[[Reviewer 1]] Thank you for your feedback. We will definitely release our code along with the camera-ready version of the manuscript. ■ **Fitting to training data:** The advantage of fitting the meta-model on sampled feature points is that the accuracy of the meta-model would not be limited by the size of the training data. However, if the meta-model is meant to be optimized w.r.t the feature distribution, then one can fit the feature distribution, say using a GAN or a kernel density function, and sample feature points from the estimated distribution to train the meta-model. Fitting the meta-model directly on training data will correspond to a 2-layer neural network with Meijer-G function as activation functions (see Figure 3). While this is very interesting, it departs from the main objective of the paper and demands a separate analysis on generalization performance, so we will add this discussion in the supplementary material. ■ **Loss function & regularization:** The loss function should be selected based on the application, e.g., if f is a classifier, then ℓ should be a cross-entropy loss. The idea of adding a regularization term is also very interesting although it is not straightforward. We will investigate using the number of poles and zeros as a penalty term as it is a natural measure of the complexity of a G function. We will add a discussion on loss functions and regularization in the final manuscript.

[[Reviewer 2]] Thank you for your helpful comments and suggestions. ■ **Interpretability of complicated functions:** As mentioned in lines 75 and 89, different functional forms are deemed interpretable in different applications. Bessel functions (and other special functions) are very common in empirical physics and material sciences (e.g. wave and field equations are modeled with such functions [3, 4]). (Please also refer to response Significance & applicability for Reviewer 3.) The theoretical justification of our framework was provided in Section 3.1, where we have shown that — based on the Kolomogorov superposition theorem — our approach can approximate any multivariate continuous function. ■ **Complexity tuning:** Our algorithm explores the Pareto front of simplicity vs. predictivity systematically in two ways: (1) it uses Bayesian optimization to conduct hyper-parameter search by picking the smallest number of poles and zeros for the Meijer-G function (i.e., simplest functional form) that best fits the model, and (2) it uses polynomial Chebyshev approximations to simplify meta-models with complex functional forms (Algorithm 1). We will emphasize this in the final manuscript. ■ **Fitting to training data:** Please kindly refer to response Fitting to training data for Reviewer 1. ■ **Loss function:** Our framework does not pose limitations on the loss function being used: any differentiable loss function (e.g., cross-entropy) can be used instead of the L-2 loss in Equation (2). ■ **Convexity:** In general, optimizing symbolic models with arbitrary non-linearity cannot be formulated as a convex optimization problem unless strict prior assumptions on the symbolic functions (e.g., linearity) are made (as in [8, 14]). This is why symbolic regression models resort to search algorithms based on genetic programming, which also does not guarantee a global solution [23-25]. Moreover, most of the competitive deep learning-based baselines such as DeepLIFT and L2X also use gradient descent. A key strength of our framework is that for the first time, flexible symbolic modeling can be conducted efficiently via gradient descent rather than exhaustive search heuristics. We believe this to be a strength of our method and not a weakness. ■ **Extra references:** We will add all the suggested references in the final the manuscript.

In addition, we have implemented two of the requested baselines and incorporated the results into Sections 5.1 and 5.2. The two baselines are: the additive GP by Duvenaud et al. and ANOVA GP by Kaufman et al.. As shown in the following Table, we found that neither baselines outperformed our model for experiment 5.2. Our interpretation for these results is that the additive GP kernel decomposition cannot capture the intricate interactions between (overlapping) feature subsets learned by the reference XGBoost model.

	AUC-ROC
SM	0.8651 ± 0.0045
Additive GP	0.8502 ± 0.0062
ANOVA GP	0.8498 ± 0.0053

[[Reviewer 3]] Thank you for your valuable comments. ■ **Significance & applicability:** As mentioned in lines 66-76 and Section 4, our method is applicable to the wide range of setups where a model's feature importance, interactions or explicit equations are essential for understanding its instance-wise predictions or uncovering the sources of its performance gain. We demonstrated the significance of our algorithm through the exemplary medical application in Section 5.2, which entailed explaining the predictions of a complex model for breast cancer, and helped recover new feature interactions that were unknown in the clinical literature. We will make sure that these aspects regarding the significance of our work are clearly stated in the camera-ready version of the paper. ■ **Empirical evaluation:** By virtue of the Kolomogorov superposition theorem [28], our algorithm can model any multivariate continuous function regardless of its dimensionality and the richness of its internal feature representations. Our algorithm is in fact more advantageous for more complex models since gradient descent is more efficient in large parameter spaces compared to black-box optimization methods which scale exponentially with the number of parameters. In the final manuscript, we will add the AUC-ROC performance of symbolic regression (SR) to Table 3. The run-time of SR on this dataset was 3.5 times longer than our algorithm. The functional form of the equation in line 267 was the same in all 5 runs, and the variability of the coefficients across runs was statistically insignificant. We will report the variance of the coefficients in line 267 in the supplementary material. ■ **Influence of hyper-parameters:** More complex models require more poles and zeros (hyper-parameters) for the corresponding meta-model. We tuned the hyper-parameters in Section 5.2 using Bayesian optimization. ■ **Related literature:** In the final version of the paper, we will make it clear that our framework does not encompass the line of research including LRP, PatternAttribution/Net, DeepTaylor, etc, and will point out to the unifying nature of the SHAP framework. ■ **Limits on symbolic expressions:** Our approach is not limited to additive meta-models: as can be seen in equation (5), our meta-models comprise composite (nested) functions of additive functions of the form $\sum_j f_j(g_1^j(x_1) + \ldots + g_n^j(x_n))$. By expanding these composite functions (e.g., using Taylor's expansion) we can recover rich multiplicative terms similar to those in the expression trees of genetic models.