Reactive Model Correction: Mitigating Harm to Task-Relevant Features via Conditional Bias Suppression

Supplementary Material

A. Method

In the upcoming section, we explore further details and derivations of our methods. Appendix A.1 provides the derivation of multi-artifact P-ClArC, as per Eq. (2). Appendix A.2 includes pseudocode for both P-ClArC and R-ClArC. Lastly, in Appendix A.3, we introduce a 3D toy experiment to illustrate the transformations induced by P- and R-ClArCs.

A.1. Derivation for multi-artifact P-ClArC

Let $\mathcal{C}' \subseteq \mathcal{C}$ represent a subset of artifacts with a size $|\mathcal{C}'| = k$, $1 \leq k \leq n$. Let $V_{\mathcal{C}'} = [v_i]_{c_i \in \mathcal{C}'}$ be the matrix comprised of the respective CAVs as column vectors. Let $Z^-_{\mathcal{C}'} = \bigcap_{c_i \in \mathcal{C}'} X^-_i$ be the intersection of negative examples, and $z^-_{\mathcal{C}'} = \frac{1}{|Z^-_{\mathcal{C}'}|} \sum_{z \in Z^-_{\mathcal{C}'}} a(z)$. Let $a_x = a(x)$ and $p_x = \tilde{h}(a_x, \mathcal{C}')$ be the P-ClArC transformation. Our objective function can be then formalized as follows:

$$\begin{aligned} &\min_{p_x} ||p_x - a_x||^2 \\ &\text{s.t. } V^T_{\mathcal{C}'}(p_x - z^-_{\mathcal{C}'}) = 0. \end{aligned} \tag{6}$$

The corresponding Lagrangian function is as follows:

$$\mathcal{L} = \frac{1}{2}||p_x - a_x||^2 + \lambda^T V^T_{\mathcal{C}'}(p_x - z^-_{\mathcal{C}'}). \tag{7}$$

Applying the Karush-Kuhn-Tucker conditions, we get:

$$\begin{aligned} p_x - a_x + V_{\mathcal{C}'}\lambda = 0; \\ V^T_{\mathcal{C}'}(p_x - z^-_{\mathcal{C}'}) = 0. \end{aligned} \tag{8}$$

Solving for λ we get:

$$\begin{aligned} &V^T_{\mathcal{C}'}p_x - V^T_{\mathcal{C}'}a_x + V^T_{\mathcal{C}'}V_{\mathcal{C}'}\lambda = 0 \iff \\ &V^T_{\mathcal{C}'}z^-_{\mathcal{C}'} - V^T_{\mathcal{C}'}a_x + V^T_{\mathcal{C}'}V_{\mathcal{C}'}\lambda = 0 \iff \\ &\lambda = (V^T_{\mathcal{C}'}V_{\mathcal{C}'})^{-1}V^T_{\mathcal{C}'}(a_x - z^-_{\mathcal{C}'}). \end{aligned} \tag{9}$$

Inserting the resulting λ into the first KKT condition we get:

$$\begin{aligned} p_x - x + V_{\mathcal{C}'}(V^T_{\mathcal{C}'}V_{\mathcal{C}'})^{-1}V^T_{\mathcal{C}'}(a_x - z^-_{\mathcal{C}'}) = 0 \iff \\ p_x = x - V_{\mathcal{C}'}(V^T_{\mathcal{C}'}V_{\mathcal{C}'})^{-1}V^T_{\mathcal{C}'}(a_x - z^-_{\mathcal{C}'}). \end{aligned} \tag{10}$$

Thus, we acquire $p_x = \tilde{h}(x)$ as the P-ClArC transformation for multiple artifacts, as described in Eq. (2).

A.2. Pseudocode for P-ClArC and R-ClArC

A detailed Algorithm for both P-ClArC and R-ClArC shown in Algorithm 1 under the common name of Class Artifact Compensation.

Algorithm 1: Class Artifact Compensation

Data: Sample x;
Model f with accessible layer l (and subnetwork f_l);
For each artifact in $\mathcal{C}$, sets of positive examples $X^+ = \{X^+_1, X^+_2, \ldots, X^+_n\}$ and negative examples $X^- = \{X^-_1, X^-_2, \ldots, X^-_n\}$;
For each artifact in $\mathcal{C}$, sets of activations of positive examples $A^+ = \{A^+_1, A^+_2, \ldots, A^+_n\}$ and activations of negative examples $A^- = \{A^-_1, A^-_2, \ldots, A^-_n\}$ in layer l;
Set of layer-l CAVs V^l for each artifact in $\mathcal{C}$.
Result: output for x according to a modified predictor f' desensitized to artifacts $\mathcal{C}$

`/* deactivate the use of` $\mathcal{C}$ `in` f `*/`
if *P-ClArC* **then**
  $Z^- = \text{mean\_of\_intersection}(A^-)$;
  $h^l_c = \text{backward\_artifact\_model}(V^l, Z^-)$;
else if *R-ClArC* **then**
  $\mathcal{C}' = \text{condition\_generating\_function}(x)$;
  $V^l_{\mathcal{C}'} = \text{subset\_by\_concept}(V^l, \mathcal{C}')$;
  $A^-_{\mathcal{C}'} = \text{subset\_by\_concept}(A^-, \mathcal{C}')$;
  $Z^-_{\mathcal{C}'} = \text{mean\_of\_intersection}(A^-_{\mathcal{C}'})$;
  $h^l_c = \text{backward\_artifact\_model}(V^l_{\mathcal{C}'}, Z^-_{\mathcal{C}'})$;
$a_x = f_L(x)$;
$f_{l'}(a_x) := h^l_c(a_x)$;
$f' = f_L \circ \cdots \circ f_{l+1} \circ f_{l'} \circ f_l \circ \cdots \circ f_1(x)$;
return $f'(x)$

A.3. 3D Toy Model

We construct a three-dimensional toy dataset comprising two classes. In Class 1, two artifacts are present. For Class 1 (red circles), we generate 500 clean samples distributed normally, with a mean at coordinates $(0, 8, 0)$ and a covariance matrix equal I, where I represents the 3×3 identity matrix. Additionally, 500 samples are created for Artifact 1 (blue diamonds), centered at $(1, 8, 8)$, with covariance matrix I. Analogously, another set of 500 samples is generated for Artifact 2 (blue diamonds), with a mean at $(1, 1, 8)$ and co-

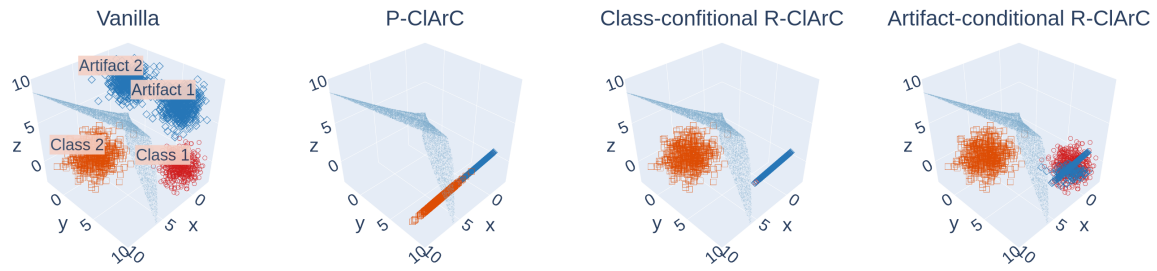


Figure 8. 3D Toy Model illustrating P-ClArC and R-ClArC transformations. The dataset includes Class 1, Class 2, and two artifacts belonging to Class 1. A three-layer feed-forward neural network is used for binary classification, with its decision boundary shown in light blue. P-ClArC shifts Class 2 samples across the decision boundary, resulting in misclassification, while both versions of R-ClArC maintain accuracy.

variance matrix I. For Class 2 (orange squares), 500 clean samples are distributed normally with a mean at $(6, 1, 1)$ and a covariance matrix $1.8 \cdot I$. The original distribution of the datapoints is represented in the "Vanilla" facet of Fig. 8.

We construct a 3-layer feed-forward neural network for binary classification with an input layer of 3 neurons, a hidden layer of 30 neurons, and an output layer of 2 neurons. The model is trained using the Adam optimizer with a learning rate of 0.01 for 5000 epochs. The decision boundary of the trained model is depicted as a light-blue surface in Fig. 8.

We compute pattern-CAVs and train two linear SVM classifiers with L2 regularization and squared hinge loss for the classification of the two artifacts. Subsequently, we apply three distinct transformations to the original data: P-ClArC (Eq. (2)), class-conditional R-ClArC (Eq. (3), Eq. (4)), and artifact-conditional R-ClArC (Eq. (3), Eq. (5)), where artifact-conditional R-ClArC utilizes the linear SVM classifier for artifact presence detection. Instead of applying the transformation to activations, we directly apply them to the input data. Fig. 8 illustrates these transformations.

In all ClArC transformations, only the data points are altered, while the decision boundary remains unchanged, as the model weights remain constant. We observe that P-ClArC uniformly transforms all samples, resulting in a significant number of Class 2 data points (orange squares) crossing the decision boundary and leading to misclassification. In contrast, class-conditional R-ClArC preserves model accuracy by leaving unchanged the data points classified by the model as Class 2. Artifact-conditional R-ClArC exclusively transforms data points classified by the SVM classifiers as Artifact 1, Artifact 2, or both, further preserving the original data distribution while maintaining accuracy.

B. Experimental Details

We outline the details of the two generated datasets for FunnyBirds, as well as the original and poisoned datasets for ISIC2019, in Appendix B.1. Appendix B.2 provides insights into the training process of Vanilla models. Information about the CAV calculation method is covered in Appendix B.3, while we evaluate CAVs in Appendix B.4. Lastly, Appendix B.5 outlines the details of the P- and R-ClArCs model correction methods and their evaluation.

B.1. Datasets

The FunnyBirds framework [19] provides a framework for the creation of controlled datasets featuring 3D-rendered birds. Each bird class comprises 5 parts, with multiple options available for each part (*e.g.*, 4 beaks, 3 eyes, etc.), which are assembled to form a bird sample. These samples are then placed within 3D scenes, where parameters such as camera position, zoom, lighting, and background objects are randomly selected for each sample.

We generated the *backdoor* FunnyBirds dataset, which comprises 2 classes of birds. The defining parts for the two classes were randomly selected. As a backdoor artifact, we randomly selected the "green box" background object (see Fig. 2) and predefined its position relative to the bird's position within the 3D scene's coordinate system. For training and validation, we created a dataset consisting of 5000 samples of each of the two classes. 33% of the labels of samples of class 0 were flipped to encourage learning the backdoor artifact. A randomly chosen 10% subset was allocated for validation. Additionally, we constructed a test set comprising 100 correctly labeled birds from each class.

The *shortcut* FunnyBirds dataset comprises 10 different classes. We incentivized the utilization of the shortcut artifact by designing classes 0 to 3 to only vary in the beak part; otherwise, the parts of other classes were randomly selected. We generated 10 different background object artifacts with predetermined positions relative to the bird (as

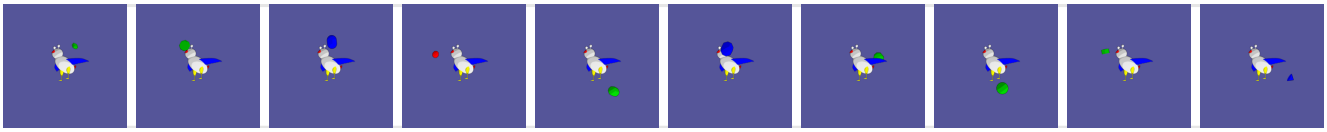


Figure 9. 10 artifacts included in Class 0 of FunnyBirds *shortcut* dataset. The positions of the 10 background objects relative to the bird objects within the scene are fixed.

depicted in Fig. 9), which were inserted into samples of class 0. Specifically, 50% of class 0 bird samples included a randomly selected number from 1 to 10 of these artifacts. We ensured an equal distribution of the total number of background artifacts between shortcut and non-shortcut samples by setting the minimum number of background objects for each sample to 10. This way, we ensured that the number of background objects was not used as a spurious feature. We generated a dataset with 1000 birds of each class, with 10% of this set allocated for validation. Additionally, we constructed a test set comprising 100 birds from each class.

For both FunnyBirds datasets, we generated binary masks that precisely localize the artifact object using the functionality of the FunnyBirds framework. These binary masks are employed to assess artifact relevance in Sec. 4.4.

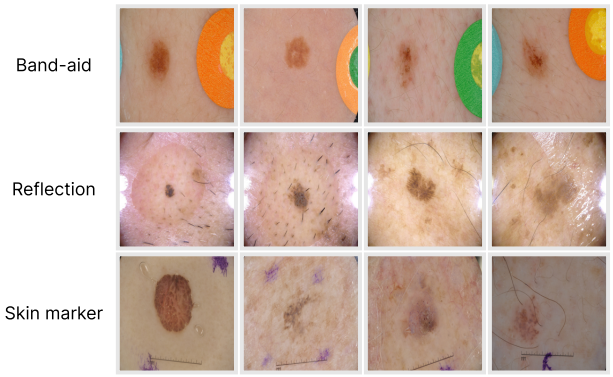


Figure 10. Examples of ISIC artifacts band-aid ("NV"), skin marker ("NV", "BKL"), and reflection ("BKL") artifacts.

The ISIC2019 [8, 9, 28] dataset consists of 25,331 samples of classes "MEL", "NV", "BCC", "AK", "BKL", "DF", "VASC", and "SCC". We used the Reveal2Revise (R2R) framework [30] to identify three artifacts naturally occurring in the dataset, strongly correlating with class labels: band-aid ("NV"), skin marker ("NV", "BKL"), and reflection ("BKL") artifacts (examples are provided in Fig. 10). Following the R2R approach, we identified artifact samples and computed artifact localization binary masks. Firstly, we obtained artifact localization heatmaps by generating Layer-wise Relevance Propagation (LRP) [4] heatmaps for SVM-CAVs [20] in the features.7 layer of VGG16 trained on ISIC2019 data (training details of the model were outlined in Appendix B.2). The LRP attribution heatmaps are computed using the $\epsilon z^+\flat$-composite [21] with the `zennit` library [2]. Secondly, we manually sorted the heatmaps to exclude those that appeared to have high attributions in regions unrelated to the artifact concept. Thirdly, pixels corresponding to positive attributions (larger than $\epsilon = 0.3$) received a value of 1 in the binary masks, while others were assigned a value of 0. These artifact binary localization masks were subsequently utilized in evaluating artifact relevance (as described in Sec. 4.4). Additionally, we utilized the CAV heatmaps to isolate the artifacts in the corresponding samples and overlay them onto clean test samples for the ISIC2019 "poisoned" setting and for the "generated" CAV datasets Sec. 4.2. Specifically, the artifact sample image was multiplied with its CAV heatmap element-wise and then added to the clean sample with the pixel values multiplied by (1 - attribution) element-wise.

Dataset	Model	Optimizer	LR	Epochs
ISIC2019	VGG16	SGD	0.05	150
	ResNet18	SGD	0.05	150
	EfficientNet-B0	Adam	0.01	150
FunnyBirds *backdoor*	VGG16	SGD	0.001	100
	ResNet18	Adam	0.001	100
	EfficientNet-B0	Adam	0.001	100
FunnyBirds *shortcut*	VGG16	SGD	0.001	100
	ResNet18	Adam	0.001	100
	EfficientNet-B0	Adam	0.001	100

Table 5. Model training details including optimizer, initial learning Rate (LR), number of epochs.

B.2. Model Training

Tab. 5 provides training details for all models and datasets, including optimizer, initial learning rate (LR), and number of epochs. The ISIC2019 models were pretrained on ImageNet [36] using weights obtained from the `Torchvision` library. The learning rate (LR) for the ISIC2019 model was divided by 10 after epochs 50 and 80 during training. Both FunnyBirds models were trained from scratch, employing early stopping based on validation set loss with a patience of 3 epochs.

Model	CAV Dataset	FunnyBirds		ISIC2019					
		"green box"		"reflection"		"band-aid"		"skin marker"	
		Pattern	Filter	Pattern	Filter	Pattern	Filter	Pattern	Filter
ResNet18	Generated	0.824	0.101	0.617	0.343	0.406	0.215	0.406	0.166
	Data Subset	0.563	0.042	0.469	0.328	0.166	0.156	0.193	0.089
VGG16	Generated	0.779	0.132	0.919	0.180	0.588	0.132	0.608	0.128
	Data Subset	0.345	0.119	0.885	0.430	0.482	0.235	0.443	0.121
EfficientNet-B0	Generated	0.859	0.040	0.655	0.093	0.454	0.054	0.440	0.088
	Data Subset	0.854	0.002	0.332	0.424	−0.077	0.080	−0.012	0.056

Table 6. Evaluating the alignment of artifact CAVs in terms of cosine similarity with the actual change in activations when the concept is added in a controlled fashion across various models, the FunnyBirds backdoor dataset, and the ISIC2019 dataset. We compare CAVs computed on original data subsets, and CAVs computed on pairs of clean and (generated) poisoned samples.

Model	FunnyBirds *backdoor*	FunnyBirds *shortcut*										ISIC2019		
	"GB"	0	1	2	3	4	5	6	7	8	9	"R"	"BA"	"SM"
ResNet18	94.1	96.3	96.2	94.0	95.6	94.9	97.5	96.3	94.9	95.6	96.3	98.4	95.9	100.0
VGG16	90.4	94.4	93.1	94.0	93.1	88.6	95.7	93.9	93.7	93.1	93.2	93.4	100.0	96.7
EfficientNet-B0	90.1	93.8	93.1	93.4	93.8	91.8	94.4	92.6	94.3	94.4	93.8	91.8	95.9	100.0

Table 7. Hold-out set accuracies of linear SVM classifiers across diverse datasets, artifacts, and models. These classifiers serve as artifact-condition-generating functions in artifact-conditional and combined R-ClArC. "GB" denotes "green box", "R" represents "reflection", "BA" indicates "band-aid", and "SM" signifies "skin marker".

B.3. CAV Calculation

For both FunnyBirds datasets, we create respectively additional 1000 negative samples of class 0 birds, as in both cases this class is associated with artifacts. In the *backdoor* FunnyBirds dataset, we then generate 1000 images with the "green box" artifact, while for the "shortcut" FunnyBirds dataset, we produce a set of 1000 positive examples for each of the 10 artifacts.

To create negative example sets for ISIC2019 for the generated CAVs, we begin by sampling 1000 non-artifact images from the classes associated with each artifact. Subsequently, we overlay the cropped-out artifacts onto these images, following the process outlined in Appendix B.1, resulting in a set of 1000 positive examples for each artifact. For dataset subset CAVs, we use all available artifact samples as positive examples and sample negative examples from the ISIC2019 non-artifact samples. We ensure that the ratio of positive to negative examples does not exceed 5.

For pattern-based CAV calculation we directly adopt the approach from [31], while for filter-based CAVs we employ linear SVMs trained with L2 regularization and squared hinge loss with class weights inversely proportional to class frequencies.

B.4. CAV Evaluation

The alignment scores for various CAVs methods were computed following the approach outlined in [31]. We present the CAV evaluation results for different model architectures, for the FunnyBirds backdoor "green box" artifact, and all examined ISIC2019 artifacts in Tab. 6.

B.5. Model Correction and Evaluation

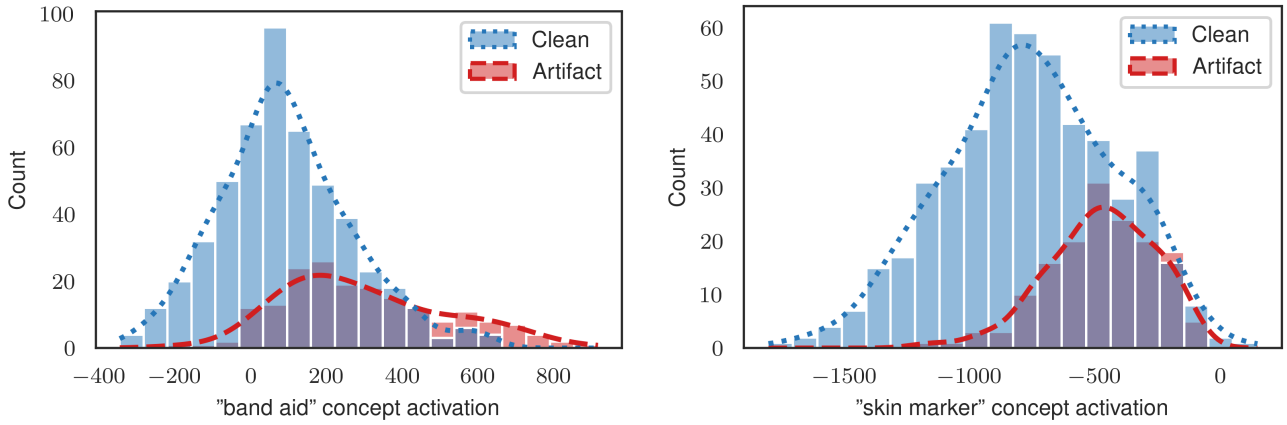
We assess artifact relevance using heatmaps computed with LRP using the $\epsilon z^+\flat$-composite [21] with the `zennit` library [2]. The procedure for generating binary localization masks is detailed in Appendix B.1. Artifact relevance is quantified as the sum of absolute attribution values within the mask divided by the sum of all absolute attribution values. To visualize the LRP heatmaps in Fig. 7, we normalize them by dividing each heatmap by its maximum absolute value.

Our evaluation encompasses P-ClArC, as well as R-ClArC with class-conditional and artifact-conditional condition-generating functions, along with their combination. Model correction is performed for all models post the last convolutional layer, utilizing pattern-based "generated" CAVs. For artifact-conditional and combined R-ClArCs, our artifact-conditioning function classifies the samples in the latent space of the last convolutional layer as well. For this, we employ linear SVMs trained with L2 regularization

and squared hinge loss, with class weights inversely proportional to class frequencies. The training data consists of all available artifact samples as positive examples and a subset of negative examples from the dataset, ensuring the positive-to-negative example ratio does not exceed 5. 20% of the training set serves as a holdout set to assess classifier accuracy. The accuracies of the resulting SVM classifiers are presented in Tab. 7.

C. Further Experiments

In the following section, we present supporting experiments aimed at testing the orthogonality of concepts (Tab. 8, Fig. 11). Additionally, we provide further heatmaps to facilitate qualitative evaluation of the R-ClArC method compared to P-ClArC (Fig. 12, Fig. 13, Fig. 14).



(a) "Band-Aid"

(b) "Skin Marker"

Figure 11. Histogram illustrating the activations of artifact CAVs for the corresponding artifact samples alongside 500 randomly selected clean samples for ISIC2019 dataset and ResNet18 model.

Artifacts	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
0	−0.33	0.12	−0.65	0.62	0.40	0.45	0.34	−0.75	0.09	0.19
1	0.59	0.07	0.79	−0.17	−0.78	−0.59	−0.77	0.47	0.48	0.01
2	0.51	−0.15	0.80	−0.53	−0.73	−0.44	−0.64	0.70	0.40	−0.23
3	−0.45	−0.08	−0.78	0.53	0.61	0.62	0.57	−0.60	−0.23	0.01
4	0.13	0.04	0.31	−0.31	−0.46	−0.12	−0.32	0.07	0.62	0.00
5	−0.23	0.37	−0.45	0.54	0.29	0.12	0.20	−0.74	0.09	0.43
6	0.54	0.13	0.77	−0.26	−0.73	−0.57	−0.74	0.49	0.37	0.07
7	−0.27	0.13	−0.03	−0.28	−0.11	0.13	0.05	−0.13	0.33	0.12
8	0.70	−0.27	0.90	−0.44	−0.80	−0.51	−0.76	0.84	0.35	−0.35
9	−0.36	0.43	−0.54	0.49	0.48	0.10	0.33	−0.73	−0.13	0.49

Table 8. Cosine similarity between artifact CAVs and the mean feature direction of each class for the FunnyBirds shortcut dataset and EfficientNet-B0. The strong relationship between artifact and class direction explains the strong negative impact of ClArC transformations on model performance. Suppressing the artifact direction results in pushing samples across the decision boundary.

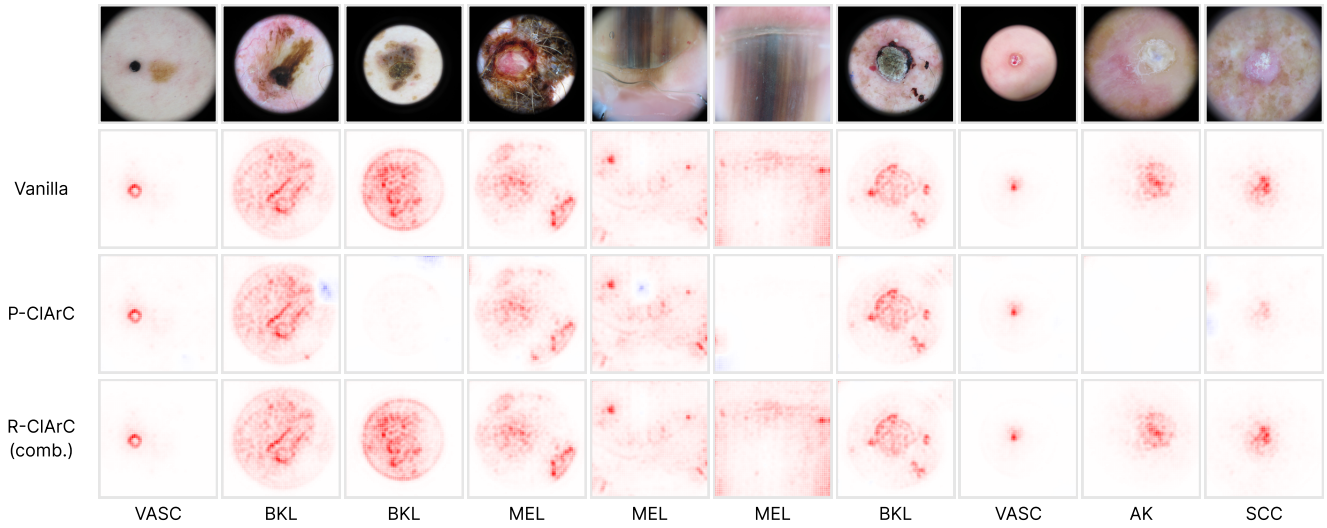


Figure 12. LRP heatmaps depicting samples with pronounced activation of the "reflection" concept for the Vanilla model and models corrected using P-ClArC and R-ClArC combining class- and artifact-conditional approaches.

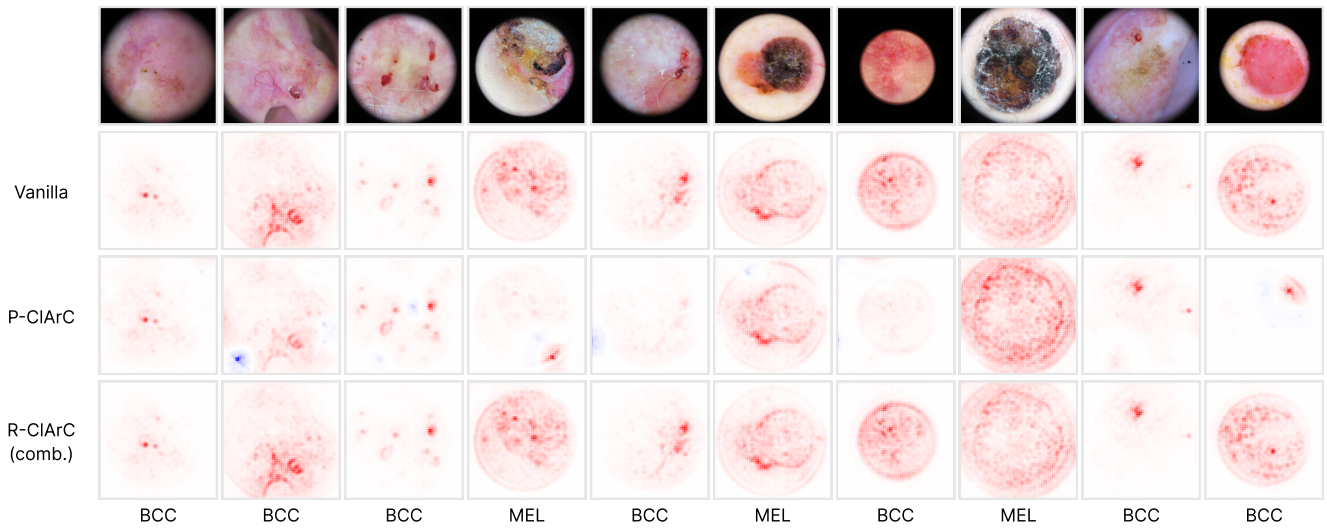


Figure 13. LRP heatmaps depicting samples with pronounced activation of the "band-aid" concept for the Vanilla model and models corrected using P-ClArC and R-ClArC combining class- and artifact-conditional approaches.

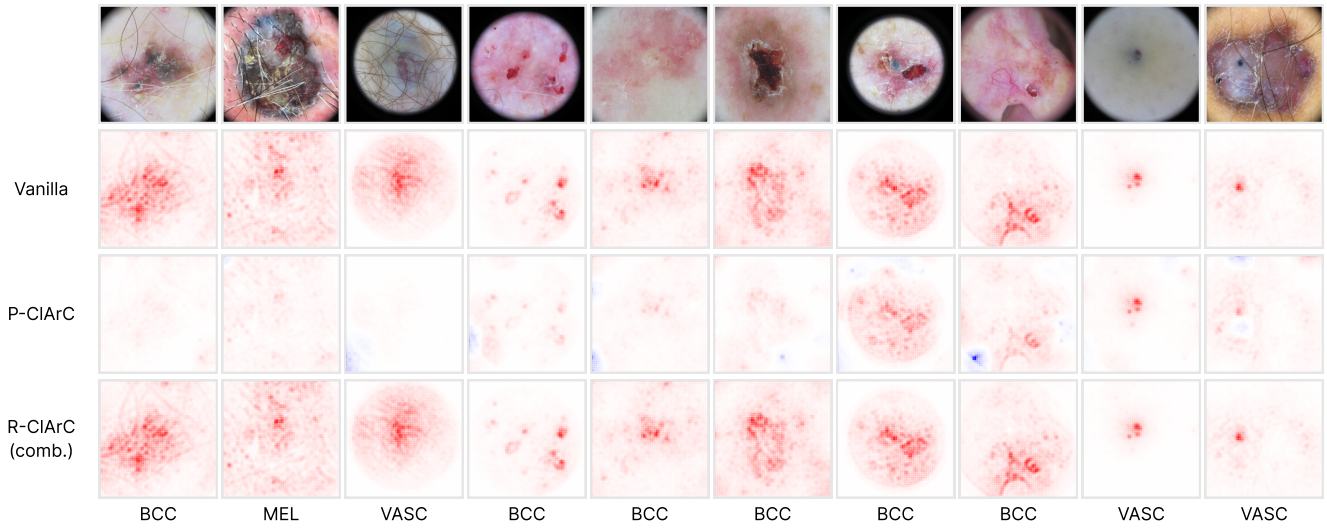


Figure 14. LRP heatmaps depicting samples with pronounced activation of the "skin marker" concept for the Vanilla model and models corrected using P-ClArC and R-ClArC combining class- and artifact-conditional approaches.