# Multiplicative-Additive Constrained Models: Toward Joint Visualization of Interactive and Independent Effects

**Fumin Wang***
Independent Researcher

## Abstract

Interpretability is one of the considerations when applying machine learning to high-stakes fields such as healthcare that involve matters of life safety. Generalized Additive Models (GAMs) enhance interpretability by visualizing shape functions. Nevertheless, to preserve interpretability, GAMs omit higher-order interaction effects (beyond pairwise interactions), which imposes significant constraints on their predictive performance. We observe that Curve Ergodic Set Regression (CESR), a multiplicative model, naturally enables the visualization of its shape functions and simultaneously incorporates both interactions among all features and individual feature effects. Nevertheless, CESR fails to demonstrate superior performance compared to GAMs. We introduce Multiplicative-Additive Constrained Models (MACMs), which augment CESR with an additive part to disentangle the intertwined coefficients of its interactive and independent terms, thus effectively broadening the hypothesis space. The model is composed of a multiplicative part and an additive part, whose shape functions can both be naturally visualized, thereby assisting users in interpreting how features participate in the decision-making process. Consequently, MACMs constitute an improvement over both CESR and GAMs. The experimental results indicate that neural network–based MACMs significantly outperform both CESR and the current state-of-the-art GAMs in terms of predictive performance.

## 1 Introdution

Neural networks (NNs) have become a cornerstone of artificial intelligence (AI), with wide-ranging applications in areas such as computer vision [12] and natural language processing [9]. Despite their strong predictive capabilities, NNs are inherently complex high-dimensional function approximators, making their decision-making processes opaque and difficult to interpret. As a result, these models are often considered "black boxes." In high-stakes domains like healthcare, this lack of transparency poses serious challenges: when errors occur, it can be difficult to identify their sources or apply targeted corrections, undermining user trust and limiting reliability. Moreover, in fields such as biology, interpretability can be even more valuable than predictive accuracy, providing novel insights and guiding scientific discovery [14].

To address the opacity of black-box models, the field of Explainable AI (XAI) has received increasing attention, resulting in a rapid expansion of related literature and techniques alongside the growth of deep learning [2]. However, most XAI methods provide post-hoc explanations, attempting to approximate an inherently uninterpretable model using surrogate models, gradients, feature importance measures, or other statistical tools [7]. For instance, saliency maps in computer vision

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

Preprint. Under review.

highlight regions or pixels that strongly influence the model's output [6]. While they offer some insight, saliency maps often fail to reveal how inputs are actually used in decision-making and can be unreliable, as different methods produce inconsistent results. In general, post-hoc explanations approximate the model rather than uncovering its true logic, and this approximation may compromise accuracy, potentially introducing misleading interpretations in high-stakes scenarios [11].

In contrast, Interpretable Machine Learning (IML) focuses on designing models whose decision-making processes are inherently understandable and trustworthy [18]. Examples include generalized linear models [13], score-based methods [21], and decision trees [16]. These models are typically structured to satisfy desirable properties such as monotonicity, additivity, sparsity, causality, or other domain-specific constraints [4]. Generalized Additive Models (GAMs) [10], as an extension of generalized linear models, capture nonlinear relationships between predictors and outcomes through flexible shape functions $f_i(*)$, typically univariate and visualizable. GAMs can be expressed as:

$$g(E[y]) = \beta + f_1(x_1) + f_2(x_2) + \cdots + f_k(x_k), \tag{1}$$

where $g(\cdot)$ is the link function, $X = (x_1, \ldots, x_k)$ represents the independent variables, $y$ is the dependent variable, and each $f_i(\cdot)$ is a shape function that facilitates interpretability by explicitly showing each feature's contribution. Although GAMs are a long-established technique, their inherent interpretability has recently rekindled interest, motivating the development of approaches such as GAMI-Net [25], Sparse Interaction Additive Networks [8], and other neural-network-based extensions. These methods primarily focus on incorporating pairwise interactions or combining GAMs with neural networks to balance interpretability and predictive accuracy, setting the stage for further advances in interpretable modeling. However, traditional additive models are typically limited to modeling independent effects and often fail to capture complex feature interactions. Even GA$^2$Ms, which incorporate pairwise interactions, are unable to represent interactions beyond two variables. This restricts their expressiveness, particularly in domains where such interactions are critical to understanding the underlying processes.

Building on recent advances, particularly the proposed Curve Ergodic Set Regression (CESR) [22], we investigate an alternative formulation that seeks to address the limitations of additive models. CESR is expressed as:

$$y_C = C \cdot \prod_{i=1}^{k} \left(1 + w_{i1} x_i^1 + \cdots + w_{in_i} x_i^{n_i}\right) = C \cdot U_1(x_1) \cdot U_2(x_2) \cdots U_k(x_k), \tag{2}$$

where each $U_i(\cdot)$ is a univariate shape function. By construction, CESR naturally incorporates both independent and higher-order interaction effects, which in principle should enable stronger predictive performance than GAMs. However, in practice, its effectiveness is limited because the coefficients of the independent terms and the interaction terms are entangled rather than separated, leading to instability during training and reduced performance.

In this paper, we propose Multiplicative-Additive Constrained Models (MACMs), which jointly determines outcomes through both multiplicative and additive parts. This model has the following characteristics:

•MACM is a constrained model designed to remain interpretable while achieving performance as close as possible to its full-complexity counterpart.

•MACM is a novel interpretable machine learning approach that considers both the independent effects of each feature on the outcome and the interactions among all features-not just pairwise interactions.

•Compared to the CESR, the introduction of additive part decouples the coefficients of independent and interaction terms, while also expanding the model's hypothesis space—thereby enhancing both accuracy and expressiveness.

•Both the multiplicative and additive parts of the model can be visualized as graphs, where their product or sum comprehensively represents the decision-making process. Unlike traditional additive models, which solely capture independent effects of features, the proposed model introduces a novel perspective. It enables users to examine, from a global viewpoint, how the interactions among all features collectively influence the model's output.

•The shape functions in this model are not restricted to polynomials; they can be any functional mappings. In this paper, we further introduce neural networks based MACMs by employing full connected layer as shape functions, enhancing the model's expressive power and accuracy.

## 2 Related Works

As Ergodic Set Regression (ESR) and Curve Ergodic Set Regression (CESR) constitute the cornerstone and motivation of this work, this section presents a concise review and description of these relatively niche models.

### 2.1 Ergodic Set Regression

Ergodic Set Regression (ESR) [22] is an enhanced multivariate polynomial regression (MPR) model [19], constructed on a specialized set of polynomial power terms $[E_N(X)]$, where $X$ denotes the input vector and $N$ is the maximum polynomial degree. All elements in this set are generated from the following function:

$$\left(1 + x_1^1 + \cdots + x_1^{n_1}\right) \cdot \left(1 + x_2^1 + \cdots + x_2^{n_2}\right) \cdot \cdots \cdot \left(1 + x_k^1 + \cdots + x_k^{n_k}\right). \tag{3}$$

The general form of ESR can then be written as:

$$ESR : [W_{lin}] \times [E_N(X)]^T, \tag{4}$$

where $[W_{lin}]$ denotes the linearized coefficients. Owing to the specific construction of $[E_N(X)]$, ESR leverages a more comprehensive set of polynomial terms compared with standard MPR under the same polynomial degree, thereby achieving higher accuracy. For example, given $X = (x_1, x_2)$, when the maximum polynomial degree is $(1, 1)$ for ESR and 1 for MPR, the term expansion in ESR is $[1, x_1, x_2, x_1x_2]$, while that in MPR is limited to $[1, x_1, x_2]$.

Despite both ESR and MPR residing in high-dimensional function spaces, ESR exhibits an inherent structural property that allows it to be readily transformed into an interpretable model.

### 2.2 Curve Ergodic Set Regression

Curve Ergodic Set Regression (CESR) is a constrained variant of ESR. Similar to ESR, CESR employs $[E_N(X)]$ as the basis for model construction; however, it introduces nonlinearity through its parameterization. By examining (3), one can naturally associate it with the following nonlinear coefficients derived from the following form:

$$(1 + w_{1,1} + \cdots + w_{1,n_1}) \cdot (1 + w_{2,1} + \cdots + w_{2,n_2}) \cdot \cdots \cdot (1 + w_{k,1} + \cdots + w_{k,n_k}). \tag{5}$$

When the nonlinear coefficient matrix is applied, the model will be interpretable, and CESR can be expressed as:

$$\begin{aligned} CESR &: [W_{nonli}] \times [E_N(X)]^T \\ &= C \cdot \prod_{i=1}^{k} \left(1 + w_{i1}x_i^1 + \cdots + w_{in_i}x_i^{n_i}\right) \\ &= C \cdot U_1(x_1) \cdot U_2(x_2) \cdot \cdots \cdot U_k(x_k), \end{aligned} \tag{6}$$

where $U_i(x_i)$ denotes the shape function of feature $x_i$. Each $U_i(\cdot)$ can be visualized as a curve, enabling intuitive interpretation of both individual contributions and the overall decision-making process. Notably, $U_i(\cdot)$ always passes through $(0, 1)$. This property arises because the bias of $U_i(\cdot)$ is extracted outside the function, so that the constant $C$ becomes the product of all biases. Consequently, $C$ serves to normalize the shape functions, avoiding misleading model behavior and eliminating the issue of infinitely many identical local minima that could otherwise obscure interpretation.

Expanding (6) while preserving all terms but reordering them yields:

$$\begin{aligned} C \cdot [1 &+ (w_{11}x_1^1 + \cdots + w_{1n_1}x_1^{n_1}) + \cdots + (w_{k1}x_k^1 + \cdots + w_{kn_k}x_k^{n_k}) \\ &+ w_{11}w_{21}x_1^1x_2^1 + \cdots + \prod_{i=1}^{k}(w_{in_i}x_i^{n_i})] \end{aligned} \tag{7}$$

In essence, as shown in 7, CESR can be regarded as a constrained subset of ESR: it remains a complex polynomial model but achieves interpretability by imposing nonlinear structure on its parameters. CESR provides users with a new perspective: it illustrates the multiplicative effect of all factors on the prediction while simultaneously retaining the power terms contained in polynomial based GAMs By contrast, in GAMs, achieving interpretability typically requires discarding all interaction terms.

As shown in (7), CESR incorporates not only multiplicative components (interaction terms) but also additive components (independent terms) that constitute GAMs. Theoretically, this enables CESR to represent more complex functions than polynomial based GAMs, thereby offering higher potential accuracy. However, due to the nonlinear parameterization, the coefficients of additive and multiplicative components are inherently coupled. For instance, $w_1$ in the additive component serves as the coefficient of $x_1$ but simultaneously contributes to the coefficients of interaction terms involving $x_1$, such as $x_1x_2, x_1x_2x_3, \ldots, \prod_{i=1}^{k}(x_i^{n_i})$. Such coupled coefficients prevent the hypothesis space of CESR from encompassing that of GAMs. In practice, as demonstrated in Table 1, this coupling often prevents CESR from outperforming polynomial based GAMs.

Table 1: AUC on Pima Indian Diabetes Dataset[5]. Under the condition that the maximum polynomial degree is set to 7, the results are obtained through 5-fold cross-validation.

| model | AUC |
|---|---|
| CESR | $0.8493 \pm 0.154$ |
| GAMs(Poly) | $0.8533 \pm 0.117$ |

## 3 Multiplicative-Additive Constrained Models

### 3.1 Decoupling of Coefficients and the Expanded Hypothesis Space

To decouple the coefficients of independent and interaction terms in CESR, we introduce an additive polynomial into the model. The resulting formulation consists of a multiplicative part and an additive part:

$$\prod_{i=1}^{k}\left(w_{i0}^{m} + w_{i1}^{m}x_i^1 + \cdots + w_{in_i}^{m}x_i^{n_i}\right) + \sum_{i=1}^{k}(w_{i0}^{a} + w_{i1}^{a}x_i^1 + \cdots + w_{in_i}^{a}x_i^{n_i}), \tag{8}$$

where $w_{ij}^{m}$ and $w_{ij}^{a}$ denote the coefficients of the multiplicative and additive parts, respectively. The polynomial order of each feature is kept consistent across the two parts, but their coefficients are parameterized independently. Expanding this model yields:

$$\begin{aligned}
&(\prod_{i=1}^{k}(w_{i0}^{m}) + \sum_{i=1}^{k}(w_{i0}^{a})) \\
&+ (w_{11}^{m} + w_{11}^{a})x_1^1 + \cdots + (w_{1n_1}^{m} + w_{1n_1}^{a})x_1^{n_1} + \cdots + (w_{kn_1}^{m} + w_{kn_k}^{a})x_k^{n_k} \\
&+ w_{11}^{m}w_{21}^{m}x_1^1x_2^1 + \cdots + \prod_{i=1}^{k}(w_{in_i}^{m}x_i^{n_i})
\end{aligned} \tag{9}$$

Here, the first row represents the constant term, the second row corresponds to the independent terms, and the third row captures the interaction terms. Compared with (7), the additive part introduces free coefficients that adjust the independent terms, thereby decoupling them from the interaction terms. This decoupling broadens the hypothesis space and enhances the representational capacity of the model. To illustrate, consider the target function:

$$y = 1 + x_1 + (1 + k)x_2 + x_1x_2, k \neq 0.$$

For CESR (or the multiplicative part alone) $\prod_{i=1}^{2}(w_{i0}^{a} + w_{i1}^{a}x_i)$ to approximate $y$ without error, it must satisfy:

$$w_{10}^{m}w_{20}^{m} = 1, w_{11}^{m}w_{20}^{m} = 1, w_{11}^{m}w_{21}^{m} = 1,$$

which further implies $w_{10}^{m}w_{21}^{m} = 1$. Thus, the expansion of the multiplicative part becomes $1 + x_1 + x_2 + x_1x_2$, leading to an error of $|kx_2|$. Since the multiplicative part cannot capture this discrepancy, CESR fails to exactly represent $y$, as well as the additive part which alone cannot

model the interaction term $x_1x_2$. By incorporating the additive part $\sum_{i=1}^{2}(w_{i0}^a + w_{i1}^a x_i)$ and setting $w_{10}^a = w_{11}^a = w_{21}^a = 0, w_{20}^a = k$, the full model becomes $1 + x_1 + x_2 + x_1x_2 + kx_2$, which exactly matches $y$. Hence, the additive part serves to decouple and correct coefficients, enabling the proposed model to represent functions that CESR cannot.

From another perspective, the introduction of the additive part enlarges the hypothesis space of the multiplicative part. Let

$$\mathcal{H}_m = \{\prod_{i=1}^{k}\left(w_{i0}^m + w_{i1}^m x_i^1 + \cdots + w_{in_i}^m x_i^{n_i}\right) | x_i \in \mathbb{R}^k, k \geq 2\}$$

and

$$\mathcal{H}_a = \{\sum_{i=1}^{k}\left(w_{i0}^a + w_{i1}^a x_i^1 + \cdots + w_{in_i}^a x_i^{n_i}\right) | x_i \in \mathbb{R}^k, k \geq 2\}$$

represent the hypothesis spaces of the multiplicative and additive part, respectively. When the number of features $k \geq 2$, $\mathcal{H}_m$ and $\mathcal{H}_a$ have little intersection. Consider the following equation:

$$\prod_{i=1}^{k}\left(w_{i0}^m + w_{i1}^m x_i^1 + \cdots + w_{in_i}^m x_i^{n_i}\right) = \sum_{i=1}^{k}\left(w_{i0}^a + w_{i1}^a x_i^1 + \cdots + w_{in_i}^a x_i^{n_i}\right).$$

To ensure formal equivalence, for any non-zero coefficient $w_{qv}^a$ of $x_q^v (v \neq 0)$, one necessary condition is

$$w_{qv}^m = w_{qv}^a, w_{qv}^m * w_{ij}^m = 0, i \neq q, i \neq 0, j = 0, \cdots, n_c.$$

This implies $w_{ij}^m = 0$ for all $i \neq q$, which means the left-hand side reduces to a univariate function, i.e., $k = 1$, contradicting the assumption $k \geq 2$. Therefore,

$$dim(\mathcal{H}_m + \mathcal{H}_a) > max\{dim(\mathcal{H}_m), dim(\mathcal{H}_a)\}.$$

Hence, $\mathcal{H}_m$ and $\mathcal{H}_a$ have little intersection when $k \geq 2$, and the essence of combining the multiplicative and additive part is to expand the hypothesis space of CESR or polnomial based GAMs, thereby granting their combination greater expressive power in theory.

### 3.2 The Form of Multiplicative-Additive Constrained Models

Unlike neural networks, which possess theoretically universal approximation capabilities [3], polynomials cannot approximate arbitrary functions with arbitrary precision [20]. This limitation inherently constrains the performance of Equation (8). To overcome this, we generalize Equation (8) by removing the restriction to polynomial bases. The resulting framework is referred to as Multiplicative-Additive Constrained Models (MACMs), defined as:

$$MACMs : \prod_{i=1}^{k} f_{mi}(x_i) + \sum_{i=1}^{k} f_{ai}(x_i), \tag{10}$$

where $f_{mi} : \mathbb{R} \to \mathbb{R}$ and $f_{ai} : \mathbb{R} \to \mathbb{R}$ denote the feature shape functions of the multiplicative and additive part, respectively. Importantly, these shape functions are not limited to polynomials and may be instantiated by arbitrary functional mappings. Since $f_{mi}$ and $f_{ai}$ are both univariate, they can be directly visualized as curves, providing an intuitive interpretation of how each feature contributes multiplicatively and additively to the prediction. Furthermore, the combination of these curves (via product and sum) faithfully reflects the model's decision-making process, thereby preserving interpretability while enhancing flexibility.

#### 3.2.1 Interpretability of MACMs

The interpretability of MACMs arises from two key aspects: the visualizable decision process and the dynamic contribution of each feature to the output. However, because MACMs involve multiplicative operations, large-magnitude feature values may cause gradient explosion during training. To mitigate this, input features are normalized prior to training using min–max scaling to the range $[-1, 1]$. Training is then performed on the normalized data, and the feature shape functions are mapped back to the original scale through a linear transformation after training.

Unlike CESR in Equation (6), the multiplicative part of MACMs does not extract constants from each $f_{mi}$. While the two forms yield identical outputs, leaving constants inside $f_{mi}$ allows the functions to belong to $\mathcal{H} = \{g : X \to \mathbb{R} \mid g(0) = 0\}$. For example, $1 + w_{i1}x_i + \cdots + w_{in_i}x_i^{n_i}$ cannot represent $x + x^2$, whereas $w_{i0}^m + w_{i1}^m x_i + \cdots + w_{in_i}^m x_i^{n_i}$ can. However, this choice may also yield misleading interpretations [22]. Consider two models:

$$y_1 = \big(10^5 + 10^5 x_1 + 10^5 x_1^2\big)(1 + x_2 + x_2^2), \quad y_2 = \big(1 + x_1 + x_1^2\big)(10^5 + 10^5 x_2 + 10^5 x_2^2).$$

Although $y_1$ and $y_2$ produce identical outputs, the curves suggest that $x_1$ dominates in $y_1$ and $x_2$ dominates in $y_2$, leading to contradictory interpretations. To resolve this, we extract constant terms for normalization. The model is transformed into the following form for visualization:

$$\begin{aligned} MACMs : &\prod_{i=1}^{k} f_{mi}(0) \prod_{i=1}^{k} \frac{f_{mi}(x_i)}{f_{mi}(0)} + \sum_{i=1}^{k} f_{ai}(0) + \sum_{i=1}^{k} \big(f_{ai}(x_i) - f_{ai}(0)\big) \\ &= C_m \prod_{i=1}^{k} U_{mi}(x_i) + C_a + \sum_{i=1}^{k} U_{ai}(x_i), \end{aligned} \tag{11}$$

where $U_{mi}$ and $U_{ai}$ are normalized shape functions of the multiplicative and additive parts, respectively. By construction, $U_{mi}(0) = 1$ and $U_{ai}$ contains no bias term. The global constant term is $C_m + C_a$. This transformation requires $f_{mi}(0) \neq 0$; otherwise, constants for that feature are left unextracted.

This normalization primarily facilitates visualization. Given these normalized functions, one can manually derive the model's output. To describe the nonlinear relationship between a feature $x_i$ and the output, we isolate $x_i$ as the independent variable:

$$MACMs : \alpha U_{mi}(x_i) + U_{ai}(x_i) + \beta,$$

where $\beta = \sum_{j \neq i} U_{aj}(x_j)$ is the additive bias and $\alpha = C_m \prod_{j \neq i} U_{mj}(x_j)$ is a dynamic scaling factor determined by the multiplicative part. Hence, the contribution of $x_i$ is dynamic: when $\alpha$ is small, the additive component $U_{ai}(x_i)$ dominates, whereas for large $\alpha$, the multiplicative part governs the nonlinear relationship. To visualize this effect, we plot dynamic curves of the form:

$$\alpha U_{mi}(x_i) + U_{ai}(x_i), \quad i = 1, \ldots, k, \tag{12}$$

where $\alpha$ varies continuously. Since infinitely many such curves exist, we sample $\alpha$ uniformly from $[\min(\alpha), \max(\alpha)]$ in 10 steps, yielding a tractable set of dynamic influence curves that reveal how each feature's relationship with the output evolves as $\alpha$ changes.

### 3.3 Neural Networks Based Multiplicative-Additive Constrained models

When the shape functions in (10) are instantiated as fully connected neural networks, the resulting models are referred to as neural networks based Multiplicative-Additive Constrained Models (MACMs(NNs)). MACMs(NNs) serve as the primary architecture studied in this paper, and their structure is illustrated in Figure 1.

Formally, $f_{mi}(x_i)$ and $f_{aj}(x_j)$ denote fully connected layers with arbitrary depth and width. The overall output is given by the sum of the multiplicative and additive part, which can be transformed into the form of (11) for visualization.

During training, the parameter updates in the multiplicative part typically lag behind those in the additive part. This effect arises because subnetworks in the multiplicative part often employ bounded activation functions (e.g., ReLU), producing outputs whose absolute values do not exceed 1. Multiplying such outputs together amplifies the risk of gradient vanishing. To address this, we introduce a scaling factor $k > 1$ to the multiplicative part, as shown in (13). This rescaling increases the initial gradient magnitude, thereby improving convergence speed while preserving the model's representational capacity.

$$MACMs(NNs) : \; k \cdot \prod_{i=1}^{k} f_{mi}(x_i) + \sum_{i=1}^{k} f_{ai}(x_i). \tag{13}$$

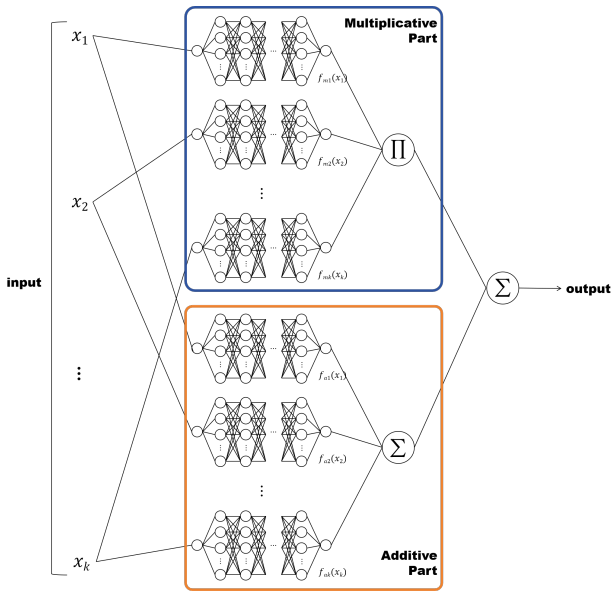


Figure 1: Architecture of MACMs(NNs), consisting of a multiplicative and an additive part. The multiplicative part is constructed as the product of multiple subnetworks, while the additive part is the sum of multiple subnetworks.

Given a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ with input features $x \in \mathbb{R}^k$ and targets $y$, the model output is defined as

$$m^\theta(x) = k^\theta \cdot \prod_{i=1}^{k} f_{mi}^\theta(x_i) + \sum_{i=1}^{k} f_{ai}^\theta(x_i).$$

For regression tasks, MACMs(NNs) employ the root mean squared error (RMSE) loss:

$$l(x, y; \theta) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(m^\theta(x_i) - y_i)^2},$$

and for binary classification, the cross-entropy loss:

$$l(x, y; \theta) = -y \log(\sigma(m^\theta(x))) - (1 - y) \log(1 - \sigma(m^\theta(x))).$$

The overall training objective is then given by

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y)\in\mathcal{D}}\big[l(x, y; \theta)\big].$$

# 4 Evaluation

## 4.1 Basellines

To evaluate the performance of MACMs and demonstrate how interpretability can be achieved, we compare MACMs(NNs) and polynomial-based MACMs (MACMs(poly)) against the following baseline models:

**Neural Additive Models (NAMs).** A neural networks based GAMs, where each input feature is processed through an exp-centered unit (ExU) before the activation function. NAMs have shown performance comparable to Explainable Boosting Machines (EBMs) [1].

**Neural Basis Models (NBMs).** Another neural networks based GAMs, which employs basis decomposition of feature functions. A small number of jointly learned basis functions are shared across all features, allowing NBMs to scale efficiently to high-dimensional and sparse data. NBMs have been shown to outperform NAMs in predictive accuracy [17].

**Prototypical Neural Additive Models (ProtoNAMs).** An extension of NAMs that incorporates prototype learning into each feature function. Prototypes represent representative samples from the data, enabling ProtoNAMs to capture more nuanced feature effects. They have been reported to outperform all existing neural network-based GAMs [23].

**Ergodic Set Regression (ESR).** A generalization of polynomial regression that includes all possible polynomial interactions across input variables. ESR serves as the full-complexity counterpart relative to CESR and MACMs(poly).

**Curve Ergodic Set Regression (CESR).** A constrained variant of ESR that retains all interaction terms while ensuring interpretability. CESR leverages nonlinear parameterizations, with model predictions jointly determined by the product of individual effect curves and a normalization constant.

**Deep Neural Networks (DNNs).** A standard fully connected network with the same number of hidden layers as MACMs(NNs). It serves as the unconstrained full-complexity counterpart, lacking the structural interpretability imposed by MACMs.

### 4.2 Datasets

**CA Housing (modified).** This dataset is a modified version of the California Housing dataset [15]. We first removed 207 samples with missing values in the `total_bedrooms` column. An additional categorical attribute, `ocean_proximity`, was added and encoded as (ISLAND: 1, NEAR OCEAN: 2, NEAR BAY: 3, 1H<OCEAN: 4, INLAND: 5). The target variable, the median house price, was scaled by dividing all values by 1000. The dataset is publicly available at Kaggle.

**Stroke Prediction.** This dataset is used to predict whether a patient is likely to have a stroke based on demographic and health-related attributes. We removed the `id` column and encoded categorical features as follows: `gender` (Female: 0, Male: 1), `ever_married` (No: 0, Yes: 1), `work_type` (Children: 0, Never_worked: 0.5, Govt_job: 0.5, Self-employed: 0.75, Private: 1), `residence_type` (Rural: 0, Urban: 1), and `smoking_status` (Never_smoked: 0, Formerly_smoked: 0.5, Smokes: 1). All missing values were removed. The prediction target is a binary variable indicating whether a patient has a stroke. The dataset is publicly available at Kaggle.

**Water Quality Prediction.** This dataset contains water quality measurements collected from the Ju River by the Langfang Environmental Monitoring Station of the Beijing Institute of Environmental Planning between August 18, 2018, and March 18, 2019 [24]. The task is to predict the `Total_Nitrogen (TN)` concentration at the next time step based on historical water quality indicators.

For all datasets, we applied only two preprocessing operations: min-max normalization and the removal of missing values. We evaluated model performance on the CA Housing (modified) and Stroke Prediction datasets, and conducted ablation studies on the Water Quality Prediction dataset.

### 4.3 Implementation Details

**MACMs(poly).** For all polynomial-based MACMs, the degree of each feature shape function was set to 12, and the scaling factor $k$ was set to 20. Models were trained using the Adam optimizer with a batch size of 1024, a fixed learning rate of 0.005, and 5000 epochs. No dropout or learning rate decay was applied.

**MACMs(NNs).** Each feature shape function was parameterized by a fully connected network with 10 hidden layers and 20 neurons per layer, using ReLU activations. The scaling factor $k$ was set to 10 for regression tasks. Regression models were trained with a batch size of 1024, a fixed learning rate of 0.0005 with exponential decay (factor 0.99 every 100 epochs), 0 dropout, and 10000 training epochs using the Adam optimizer. For binary classification, the scaling factor was increased to $k = 1000$, a sigmoid function was applied to the outputs, the learning rate was set to 0.00005 with decay factor 0.995 every 10 epochs, and models were trained for 2000 epochs; all other hyperparameters remained the same as in the regression setting.

For all datasets, 80% of the samples were used for training and the remaining 20% for testing, with data randomly shuffled during loading. Model performance was evaluated using RMSE for regression and AUC for binary classification, with reported results averaged over 5-fold cross-validation.

### 4.4 Performance of MACMs and Visualization

Table 2 presents the predictive performance of MACMs. MACMs(poly) achieve results intermediate between CESR and the full-complexity ESR, whereas MACMs(NNs) attain performance between the multiplicative part alone and their full-complexity counterpart, DNNs, consistent with our expectations. We further compared MACMs(NNs) against mainstream neural network-based GAMs. Since the hypothesis space of MACMs(NNs) can be regarded as the union of the multiplicative and additive hypothesis spaces, the model benefits from this enlarged space, yielding notable performance gains over NAMs, NBMs, and ProtoNAM, even without exhaustive hyperparameter tuning. This advantage is particularly pronounced in regression tasks.

Figure 2 visualizes the learned shape functions of MACMs(NNs) on the CA Housing (modified) dataset. In the plots, the green thin lines correspond to the shape functions obtained from each fold of 5-fold cross-validation, while the orange thick line represents the averaged shape function across all folds. In this experiment, with $C = 51.89$ and nearly all shape function values positive, we marked the reference line $y = 1$ within the multiplicative part. From a microscopic perspective, Housing Median Age and Total Bedrooms exhibit an overall positive correlation with the multiplicative part, whereas Population demonstrates a negative correlation. In contrast, within the additive part, the monotonicity often differs: Housing Median Age tends to show a negative contribution, while Median Income exhibits a positive contribution.

Figure 3 presents the dynamic shape functions of all features, following Equation (12). As each feature has infinitely many dynamic curves, we display 10 uniformly sampled curves for representative illustration. The plots reveal that, although the shape functions vary dynamically, their overall forms remain structured. With increasing $\alpha$, the dynamic shape functions increasingly resemble those of the multiplicative part, illustrating how the multiplicative part progressively dominates the feature contributions.

Table 2: Performance of MACMs and baselines. All results are reported as the mean of 5-fold cross-validation. Lower RMSE values indicate better performance, whereas higher AUC values are preferred.

| | CA Housing(modified) | CA Housing | Stroke | Water Quality |
|---|---|---|---|---|
| **Model** | RMSE | RMSE | AUC | RMSE |
| NAMs | 56.5699±0.6841 | 56.4011±0.6131 | 0.8190±0.0227 | 0.4757±0.0402 |
| CESR | 64.7516±2.8115 | 64.1922±0.9096 | 0.8104±0.1055 | 0.7881±0.0504 |
| MACMs(poly) | 61.0028±1.3343 | 62.2639±1.0108 | 0.8170±0.0335 | 0.5981±0.0874 |
| NBMs | 56.1521±0.8341 | 56.2230±0.5366 | 0.8220±0.0174 | 0.4732±0.0493 |
| ProtoNAM | 55.1818±0.7710 | 55.2775±0.2916 | **0.8244±0.0281** | 0.4605±0.0220 |
| MACMs(NNs) | **53.4050±1.3993** | **52.6411±1.8121** | 0.8211±0.0662 | **0.4036±0.0938** |
| ESR | 49.7763±0.8667 | 49.4074±0.6692 | 0.7582±0.0122 | 0.3417±0.0297 |
| DNN | 49.0046±1.2313 | 49.2913±0.5592 | 0.8133±0.0209 | 0.3592±0.0707 |
| MP(poly based) | 63.9860±0.9295 | 64.5857±1.5560 | 0.8159±0.0747 | 0.8834±0.0744 |
| AP(poly based) | 63.9305±3.1625 | 65.2596±0.7027 | 0.8135±0.0324 | 0.7818±0.1597 |
| MP(NNs based) | 56.9609±2.1844 | 57.9676±2.5309 | 0.8187±0.0372 | 0.7153±0.0681 |
| AP(NNs based) | 60.2258±1.9126 | 59.3957±1.2092 | 0.8182±0.0436 | 0.5730±0.0716 |

### 4.5 Ablation Research

The purpose of our ablation experiments is to disentangle and analyze the individual contributions of the multiplicative and additive part in MACMs. Specifically, we aim to identify which part is more essential for the overall performance of the model, and to assess the impact of substituting

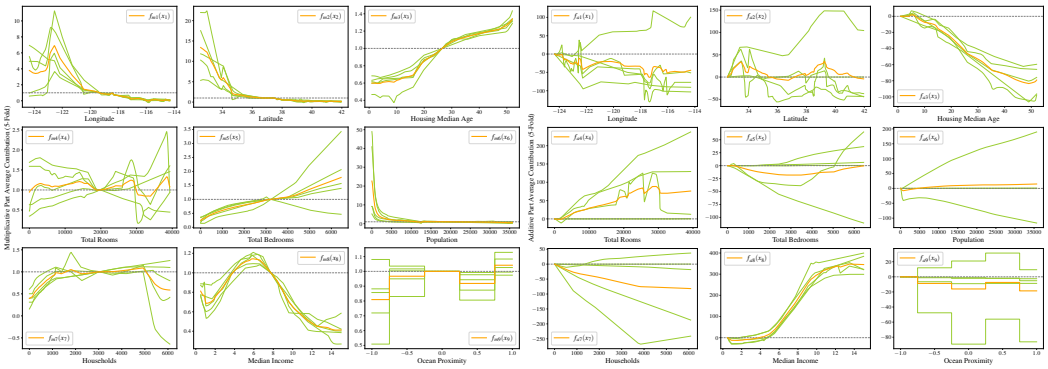

Figure 2: Multiplicative Part and Additive Part Shape Functions of MACMs(NNs) on CA Housing(modified).

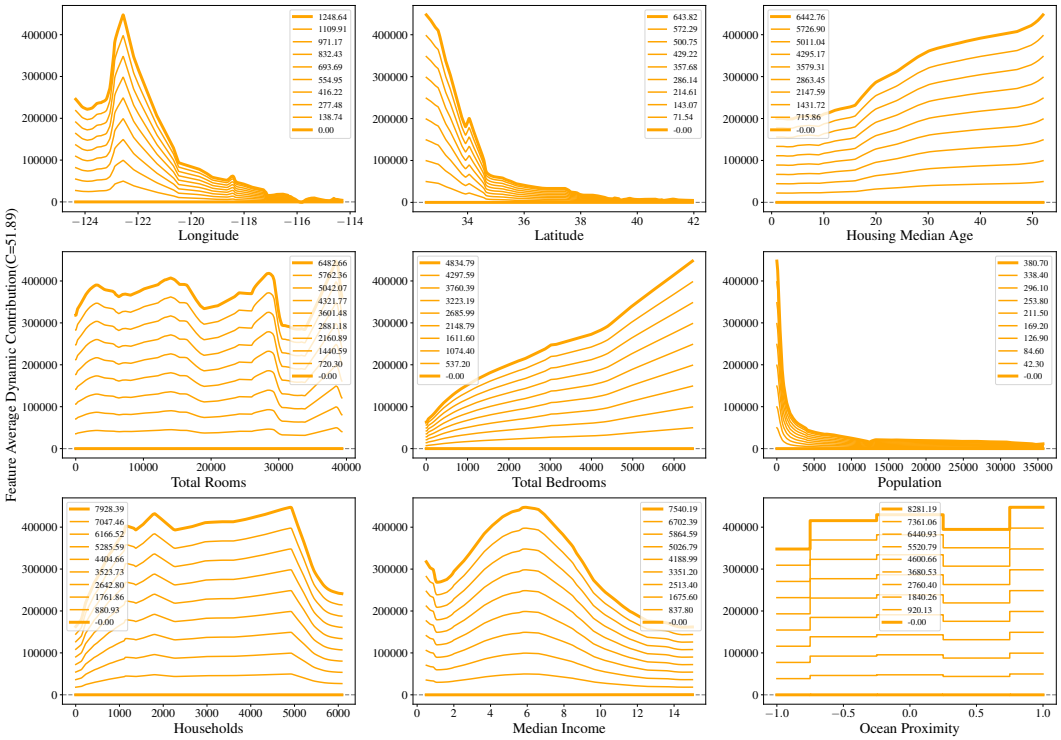


Figure 3: Multiplicative Part and Additive Part Shape Functions of MACMs(NNs) on CA Housing(modified)

polynomial-based shape functions with neural network parameterizations. The results of the ablation experiments are reported in Table 2.

The Multiplicative Part (NNs) and Additive Part (NNs) perform worse than MACMs (NNs) across all datasets, indicating that both parts are indispensable for achieving the high performance of MACMs (NNs), and also demonstrating the success of our strategy in CESR to add an additive part in order to decouple constrained parameters and expand the hypothesis space.

For both MACMs (Poly) and MACMs (NNs), the performance difference between the multiplicative and additive part is not substantial. Specifically, the multiplicative part outperforms on the CA Housing and CA Housing (modified) datasets, while the additive part yields better performance on the Water Quality dataset. This further highlights that the multiplicative part in MACMs should not be regarded solely as an interaction effect; rather, it is jointly constituted by both the multiplicative and additive part.

Across all experiments, MACMs (NNs) achieve better performance than MACMs (Poly), with the improvement being especially significant in regression tasks. This finding suggests that substituting polynomials with more complex fully connected layers effectively boosts the performance of MACMs.

## 5 Discussion

The proposed MACMs provide new insights into the development of interpretable models. While originally motivated by the enhancement of multiplicative models, the framework also reveals an unexpected connection to GAMs. Unlike recent advances in GAMs that primarily focus on refining shape functions within an unchanged hypothesis space, MACMs fundamentally expand the hypothesis space by explicitly incorporating higher-order interactions beyond pairwise terms. This expansion explains the notable performance improvements observed in our experiments and suggests that MACMs can serve as a bridge between multiplicative models and GAM-based approaches.

Despite these advantages, several limitations remain. First, the pursuit of greater model complexity inevitably introduces a trade-off between accuracy and interpretability. The dynamic shape functions in MACMs offer only semi-global interpretability, which, although helpful, falls short of the full transparency provided by classical GAMs. Second, the multiplicative component exhibits sensitivity to the number of features. Without careful adjustment of the scaling factor $k$, the model may become unbalanced, leading to insufficient learning in the multiplicative component while the additive part dominates.

## 6 Conclusion and Future Work

This work introduces a novel interpretable model—Multiplicative-Additive Constrained Models (MACMs), which consist of a multiplicative part and an additive part. By incorporating an additive component into a multiplicative model called Curve Ergodic Set Regression (CESR), we decouple the coefficients of the multiplicative and additive components and thereby expand the model's hypothesis space. MACMs permit the construction of shape functions using arbitrary mapping forms. In this work, we primarily instantiate MACMs with shape functions parameterized by fully connected neural networks, referred to as MACMs (NNs). Given that MACMs (NNs) include an additive component, we further evaluate their performance in comparison with representative NNs-based GAMs. The experimental results indicate that our proposed model yields a notable improvement over CESR. Furthermore, MACMs (NNs) surpass the state-of-the-art NNs-based GAMs, including NBMs and ProtoNAM, in terms of performance.

In the future, we aim to develop more effective interpretability techniques for MACMs. For example, overlaying raw data samples on dynamic shape function plots could help reveal which patterns between features and outputs are more prominent. Another important direction is to design an automatic adjustment mechanism for the scaling factor. Furthermore, we plan to explore constructing models with two-dimensional shape functions, particularly in the context of MACMs built upon ESR, where the number of parameters could be significantly smaller than that of GAMs. Most importantly, we will enhance MACMs with various optimization strategies, such as grid search for hyperparameter tuning, adding output regularization, or incorporating shape function designs from NBMs and ProtoNAMs, in order to further improve the model's performance.

## References


[1] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *Advances in neural information processing systems*, 34:4699–4711, 2021.

[2] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[3] A.R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

[4] Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8):832, 2019.

[5] Victor Chang, Jozeene Bailey, Qianwen Xu, and Zhili Sun. Pima indians diabetes mellitus classification based on machine learning (ml) algorithms. *Neural Computing and Applications*, 35:16157–16173, 03 2022.

[6] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems*, 32, 2019.

[7] Rudresh Dwivedi, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, et al. Explainable ai (xai): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9):1–33, 2023.

[8] James Enouen and Yan Liu. Sparse interaction additive networks via feature interaction detection and sparse selection. *Advances in Neural Information Processing Systems*, 35:13908–13920, 2022.

[9] Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, et al. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*, 2024.

[10] Trevor J Hastie. Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge, 2017.

[11] Himabindu Lakkaraju and Osbert Bastani. " how do i fool you?" manipulating user trust via misleading black box explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 79–85, 2020.

[12] Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24142–24153, June 2024.

[13] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135(3):370–384, 1972.

[14] Gherman Novakovsky, Nick Dexter, Maxwell W Libbrecht, Wyeth W Wasserman, and Sara Mostafavi. Obtaining genetics insights from deep learning via explainable artificial intelligence. *Nature Reviews Genetics*, 24(2):125–137, 2023.

[15] R Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997.

[16] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1:81–106, 1986.

[17] Filip Radenovic, Abhimanyu Dubey, and Dhruv Mahajan. Neural basis models for interpretability. *Advances in Neural Information Processing Systems*, 35:8414–8426, 2022.

[18] Cynthia Rudin, Chaofan Chen, Zhi Chen, Haiyang Huang, Lesia Semenova, and Chudi Zhong. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistic Surveys*, 16:1–85, 2022.

[19] Priyanka Sinha. Multivariate polynomial regression in data mining: Methodology, problems and solutions. *International Journal of Scientific and Engineering Research*, 4, 12 2013.

[20] M. H. Stone. Applications of the theory of boolean rings to general topology. *Transactions of the American Mathematical Society*, 41(3):375–481, 1937.

[21] Berk Ustun and Cynthia Rudin. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, 102:349–391, 2016.

[22] Fumin Wang, Hongxia Xu, and Jianzhuo Yan. Ergodic set regression models. In *2022 4th International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, pages 188–196. IEEE, 2022.

[23] Guangzhi Xiong, Sanchit Sinha, and Aidong Zhang. Protonam: Prototypical neural additive models for interpretable deep tabular learning. *arXiv preprint arXiv:2410.04723*, 2024.

[24] Jianzhuo Yan, Qingcai Gao, Yongchuan Yu, Lihong Chen, Zhe Xu, and Jianhui Chen. Combining knowledge graph with deep adversarial network for water quality prediction. *Environmental Science and Pollution Research*, 30(4):10360–10376, 2023.

[25] Zebin Yang, Aijun Zhang, and Agus Sudjianto. Gami-net: An explainable neural network based on generalized additive models with structured interactions. *Pattern Recognition*, 120:108192, 2021.

# A Appendix

## A.1 Additional visualization

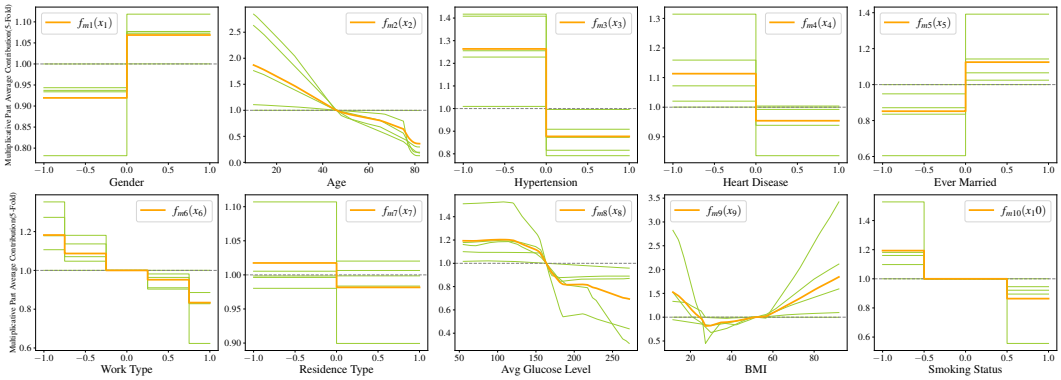


Figure 4: Multiplicative Part Shape Functions of MACMs(NNs) on Stroke Prediction.

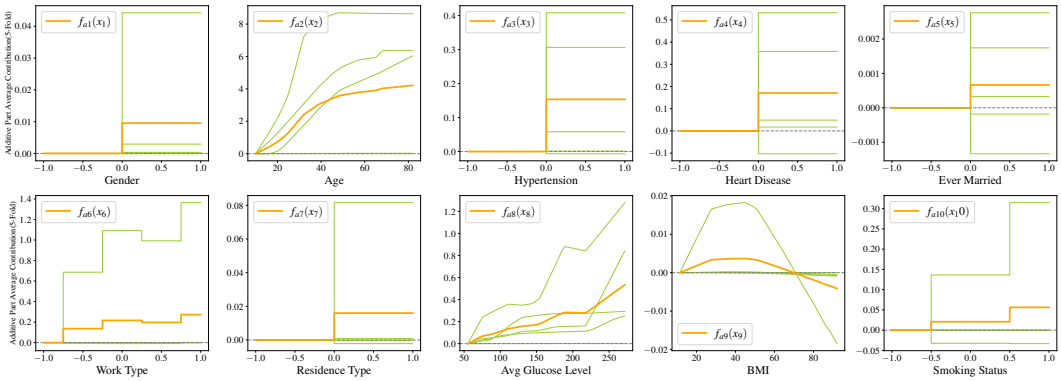


Figure 5: Additive Part Shape Functions of MACMs(NNs) on Stroke Prediction.

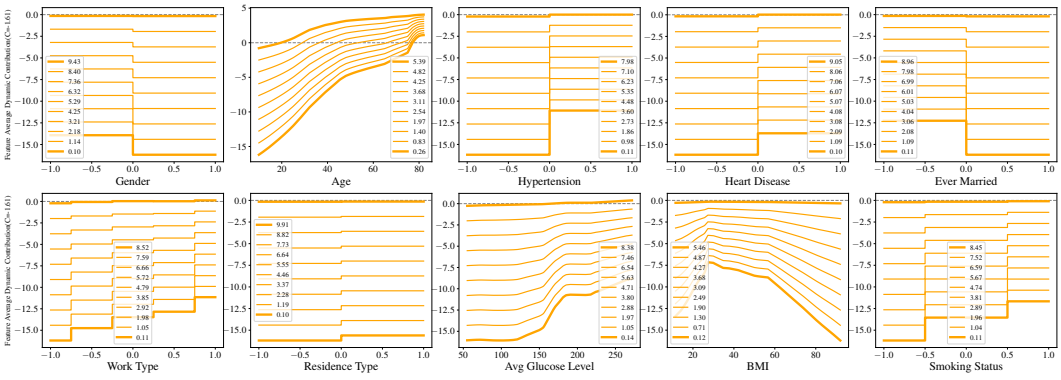


Figure 6: Feature average dynamic contribution shape functions of MACMs(NNs) on Stroke Prediction.

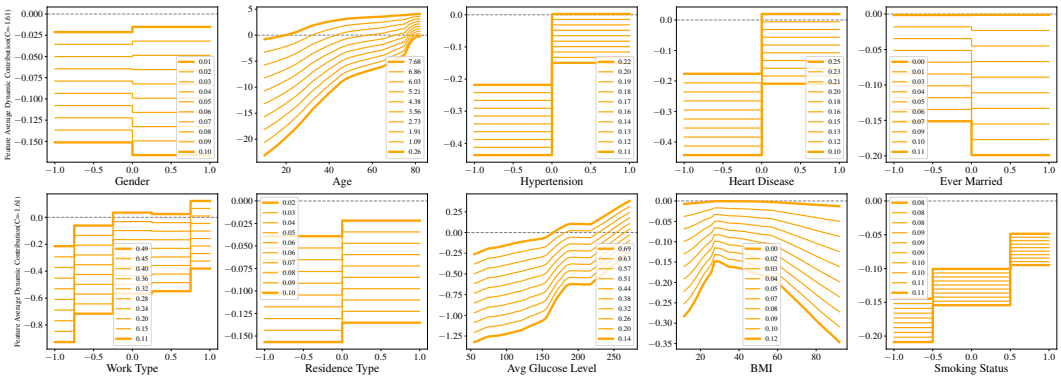


Figure 7: Feature average dynamic contribution shape functions in the first decile interval of MACMs(NNs) on Stroke Prediction.