Mind the box: l_1-APGD for sparse adversarial attacks on image classifiers

Francesco Croce   Matthias Hein

University of Tübingen

International Conference on Machine Learning, 2021

Projected gradient descent (PGD) is commonly used for l_p-bounded adversarial attacks on image classifiers. It maximizes a loss L with the iterative scheme

$$u^{(i+1)} = x^{(i)} + \eta^{(i)} \cdot s(\nabla L(x^{(i)})) \quad (1)$$

$$x^{(i+1)} = P_S(u^{(i+1)}), \quad (2)$$

on the feasible set S, with P_S the projection onto S.

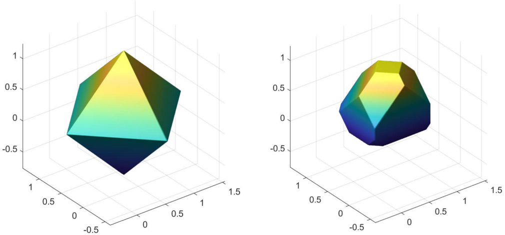
Note: unlike for the l_∞- and l_2-threat models, for PGD wrt l_1 there is **no standard version**, and the existing ones are less effective than other attacks.

For l_1 we need to explicitly consider the role of the image domain $[0, 1]^d$!

Then, we introduce an **adaptive** version of PGD, l_1-APGD, specific for the effective threat model l_1-ball $\cap [0, 1]^d$, which achieves SOTA performance.

Components of l_1-APGD

Projection step: existing versions of PGD for l_1 project first onto the l_1-ball $B_1(x, \epsilon)$, then clip to $[0, 1]^d$ (approximated projection).



Proposition 1

The projection problem onto $S = B_1(x, \epsilon) \cap [0, 1]^d$ can be solved in $O(d \log d)$.

Using the **exact projection** allows to better explore the feasible set compared to the approximated one, improving the performance of the attacks.

Update step: in PGD-based attacks the update step is usually done in the **steepest descent direction**. For the l_1-ball $\cap[0,1]^d$-threat model, we get

Proposition 2

Let $z_i = \max\{(1-x_i)\,sign(w_i), -x_i\,sign(w_i)\}$, π *the ordering such that* $|w_{\pi_i}| \geq |w_{\pi_j}|$ *for* $i > j$ *and* k *the smallest integer for which* $\sum_{i=1}^{k} z_{\pi_i} \geq \epsilon$. *The steepest descent direction in* $B_1(x,\epsilon) \cap H$ *is given elementwise by*

$$\delta^*_{\pi_i} = \begin{cases} z_{\pi_i} \cdot sign(w_{\pi_i}) & \text{for } i < k, \\ (\epsilon - \sum_{i=1}^{k-1} z_{\pi_i}) \cdot sign(w_{\pi_k}) & \text{for } i = k, \\ 0 & \text{for } i > k \end{cases}. \tag{3}$$

The sparsity of the steepest descent direction depends on the current point. Then, l_1-APGD uses updates with **adaptive sparsity**, unlike existing methods.

Experiments

We also adapt the **black-box** Square Attack (Andriushchenko et al., 2020) to l_1.

Table 1. **Low Budget ($\epsilon = 12$):** Robust accuracy achieved by the SOTA l_1 -adversarial attacks on various models for CIFAR-10 in the l_1-threat model with radius $\epsilon = 12$ of the l_1-ball. The statistics are computed on 1000 points of the test set. PA and Square are black-box attacks. The budget is 100 iterations for white-box attacks ($\times 9$ for EAD and +10 for B&B) and 5000 queries for our l_1-Square-Attack.

model	clean	EAD	ALMA	SLIDE	B&B	FAB^T	APGD$_{CE}$	PA	Square
APGD-AT **(ours)**	87.1	64.6	65.0	66.6	62.4	67.5	**61.3**	79.7	71.8
(Madaan et al., 2021)	82.0	55.3	58.1	56.1	55.2	56.8	**54.7**	73.1	62.8
(Maini et al., 2020) - AVG	84.6	51.8	54.2	53.8	52.1	61.8	**50.4**	77.4	68.4
(Maini et al., 2020) - MSD	82.1	51.6	55.4	53.2	50.7	54.6	**49.7**	72.7	63.5
(Augustin et al., 2020)	91.1	48.9	50.7	48.8	42.1	50.4	**37.1**	73.2	56.8
(Engstrom et al., 2019) - l_2	91.5	40.3	46.4	35.1	36.8	39.9	**30.2**	71.7	52.7
(Rice et al., 2020)	89.1	37.7	45.2	32.3	35.2	37.0	**27.1**	70.5	50.3
(Xiao et al., 2020)	79.4	44.9	74.5	33.3	72.6	78.9	41.4	36.2	**20.2**
(Kim et al., 2020)*	81.9	26.7	31.8	25.1	23.8	32.4	**18.9**	54.9	36.0
(Carmon et al., 2019)	90.3	25.1	18.4	19.7	18.7	31.1	**13.1**	60.8	34.5
(Xu & Yang, 2020)	83.8	20.1	24.0	18.2	14.7	27.8	**10.9**	57.0	32.0
(Engstrom et al., 2019) - l_∞	88.7	14.5	19.4	14.2	12.2	20.9	**8.0**	57.6	28.0

l_1-APGD outperforms the competitors, especially with low computational budget, and l_1-Square Attack gets better results than the existing black-box methods!

Thanks to l_1-APGD and l_1-Square Attack we can extend AutoAttack (Croce & Hein, 2020) to the l_1-threat model, to test robustness with no parameter tuning!

model	clean	EAD	ALMA	SLIDE	B&B	APGD$_{CE+T}$	WC	AA	rep.
APGD-AT **(ours)**	87.1	63.3	61.4	65.9	59.9	60.3	**59.7**	60.3	-
(Madaan et al., 2021)	82.0	54.5	54.3	55.1	51.9	51.9	**51.8**	51.9	55.0**
(Maini et al., 2020) - AVG	84.6	50.0	49.7	52.3	49.0	**46.8**	47.3	**46.8**	54.0
(Maini et al., 2020) - MSD	82.1	50.1	49.8	51.7	47.7	**46.5**	46.8	**46.5**	53.0
(Augustin et al., 2020)	91.1	46.0	42.9	41.5	32.9	31.1	31.9	**31.0**	-
(Engstrom et al., 2019) - l_2	91.5	36.4	34.7	30.6	27.5	27.0	27.1	**26.9**	-
(Rice et al., 2020)	89.1	33.9	32.4	28.1	24.2	24.2	**23.7**	24.0	-
(Xiao et al., 2020)	79.4	34.4	75.0	22.5	59.3	27.2	20.2	**16.9**	-
(Kim et al., 2020)*	81.9	24.4	22.9	19.9	15.7	15.4	**15.1**	**15.1**	81.18
(Carmon et al., 2019)	90.3	26.2	13.6	13.6	10.4	**8.3**	8.5	**8.3**	-
(Xu & Yang, 2020)	83.8	18.1	14.5	13.9	7.8	7.7	**6.9**	7.6	59.63
(Engstrom et al., 2019) - l_∞	88.7	12.5	10.0	8.7	5.9	**4.9**	5.1	**4.9**	-

l_1-AutoAttack improves the evaluation of robustness wrt l_1 on many classifiers!

Code available at https://github.com/fra31/auto-attack.