We thank all reviewers for their helpful and constructive feedback. We will address all minor issues as everyone mentioned. From both positive and negative reviews, we believe all reviewers read our paper carefully, and we appreciate it. Should we address your main concerns well, we hope you improve your score accordingly.

To all, particularly to R3 who misunderstands our error assumption and derivation of the objective. In statistical learning, a training example (x, y) is a realized random variable (X, Y) drawn from some unknown probability distribution $\mathbb{P}(x, y)$. In regression, we do not need to model $\mathbb{P}$, but rather only $p(y|x)$. For example, in l_2 regression, we assume $p(y|x)$ is Gaussian: $p(y|x) = \mathcal{N}(y; g(x), \sigma)$, where $g(x)$ is the mean and σ the standard deviation (STD). Under this assumption, the error $g(X) - Y$ is Gaussian, irrespective of what the marginal distribution of X is. In contrast, our key idea is to **avoid** making assumptions on $p(y|x)$. Instead, we only assume **the error** $g(X) - Y$ is Gaussian. Such an assumption is weaker: if $p(y|x)$ is Gaussian, then the error must be Gaussian; however, if the error is Gaussian, $p(y|x)$ does not have to be Gaussian. For example, for Gaussian mixture $p(y|x) = \sum_{i=1}^{k} w_i N(y; g_i(x), \sigma)$, the error $\epsilon(X, Y) = g_i(X) - Y$ is Gaussian with zero mean and variance σ^2: $p(\epsilon(x, y)) = N(\epsilon(x, y); 0, \sigma)$.

Then we propose to use a function $f_\theta(X, Y)$ to approximate the error $\epsilon(X, Y)$. This can be accomplished by maximizing the likelihood that $f_\theta(X, Y)$ is a zero-mean Gaussian for the given data. This objective has a trivial solution. Hence, the implicit function theorem is applied to ensure there exists some implicit function to express y in terms of x around each training point, and results in the additional term $(\frac{\partial f_\theta}{\partial y} + 1)^2$. As R2 pointed out, we could add a parameter to weight the two parts differently. For this work, we opted for the simplest approach with fewer hyperparameters.

To R2, R3, R4. Let us clarify how to make predictions and S_{local}. The loss l_θ is used, instead of finding $f_\theta(x, y) = 0$, because it encodes the full constraints during learning that were used to identify the modes. Finding points y that have low $l_\theta(x, \cdot)$ ensures we find the most plausible set of conditional modes. As mentioned in the text, arguably one of the most important limitations of this approach is that it might find spurious modes. We address this issue explicitly, in Section 3.2, both proposing a method to reduce the likelihood of spurious modes and showing that the simpler approach is often itself quite robust to spurious modes. The strategy to avoid such spurious modes relies on using l_θ for prediction, to sufficiently constrain the set of possible candidates. As for S_{local}, our idea is to take advantage of the residual. It is different with conventional l_2 regression, where the prediction function is fixed after training. Our training process can be thought of as constructing many implicit prediction functions. Those prediction functions are defined by both parameters θ and the input (x, y) itself. When searching for y given an input x, it is unlikely to reconstruct an prediction function which is exactly the same as one of those learned during training. That's why S_{local} makes sense: those modes with higher likelihood should have lower residual.

To R2, R3. L106 negative sampling. We actually meant negative sampling is problematic and we avoid it.

To R2. For some of your remaining concerns. 1) Thm 1, out of support issue. In theory, Thm1 tells that the spurious mode should be mostly eliminated by small enough u. In practice, the empirical results show that the gradient condition by itself often fixes the problem of spurious modes. 2) We use KDE to learn the joint distribution and use the same way to find maxima for each x. 3) Hyperparameter tuning. There is no standard way to do model selection in modal regression. Hence the best thing we can do is to ensure fair comparison: we optimize each algorithm's testing error by choosing best parameter setting. In fact, we sweep over larger range of hyperparameters for our competitors. We can definitely include some discussions. 4) High-frequency dataset. Yes, we use global mode. Source of randomness: both. 5) Where f_θ converges to. At the modes, f_θ is almost zero. The objective pushes f_θ outside of the modes to be non-zero, and encourages f_θ to have a derivative of 1 as much as possible between modes. We have figures showing f_θ and can definitely add them.

To R3. 1) EBMs are clearly different. They model the joint distribution of (X, Y), but we model the error. You also claim that our method has the same problem as EBMs:"negative sampling of (x, y) pairs ... " However, we **never** do negative sampling, it only appears one time in the paper. Our approach actually avoid negative sampling. 2) Other probabilistic models. Like other regression approaches, we are motivated by the principle: model only what you need, and not more. MDN is a reasonably representative probabilistic model—which is why its chosen for the experiments—but we can highlight a few more models in the intro that could be used for this problem. 3) We will cite the given references, including about score matching which is different but relevant.

To R3, R4. There is some concern about the efficiency of the approach. In this first work, our primary goal was to investigate the viability of this (first) parametric approach to modal regression. We have not yet focused on smarter algorithms for obtaining y during prediction. The method intuitively scales well with increased dimensionality in x and dataset size, in contrast to previous nonparametric modal regression algorithms. This already is a victory, and facilitates the use of modal regression in a broader range of settings. For higher-dimensional outputs, we do at least have an obvious strategy of gradient descent to search for minimal y; but more work needs to be done to understand scalability for higher-dimensional outputs.