL	65.3 ± 0.4	54.4 ± 0.5	76.5 ± 0.1	78.4 ± 0.5	68.7
MMD	60.4 ± 0.2	53.3 ± 0.3	74.3 ± 0.1	77.4 ± 0.6	66.3
DANN	59.9 ± 1.3	53.0 ± 0.3	73.6 ± 0.7	76.9 ± 0.5	65.9
CDANN	61.5 ± 1.4	50.4 ± 2.4	74.4 ± 0.9	76.6 ± 0.8	65.8
MTL	61.5 ± 0.7	52.4 ± 0.6	74.9 ± 0.4	76.8 ± 0.4	66.4
SagNet	63.4 ± 0.2	54.8 ± 0.4	75.8 ± 0.4	78.3 ± 0.3	68.1
ARM	58.9 ± 0.8	51.0 ± 0.5	74.1 ± 0.1	75.2 ± 0.3	64.8
VREx	60.7 ± 0.9	53.0 ± 0.9	75.3 ± 0.1	76.6 ± 0.5	66.4
RSC	60.7 ± 1.4	51.4 ± 0.3	74.8 ± 1.1	75.1 ± 1.3	65.5
Ours	56.3± 0.1	66.7 ± 0.7	79.2± 0.5	81.3 ± 0.4	70.9

F.5. Few-shot transfer learning

F.6. Few-shot transfer learning.

We test the adaptability of the feature space to new domains with limited labeled samples in few-shot transfer learning setting in Append F.5. For transfer learning tasks, we fit a linear head using the available limited supervised data. The sparsity penalty α is set to the value used in training; the feature sharing parameter β is defaulted to zero unless specified. We compare the results with ERM in Table 11, averaged by domains in each benchmark dataset. The full scores for each domain are in Appendix F.5 for 1-shot, 5-shot, and 10-shot setting, reporting the mean accuracy and standard deviations over 100 draws. Our approach achieves consistently higher accuracy than ERM, showing the better adaptation capabilities of our minimal and sufficiently sparse feature space.

Results on few-shot transfer learning on datasets `PACS`,`VLCS`,`OfficeHome`,`Waterbirds` in Tables 12,13,14 and 15.

F.7. Feature sharing on `PACS`

See Figure 9 for additional results on all domains in `PACS`.

Table 11: Quantitative results for few-shot transfer learning, with our method consistently outperforming ERM across all sample sizes and data sets.

N-shot/Algorithm	**OOD accuracy (averaged by domains)**			
1-shot	PACS	VLCS	OfficeHome	Waterbirds
ERM	80.5	59.7	56.4	79.8
Ours	**81.5**	**68.2**	**58.4**	**88.4**
5-shot				
ERM	87.1	71.7	75.7	79.8
Ours	**88.3**	**74.5**	**77.0**	**87.6**
10-shot				
ERM	87.9	74.0	81.0	84.2
Ours	**90.4**	**77.3**	**82.0**	**89.2**

Table 12: Results few-shot transfer learning on PACS

Dataset/Algorithm	**OOD accuracy (by domain)**				
PACS 1-shot	S	A	P	C	Average
ERM	72.3 ± 0.3	80.4 ± 0.09	93.3 ± 4.1	75.8± 2.6	80.5
Ours	**75.4** ± 3	**81.7**± 0.8	**98.0** ± 0.8	**71** ± 5.2	**81.5**
PACS 5-shot	S	P	A	C	Average
ERM	84.9± 1.1	85.7 ± 0.08	98.6 ± 0.0	79.1 ± 0.9	87.1
Ours	**85.0** ± 0.1	**86.7**± 0.8	**97.8** ± 0.1	**83.5** ± 0.1	**88.3**
PACS 10-shot	S	P	A	C	Average
ERM	81.0 ± 0.1	88.9 ± 0.1	97.4 ± 0.0	84.2 ± 0.9	87.9
Ours	**86.2** ± 0.5	**90.0** ± 0.8	**98.9** ± 0.1	**86.6** ± 0.1	**90.4**

Table 13: results few-shot transfer learning on VLCS

Dataset/Algorithm	**OOD accuracy (by domain)**				
VLCS 1-shot	C	L	V	S	Average
ERM	98.9 ± 0.4	32.7 ± 16.2	59.8 ± 10.7	47.5 ± 11.2	59.7
Ours	**98.6** ± 0.3	**51.0** ± 4.9	**61.2** ± 9.8	**61.9** ± 9.7	**68.2**
VLCS 5-shot	C	L	V	S	Average
ERM	99.4 ± 0.2	50.0 ± 6.2	71.9 ± 3.2	65.3 ± 2.8	71.7
Ours	**98.9** ± 0.4	**56.0** ± 6.2	**73.4** ± 1.4	**69.8** ± 2.0	**74.5**
VLCS 10-shot	C	L	V	S	Average
ERM	99.5 ± 0.2	52.6 ± 5.0	74.8 ± 3.8	69.1 ± 2.4	74.0
Ours	**99.1** ± 0.2	**65.0** ± 6.2	**74.4** ± 1.9	**70.8** ± 2.3	**77.3**

F.8. Task similarity

We show that our method enables direct extraction of a task representation and a metric for task similarity from our model and its feature space. We propose to use the coefficients of the fitted linear heads $f_{\phi_t^*}$ on a given task as a *representation for that task*. Specifically we transform the optimal coefficients ϕ^* in a M-dimensional vector space (here M is the number of features) by simply computing $\sum_c |\phi^*_{t,m,c}|$, and discretize them by a threshold ϵ. The resulting binary vectors, together with

Table 14: results few-shot transfer learning on OfficeHome

Dataset/Algorithm	**OOD accuracy (by domain)**				
OfficeHome 1-shot	C	A	P	R	Average
ERM	40.2 ± 2.4	52.7 ± 2.6	68.1 ± 1.7	64.6 ± 1.8	56.4
Ours	**41.4** ± 1.7	**54.5** ± 2.0	**68.5** ± 2.7	**69.0** ± 1.5	**58.4**
OfficeHome 5-shot	C	A	P	R	Average
ERM	63.2 ± 0.4	73.3 ± 0.8	84.1 ± 0.4	82.0 ± 0.8	75.7
Ours	**66.2** ± 1.2	**75.1** ± 1.0	**83.6** ± 0.5	**83.1** ± 0.8	**77.0**
OfficeHome 10-shot	C	A	P	R	Average
ERM	71.1 ± 0.4	80.5 ± 0.5	87.5 ± 0.3	84.9 ± 0.5	81.0
Ours	**72.2** ± 1.2	**81.8** ± 0.5	**87.5** ± 0.2	**86.3** ± 0.4	**82.0**

Table 15: results few-shot transfer learning Waterbirds

Dataset/Algorithm	**OOD accuracy (by domain)**				
Waterbirds 1-shot	LL	LW	WL	WW	Average
ERM	99.1 ± 1.1	43.8 ± 16.5	79.5 ± 10.2	86.7 ± 8.2	79.8
Ours	**95.2** ± 8.1	**81.9** ± 9.5	**80.7** ± 5.5	**95.9** ± 1.2	**88.4**
Waterbirds 5-shot	LL	LW	WL	WW	Average
ERM	96.3 ± 5.0	58.7 ± 17.2	80.1 ± 12.6	84.1 ± 12.7	79.8
Ours	**98.8** ± 1.8	**75.4** ± 9.0	**81.6** ± 14.0	**94.8** ± 1.8	**87.6**
Waterbirds 10-shot	LL	LW	WL	WW	Average
ERM	94.2 ± 4.2	73.0 ± 11.6	80.4 ± 6.3	89.3 ± 3.3	84.2
Ours	**98.2** ± 0.9	**82.6** ± 5.9	**80.7** ± 6.3	**95.5** ± 1.4	**89.2**

a distance metric (we choose the Hamming distance), form a discrete metric space of tasks. We preliminary verify how the proposed representation and metric behave on `MiniImagenet` (Vinyals et al., 2016) below.

We sample 160 tasks from 10 groups from , where each group has the same class support, i.e. $t_1, t_2 \in G_i \mapsto Supp(t_1) == Supp(t_2) \forall i$. We then fit the linear heads independently on each task (i.e. not using the feature sharing regularizer). Then we compute the discrete task representation and project the resulting vector space in a two dimensional vector space using tSNE (Wattenberg et al., 2016). The clusters obtained in this space correspond exactly to the group identities (visualized in color in Figure 10).

F.9. Comparison with metalearning baselines

In Table 16, we further compare our method on meta learning benchmarks, namely `Mini Imagenet` (Vinyals et al., 2016) and CIFAR-FS (Bertinetto et al., 2019) with different approaches in the literature based on meta learning (Snell et al., 2017; Oreshkin et al., 2018; Dhillon et al., 2020; Lachapelle et al., 2022a).

In Figure 11 we compare the predicting performance of our method and capacity to leverage shared knowledge between task, comparing with backbone trained with protopical network approach. We sample a set of task with different overlap, where the overlap between two task t_1, t_2 is defined as $sim(t_1, t_2) = \frac{Supp(t_1) \cap Supp(t_2)}{Supp(t_1) \cup Supp(t_2}$ indicating with $Supp(t_i)$ the support over classes in task t_i. We show that other than reaching a much higher accuracy the features of our model are able to be clustered at test time enabling to reach better performance on unseen task. As a matter of fact we can use the feature sharing regularizer at test time showing that there is a increasing trend in the performance, while the prototypical networks features just decreases being unable to share information across tasks at test time.

Table 16: Meta learning baselines, including concurrent work (Lachapelle et al., 2022a) which we significantly outperform.

	Architecture	Cifar-FS (1 shot)	Cifar-FS(5 shot)	MiniImagenet(1 shot)	MiniImagenet (5 shot)
MAML	Conv32(x4)	-	-	48.7±1.84	63.11±0.66
Prototypical Net	Conv64(x4)	-	-	49.42±0.78	68.20±0.66
TADAM	ResNet12	-	-	58.5 ±0.56	76.7 ±0.3
MetaOptNet	ResNet12	72.0 ± 0.7	84.2 ± 0.5	**62.64**±0.61	**78.63**±0.46
MetaBaseline	WRN 28-10	**76.58**±0.68	85.79±0.5	59.62 ±0.66	78.17 ±0.49
Lachapelle et al(Lachapelle et al., 2022a)	ResNet12	-	-	54.22 ± 0.6	70.01 ± 0.51
Ours*	ResNet12	75.1 ±0.4	**86.9** ±0.19	60.1 ± 2	76.6 ± 0.1

F.10. Sharing features at test time

Features can be enforced to be shared also at test time, simply by setting $\beta > 0$ to fit the linear head on top of the learned feature space. We observe the benefits of utilizing the feature sharing penalty at test time on the `Camelyon17` dataset in the fourth row of Table 17.

As highlighted in the main paper, retaining features which are shared across the training domains and cutting the ones that are domain-specific enable to perform better at test time, at the expenses of lower performance near the training distribution.

We analyzed in more depth this phenomenon in Figure 11. For this experiment we trained our model and a Prototypical network (Snell et al., 2017) one on the `MiniImagenet` dataset. Then we sampled 5 groups of tasks according to an average overlap measure between tasks. Between two task t_1, t_2 the overlap is defined as $sim(t_1, t_2) = \frac{Supp(t_1) \cap Supp(t_2)}{Supp(t_1) \cup Supp(t_2}$. each group is made of 10 task. We then plot the performance at test time increasing the regularization parameter β, weighting the feature sharing. The outcome of the experiment is twofold: (i) we observe an increase in performance at test time, especially when tasks shows maximal overlap (i.e. they share more features) (ii) this is not the case with the pretrained backbone of (Snell et al., 2017) which shows almost monotonical decrease in the performance, i.e. enforcing the minimality property during training enables to use it as well at test time.

Further analysis on different datasets, and also on tuning strategies on the regularization parameter are promising directions for future work, to better understand when and how enforcing feature sharing is beneficial at test time.

Table 17: Camelyon17 quantitative results: we report accuracy both on ID and OOD splits. We show (last row) that feature sharing at test time, leads to more robust features on OOD test data.

	Validation(ID)	Validation (OOD)	Test (OOD)
ERM	93.2	84	70.3
CORAL	95.4	86.2	59.5
IRM	91.6	86.2	64.2
Ours	**93.2**±0.3	**89.9**±0.6	74.1±0.2
Ours($\beta > 0$ test)	90.4±0.2	84.01±0.9	**85.5**±0.6

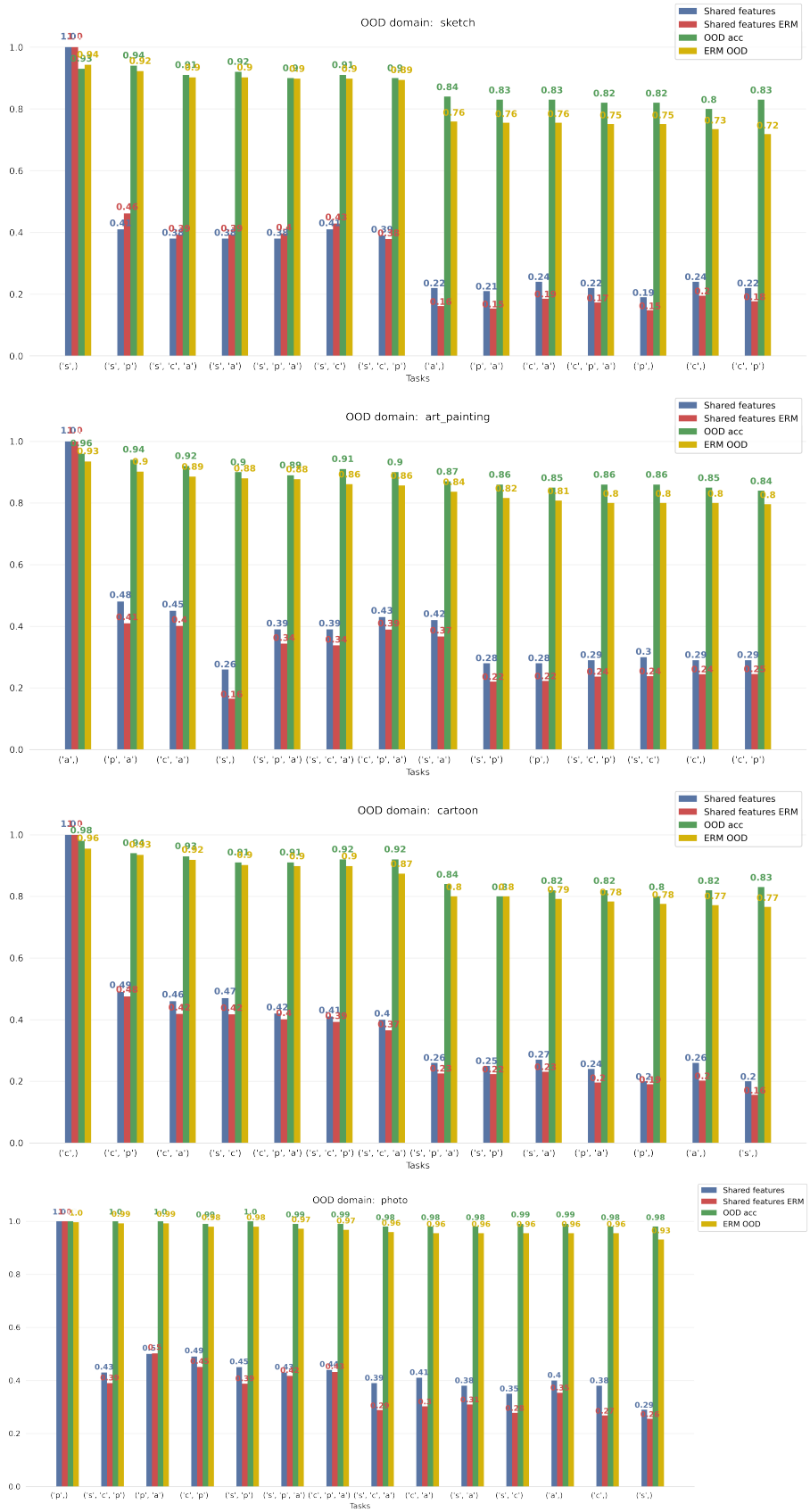


Figure 9: Additional results for all domains in PACS, separated by domain. The overall message of Figure 7 appear consistent across all domains.

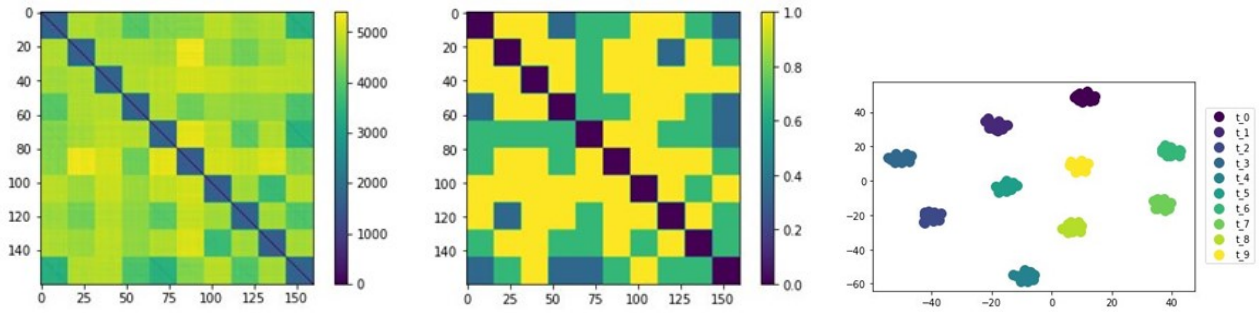


Figure 10: *Task Similarity.* We visualize the tSNE of the discrete task representation and observe that the clusters in this space corresponds to group identities.

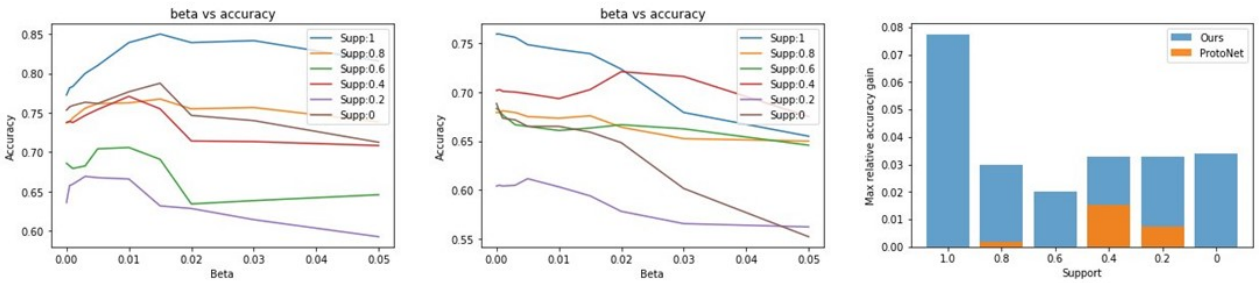


Figure 11: Enforcing feature sharing at test time. Our approach (on the left) is able to benefit from the feature sharing constraint at test time, while using the prototypical network backbone performance monotonically decrease (center). On the right we show the maximal performance gain for each group of tasks for the two approaches.