We would like to thank all the reviewers for their thoughtful feedback. We are happy that reviewers found our work novel, the paper well-written, and our empirical evaluation well-organized and interesting. We address your comments and questions below.

Scalability.

(R1, 2, 3) A primary concern is the scalability of our approach. The following reasons elucidate why the paper is interesting and state-of-the-art with its current degree of scalability.

1. Prior work [12, 53] guarantees monotonicity by restricting the hypothesis class, making it easier to scale. For example, linear models restricted to positive weights are very scalable and monotonic. However, restricting the hypothesis class leads to less expressive models, which can reduce their practical applicability. Ours is the **first work** to guarantee monotonicity for ReLU networks, without restricting the model architecture. Providing such guarantees for arbitrary ReLU networks, even for the size used in our paper, is a challenging problem.
2. Our experiments use real-world models and sizes that are state-of-the-art and have natural monotonicity constraints. Model architectures were chosen based on grid search and not based on their simplicity. From Figure 3, we can see that average monotonic prediction time for these models is less than 1 second.
3. When violating monotonicity leads to safety, ethical or legal problems, and PR disasters, the question is not whether we can scale monotonicity enforcement, but whether it is safe to use machine learning at all. In this context, the computational price of enforcing monotonicity, even if it ends up being significant, is entirely warranted.
4. Our algorithm relies on off-the-shelf SMT solvers; as there are improvements in automated solvers, our approach will directly benefit from the improvements. Future works can also investigate other efforts to improve scalability in solvers. As a concrete example, as (R4) suggested, one can compare prediction performance when using MaxSAT instead of OMT.

Robustness.

(R2) "This algorithm also appears to lack robustness". We respectfully disagree, in fact our approach **improves robustness**. First, we want to clarify that the monotonic envelope is constructed on the learned function and not on the data. Therefore, individual data outliers will not affect it too much. Second, if the function to be learned is naturally monotonic, enforcing invariants counteracts noise and outliers, leading to improved robustness. To illustrate the improvement, we added five outlier points to the WEIGHT feature of AUTO-MPG regression dataset. Table 1 shows that our approach produces **more robust models** with COMET improving baseline MSE. Since this is such an appealing advantage of our proposed technique, we will add these robustness experiments to the final paper.

Table 1: COMET is more robust than the baseline model (NN_b)

Model	Without Outliers	With Outliers
NN_b	9.33±3.22	13.54±4.65
COMET	8.92±2.93	10.54±1.98

(R2) "Data augmentation could cause dangerous drift away from the original training data." Our approach guards against this in multiple ways. First, data augmentation with counterexamples is recomputed for each batch at every epoch. This ensures that: 1) an incorrect old counterexample does not burden the learning, and 2) learning incorporates multiple counterexamples at a time and so is less sensitive to any particular one. Second, the labeling heuristic for counterexamples (see lines 259-267 Section 4.1, page 6) provides a smoother loss with respect to monotonicity. As evidenced in our evaluation (see Table 2), there is no drift in the model quality. The quality of our model is similar or better than a model trained without monotonicity constraints.

Other Questions and Comments.

(R1, 3) "The presented method can only be applied to networks with ReLU activations." As noted by R2, other than the SMT encoding, the approach is **not limited** to ReLU activation functions. For other activation functions, given the encoding and an SMT solver with appropriate theory, we can directly use our algorithms. It is a non-trivial interesting future research to encode non-linear activation functions such as tanh without approximations in SMT, and we will add more discussion in the paper.

(R1, 2, 3, 4) Additional Feedback: We thank the reviewers for further suggestions. We will take these into account for improving the final draft of the paper. As suggested by (R4), we will simplify definitions and add pseudo-code to detail our algorithms further. Also, note that our code will be publicly available for reproducibility.