A. Details about Model-agnostic Invariance-based Attacks

Here, we give details about our model-agnostic invariance-based adversarial attacks on MNIST.

Generating ℓ_0-invariant adversarial examples. Assume we are given a training set $\mathcal{X}$ consisting of labeled example pairs $(\hat{x}, \hat{y})$. As input our algorithm accepts an example x with oracle label $\mathcal{O}(x) = y$. Image x with label $y = 8$ is given in Figure 4 (a).

Define $\mathcal{S} = \{\hat{x} : (\hat{x}, \hat{y}) \in \mathcal{X}, \hat{x} \neq y\}$, the set of training examples with a different label. Now we define $\mathcal{T}$ to be the set of transformations that we allow: rotations by up to 20 degrees, horizontal or vertical shifts by up to 6 pixels (out of 28), shears by up to 20%, and re-sizing by up to 50%.

We generate a new augmented training set $\mathcal{X}^* = \{t(\hat{x}) : t \in \mathcal{T}, \hat{x} \in \mathcal{S}\}$. By assumption, each of these examples is labeled correctly by the oracle. In our experiments, we verify the validity of this assumption through a human study and omit any candidate adversarial example that violates this assumption. Finally, we search for

$$x^* = \arg\min_{x^* \in \mathcal{X}^*} \|x^* - \hat{x}\|_0.$$

By construction, we know that x and x^* are similar in pixel space but have a different label. Figure 4 (b-c) show this step of the process. Next, we introduce a number of refinements to make x^* be "more similar" to x. This reduces the ℓ_0 distortion introduced to create an invariance-based adversarial example—compared to directly returning x^* as the adversarial example.

First, we define $\Delta = |x - x^*| > 1/2$ where the absolute value and comparison operator are taken element-wise. Intuitively, Δ represents the pixels that substantially change between x^* and x. We choose $1/2$ as an arbitrary threshold representing how much a pixel changes before we consider the change "important". This step is shown in Figure 4 (d). Along with Δ containing the *useful* changes that are responsible for changing the oracle class label of x, it also contains irrelevant changes that are superficial and do not contribute to changing the oracle class label. For example, in Figure 4 (d) notice that the green cluster is the only semantically important change; both the red and blue changes are not necessary.

To identify and remove the superficial changes, we perform spectral clustering on Δ. We compute Δ_i by enumerating all possible subsets of clusters of pixel regions. This gives us many possible **potential** adversarial examples $x_i^* = x + \Delta_i$. Notice these are only potential because we may not actually have applied the necessary change that actually modifies the class label.

We show three of the eight possible candidates in Figure 4. In order to alleviate the need for human inspection of each candidate x_i^* to determine which of these potential adversarial examples is actually misclassified, we follow an approach from Defense-GAN (Samangouei et al., 2018) and the Robust Manifold Defense (Ilyas et al., 2017): we take the generator from a GAN and use it to assign a likelihood score to the image. We make one small refinement, and use an AC-GAN (Mirza & Osindero, 2014) and compute the class-conditional likelihood of this image occurring. This process reduces ℓ_0 distortion by 50% on average.

As a small refinement, we find that initially filtering $\mathcal{X}$ by removing the 20% least-canonical examples makes the attack succeed more often.

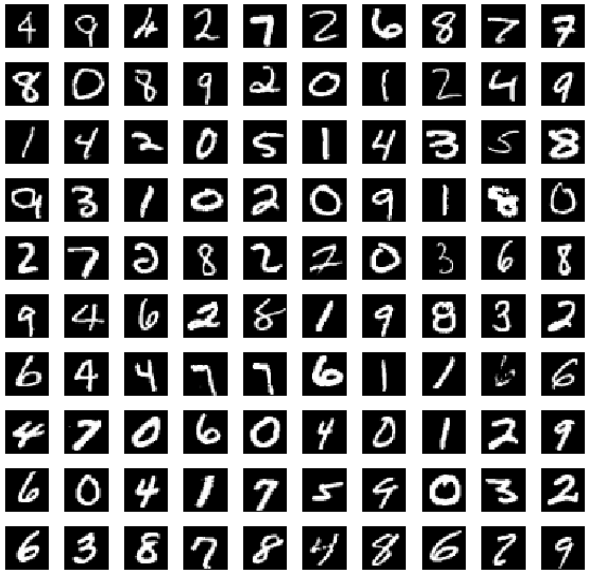
Generating ℓ_∞-invariant adversarial examples. Our approach for generating ℓ_∞-invariant examples follows similar ideas as for the ℓ_0 case, but is conceptually simpler as the perturbation budget can be applied independently for each pixel (our ℓ_∞ attack is however less effective than the ℓ_0 one, so further optimizations may prove useful).

We build an augmented training set $\mathcal{X}^*$ as in the ℓ_0 case. Instead of looking for the closest nearest neighbor for some example x with label $\mathcal{O}(x) = y$, we restrict our search to examples $x^* \in \mathcal{X}^*$ with specific target labels y^*, which we've empirically found to produce more convincing examples (e.g., we always match digits representing a 1, with a target digit representing either a 7 or a 4). We then simply apply an ℓ_∞-bounded perturbation to x by interpolating with x^*, so as to minimize the distance between x and the chosen target example x^*.

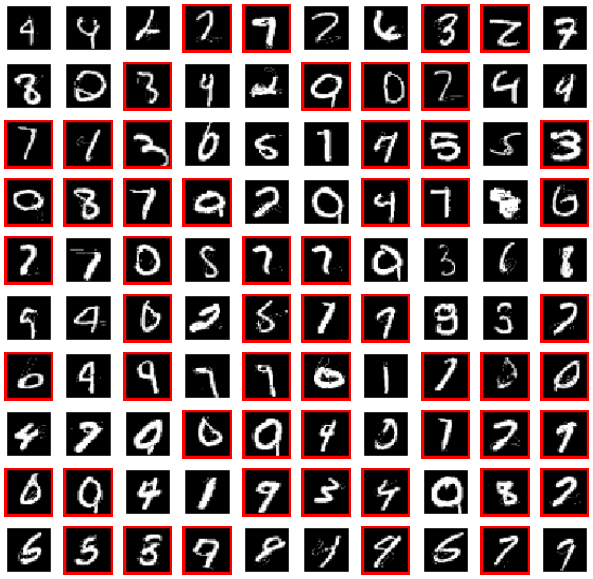
B. Complete Set of 100 Invariance Adversarial Examples

Below we give the 100 randomly-selected test images along with the invariance adversarial examples that were shown during the human study.

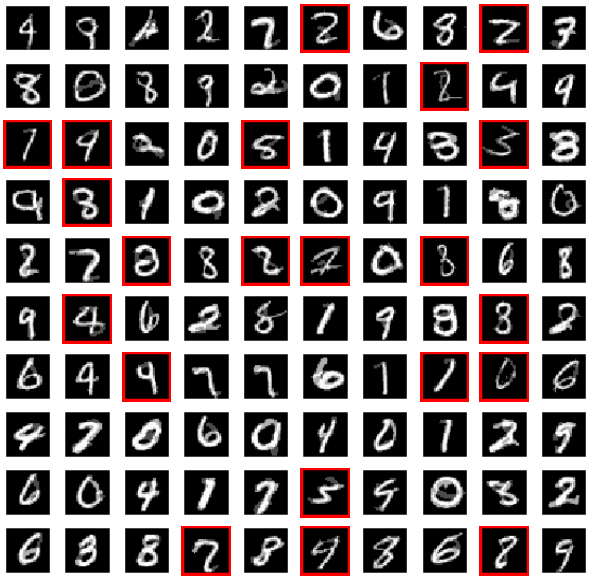
B.1. Original Images



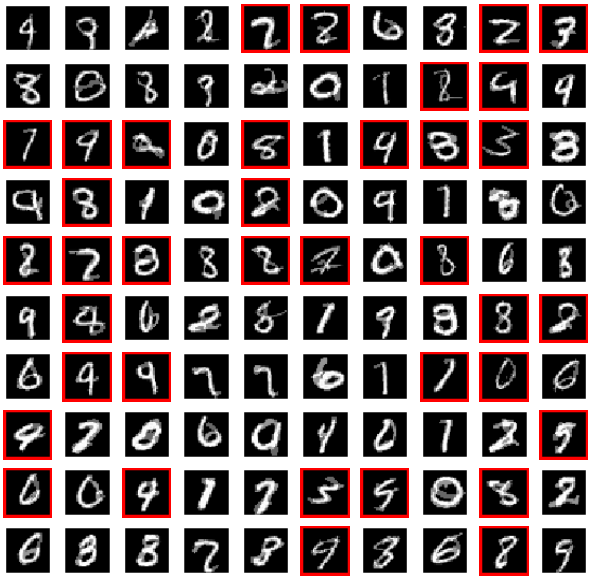
B.2. ℓ_0 Invariance Adversarial Examples



B.3. ℓ_∞ Invariance Adversarial Examples ($\varepsilon = 0.3$)



B.4. ℓ_∞ Invariance Adversarial Examples ($\varepsilon = 0.4$)



Agreement between model and the original MNIST label, for sensitivity-based adversarial examples						
Model:	**Undefended**	**ℓ_0 Sparse**	**Binary-ABS**	**ABS**	**ℓ_∞ PGD** ($\epsilon = 0.3$)	**ℓ_2 PGD** ($\epsilon = 2$)
ℓ_0 Attack ($\epsilon = 25$)	0%	45%	63%	43%	0%	40%
ℓ_∞ Attack ($\epsilon = 0.3$)	0%	8%	77%	8%	92%	1%
ℓ_∞ Attack ($\epsilon = 0.4$)	0%	0%	60%	0%	7%	0%

Table 3: Robust model accuracy with respect to the original MNIST labels under different threat models. For ℓ_∞ attacks, we use PGD (Madry et al., 2017). For ℓ_0 attacks, we use the PointwiseAttack of (Schott et al., 2019).

C. Details on Trained Models

In Section 4, we evaluate multiple models against invariance adversarial examples. Table 2 gives results for models taken from prior work. We refer the reader to these works for details. The undefended model is a ResNet-18.

Table 3 reports the standard test accuracy of these models against sensitivity-based adversarial examples. That is, the model is considered correct if it classifiers the adversarial example with the original test-set label of the unperturbed input. To measure ℓ_0 robustness, we use the PointwiseAttack of (Schott et al., 2019) repeated 10 times, with $\epsilon = 25$. For ℓ_∞ robustness, we use PGD with 100 iterations for $\epsilon = 0.3$ and $\epsilon = 0.4$. For the ABS and Binary-ABS models, we report the number from (Schott et al., 2019), for PGD combined with stochastic gradient estimation.

Trading Perturbation-Robustness and Invariance Robustness. The adversarially-trained models in Figure 6 use the same architecture as (Madry et al., 2017). We train each model for 10 epochs with Adam and a learning rate of 10^{-3} reduced to 10^{-4} after 5 epochs (with a batch size of 100). To accelerate convergence, we train against a weaker adversary in the first epoch (with $1/3$ of the perturbation budget). For training, we use PGD with 40 iterations for ℓ_∞ and 100 iterations for ℓ_1. For ℓ_∞-PGD, we choose a step-size of $2.5 \cdot \varepsilon/k$, where k is the number of attack iterations. For the models trained with ℓ_1-PGD, we use the Sparse ℓ_1-Descent Attack of Tramèr & Boneh (2019), with a sparsity fraction of 99%.

Below, we report the robust accuracy of these models against sensitivity-based adversarial examples, in the sense of equation 1.

	ϵ for ℓ_∞-PGD training			
Attack	0.1	0.2	0.3	0.4
PGD $\epsilon = 0.3$	0%	6%	92%	93%
PGD $\epsilon = 0.4$	0%	0%	7%	90%

Table 4: Robust model accuracy with respect to the original MNIST label for models trained against ℓ_∞ attacks.

	ϵ for ℓ_1-PGD training		
Attack	5	10	15
ℓ_0-PointwiseAttack ($\epsilon = 25$)	41%	59%	65%

Table 5: Robust model accuracy with respect to the original MNIST label for models trained against ℓ_1 attacks, and evaluated against ℓ_0 attacks.

The Role of Data Augmentation. The models in Figure 7 and Figure 8 are trained against an adversary that first rotates and translates an input (using the default parameters from (Engstrom et al., 2019b)) and then adds noise of ℓ_∞-norm bounded by ε to the transformed input. For training, we sample 10 spatial transformations at random for each input, apply 40 steps of ℓ_∞-PGD to each transformed input, and retain the strongest adversarial example. At test time, we enumerate all possible spatial transformations for each input, and apply 100 steps of PGD to each.

When training against an adversary with $\varepsilon \geq 0.25$, a warm-start phase is required to ensure training converges. That is, we first trained a model against an $\varepsilon = 0.2$ adversary, and then successively increases ε by 0.05 every 5 epochs.

D. Proof of Lemma 4

We recall and prove Lemma 4 from Section 3:

Lemma. *Constructing an oracle-aligned distance function that satisfies Definition 3 is as hard as constructing a function f so that $f(x) = \mathcal{O}(x)$, i.e., f perfectly solves the oracle's classification task.*

Proof. We first show that if we have a distance function `dist` that satisfies Definition 3, then the classification task can be perfectly solved.

Let x be an input from class y so that $\mathcal{O}(x) = y$. Let $\{x_i\}$ be any (possibly infinite) sequence of inputs so that $\texttt{dist}(x, x_i) < \texttt{dist}(x, x_{i+1})$ but so that $\mathcal{O}(x_i) = y$ for all x_i. Define $l_x = \lim_{i\to\infty} \texttt{dist}(x, x_i)$ as the distance to the furthest input from this class along the path x_i.

Assume that $\mathcal{O}$ is not degenerate and there exists at least

one input z so that $\mathcal{O}(z) \neq y$. If the problem is degenerate then it is uninteresting: *every* function $\texttt{dist}$ satisfies Definition 3.

Now let $\{z_i\}$ be any (possibly infinite) sequence of inputs so that $\texttt{dist}(x, z_i) > \texttt{dist}(x, z_{i+1})$ and so that $\mathcal{O}(z_i) \neq y$. Define $l_z = \lim_{i\to\infty} \texttt{dist}(x, z_i)$ as the distance to the closest input along z. But by Definition 3 we are guaranteed that $l_z > l_x$, otherwise there would exist an index I such that $\texttt{dist}(x, x_I) \geq \texttt{dist}(x, z_I)$ but so that $\mathcal{O}(x) = \mathcal{O}(x_I)$ and $\mathcal{O}(x) \neq \mathcal{O}(z_I)$, contradicting Definition 3. Therefore for any example x, *all* examples x_i that share the same class label are closer than *any* other input z that has a different class label.

From here it is easy to see that the task can be solved trivially by a 1-nearest neighbor classifier using this function $\texttt{dist}$. Let $S = \{(\alpha_i, y_i)\}_{i=1}^{C}$ contain exactly one pair (z, y) for every class. Given an arbitrary query point x, we can therefore compute the class label as $\arg\min \texttt{dist}(x, \alpha_i)$, which must be the correct label, because of the above argument: the closest example from any (incorrect) class is different than the furthest example from the correct class, and so in particular, the closest input from S *must* be the correct label.

For the reverse direction, assume we have a classifier $f(x)$ that solves the task perfectly, i.e., $f(x) = \mathcal{O}(x)$ for any $x \in \mathbb{R}^d$. Then the distance function defined as

$$\texttt{dist}(x, x') = \begin{cases} 0 & \text{if } f(x) = f(x') \\ 1 & \text{otherwise} \end{cases}$$

is aligned with the oracle. □

E. Proofs for the Overly-Robust Features Model

We recall the binary classification task from Section 5. Unlabeled inputs $x \in \mathbb{R}^{d+2}$ are sampled from some distribution $\mathcal{D}_k^*$ parametrized by $k > 1$ as follows:

$$z \overset{\text{u.a.r}}{\sim} \{-1, 1\}, \quad x_1 = z/2$$
$$x_2 = \begin{cases} +z & \text{w.p. } \frac{1+1/k}{2} \\ -z & \text{w.p. } \frac{1-1/k}{2} \end{cases}, \; x_3, \ldots, x_{d+2} \overset{\text{i.i.d}}{\sim} \mathcal{N}(\frac{z}{\sqrt{d}}, k) \; .$$

The oracle label for an input x is $y = \mathcal{O}(x) = \texttt{sign}(x_1)$. Note that for $k \gg 1$, features $x_2, \ldots, x_{d+2}$ are only weakly correlated with the label y. The oracle labels are robust to ℓ_∞-perturbations bounded by $\varepsilon = 1/2$:

Claim 5. *For any $x \sim \mathcal{D}^*$ and $\Delta \in \mathbb{R}^{d+2}$ with $\|\Delta\|_\infty < 1/2$, we have $\mathcal{O}(x) = \mathcal{O}(x + \Delta)$.*

Recall that we consider that a model is trained and evaluated on sanitized and labeled data from this distribution. In this data, the "noise" features $x_2, \ldots, x_{d+2}$ are more strongly correlated with the oracle labels y, and there is a small amount of label noise attributed to mistakes in the data labeling process. Specifically, we let $\alpha > 0$ and $\delta > 0$ be small constants, and define $\mathcal{D}$ as the following distribution:

$$x \sim \mathcal{D}_{1+\alpha}^*, \quad y = \begin{cases} +\mathcal{O}(x) & \text{w.p. } 1-\delta \\ -\mathcal{O}(x) & \text{w.p. } \delta \end{cases} .$$

We first show that this sanitization introduces spurious weakly robust features. Standard models trained on $\mathcal{D}$ are thus vulnerable to sensitivity-based adversarial examples.

Lemma 6. *Let $f(x)$ be the Bayes optimal classifier on $\mathcal{D}$. Then f agrees with the oracle $\mathcal{O}$ with probability at least $1-\delta$ over $\mathcal{D}$ but with 0% probability against an ℓ_∞-adversary bounded by some $\varepsilon = O(d^{-1/2})$.*

Proof. The first part of the lemma, namely that f agrees with the oracle $\mathcal{O}$ with probability at least $1-\delta$ follows from the fact that for $(x, y) \sim \mathcal{D}$, $\texttt{sign}(x_1) = y$ with probability $1-\delta$, and $\mathcal{O}(x) = \texttt{sign}(x_1)$. So a classifier that only relies on feature x_1 achieves $1-\delta$ accuracy. To show that the Bayes optimal classifier for $\mathcal{D}$ has adversarial examples, note that this classifier is of the form

$$\begin{aligned} f(x) &= \texttt{sign}(w^T x + C) \\ &= \texttt{sign}(w_1 \cdot x_1 + w_2 \cdot x_2 + \sum_{i=3}^{d+2} w_i \cdot x_i + C) \; , \end{aligned}$$

where w_1, w_2, C are constants, and $w_i = O(1/\sqrt{d})$ for $i \geq 3$. Thus, a perturbation of size $O(1/\sqrt{d})$ applied to features $x_3, \ldots, x_{d+2}$ results in a change of size $O(1)$ in $w^T x + C$, which can be made large enough to change the output of f with arbitrarily large probability. As perturbations of size $O(1/\sqrt{d})$ cannot change the oracle's label, they can reduce the agreement between the classifier and oracle to 0%. □

Finally, we show that there exists an overly-robust classifier on $\mathcal{D}$ that is vulnerable to invariance adversarial examples:

Lemma 7. *Let $f(x) = \texttt{sign}(x_2)$. This classifier has accuracy above $1-\alpha/2$ on $\mathcal{D}$, even against an ℓ_∞ adversary bounded by $\varepsilon = 0.99$. Under such large perturbations, f agrees with the oracle with probability 0%.*

Proof. The robust accuracy of f follows from the fact that $f(x)$ cannot be changed by any perturbation of ℓ_∞ norm strictly below 1, and that for $(x, y) \sim \mathcal{D}$, we have $x_2 = y$ with probability $\frac{1+1/(1+\alpha)}{2} \geq 1 - \alpha/2$. For any $(x, y) \sim \mathcal{D}$, note that a perturbation of ℓ_∞-norm above $1/2$ can always flip the oracle's label. So we can always find a perturbation Δ such that $\|\Delta\|_\infty \leq 0.99$ and $f(x+\Delta) \neq \mathcal{O}(x+\Delta)$. □