Supplementary Material: Adversarial Active Learning for Deep Networks

We point out specific cases in which we can obtain a significant improvement in the labeled data sample complexity using adversarial active learning for linear classifiers. We restrict our case of study to a specific case; which is when the data instances are drawn from the unit ball in $\mathbb{R}^2$ and their labels are drawn from ± 1. Notice that our proof may be extended to other distributions as long as they are uniformly distributed along with any dimension (*such as isotropic gaussian*). Throughout this section, our goal is to find a linear classifier f going through the origin, so that its expected true loss is as small as possible. The error is induced by the classification rule $2\mathbb{I}(f(x) \geq 0) - 1$ where $\mathbb{I}(\cdot)$ is the set of indicator functions. We consider the following classification error loss defined as l(f(x),y)=1 if $yf(x) \leq 0$ and $l(f(x), y) = 0$ otherwise.

Firstly, we detail our strategy when the labels are consistent with a linear separator going through the origin. While we knew already from the literature that active learning is highly beneficial for such a case, ensuring a need of $\tilde{\mathcal{O}}(d\ \ln(\frac{1}{\epsilon}))$ labeled examples, given ϵ as the error rate and *d* the dimension, we will see how adversarial active queries help to diminish the effective numbers of labels queried.

Indeed, (Balcan, Broder, and Zhang 2007) demonstrated that to obtain an exponential improvement in the label sampled complexity, one needs to sample the examples from a subregion carefully chosen and not from the entire region of uncertainty. When sampling uniformly along the unit ball, few samples lie in such low confidence regions. Although, to achieve an error rate of $2^{-(k+1)}$ at the k-th iteration, we still need to add $\tilde{\mathcal{O}}(2^k d)$ unlabeled samples[1], we can automatically guess the ground-truth labels of the majority of them. Given the current linear classifier c_k consistent on the labeled examples at iteration k and a given threshold b_k, every unlabeled sample x_k lying further from the decision boundary than b_k is necessarily predicted correctly by the current classifier c_k. This result relies on the assumption made on the data distribution and its separability using a linear classifier (Balcan, Broder, and Zhang 2007). When sampling uniformly queries and considering $b_k = 2^{-k}\pi$, we can estimate the probability for any sample x to be part of the low confidence regions as $p(\mid c_k \cdot x \mid \leq b_k) = \tilde{\mathcal{O}}(2^{-k}\sqrt{d})$. Hence, in the original strategy proposed in (Balcan, Broder, and Zhang 2007), a human annotator effectively annotates $\tilde{\mathcal{O}}(d^{\frac{3}{2}})$ unlabeled samples at each iteration to obtain an exponential improvement in the error rate.

Here we argue how adversarial queries may help to reduce the number of effective labels at any iteration $k > 1$.

Transferable adversarial attacks

When it comes to deep networks, their adversarial attacks can transfer across many other models: adversarial examples generated for a specific model will often mislead other unseen networks. Such a property is commonly known as transferability. However, transferability has been mainly observed empirically (Goodfellow, Shlens, and Szegedy 2015). Up to our knowledge, how to understand the underlying phenomena and how to defend against them effectively are still open questions. Meanwhile, (Tanay and Griffin 2016) have investigated the phenomenon of adversarial examples for binary linear classifiers. They proposed a new taxonomy to classify adversarial attacks: they defined the notion of *adversarial strength* and show that it can be reduced to the deviation angle between the classifier considered and the nearest centroid classifier (*i.e the bissecting hyperplane between positive and negative samples*).

The probability of transferability of an adversarial attack directly depends on the level of regularization used; more specifically to the deviation angle between the classifier and the bissecting hyperplane between positive and negative samples.

Based on the notion of *adversarial strength*, we define **weak** adversarial examples. Weak adversarial examples will not transfer to any other consistent classifier, other than the one they have been designed for. They result from a lack of regularization, which can be improved by adding the adversarial sample to the training set. Similarly, as for DFAL, we can use twice the same label for any sample and its weak adversarial counterpart. If one is able to design weak adversarial examples given a labeled sample x, then we can increase the training set without corrupting it. Eventually, the weak adversarial sample will have the same label as x.

We detail the procedure to build weak adversarial attacks for linear classifiers in Theorem 1. To build our adversarial attacks, we stick to the standard of the litterature by adding a

[1] according to the VC dimension of linear classifiers

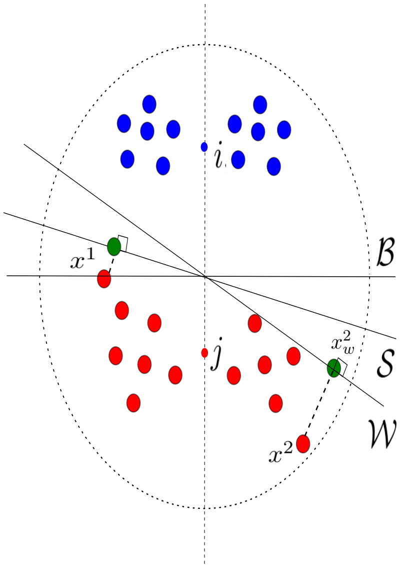

Figure 1: Toy problem: learning a linear separator that predicts with no error the labels of positive instances ●, and negative instances ●. We illustrate the notion of weak adversarial examples ● on two samples x^1 and x^2.

perturbation along the gradient direction (Goodfellow, Shlens, and Szegedy 2015). The strength of the adversarial example is directly impacted by the deviation angle and the magnitude of the perturbation. We illustrate our strategy on a 2-dimensional toy problem in Figure 1. Consider instances distributed in a circle such that positive and negative points may be well separated given a linear classifier going through the origin. $\mathcal{B}$, in accordance with Definition 1, is the bissecting line between positive and negative points. For a sake of clarity, we centered $\mathcal{B}$ to go through the origin. Several optimal separators coexist. Among them, we consider the one which maximizes its angle with $\mathcal{B}$ (*i.e $\mathcal{W}$ in def 2*), and the one wich minimizes it (*i.e $\mathcal{S}$ in Definition 2*). Every other solution necessarily lies between $\mathcal{W}$ and $\mathcal{S}$.

We describe for two points x^1 and x^2 in Figure 1 how to build their weak adversarial counterparts, based on Definition 2. Notice that a necessary condition is that both points x^1 and x^2 considered are well predicted by our strong and weak classifiers. The mirror projection of x^1 given $\mathcal{S}$ will lie in the hypothesis space (*a.k.a in the area between $\mathcal{W}$ and $\mathcal{S}$*). When it comes to x^2, projecting it on $\mathcal{W}$ ensures that every consistent classifier will predict x^2_w as a negative instance.

Definition 1 (Bissecting Hyperplane) *According to (Tanay and Griffin 2016), we define the bissecting hyperplane $\mathcal{B}$ as a unique linear separator of unit vector* $\mathbf{b}$ *and bias b_0 such that $\mathcal{B}$ reflects the mean of positive instances on the mean of negative instances. Note that $\mathcal{B}$ is not necessarily part of the hypothesis space, nor $\mathcal{B}$ minimizes the error on $X \times Y$.*

$$j = i - 2(i \cdot \mathbf{b} + b_0)\mathbf{b} \; s.t \;\; i = \mathbb{E}[X \mid Y = 1], \; j = \mathbb{E}[X \mid Y = -1]$$

Definition 2 (Transferable adversarial attacks) *Given $\mathcal{L}(X \times Y)$ the set of optimal classifiers given the task at hand, we define two boundary classifiers: $\mathcal{S}$ the strong linear classifier of unit vector* $\mathbf{s}$*, and $\mathcal{W}$ the weak linear classifier of unit vector* $\mathbf{w}$*. $\mathcal{S}$ is consistent with the training set and minimizes the deviation angle with $\mathcal{B}$. $\mathcal{W}$ is consistent with the training set and maximizes the deviation angle with $\mathcal{B}$.*

$$\mathcal{L}(X \times Y) = \{\mathcal{C} \mid \forall\, (x, y) \in X \times Y \; y(x \cdot \mathbf{c}) > 0\} \quad (1)$$

$$\mathcal{S} = argmin_{\mathcal{S} \in \mathcal{L}(X \times Y)} \mathbf{s} \cdot \mathbf{b} \quad (2)$$

$$\mathcal{W} = argmax_{\mathcal{W} \in \mathcal{L}(X \times Y)} \mathbf{w} \cdot \mathbf{b} \quad (3)$$

$$(4)$$

$\forall (x, y) \in X \times Y$ we define its weak adversarial attack, $\tilde{x}_w$, based on the following:
- *if* $|\mathbf{w} \cdot x| \leq |\mathbf{s} \cdot x|$*:* $\tilde{x}_w = x - (x \cdot \mathbf{w})\mathbf{w}$
- *if* $|\mathbf{s} \cdot x| \leq |\mathbf{w} \cdot x|$*:* $\tilde{x}_w = x - (x \cdot \mathbf{s})\mathbf{s}$

Theorems 1 (Weak Adversarial Examples) $\forall (x, y) \in X \times Y$, $y(\tilde{x}_w \cdot \mathbf{c}) \geq 0$

Notice that our definition of adversarial attacks does not match exactly the common definition as we do not restrict our adversarial attacks to be close to their target sample anymore.

Proof of Theorem 1

♠ We consider samples from the unit ball from a binary task. The dataset is centered around the origin We also assume that the task at hand is linearly separable by a normalized linear classifier going through the origin. Given i, j the mean of respectively positive and negative points in X; we can deduce the nearest centroid classifier $\mathcal{B}$ with unit vector $\mathbf{b}$ and bias b_0: the bissecting hyperplane separating at best i and j. Notice that $\mathcal{B}$ is not necessarily minimizing the error. However, necessarily, $\mathcal{B}$ predicts i as positive and j as negative.

Since the problem is consistent with linear classifiers without bias, we denote by $\mathcal{L}$ the set of optimal classifiers of norm 1 and going through the origin: $\mathcal{L} = \{\mathcal{W} \mid \forall x \in X, \; y(x)\langle w, x\rangle > 0\}$. Also among those classifiers, we call *weak classifier $\mathcal{W}$*, the classifier minimizing the error with the largest deviation angle given $\mathbf{b}$, in accordance with the Definition 2. Moreover, we denote *strong classifier $\mathcal{S}$*, the classifier minimizing the error with the smallest deviation angle given $\mathbf{b}$.

Firstly, we demonstrate that what we call the *deviation angle* (the angle between a classifier in $\mathcal{L}$ and $\mathcal{B}$) lies in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$, as detailed in Lemma 1.

Lemma 1 (Deviation Angle) *$\forall \mathcal{C} \in \mathcal{L}$ we can express its unit vector* $\mathbf{c}$ *given the unit vector* $\mathbf{b}$ *of the bissecting angle and a unit vector* $\mathbf{b}^{\perp}_{c}$*:*

$$\mathbf{c} = cos(\delta_c)\mathbf{b} + sin(\delta_c)\mathbf{b}^{\perp}$$

Thus we obtain $cos(\delta_c) \geq 0$ which implies that the deviation angle lies in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$.

♣ By linearity any optimal classifier predicts i as positive and j as negative: $\mathbf{c} \cdot i \geq 0$, $\mathbf{c} \cdot j \leq 0$. Also, since $\mathcal{B}$ is the bissecting hyperplane, we can express i, and j using $\mathbf{b}$ and the signed distance to the hyperplane $d(\cdot, \mathcal{B})$. Note that since $\mathcal{B}$ predicts i as positive we have $d(i, \mathcal{B}) > 0$.

$$j = i - 2d(i, \mathcal{B})\mathbf{b} \quad (5)$$
$$i = j - 2d(j, \mathcal{B})\mathbf{b} \quad (6)$$
$$(7)$$

Thus, for the orthogonal vector $\mathbf{b}^{\perp}$ to $\mathbf{b}$, we have : $(i \cdot \mathbf{b}^{\perp}) = (j \cdot \mathbf{b}^{\perp})$. Eventually we can express the dot product $c \cdot (i - j)$, which is positive, using $\mathbf{b}$ and $\mathbf{b}_c^{\perp}$.

$$\begin{aligned} \mathbf{c} \cdot (i - j) &= 2d(i, \mathbf{B})\mathbf{c} \cdot \mathbf{b} && (8) \\ &= 2d(i, \mathbf{B})(cos(\delta_c)\mathbf{b} \cdot \mathbf{b} + sin(\delta_c)\mathbf{b}_c^{\perp} \cdot \mathbf{b}) && (9) \\ &= 2d(i, \mathbf{B})cos(\delta_c) && (10) \end{aligned}$$

Finally, equation implies $cos(\delta_c) \geq 0$

Weak adversarial attack: We prove it by contradiction. Assume $\exists\, x$ s.t $\exists\, \mathcal{C} \in \mathcal{L}$ which wrongly predicts x_w. Thus it implies $(\mathbf{c} \cdot x_w)(\mathbf{c} \cdot x) < 0$ Without loss of information, we assume that the *weak classifier* is the closest boundary to x. Thus $x_w = x - (\mathbf{w} \cdot x)\mathbf{w}$.

$(\mathbf{c} \cdot x_w)(\mathbf{c} \cdot x) < 0$

$(\mathbf{c} \cdot x)(\mathbf{c} \cdot x) - (x \cdot \mathbf{w})(\mathbf{c} \cdot \mathbf{w})(\mathbf{c} \cdot x) < 0$

$(\mathbf{c} \cdot x)^2 < [cos(\delta_w)cos(\delta_c) + sin(\delta_w)sin(\delta_c)](x \cdot \mathbf{w})(\mathbf{c} \cdot x)$

$(\mathbf{c} \cdot x)^2 < cos(\delta_w - \delta_c)(x \cdot \mathbf{w})(\mathbf{c} \cdot x)$

Because $\mathcal{W}$ and $\mathcal{C}$predict the same label

for x we obtain a necessary condition:

$$cos(\delta_w - \delta_c) \geq \frac{(\mathbf{c} \cdot x)^2}{(x \cdot \mathbf{w})(\mathbf{c} \cdot x)}$$

$$cos(\delta_w - \delta_c) \geq \frac{\mathbf{c} \cdot x}{x \cdot \mathbf{w}}$$

Because we picked $\mathbf{w}$ instead of $\mathbf{s}$ as $\mathbf{w}$ minimizes its distance with the sample x, then $\frac{\mathbf{c} \cdot x}{x \cdot \mathbf{w}} > 1$ which contradicts the previous inequality. Thus any other classifier than $\mathcal{W}$ will predict the same label for both x, x_w. When it comes to $\mathcal{W}$, x_w lies on the boundary, thus it can be assumed to share the same label.

Label Complexity on the unit ball

Here we argue how weak adversarial queries help to reduce the number of effective labels at any iteration $k > 1$. Our active learning strategy consists in adding also weak adversarial instances to the training set when it is relevant, as proposed for deep networks with DFAL. Thus we will reduce the effective need of queries by a ratio of two at best. Indeed, weak adversarial instances are relevant if and only if the sample queried is already well predicted by the current weak and strong classifiers. In Theorem 2, we describe further the expected improvement in terms of human annotations.

A first observation is that projecting the unit ball according to any hyperplane going through the origin corresponds to the identity mapping. Consequently, when adding weak adversarial examples in the training set, we do not modify the underlying distribution of the instance space. Moreover, the main advantage of our adversarial examples is that for any instance lying in the low confidence region, its weak adversarial examples will also lie in that subregion (*Lemma 2*). It means that when using adversarial queries, we respect the *i.i.d* assumption, and query relevant samples, as illustrated in Figure 2. Finally the number of artificial queries that can be added mostly depend on the generalization error at the current iteration: when a sample query is correctly predicted, we can add its weak adversarial attacks.

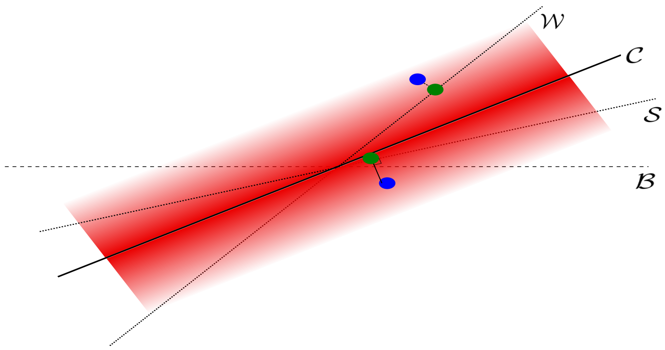

Figure 2: Repartition of weak examples ● for samples ● lying in the low confidence subregion of a consistent classifier $\mathcal{C}$

Lemma 2 (Low confidence region) $\forall (x, y) \in X \times Y$, $\forall \mathcal{C} \in \mathcal{L}(X \times Y)$*:* $\forall \alpha \in \mathbb{R}^+$ *such that* $\mid \mathbf{c} \cdot x \mid \leq \alpha$ *then we have* $\mid \mathbf{c} \cdot \tilde{x}_w \mid \leq \alpha$

Theorems 2 (Convergence of adversarial queries) *Given* $n = \tilde{\mathcal{O}}(d^{\frac{3}{2}})$ *the effective number of labels to query at iteration k. We denote the generalization error at step k,* $p_k = 2^{-(k+1)}$.

Using our adversarial strategy (adding both $\tilde{x}_w$ *and* $\tilde{x}_s$*), we can reduce the effective number* m_k *of labels with high probability* $\delta > 0$ *up to:*

$$m_k = \min\{m \geq \frac{n}{2} \mid \binom{m-1}{m-\frac{n}{2}}(1-p_k)^{\frac{n}{2}} p_k^{m-\frac{n}{2}} \geq \delta\} \quad (11)$$

Proof of Theorem 2

♠ Theorem 2 results from the number of successes from a Bernouilli law. If we assume that the probability of misclassification of $\mathcal{W}$ and $\mathcal{S}$ are independent, then we have probability $\frac{p_k}{2}$ to be able to build a query for an unlabeled sample at step k. In this case, we will add two samples to the training set instead of one. Consequently, Theorem 2 relies on Lemma 2.

Lemma 2 Consider a threshold α so that $|\mathbf{c} \cdot x| \leq \alpha$. Without loss of information, we assume that the *weak classifier* is the closest boundary to x. Thus $x_w = x - (\mathbf{w} \cdot x)\mathbf{w}$.

- Without loss of information we assume $(\mathbf{c} \cdot x) \geq 0$

$$
\begin{aligned}
cos(\delta_c)cos(\delta_w)(\mathbf{w}\cdot x) &\geq 0\\
(\mathbf{c}\cdot x) - cos(\delta_c)cos(\delta_w)(\mathbf{w}\cdot x) &\leq \alpha\\
(\mathbf{c}\cdot x) - (\mathbf{w}\cdot\mathbf{c})(\mathbf{w}\cdot x) &\leq \alpha\\
(\mathbf{c}\cdot x_w) &\leq \alpha
\end{aligned}
$$

References

[Balcan, Broder, and Zhang 2007] Balcan, M.-F.; Broder, A.; and Zhang, T. 2007. Margin based active learning. *Learning Theory* 35–50.

[Goodfellow, Shlens, and Szegedy 2015] Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. *ICLR 2015*.

[Tanay and Griffin 2016] Tanay, T., and Griffin, L. 2016. A boundary tilting persepective on the phenomenon of adversarial examples. *arXiv preprint arXiv:1608.07690*.