Appendix

A Computation of $\bar{x}'$

Algorithm 1 Convex hull projection

procedure PROJECTION(X, Y)
  Dataset $X \in \mathbb{R}^{l \times d}$ ▷ l samples with extrinsic dimension d
  Labels $Y \in \mathbb{N}^l$
  for j in $1 \cdots l$ **do**: ▷ Iterate over all samples in X
    $x'_j \leftarrow X[j]$
    $y'_j \leftarrow Y[j]$
    for y in unique(Y) $\setminus$ y'_j **do**: ▷ Iterate over unique labels in Y other than y'_j
      $\{x_i\}_{i=1}^d \leftarrow d$-NearestNeighbours$(x', y, d)$ ▷ NNs of x'_j with label y
      $\mathcal{S}_j \leftarrow \{x_1, ..., x_d, x'_j\}$ ▷ Lin. sep. set of cardinality $|\mathcal{S}_j| = d + 1$
      $h \leftarrow$ SupportVectorClassifier$(\mathcal{S})$ ▷ Get max-margin hyperplane
      $\hat{x} \leftarrow$ OrthogonalProjection(x', h) ▷ Project on max-margin hyperplane
      $\bar{x} \leftarrow$ Reflection$(\hat{x}, h)$ ▷ Reflect around projection
      $r'_j(y) \leftarrow ||x'_j - \bar{x}'_j||_2$ ▷ r'_j for x'_j and class y
      $r_j(y) \leftarrow ||x'_j - x_1||_2$ ▷ r_j for x'_j and class y
    $r'_j \leftarrow \min_i(\{r'_j(y_i)\})$ ▷ Minimal r'_j over all classes
    $r_j \leftarrow \min_i(\{r_j(y_i)\})$ ▷ Minimal r_j over all classes
  $R^* \leftarrow \min_{j \in 1,...,l}(\{r'_j\})$ ▷ Minimal r'_j over all samples

We consider all samples $x'_j \in X$ for $j = 1, ..., l$. The orthogonal projection of x'_j onto the convex hull of its d nearest neighbours of another class label $\mathcal{C}(\{x_i\}_{i=1}^d)$ is defined as

$$\bar{x}'_j \coloneqq \text{argmin}_{\hat{x} \in \mathcal{C}(\{x_i\}_{i=1}^d)} ||x'_j - \hat{x}||_2 \text{ s.t.}$$
$$\hat{x} = \sum_{i=1}^{d} w_i x_i,\ 0 \leq w_i \leq 1,\ \sum_{i=1}^{d} w_i = 1$$

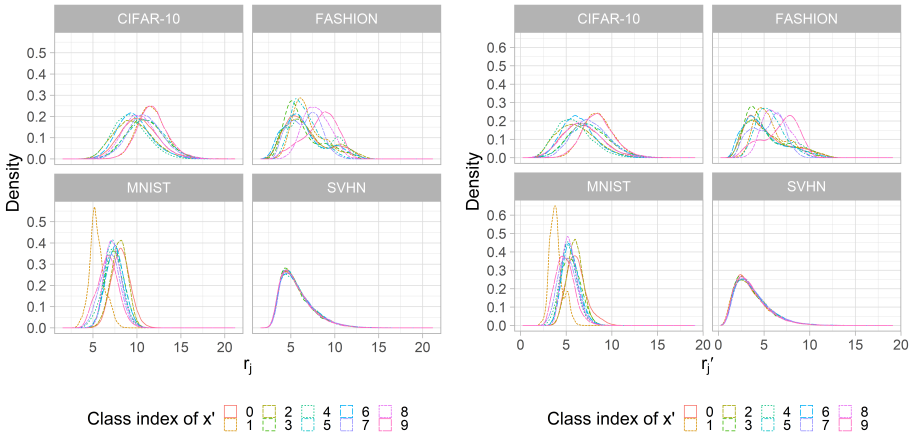
We derive $\bar{x}'_j$ by training a support vector classifier [75] (SVC) on the corresponding set $\mathcal{S}_j \coloneqq \{x_1, ..., x_d, x'_j\}$. As the cardinality of $|\mathcal{S}_j|$ is equal to the VC-dimension of a linear classifier, i.e. $d + 1$, the SVC obtains zero error and maximizes the margin m between x'_j and $\{x_i\}_{i=1}^d$. It simply follows that $2m = r'_j \leq r_j$, where $r_j = 2m$ implies that $\bar{x}'_j = x_1$. Computation of $\bar{x}'_j$ is straightforwardly done by orthogonally projecting x'_j onto the hyperplane learned by the SVC. Then, the reflection of x'_j around this projection yields $\bar{x}'_j$ which implies $2m = r'_j$. This computation is efficient and deterministic. It is outlined in Algorithm 1.

B Computation of $\bar{x}'$ for $\mathcal{S}_j \neq d + 1$

Defining $\mathcal{S}_j$ such that $|\mathcal{S}_j| > d + 1$ might results in sets that are not linearly separable and our analysis is not be applicable to those. Defining $\mathcal{S}_j$ such that

Table 6: Results for $|\mathcal{S}_j| = 2$. The bound r'_j is vacuous and the results falsely contradict the experimental findings in Section 5.

	R	$0.5R$	R^*	R^{crit}	$\{\bar{x}\}^{\text{crit}}_{\text{local}}$	$\frac{\lvert\{\bar{x}\}^{\text{crit}}_{\text{local}}\rvert}{l(c-1)}$	$\{\bar{x}\}^{\text{crit}}_{\text{global}}$	$\frac{\lvert\{\bar{x}\}^{\text{crit}}_{\text{global}}\rvert}{l(c-1)}$
CIFAR-10	2.751	1.375	2.313	1.682	0	0.000	0	0.000



(a) Density of r_j for all $x'_j \in X$. (b) Density of r'_j for all $x'_j \in X$.

Fig. 6: Distributions of r_j and r'_j.

$|\mathcal{S}_j| < d + 1$ results in a vacuous r'_j because it might not be the smallest lower bound any more. For illustration we display the results for CIFAR-10 in Table 6 where $|\mathcal{S}_j| = 2$, so x'_j is projected onto the line segment of its two nearest neighbours x_1 and x_2. As one can see, r'_j is a vacuous bound on the perturbation magnitude and the results falsely contradict the experimental findings of Section 5.

C Additional statistics

In Figure 6 we display the densities of r_j and r'_j for all used datasets. One can observe that the distributions have similar shapes but different modes, except for the SVHN dataset.

In Figure 7 we display the empirical cumulative distribution function of r_j^{crit} for all datasets.

In Figure 8 we display for every sample $x'_j \in \{\bar{x}\}^{\text{crit}}_{\text{global}}$ the label of x'_j and the distribution of labels of the corresponding $\bar{x}'_j$ inherited from the label of the samples $\{x_i\}_{i=1}^d$. For SVHN one can observe that the critical samples lie on close proximity to several if not all samples of other classes. This uniform distribution of distances is also visible in Figures 6 and 7. Contrary, for CIFAR-10 only

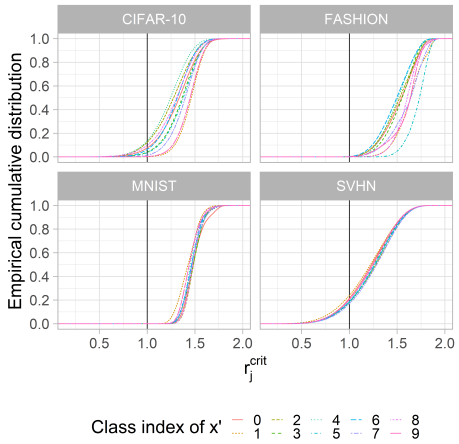


Fig. 7: Cumulative distribution of r_j^{crit} for all $x'_j \in X$.

the classes *airplane*, *bird*, *deer*, *frog* and *ship* contain critical samples $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ and the nearest neighbours come from a few dominating classes. For the critical samples with labels *bird* and *ship*, for example, the corresponding $\{x_i\}_{i=1}^{d}$ are of classes *airplane* with a relative frequency of 41.7% and 43.6%, respectively. This might be due to the common uniform background of shades of blue or grey shared by these samples. This observation highlights that robust radii need to be chosen class dependent.

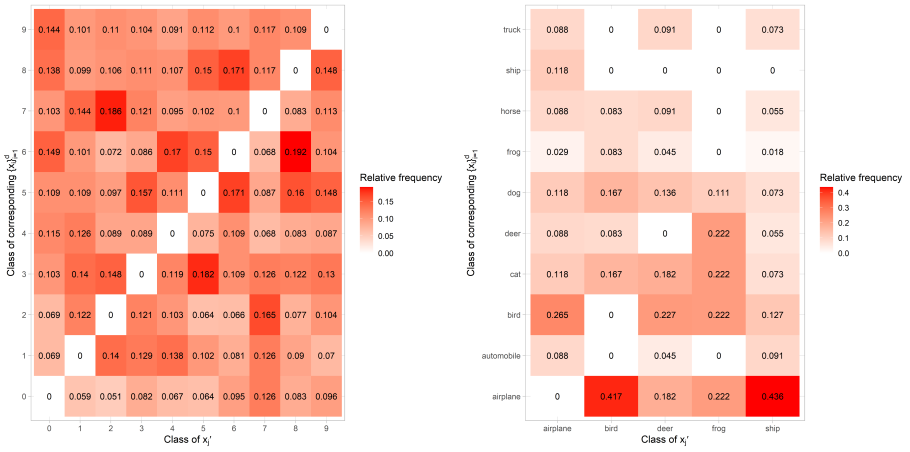
D Relationship with DeepFool

An adversarial example x_j^{adver} of model f is defined as a sample for which $f(x_j) \neq f(x_j^{\text{adver}})$ while $||x_j - x_j^{\text{adver}}||_p \leq \delta$. DeepFool [76] is an algorithm that finds an adversarial examples while minimizing the perturbation δ.

If we assume to have a *perfectly* robust model f with robust radius of $\geq 0.5r'_j$ for every sample (for complex datasets: $= 0.5r'_j$) then the closest point on the decision boundary is precisely in the middle of the line segment between x'_j and $\bar{x}'_j$, with distance $0.5r'_j$ from both. As the model's gradient points towards the direction of steepest ascent, it points towards the closest point on the decision boundary. As the model is perfectly robust the minimum distance to this point is $0.5r'_j$ and the point on the decision boundary is exactly the aforementioned middle of the line segment between x'_j and $\bar{x}'_j$. Thus, this middle of the line segment should also be the adversarial sample x_j^{adver} that is found by DeepFool, as DeepFool minimizes the introduced perturbation magnitude. As a result, more robust models should have the vectors $\bar{x}'_j - x'_j$ and $x_{\text{adver}} - x'_j$ being more aligned.

We compute the cosine similarities

$$\tilde{c}_j = \text{cosine}(x_{\text{adver}} - x'_j, \bar{x}'_j - x'_j) \in [-1, 1] \tag{14}$$



(a) SVHN. (b) CIFAR-10.

Fig. 8: Class pairs for x'_j and their corresponding $\bar{x}'_j$ for $x'_j \in \{\bar{x}\}^{\text{crit}}_{\text{global}}$.

for all samples $x_j \in X$ and display the distributions for the robust and non-robust neural network pairs. We observe that for the robust models the distributions contain significantly more positive values, even though those are nowhere near being perfectly robust. Thus, our method could be used in conjunction with DeepFool to investigate the distance and shape of the decision boundary around the critical points which are those that determine the introduction of label noise and the complexity of the decision boundary.

E Labelling Errors in SVHN

SVHN's [70] original train set contains three wrongly labelled samples that need to be removed before the computations in Section 4. Those are displayed in Figure 10.

F Additional Experimental Results for SVHN

In Figure 11 we present the results for SVHN when $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ are added to the original train set. As $|\{\bar{x}\}^{\text{crit}}_{\text{global}}| = 2,501$ (see Table 1), the addition of globally critical points to the train set make a non-negligible difference. Thus, generalization performance is decreased when perturbation magnitudes $\delta \geq R^*$ are introduced. This affirms the hypothesis in Section 5 that the reason for the *visibly* unchanged performance for CIFAR-10 is the comparably low number of $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ (see Figure 14). Thus, robust training for $\delta \geq R^*$, so the addition of $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ to the train set, also negatively affects CIFAR-10 training.

In Figure 12 we display the results when $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ are added to the original train set. We observe the same results as for the the addition of $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ for

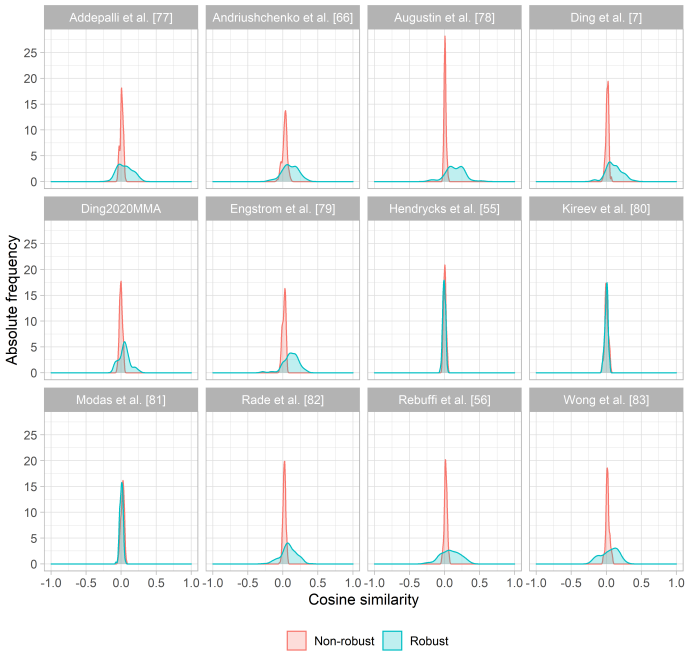


Fig. 9: Distribution of cosine similarities between $\bar{x}'_j - x'_j$ and $x_{\text{adver}} - x'_j$.

SVHN and the addition of $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ for CIFAR-10 to the original train sets. In all cases generalization performance deteriorates.

G Additional Experimental Results for FASHION

The FASHION dataset does not contain any $\{\bar{x}\}^{\text{crit}}_{\text{global}}$, so $0.5R$ is its actual robust radius. In Figure 13 we observe no difference in train and test performance when the $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ are added to the original train set. This result is expected as $R^* > 0.5R$ and the dataset is known to be simple and well separated which is further proven in Figure 18 as there is no class change between x'_j and their associated $\bar{x}'_j$.

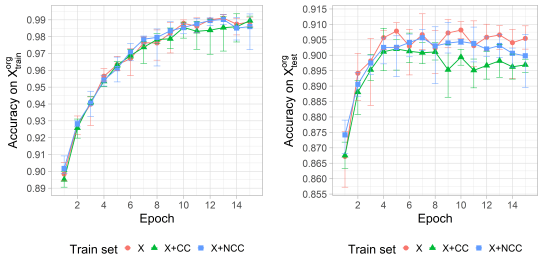
H Additional Experimental Results for CIFAR-10

In Figure 14 we display the results when trained with the addition of $\{\bar{x}\}^{\text{crit}}_{\text{global}}$. As $|\{\bar{x}\}^{\text{crit}}_{\text{global}}| = 132$, no difference in train and test performance is measurable.

In Table 7 we display predictions and confidences on $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ and observe the same results as described in Section 4. Firstly, robust models predict more images to have undergone a class change than non-robust models and thus having a more complex decision boundary. Secondly, robust models display significantly

(a) (b) (c)

Fig. 10: Labelling errors in the original SVHN train set. (a) Index : 11933, label : 5. (b) Index : 25235, label : 9. (c) Index : 65043, label : 1.



(a) (b)

Fig. 11: Results for SVHN. Error-bars denote minimum and maximum over five runs. (a) Mean accuracy on $X^{\text{org}}_{\text{train}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{global}}$. (b) Mean accuracy on $X^{\text{org}}_{\text{test}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{global}}$.

lower confidences on low-density samples compared to their non-robust counter parts and so a better calibrated.

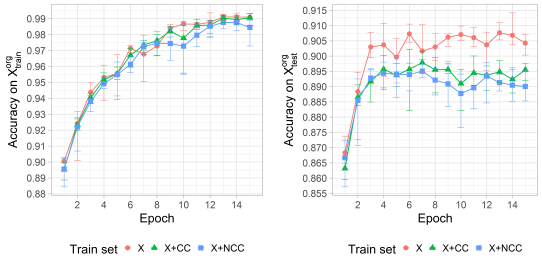
In Figure 15 we display example images from CIFAR-10 when applied the noise- and blur-corruptions provided by Hendrycks et al. [14].

I Further Examples of $\bar{x}'$

For SVHN we display random $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ Figure 16 and random $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ in Figure 17. For FASHION we display random $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ in Figure 18 and for CIFAR-10 we display random $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ in Figure 19.

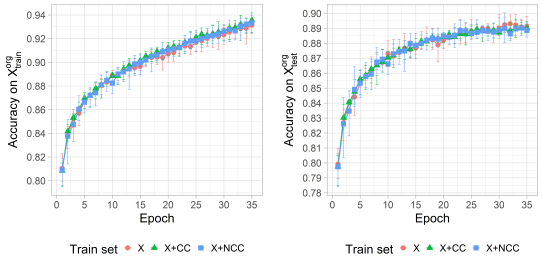
J Choice of the Robust Models

The particular choice of robust models was made because of computational reasons. The provided architectures have moderate numbers of parameters so retraining to remove their robust representation can be done on a single standard



(a) (b)

Fig. 12: Results for SVHN. Error-bars denote minimum and maximum over five runs. (a) Mean accuracy on $X^{\text{org}}_{\text{train}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{local}}$. (b) Mean accuracy on $X^{\text{org}}_{\text{test}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{local}}$.



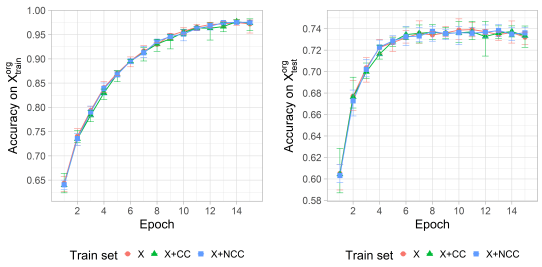
(a) (b)

Fig. 13: Results for FASHION. Error-bars denote minimum and maximum over five runs. (a) Mean accuracy on $X^{\text{org}}_{\text{train}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{local}}$. (b) Mean accuracy on $X^{\text{org}}_{\text{test}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{local}}$.

GPU with vanilla mini-batch gradient descent and Adam [73]. The robust pre-trained models are obtained from Croce et al. [72].

K Architecture of Convolutional Networks

The convolutional networks used for the experiments in Section 5 consist of five convolutional layers with batch normalization [84] and a single fully-connected layer with ReLU [85–87] activations implemented in pytorch [88]. Training was done with Adam [73] for 35 epochs and a batch-size of 128. No data augmentation or pre-training is used as to not distort the results.



(a) (b)

Fig. 14: CIFAR-10. Error-bars denote minimum and maximum over five runs. (a) Mean accuracy on $X^{\text{org}}_{\text{train}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{global}}$. (b) Mean accuracy on $X^{\text{org}}_{\text{test}}$ during training with $\{\bar{x}\}^{\text{crit}}_{\text{global}}$.

Fig. 15: Examples images for the noise and blur perturbations provided by Hendrycks et al. [14] in Table 5. Images on the far left side are the original ones.

Fig. 16: Example image-pairs of $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ (right) their associated x'_j (left) for SVHN. Multiple x'_j are associated with elements from $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ for different classes.

Table 7: Predictions and confidences of model f for $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ for CIFAR-10. Confidence values are reported as: mean $\pm$ standard deviation. NCC denotes no predicted class change by f and CC denotes a predicted class change between x'_j and $\bar{x}'_j \in \{\bar{x}\}^{\text{crit}}_{\text{local}}$.

Model f		$f(x'_j) = f(\bar{x}'_j)$ (NCC)		$f(x'_j) \neq f(\bar{x}'_j)$ (CC)	
		Fraction	Confidence	Fraction	Confidence
Addepalli et al. [77]	Non-robust	0.43	0.872 ± 0.169	0.57	0.796 ± 0.194
	Robust	0.58	0.447 ± 0.144	0.42	0.331 ± 0.092
Andriushchenko et al. [66]	Non-robust	0.32	0.913 ± 0.148	0.68	0.824 ± 0.184
	Robust	0.62	0.479 ± 0.164	0.38	0.370 ± 0.115
Augustin et al. [78]	Non-robust	0.32	0.849 ± 0.177	0.68	0.784 ± 0.190
	Robust	0.62	0.373 ± 0.162	0.38	0.270 ± 0.102
Ding et al. [7]	Non-robust	0.34	0.864 ± 0.169	0.66	0.798 ± 0.190
	Robust	0.57	0.904 ± 0.151	0.43	0.774 ± 0.194
Engstrom et al. [79]	Non-robust	0.34	0.839 ± 0.179	0.66	0.776 ± 0.191
	Robust	0.57	0.495 ± 0.182	0.43	0.372 ± 0.120
Hendrycks et al. [55]	Non-robust	0.32	0.898 ± 0.143	0.68	0.847 ± 0.165
	Robust	0.40	0.869 ± 0.171	0.61	0.788 ± 0.197
Kireev et al. [80]	Non-robust	0.30	0.827 ± 0.181	0.70	0.785 ± 0.193
	Robust	0.37	0.838 ± 0.188	0.63	0.756 ± 0.212
Modas et al. [81]	Non-robust	0.34	0.921 ± 0.138	0.66	0.868 ± 0.167
	Robust	0.46	0.528 ± 0.165	0.54	0.420 ± 0.148
Rade et al. [82] (*ddpm*)	Non-robust	0.35	0.895 ± 0.146	0.64	0.850 ± 0.167
	Robust	0.65	0.615 ± 0.181	0.35	0.446 ± 0.137
Rade et al. [82] (*extra*)	Non-robust	0.29	0.878 ± 0.160	0.71	0.805 ± 0.179
	Robust	0.53	0.490 ± 0.146	0.47	0.380 ± 0.103
Rebuffi et al. [56]	Non-robust	0.30	0.841 ± 0.170	0.70	0.780 ± 0.180
	Robust	0.65	0.569 ± 0.185	0.35	0.410 ± 0.128
Rice et al. [67]	Non-robust	0.35	0.906 ± 0.149	0.65	0.843 ± 0.185
	Robust	0.57	0.626 ± 0.196	0.43	0.466 ± 0.156
Wong et al. [83]	Non-robust	0.35	0.888 ± 0.158	0.65	0.816 ± 0.182
	Robust	0.56	0.535 ± 0.179	0.44	0.414 ± 0.131

Fig. 17: Example image-pairs of $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ (right) their associated x'_j (left) for CIFAR-10. Multiple x'_j are associated with elements from $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ for different classes.

Table 8: Predictions and confidences of model f for $\{\bar{x}\}^{\text{crit}}_{\text{global}}$ for CIFAR-10. Confidence values are reported as: mean ± standard deviation. NCC denotes no predicted class change by f and CC denotes a predicted class change between x'_j and $\bar{x}'_j \in \{\bar{x}\}^{\text{crit}}_{\text{global}}$. Full version of Table 2.

Model f		$f(x'_j) = f(\bar{x}'_j)$ (NCC)		$f(x'_j) \neq f(\bar{x}'_j)$ (CC)	
		Fraction	Confidence	Fraction	Confidence
Addepalli et al. [77]	Non-robust	0.64	0.925 ± 0.134	0.36	0.817 ± 0.170
	Robust	0.91	0.499 ± 0.128	0.09	0.281 ± 0.039
Andriushchenko et al. [66]	Non-robust	0.62	0.887 ± 0.154	0.38	0.708 ± 0.192
	Robust	0.79	0.535 ± 0.164	0.21	0.373 ± 0.098
Augustin et al. [78]	Non-robust	0.58	0.813 ± 0.192	0.42	0.666 ± 0.169
	Robust	0.95	0.314 ± 0.162	0.05	0.243 ± 0.051
Ding et al. [7]	Non-robust	0.52	0.913 ± 0.171	0.48	0.826 ± 0.174
	Robust	0.93	0.979 ± 0.057	0.07	0.791 ± 0.113
Engstrom et al. [79]	Non-robust	0.56	0.757 ± 0.215	0.44	0.571 ± 0.181
	Robust	0.86	0.537 ± 0.189	0.14	0.338 ± 0.113
Hendrycks et al. [55]	Non-robust	0.50	0.907 ± 0.144	0.50	0.778 ± 0.173
	Robust	0.76	0.905 ± 0.150	0.24	0.749 ± 0.161
Kireev et al. [80]	Non-robust	0.50	0.814 ± 0.195	0.50	0.726 ± 0.200
	Robust	0.61	0.905 ± 0.151	0.39	0.691 ± 0.209
Modas et al. [81]	Non-robust	0.60	0.900 ± 0.155	0.40	0.768 ± 0.176
	Robust	0.68	0.632 ± 0.155	0.32	0.419 ± 0.125
Rade et al. [82] (*ddpm*)	Non-robust	0.51	0.815 ± 0.178	0.49	0.601 ± 0.190
	Robust	0.95	0.670 ± 0.176	0.05	0.488 ± 0.068
Rade et al. [82] (*extra*)	Non-robust	0.52	0.867 ± 0.181	0.48	0.608 ± 0.182
	Robust	0.75	0.603 ± 0.153	0.25	0.413 ± 0.102
Rebuffi et al. [56]	Non-robust	0.50	0.844 ± 0.177	0.50	0.650 ± 0.171
	Robust	0.94	0.643 ± 0.202	0.06	0.389 ± 0.083
Rice et al. [67]	Non-robust	0.54	0.863 ± 0.168	0.46	0.677 ± 0.204
	Robust	0.92	0.635 ± 0.200	0.08	0.401 ± 0.072
Wong et al. [83]	Non-robust	0.43	0.873 ± 0.183	0.57	0.707 ± 0.185
	Robust	0.74	0.645 ± 0.187	0.26	0.458 ± 0.095

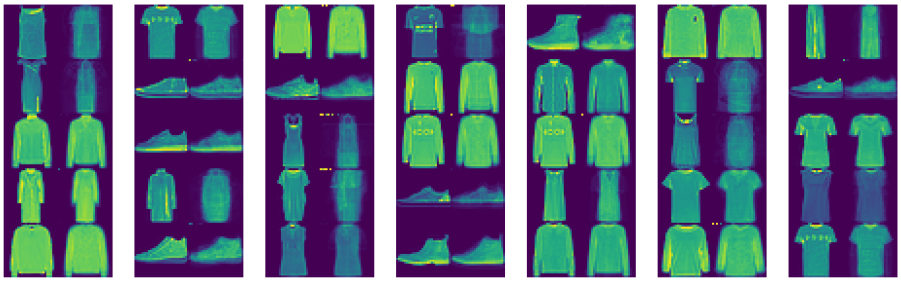


Fig. 18: Example image-pairs of $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ (right) their associated x'_j (left) for FASHION. Multiple x'_j are associated with elements from $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ for different classes.

Fig. 19: Example image-pairs of $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ (right) their associated x'_j (left) for CIFAR-10. Multiple x'_j are associated with elements from $\{\bar{x}\}^{\text{crit}}_{\text{local}}$ for different classes.