We thank the reviewers for their efforts. Below we address the main comments.

Reviewer #1

The reviewer asked how the techniques in this paper differ from the prior work that established the hardness of neural networks under the RSAT assumption. First, we show that learning neural networks is hard already for networks with a special structure that we call sign-CNN. It requires a reduction from the RSAT problem that is different from the reduction used in prior work. We do use in this reduction a Lemma from [17], so we do not repeat the work that has been done there. Second, we show that the properties of sign-CNNs allow us to transform an unknown sign-CNN to a random network, by a multiplication with a random matrix that has a special structure. This method is new, and does not appear in prior work. Finally, in order to bound the support of the input distribution, we need to analyze the singular values of our special-structure random matrices.

The reviewer also commented on the presentation. We did make much effort to present the results and proofs in a simple and clear way. In the "proof ideas" and "proof structure" sections we give a high-level overview of the proofs. In the appendix, we first establish some needed lemmas, and then each theorem is proved in a separate subsection. Since all theorems require common lemmas and constructions, skipping directly from the "results" section to the subsection in the appendix where the theorem is proved is not possible. Also, for the same reason, giving a sketch of each proof in the "results" section is not possible. Instead, the "proof ideas" section is essentially a sketch of the proofs.

Reviewer #2

The reviewer asked about the worst-case nature of the input distribution. Prior hardness results for learning neural networks assume that both the input distribution and the weights are worst-case. We show that the problem is hard already for simple networks with natural weights, but the worst-case nature of the input distribution remains. Thus, we show that even very strong assumptions on the network are not sufficient for efficient learning, and therefore that assumptions on the input distribution are necessary. We believe that positive results on the learnability of neural networks would require a combination of assumptions on the weights and on the input distribution.

Reviewer #4

The reviewer raised three concerns:

1. "The assumption that the adversary can choose a worst-case input distribution is too strong and impractical":
   The standard PAC-learning framework requires an algorithm that learns successfully for every input distribution and every hypothesis in the class. Hence, the assumption that the adversary can choose a worst-case input distribution is the standard assumption in PAC learning. While prior hardness results for learning neural networks assume worst-case input distribution and worst-case weights, we show that the problem is hard already for networks with natural weights (but the worst-case nature of the input distribution remains). The practical importance of our result, is that it suggests that in order to establish efficient algorithms for learning neural networks, or to show polynomial-time guarantees for existing algorithms, assumptions on the network's architecture and weights are not sufficient, and assumptions on the input distribution are necessary. Hence, our result may help focussing the efforts on directions where positive results are possible. We believe that a combination of assumptions on the weights and on the input distribution is necessary in order to obtain positive results.
2. "The weights on the second layer are all 1. This assumption breaks the "naturalness" of the network. Is this essential or can it be generalized?":
   The choice to focus on networks where the weights in the second layer are all 1 is not essential. Our results hold also where the weights in the second layer have other fixed values, and also in the case where they are random. We may add a remark on this point.
3. "The result only holds if the hidden dimension $m = O(log^2(n))$, which makes the result less interesting. Is there a way to prove similar results with more hidden nodes?":
   The results do extend to more hidden nodes. As we mention in page 4 (lines 168-170), the results hold (with minor changes) for any $m = \omega(log(n))$, including networks with many hidden neurons. We chose to focus on $m = O(log^2(n))$ since we wanted to show hardness of learning already where the number of hidden neurons is relatively small.

Reviewer #5

We thank the reviewer for his/her comments and will fix accordingly.