Appendix

We present the results of additional experiments in Appendix A followed by details of datasets used in our work along with implementation details of our algorithm and baselines in Appendix B.

A. Additional experiments

Here we present the results omitted in the main paper due to space limitations. In App. A.1, we provide additional empirical results for the comparison between TT-NSS and the confidence-based abstaining mechanism on the VLCS dataset and the multi-source domain setting. In App. A.2, we provide additional empirical results demonstrating the effectiveness of models trained with NSS when evaluated with TT-NSS and the confidence-based abstaining mechanism on different datasets in both single and multi-domain settings. Finally, in App. A.3, we show the effect of using different values of n in TT-NSS.

A.1. Additional results on the comparison of TT-NSS and confidence-based abstaining

Here we present results on the comparison of TT-NSS and confidence-based abstaining using the AUC metric, and present results on the VLCS dataset both in single and multi-domain settings. Our results in Tables 3, 4 and 5, 6 show that similar to the results presented in Fig. 2 in the main paper, the AUC for the accuracy versus the percentage of abstained samples curve is significantly better for TT-NSS compared to confidence-based abstaining in the single domain setting and is competitive on the multi-domain setting. The advantage of TT-NSS becomes clear when evaluated on data from Wikiart and corrupted domains. This advantage of TT-NSS holds regardless of the training method used for training the DG classifier or the dataset used.

In Figs. 6 and 7, we show the full accuracy versus percentage of abstained sample curves for classifiers trained on PACS and VLCS dataset in both the single and multi-domain setting. The results show that the performance of the DG classifier when evaluated with TT-NSS remains better or competitive with the performance of the confidence-based abstaining method for most domains and most of the range of abstaining rates.

A.2. Additional results on the effectiveness of NSS at improving risk-averse predictions

In this section, we present additional empirical results on the effectiveness of training DG models with NSS (combined with ERM) on different datasets and settings. Similar to the results in Sec. 4.2 in the main paper, we observe that models trained with NSS achieve consistently better AUC than models trained with ERM on different variants of PACS, VLCS, and OfficeHome datasets as shown in Table 7. The highest improvement is observed when classifiers are evaluated on test sets corrupted with severity 5 corruptions. Similar to Fig. 3, we observe in Fig. 8, 9, 10 that NSS trained models achieve better accuracy on non-abstained samples on most domains compared to the models trained with ERM. Incorporating NSS with ERM makes the performance similar to that of other SOTA DG methods such as RSC and SagNet. Due to the versatility of NSS to be combined with any DG method, training classifiers with RSC and SagNet in conjunction with NSS could lead to further improvement in the accuracy of the classifier trained with these SOTA DG methods on non-abstained samples when evaluated with TT-NSS. Lastly, classifiers trained with NSS also perform better in terms of risk-averse predictions when using the confidence-based abstaining mechanism as shown in Tables 8 and 9. As mentioned in Sec. 4.2, TT-NSS remains superior in the presence of severe shifts such as those induced by adding severity 5 corruptions for all the datasets in both single and multi-domain settings.

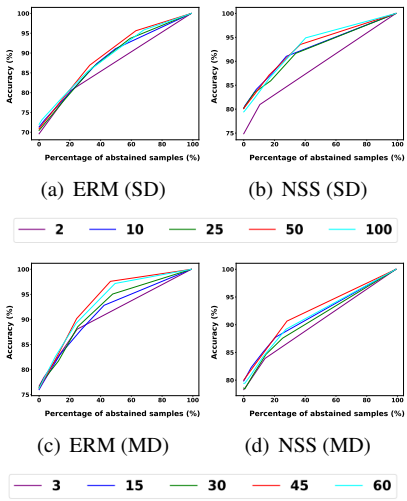


Figure 5. The performance of TT-NSS is not significantly affected by the value of n beyond $n = 10$ for single (SD) (a, b) and $n = 15$ in multiple (MD) (c, d) source domain settings. For the SD setting, the classifier is trained on the Cartoon domain and evaluated on the rest, and for the MD setting, the classifier is evaluated on the Cartoon domain after training on the rest of the domains in PACS.

A.3. Effect of number of styles

Here we evaluate the effect of using different number of re-stylizations of a single test image, n, in TT-NSS using a subsample (see App. B.2) of the PACS dataset (original style). Results in Fig. 5 show that in both single and multi-source domain settings, using a large value of n leads to only a small improvement in the accuracy of non-abstained samples at higher abstaining rates whereas performance at

lower abstaining rates remains similar for different values of n. Since using a larger value of n can slow down the inference, we use n as 10 and 15 (5 per domain) in the single and multiple source domain settings. Evaluating a single test sample with TT-NSS using 15 styles increases the inference cost by a mere 0.26 seconds on our hardware, showing the potential of TT-NSS to produce risk-averse predictions without sacrificing inference efficiency.

B. Dataset and experimental details

All codes were written in Python using Tensorflow/Pytorch and were run on an AMD EPYC 7J13 CPU with 200 GB of RAM and an Nvidia A100 GPU. Implementation and hyperparameters are described below.

B.1. Dataset description

In this work, we use three popular benchmark datasets along with their stylized and corrupted versions to evaluate the performance of various methods. For single source domain setting, we use 90% of the data for training and 10% for hyperparameter tuning, and for multi-domain setting, we use 80% of the data for training and 20% for hyperparameter tuning.

PACS [47]: This dataset contains images from four domains Art, Cartoons, Photos, and Sketches. It contains 9991 images belonging to 7 different classes.

VLCS [22]: This dataset contains images from four domains Caltech101, LabelMe, SUN09, PASCAL VOC 2007. It contains 10729 images belonging to 5 different classes.

Office-Home [71]: This dataset contains images from four domains Art, Clipart, Product, and Real. It contains 15588 images belonging to 65 different classes.

B.2. Details of the subsample used for reporting the evaluation results in App. A.3

As mentioned in Sec. 4, we use a subsample of the PACS, VLCS, and OfficeHome datasets to present the results of using TT-NSS and confidence-based abstaining on corrupted variants of the datasets, and for the experiment in App. A.3 with different values of n in TT-NSS. For reporting the results on the corrupted version of the dataset we used 10 images per class from VLCS/PACS and 2 images per class from the OfficeHome dataset. We report average results over all 10 corruption types for this experiment.

For the experiment in App. A.3 we used the following subsample. For the single source domain setting, we report the results on a balanced subsample of the dataset containing 50 images from each class and each target domain for PACS. For the multi-domain setting, we use 100 images for each class of the target domain for PACS. For classes with fewer samples, we use all the samples from that class

Table 2. Results on single and multi-domain generalization settings using ResNet50 as the backbone on the PACS dataset using RSC [37] and SagNet [55]. The original work, RSC [37], only reports multi-domain results (presented without *) while SagNet [55], only reported results based on the ResNet-18 backbone in the original paper. We used their official implementation using ResNet-50 as the backbone to obtain results for both single and multi-domain settings (reported with *) (see details in Appendix B.3.1).

DG Setting	Methods	A	C	P	S	Avg.
Single	RSC*	72.55	77.30	47.88	57.54	63.82
	SagNet*	77.45	78.36	52.39	53.96	65.54
Multi	RSC	87.89	82.16	97.92	83.35	87.83
	RSC*	85.79	79.60	95.03	81.52	85.49
	SagNet*	86.00	81.29	97.47	80.72	86.37

B.3. Experimental details

B.3.1 Reproducing the baselines

For the RSC [37] method, we independently run the code using the official implementation published by the authors, using different configurations (https://github.com/DeLightCMU/RSC). We trained both multi-domain and single-domain RSC [37] classifiers with the same hyperparameters except for smaller batch size 2 and a learning rate of 0.0001 on one random seed. For the SagNet [55], we reproduce their open-source implementation code with the default configuration on three different random seeds (https://github.com/hyeonseobnam/sagnet). We use the official train and test split of PACS for all three methods. Table 2 shows our reproduced results and the results the authors reported in their papers.

B.3.2 Training classifiers with NSS

To train the classifiers with NSS, we incorporate style augmentation and style consistency losses computed on stylized versions of the source domain images generated through the AdaIN decoder. We additionally incorporate the ERM training loss which minimizes the misclassification of original source domain samples. As mentioned in Sec. 3 other losses used in specific DG algorithms can also be incorporated to improve the quality of risk-averse predictions from classifiers trained with those methods. To compute the style consistency loss we use four different styles for every sample in the batch and use a batch size of 16. These losses are then used to fine-tune the ResNet50 backbone augmented with a fully connected layer used for classification. For the multi-domain setting, the classifier that achieves the highest accuracy on the validation set is used for final evaluation whereas for the single source domain setting, the classifier at the last step is used for final evaluation.

Table 3. Comparison of the area under the accuracy versus percentage of abstained samples curve for TT-NSS and the confidence-based abstaining mechanism in a **single** domain setting on different variations of the **PACS** dataset. The training domain is denoted in the columns.

		A	C	P	S
Alg.	Evaluation	Original Style			
ERM	Confidence	**0.882**	0.875	0.634	0.707
	TT-NSS	0.875	0.878	**0.662**	0.702
RSC	Confidence	**0.892**	0.899	**0.705**	0.779
	TT-NSS	0.858	**0.912**	0.682	**0.794**
SagNet	Confidence	**0.913**	**0.91**	**0.741**	0.758
	TT-NSS	0.889	0.88	0.726	**0.771**
		Wikiart Style			
ERM	Confidence	0.84	0.757	0.609	0.558
	TT-NSS	**0.854**	**0.816**	**0.643**	**0.626**
RSC	Confidence	0.823	0.766	0.63	0.662
	TT-NSS	**0.835**	**0.887**	**0.654**	**0.733**
SagNet	Confidence	0.871	0.8	0.683	0.61
	TT-NSS	0.875	**0.813**	**0.692**	**0.718**
		Corrupted with severity 3			
ERM	Confidence	0.832	0.709	0.613	**0.612**
	TT-NSS	**0.886**	**0.812**	**0.622**	0.545
RSC	Confidence	0.871	0.667	0.673	0.62
	TT-NSS	**0.901**	**0.86**	**0.682**	**0.699**
SagNet	Confidence	0.903	0.78	0.725	0.629
	TT-NSS	0.901	**0.794**	**0.731**	**0.667**
		Corrupted with severity 5			
ERM	Confidence	0.696	0.579	0.418	**0.479**
	TT-NSS	**0.834**	**0.708**	**0.519**	0.468
RSC	Confidence	0.728	0.449	0.564	0.465
	TT-NSS	**0.863**	**0.776**	**0.626**	**0.613**
SagNet	Confidence	0.786	0.576	0.565	0.485
	TT-NSS	**0.855**	**0.686**	**0.666**	**0.59**

Table 4. Comparison of the area under the accuracy versus percentage of abstained samples curve for TT-NSS and the confidence-based abstaining mechanism in a **single** domain setting on different variations of the **VLCS** dataset. The training domain is denoted in the columns.

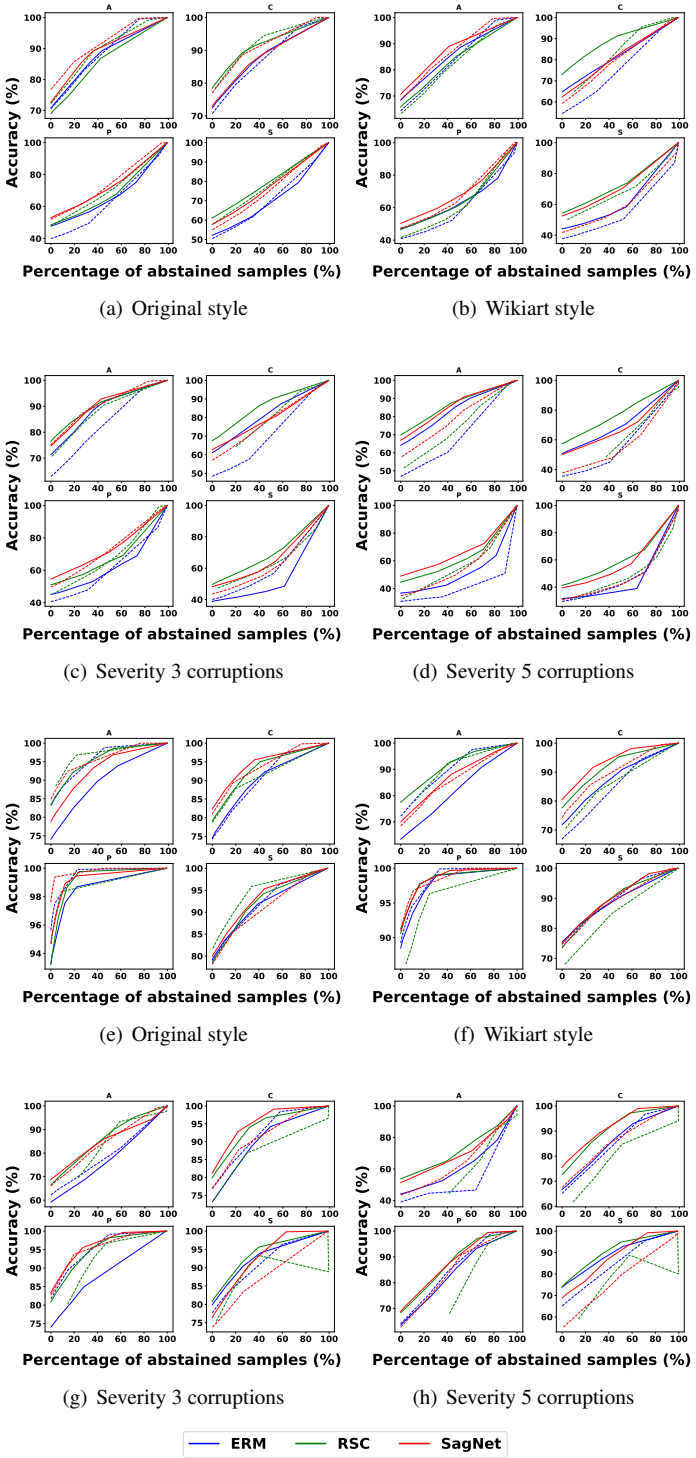
		A	C	P	S
Alg.	Evaluation	Original Style			
ERM	Confidence	**0.653**	0.68	0.806	0.715
	TT-NSS	0.567	**0.724**	**0.851**	**0.751**
		Wikiart Style			
ERM	Confidence	0.426	0.584	0.763	0.679
	TT-NSS	**0.477**	**0.682**	**0.785**	**0.704**
		Corrupted with severity 3			
ERM	Confidence	**0.504**	0.381	**0.734**	0.468
	TT-NSS	0.468	**0.551**	0.689	**0.471**
		Corrupted with severity 5			
ERM	Confidence	**0.433**	0.329	0.563	0.346
	TT-NSS	0.411	**0.439**	0.567	**0.415**

Table 5. Comparison of the area under the accuracy versus percentage of abstained samples curve for TT-NSS and the confidence-based abstaining mechanism in a **multi-**domain setting on different variations of the **PACS** dataset. The domain used for evaluation is denoted in the columns.

		A	C	P	S
Alg.	Evaluation	Original Style			
ERM	Confidence	**0.95**	0.902	0.986	0.915
	TT-NSS	0.893	0.9	0.978	0.911
RSC	Confidence	0.925	0.908	0.978	**0.936**
	TT-NSS	**0.948**	**0.926**	0.983	0.917
SagNet	Confidence	**0.951**	0.932	0.988	0.905
	TT-NSS	0.927	0.939	0.984	**0.925**
		Wikiart Style			
ERM	Confidence	**0.898**	0.85	0.975	0.892
	TT-NSS	0.816	**0.876**	0.97	0.886
RSC	Confidence	0.81	0.842	0.915	0.828
	TT-NSS	**0.911**	**0.916**	**0.976**	**0.891**
SagNet	Confidence	0.858	0.898	0.977	0.886
	TT-NSS	0.869	**0.933**	0.977	0.897
		Corrupted with severity 3			
ERM	Confidence	**0.79**	**0.918**	**0.947**	0.909
	TT-NSS	0.771	0.898	0.878	**0.923**
RSC	Confidence	0.673	0.868	0.802	0.851
	TT-NSS	**0.856**	**0.934**	**0.941**	**0.933**
SagNet	Confidence	0.842	0.913	0.948	0.873
	TT-NSS	0.845	**0.948**	0.953	**0.924**
		Corrupted with severity 5			
ERM	Confidence	0.539	0.85	**0.852**	0.845
	TT-NSS	**0.621**	0.856	0.837	**0.888**
RSC	Confidence	0.405	0.734	0.505	0.673
	TT-NSS	**0.719**	**0.904**	**0.875**	**0.903**
SagNet	Confidence	0.649	0.855	0.845	0.764
	TT-NSS	**0.696**	**0.914**	**0.878**	**0.877**

Table 6. Comparison of the area under the accuracy versus percentage of abstained samples curve for TT-NSS and the confidence-based abstaining mechanism in a **multi-**domain setting on different variations of the **VLCS** dataset. The domain used for evaluation is denoted in the columns.

		A	C	P	S
Alg.	Evaluation	Original Style			
ERM	Confidence	0.986	0.752	**0.88**	**0.831**
	TT-NSS	0.968	**0.772**	0.86	0.776
		Wikiart Style			
ERM	Confidence	**0.954**	0.747	0.815	**0.691**
	TT-NSS	0.941	0.744	**0.822**	0.678
		Corrupted with severity 3			
ERM	Confidence	**0.908**	**0.601**	0.678	**0.599**
	TT-NSS	0.785	0.553	**0.692**	0.476
		Corrupted with severity 5			
ERM	Confidence	**0.775**	**0.526**	0.483	**0.427**
	TT-NSS	0.626	0.477	**0.54**	0.388



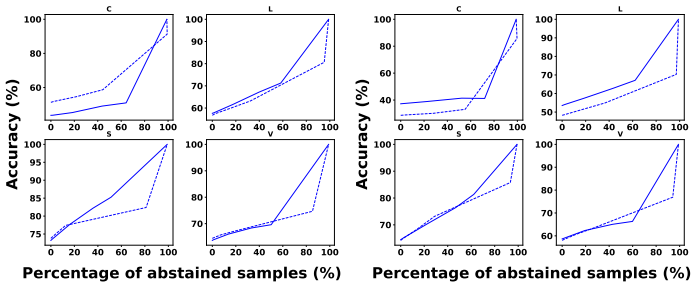
(a) Original style (b) Wikiart style

(c) Severity 3 corruptions (d) Severity 5 corruptions

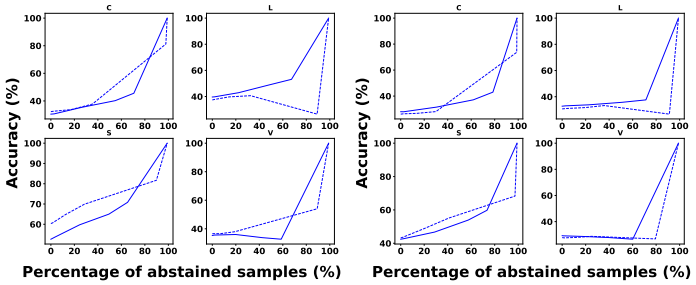
(e) Original style (f) Wikiart style

(g) Severity 3 corruptions (h) Severity 5 corruptions

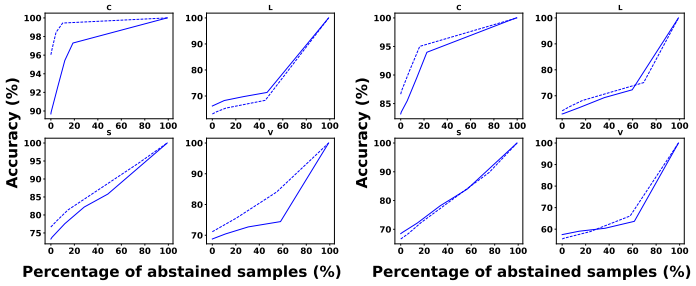
Figure 6. Comparison of TT-NSS (solid lines) and confidence-based method (dashed lines) in a **single** (a-d) and **multi**-source (e-h) domain setup on classifiers trained with ERM. The graphs show accuracy vs. abstained points on different variants of the **PACS** dataset ((a,e) original, (b,f) wikiart, (c,d,g,h) corrupted), and different source/target domains. In most domains, the accuracy of the TT-NSS (solid line) is similar to or better than the corresponding accuracy of the confidence-based method (dashed line) for most of the range of the percentage of abstained samples. (Note: The source domain from PACS used for training is denoted in the title and the target domain used for evaluation is denoted in the title in the bottom row.)



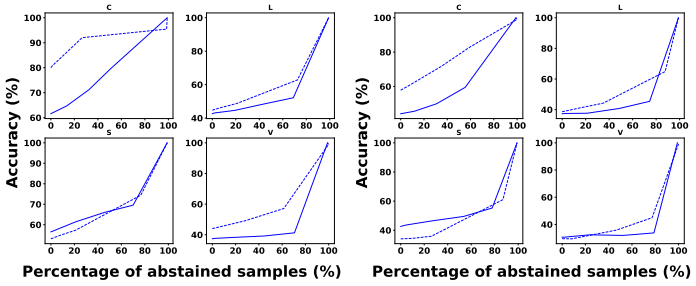
(a) Original style (b) Wikiart style



(c) Severity 3 corruptions (d) Severity 5 corruptions

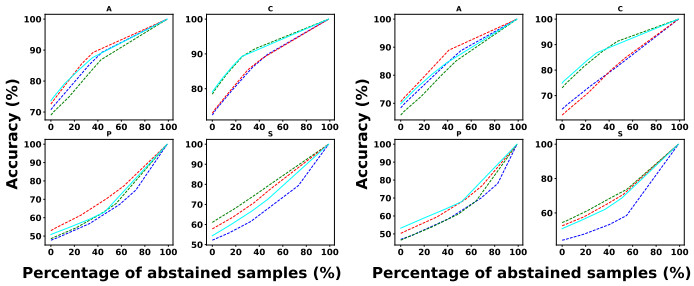


(e) Original style (f) Wikiart style

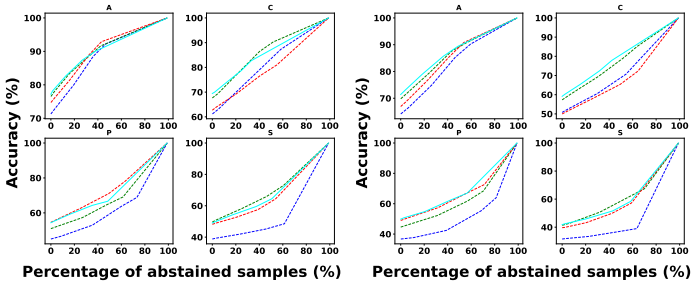


(g) Severity 3 corruptions (h) Severity 5 corruptions

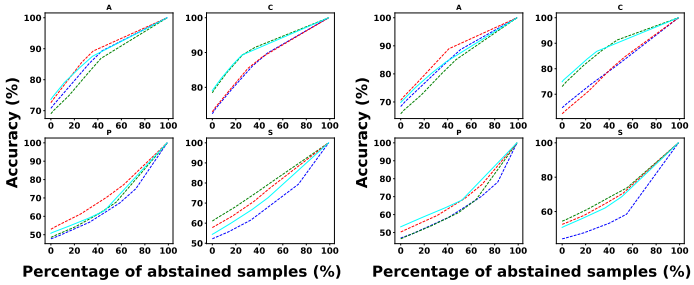
Figure 7. Comparison of TT-NSS (solid lines) and confidence-based method (dashed lines) in a **single** (a-d) and **multiple** (e-h) source domain setup on classifiers trained with ERM. The graphs show accuracy vs. abstained points on different variants of the **VLCS** dataset ((a,e) original, (b,f) wikiart, (c,d,g,h) corrupted), and different source/target domains. In most domains, the accuracy of the TT-NSS (solid line) is similar to or better than the corresponding accuracy of the confidence-based method (dashed line) for most of the range of the percentage of abstained samples. (Note: The source domain from VLCS used for training is denoted in the title in the top row and the target domain used for evaluation is denoted in the title in the bottom row.)



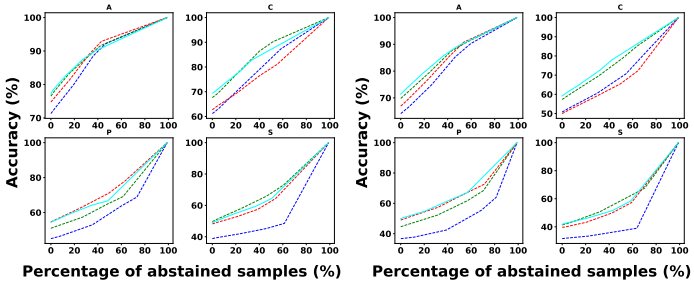
(a) Original style (b) Wikiart style



(c) Severity 3 corruptions (d) Severity 5 corruptions

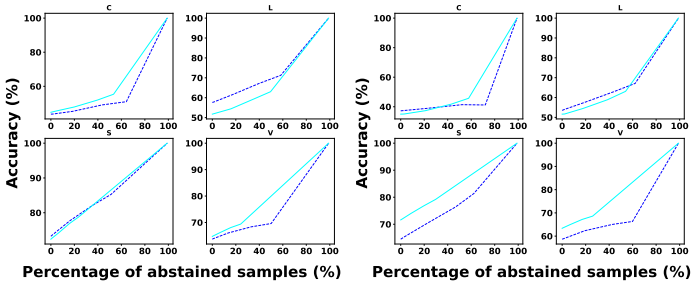


(e) Original style (f) Wikiart style

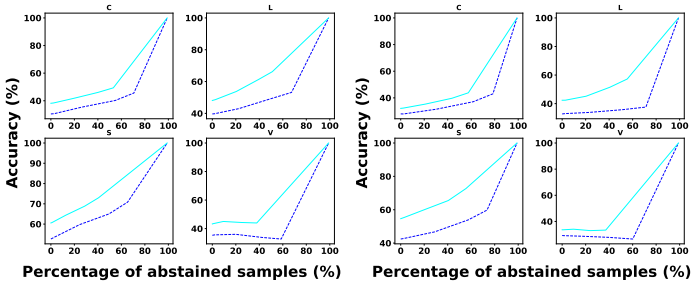


(g) Severity 3 corruptions (h) Severity 5 corruptions

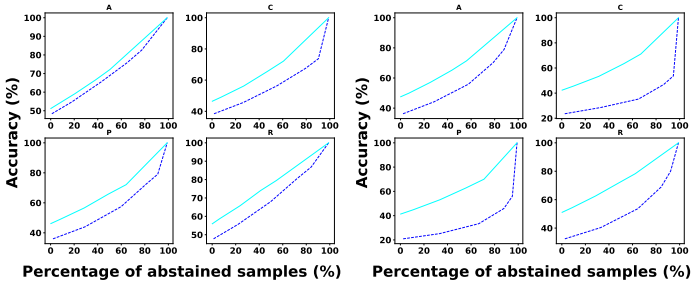
Figure 8. Effectiveness of using NSS (with ERM) (solid lines) at improving the ability of DG classifiers to produce risk-averse predictions when evaluated with TT-NSS in comparison to that of other DG methods (dashed lines) in a **single** (a-d) and **multi**-domain (e-h) setup. NSS-trained classifiers achieve significantly better accuracy on non-abstained samples compared to classifiers trained with ERM and achieve competitive performance to models trained with RSC and SagNet at different abstaining rates on variants of the **PACS** dataset in a multi-source domain setup. (See Fig. 6 for the explanation of settings.)



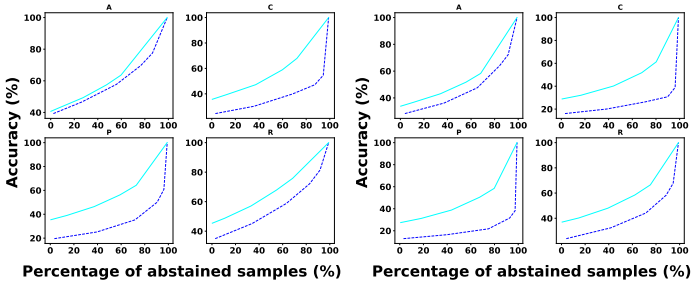
(a) Original style (b) Wikiart style



(c) Severity 3 corruptions (d) Severity 5 corruptions

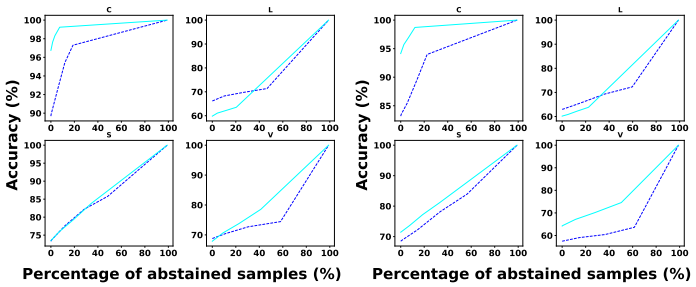


(e) Original style (f) Wikiart style

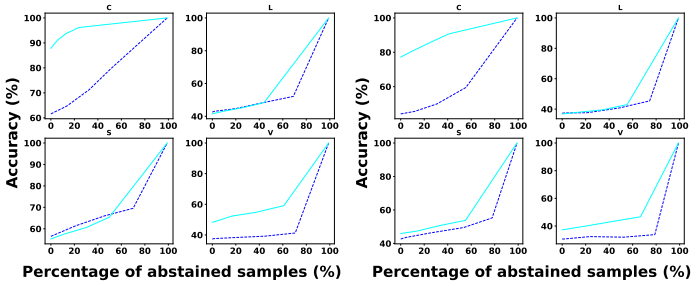


(g) Severity 3 corruptions (h) Severity 5 corruptions

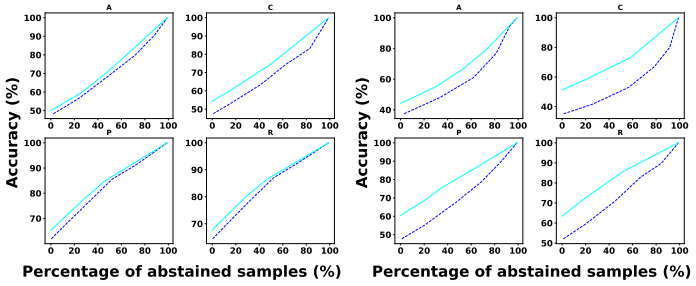
Figure 9. Effectiveness of using NSS (with ERM as the base DG method) (solid lines) at improving the ability of DG to produce risk-averse predictions when evaluated with TT-NSS making it superior or competitive to classifiers trained with ERM (dashed lines) on variants of the **VLCS** (a-d) and **OfficeHome** (e-h) dataset in a **single** source domain setup. (See Fig. 2 for the explanation of settings.)



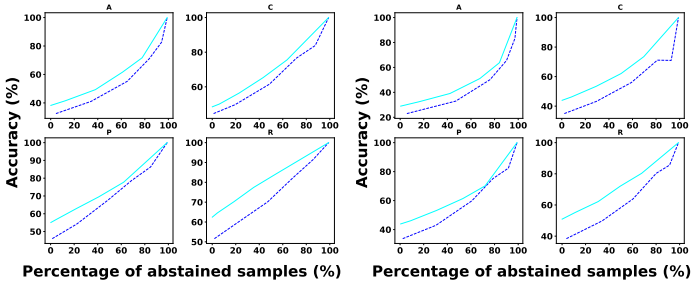
(a) Original style (b) Wikiart style



(c) Severity 3 corruptions (d) Severity 5 corruptions



(e) Original style (f) Wikiart style



(g) Severity 3 corruptions (h) Severity 5 corruptions

Figure 10. Effectiveness of using NSS (with ERM as the base DG method) (solid lines) at improving the ability of DG to produce risk-averse predictions when evaluated with TT-NSS making it superior or competitive to classifiers trained with ERM (dashed lines) on variants of the **VLCS** (a-d) and **OfficeHome** (e-h) dataset in a **multi**-source domain setup. (See Fig. 6 for the explanation of settings.)

Table 7. Effectiveness of NSS at producing a better AUC score compared to classifiers trained with ERM in a **multiple** source domain setting on PACS, VLCS, and OfficeHome datasets and their variations when evaluated with TT-NSS. (The target domain used for evaluation is denoted in the columns).

		PACS				VLCS				OfficeHome			
Dataset Variation	Alg.	A	C	P	S	C	S	L	V	A	C	P	R
Original Style	ERM	0.893	**0.9**	0.978	0.911	0.968	**0.772**	0.86	0.776	0.683	0.679	0.815	0.83
	NSS	**0.95**	0.884	**0.98**	0.914	**0.985**	0.769	0.865	**0.818**	**0.72**	**0.749**	**0.836**	**0.849**
Wikiart Style	ERM	0.816	0.876	0.97	0.886	0.941	0.744	0.822	0.678	0.578	0.534	0.692	0.726
	NSS	**0.926**	0.869	0.971	**0.909**	**0.98**	**0.766**	**0.85**	**0.775**	**0.667**	**0.713**	**0.798**	**0.825**
Corrupted with severity 3	ERM	0.771	0.898	0.878	0.923	0.785	0.553	0.692	0.476	0.5	0.64	0.677	0.715
	NSS	**0.889**	**0.933**	**0.943**	**0.933**	**0.959**	**0.605**	**0.706**	**0.632**	**0.587**	**0.697**	**0.738**	**0.812**
Corrupted with severity 5	ERM	0.621	0.856	0.837	0.888	0.626	0.477	0.54	0.388	0.387	0.53	0.554	0.59
	NSS	**0.792**	0.854	**0.88**	**0.902**	**0.898**	**0.53**	**0.611**	**0.517**	**0.473**	**0.648**	**0.628**	**0.721**

Table 8. Effectiveness of NSS at producing a better AUC score compared to classifiers trained with ERM in a **single** source domain setting on PACS, VLCS, and OfficeHome datasets and their variations when evaluated with the confidence-based abstaining mechanism. (The source domain used for training is denoted in the columns).

		PACS				VLCS				OfficeHome			
Dataset Variation	Alg.	A	C	P	S	C	L	S	V	A	C	P	R
Original Style	ERM	0.882	0.875	0.634	**0.707**	0.653	0.68	0.806	0.715	0.743	0.717	0.699	**0.789**
	NSS	**0.907**	**0.923**	**0.733**	0.665	**0.671**	0.687	**0.838**	**0.74**	0.739	0.72	0.708	0.778
Wikiart Style	ERM	0.84	0.757	0.609	**0.558**	0.426	0.584	0.763	0.679	0.545	0.364	0.334	0.484
	NSS	**0.871**	**0.885**	**0.672**	0.526	**0.535**	**0.655**	**0.816**	**0.722**	**0.705**	**0.658**	**0.64**	**0.749**
Corrupted with severity 3	ERM	0.832	0.709	0.613	**0.612**	0.504	0.381	0.734	0.468	0.596	0.412	0.411	0.586
	NSS	**0.871**	**0.865**	**0.754**	0.549	**0.592**	**0.631**	**0.771**	**0.522**	**0.666**	**0.586**	**0.566**	0.595
Corrupted with severity 5	ERM	0.696	0.579	0.418	**0.479**	0.433	0.329	0.563	0.346	0.416	0.243	0.223	0.388
	NSS	**0.769**	**0.746**	**0.667**	0.434	0.454	**0.576**	**0.635**	**0.4**	**0.546**	**0.49**	**0.415**	**0.42**

Table 9. Effectiveness of NSS at producing a better AUC score compared to classifiers trained with ERM in a **multiple** source domain setting on PACS, VLCS, and OfficeHome datasets and their variations when evaluated with the confidence-based abstaining mechanism. (The target domain used for evaluation is denoted in the columns).

		PACS				VLCS				OfficeHome			
Dataset Variation	Alg.	A	C	P	S	C	L	S	V	A	C	P	R
Original Style	ERM	0.95	0.902	0.986	0.915	0.986	**0.752**	**0.88**	0.831	0.802	0.721	0.889	**0.905**
	NSS	0.955	0.896	0.985	0.922	0.987	0.706	0.86	0.829	0.783	**0.767**	0.876	**0.884**
Wikiart Style	ERM	0.898	0.85	0.975	0.892	0.954	**0.747**	0.815	0.691	0.601	0.588	0.726	0.796
	NSS	**0.927**	**0.898**	0.982	**0.92**	**0.982**	0.705	**0.833**	**0.781**	**0.707**	**0.747**	**0.829**	**0.838**
Corrupted with severity 3	ERM	0.79	0.918	0.947	0.909	0.908	0.601	0.678	0.599	0.529	0.584	0.74	0.717
	NSS	**0.887**	0.909	0.955	**0.922**	**0.966**	0.594	**0.735**	**0.627**	**0.647**	**0.735**	**0.775**	**0.808**
Corrupted with severity 5	ERM	0.539	0.85	0.852	0.845	0.775	**0.526**	0.483	0.427	0.362	0.475	0.581	0.551
	NSS	**0.735**	**0.881**	**0.887**	0.833	**0.91**	0.508	**0.621**	**0.44**	**0.528**	**0.66**	**0.672**	**0.688**