# Statistically Robust Sparse High-order Interaction Model

**Diptesh Das**[1*], **Ichiro Takeuchi**[2, 3], **Koji Tsuda**[1, 3, 4†]

[1]Department of Computational Biology and Medical Sciences, The University of Tokyo, Japan.
[2]Department of Mechanical Systems Engineering, Nagoya University, Japan.
[3]RIKEN Center for Advanced Intelligence Project, Japan.
[4]Center for Basic Research on Materials, National Institute for Materials Science, Japan.

## Abstract

Deep learning models often achieve high accuracy but lack interpretability, making them unsuitable for critical applications such as medical diagnosis, biomolecule design, criminal justice, etc. The Sparse High-order Interaction Model (SHIM) addresses this limitation by providing both transparency and predictive reliability. However, real-world data often contain outliers, which can distort model performance. To overcome this, we propose Huberized-SHIM, an extension of SHIM that integrates Huber loss-based robust regression to mitigate the impact of outliers. We introduce a homotopy-based exact regularization path algorithm and a novel tree-pruning criterion to efficiently manage interaction complexity. Additionally, we incorporate the conformal prediction framework to enhance statistical reliability. Empirical evaluations on synthetic and real-world datasets demonstrate the superior robustness and accuracy of Huberized-SHIM in high-stakes decision-making contexts.

## 1 Introduction

While deep neural networks and other black-box models often achieve high predictive accuracy, their lack of interpretability makes them less reliable (Rudin 2019; Das 2019). Consequently, in critical applications like medical diagnosis, biomolecule design, criminal justice, etc. where transparency is essential for decision-making, models with greater interpretability and high accuracy are preferred. The sparse high-order interaction model (SHIM) (Suzumura et al. 2017; Das et al. 2019; Das 2019; Das et al. 2022; Das, Ndiaye, and Takeuchi 2024) offers both interpretability and strong predictive performance, making it a suitable choice for such tasks. Considering a regression problem of $m$ original covariates $z_1, \ldots, z_m$ and response $y$, an example SHIM up to $4^{th}$ order interactions can be written as

$$y = \beta_1 z_2 + \beta_2 z_3 + \beta_3 z_2 z_5 + \beta_4 z_1 z_3 z_4 z_6,$$

where $\beta$'s are the regression coefficients. A SHIM has significant practical applications. For example, identifying complex genotypic traits related to HIV-1 drug resistance (Saigo, Uno, and Tsuda 2007; Das et al. 2022; Das, Ndiaye, and Takeuchi 2024) where a combination of multiple mutations, along with certain key single mutations provides the most accurate representation of the intricate biological mechanisms underlying drug resistance (Vivet-Boudou et al. 2006; Iversen et al. 1996; Rhee et al. 2006) or recognizing patterns of epistasis where the interdependence of mutations is crucial for understanding the relationship between genotype and phenotype (Poelwijk, Socolich, and Ranganathan 2019; Fannjiang et al. 2022). Key protein characteristics, such as folding, biochemical function, and evolvability, emerge from a network of cooperative energetic interactions among amino acid residues. Identifying epistasis plays a significant role in reconstructing phylogenetic trees and assessing the evolutionary potential of antibiotic resistance genes and viruses. Additionally, in protein engineering and directed evolution, insights into epistatic structures can aid in selecting optimal templates, targeting mutations in highly epistatic regions, and identifying cooperative units for DNA shuffling experiments. Another example is criminal recidivism prediction that aims to determine the likelihood of an individual being arrested within a specific period after their release from jail or prison (Larson et al. 2016; Angelino et al. 2018). In such cases, where predictions directly impact human lives, a model that is both highly accurate and interpretable is essential for ensuring fairness and transparency in decision-making (Rudin 2019; Huang, Das, and Tsuda 2023; Das et al. 2019; Das 2019).

However, real-world data are often contaminated with outliers and the presence of outliers can highly influence the data-driven modeling. For example, (Reichel 2025) recently studied that how the presence of a single outlier can cause an otherwise insignificant coefficient to appear statistically significant in finite-sample inference. To counter this generally robust regression model (Wilcox 1996) is used which instead of of automatically removing outliers, helps mitigate their impact (Tsukurimichi et al. 2022). Robust regression modifies the loss function to downweight the effect of extreme residuals and a common choice is Huber loss, which combines squared loss for small residuals and absolute loss for large residuals (Huber 1964; Owen 2007; Huber and Ronchetti 2011). Huber loss-based regression models have been successfully used in biology (Deng et al. 2021; Deutelmoser et al. 2021), medicine (Normolle 1993), medical diagnosis (Karim et al. 2023), finance (He et al. 2021;

*Corresponding author: diptesh.das@edu.k.u-tokyo.ac.jp
†Corresponding author: tsuda@k.u-tokyo.ac.jp

Pervez and Ali 2024), and others (Das 2023; Korgialas and Kotropoulos 2023). In this paper we extend SHIM and proposed Huberized-SHIM to counter the effect of outliers so that it can be used reliably even in the presence of outliers. We provided a homotopy-based exact regularization path following algorithm to compute the entire regularization path of Huberized-SHIM. We derived a novel branch and bound tree pruning criteria essential for fitting a SHIM which is otherwise intractable due the combinatorial explosion of the interaction terms. Furthermore, we integrated conformal prediction framework to demonstrate the statistical efficiency of proposed Huberized-SHIM over SHIM. We demonstrated the computational and statistical efficiency of the proposed framework using synthetic and real world data.

## 2 Problem Statement

Consider a regression problem with a response vector $y \in \mathbb{R}^n$ and $m$ original covariate vectors $z_1, \ldots, z_m$, where $z_j \in \mathbb{R}^n$ and $j \in [m]$. A high-order interaction model up to the $d^{\text{th}}$ order is then written as follows:

$$y = \sum_{j_1 \in [m]} \theta_{j_1} z_{j_1} + \sum_{\substack{(j_1,j_2)\in[m]\times[m] \\ j_1 \neq j_2}} \theta_{j_1,j_2} z_{j_1} z_{j_2} + \cdots + \sum_{\substack{(j_1,\ldots,j_d)\in[m]^d \\ j_1 \neq \ldots \neq j_d}} \theta_{j_1,\ldots,j_d} z_{j_1} \cdots z_{j_d} + \epsilon, \tag{1}$$

where $z_{j_1} \cdots z_{j_d}$ is the element-wise product, scalar $\theta$ represents the coefficient and $\epsilon$ is the noise. In this study, we mainly consider each element of the original covariate vector $z_j \in \{0,1\}^n$. However, our model is equally applicable to covariate vectors defined in the domain $[0,1]^n$. To simplify the notation, it is convenient to write the high-order interaction model in (1) using the following matrix of concatenated vectors of all high-order interactions:

$$X = [\underbrace{z_1, \ldots, z_m}_{1^{\text{st}} \text{ order}}, \cdots, \underbrace{z_1 \ldots z_d, \ldots, z_{m-d+1} \ldots z_m}_{d^{\text{th}} \text{ order}}] \in \mathbb{R}^{n \times p},$$

where $p := \sum_{\kappa=1}^{d} \binom{m}{\kappa}$, considering up to $d^{th}$ order interactions. Similarly, the coefficient vector associated with all possible high-order interaction terms can be written as follows:

$$\beta := [\underbrace{\theta_1, \ldots, \theta_m}_{1^{\text{st}} \text{ order}}, \cdots, \underbrace{\theta_{1,\ldots,d}, \ldots, \theta_{m-d+1,\ldots,m}}_{d^{\text{th}} \text{ order}}]^\top \in \mathbb{R}^p.$$

The high-order interaction model (1) is then simply written as a linear model $y = X\beta + \epsilon$. Unfortunately, $p$ can be prohibitively large unless both $m$ and $d$ are fairly small. In the SHIM, we consider a sparse estimation of a high-order interaction model. An example of a SHIM is as follows:

$$y = \theta_2 z_2 + \theta_3 z_3 + \theta_{2,6} z_2 z_6 + \theta_{1,2,4,6} z_1 z_2 z_4 z_6 + \epsilon.$$

## 3 Proposed Method

We propose a *homotopy-mining* method to compute the exact regularization path of Huberized-SHIM. The homotopy method refers to an optimization framework for solving a sequence of parameterized optimization problems. In robust (Huberized) SHIM we solve the following optimization problem:

$$\beta(\lambda) \in \underset{\beta \in \mathbb{R}^p}{\arg\min} \sum_{i=1}^{n} L(r_i(\lambda)) + \lambda ||\beta||_1 + \frac{1}{2}\alpha ||\beta||_2^2, \tag{2}$$

where $r_i(\lambda) = y_i - X_i^\top \beta(\lambda)$ is the residual, $\lambda$ and $\alpha$ are the regularization parameters of $\ell_1$ and $\ell_2$ penalty terms. The loss $L(\cdot)$ is the Huber loss:

$$L(r_i(\lambda)) = \begin{cases} \frac{1}{2} r_i^2(\lambda), & \text{if} |r_i(\lambda)| \leq \delta, \\ \delta |r_i(\lambda)| - \frac{\delta^2}{2}, & \text{otherwise.} \end{cases}$$

where $\delta \geq 0$ is a hyperparameter. We further define two new parameters $a$ and $s$ as stated in (3) to redefine the optimization problem (2):

$$a_i(\lambda) = \begin{cases} 1, & \text{if} |r_i(\lambda)| \leq \delta, \\ 0, & \text{otherwise.} \end{cases}, \quad \text{and}$$

$$s(r_i(\lambda)) = \begin{cases} \pm 1, & \text{if } r_i(\lambda) \neq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Now, we can rewrite the loss in (2):

$$\sum_{i=1}^{n} L(r_i(\lambda)) = \sum_{i=1}^{n} \frac{1}{2} a_i(\lambda) r_i^2(\lambda) + \delta \sum_{i=1}^{n} (1 - a_i(\lambda)) |r_i(\lambda)|.$$

**Optimality conditions.** At optima we can write

$$X^\top h(\lambda) - \alpha \beta(\lambda) = \lambda s(\beta(\lambda)), \tag{4}$$

where $h(\lambda) = a(\lambda) \odot r(\lambda) + \delta(1 - a(\lambda)) \odot s(r(\lambda))$ and $\forall \ell \in [p]$,

$$s_\ell(\beta(\lambda)) \in \begin{cases} \{-1, +1\}, & \text{if } \beta_\ell(\lambda) \neq 0, \\ [-1, +1], & \text{if } \beta_\ell(\lambda) = 0, \end{cases} \tag{5}$$

and $\odot$ represents element-wise vector product. Let's define the active set

$$\mathcal{A}_\lambda = \{\ell \in [p] : |x_\ell^\top h(\lambda) - \alpha \beta_\ell| = \lambda\}, \tag{6}$$

where $[p] = \{1, 2, \ldots, p\}$ is the set of indices of all possible interaction terms of a SHIM, and the non-active set can be defined as the complement of the active set:

$$\mathcal{A}_\lambda^c = \{[p] \setminus \mathcal{A}_\lambda\}.$$

The solutions $\beta(\lambda)$ of (2) at different values of $\lambda$ is called the regularization path (or $\lambda$-path) and the regularization path ($\lambda \mapsto \beta(\lambda)$) of the Huberized-SHIM can be shown to be piecewise linear as stated in Proposition 1.

**Proposition 1.** *If $\beta(\lambda)$'s have the same sign between two points $\lambda_1$ and $\lambda_2$, that is sign($\beta(\lambda_1)$)=sign($\beta(\lambda_2)$)=sign($\beta(\lambda)$), $\forall \lambda \in [\lambda_1, \lambda_2)$, then $\mathcal{A}_\lambda = \mathcal{A}_{\lambda_1}$. Furthermore, assuming that $X_{\mathcal{A}_\lambda}^\top X_{\mathcal{A}_\lambda}$ is invertible and there is no "knot-crossing" for any instance $i \in [n]$ such that the values of $a(\lambda)$ and $\big(1 - a(\lambda)\big) s\big(r(\lambda)\big)$ remain the same for all $\lambda \in [\lambda_1, \lambda_2)$, we have the linear relations*

$$\beta_{\mathcal{A}_\lambda}(\lambda_2) = \beta_{\mathcal{A}_\lambda}(\lambda_1) + (\lambda_1 - \lambda_2) \psi_{\mathcal{A}_\lambda}(\lambda),$$
$$\lambda_2 s\big(\beta_{\mathcal{A}_\lambda^c}(\lambda_2)\big) = \lambda_1 s\big(\beta_{\mathcal{A}_\lambda^c}(\lambda_1)\big) + (\lambda_1 - \lambda_2) \gamma_{\mathcal{A}_\lambda^c}(\lambda),$$

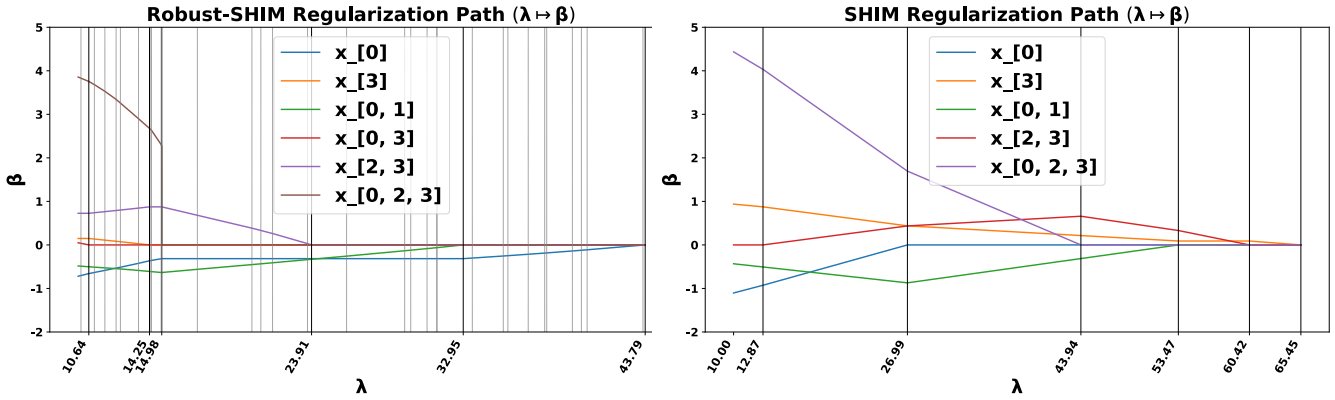


Figure 1: The entire regularization path of Robust-SHIM (left) and SHIM (right). In robust-SHIM, there exists a third event "knot-crossing" which are represented by dotted grey vertical lines. In both the plots, the solid black vertical lines represent either feature "addition" or "deletion". Both the plots have been generated using the same true model $y = -x_1 + x_3 - 2x_1x_3 + 7x_1x_3x_4$ with same setting of regularization parameters $\lambda_{target} = 10, \alpha = 0.001$, with $\delta = 1.5$ for robust-SHIM. The choice of this true model is arbitrary and the proposed method should work with any chosen model.

*where the direction vectors $\psi$ and $\gamma$ are defined as*

$$\psi_{\mathcal{A}_\lambda}(\lambda) = \Big(\alpha \mathbb{I}_{|\mathcal{A}_\lambda|} + X_{\mathcal{A}_\lambda}^\top (a(\lambda) \odot X_{\mathcal{A}_\lambda})\Big)^{-1} s\big(\beta_{\mathcal{A}_\lambda}(\lambda)\big),$$
$$\gamma_{\mathcal{A}_\lambda^c}(\lambda) = -X_{\mathcal{A}_\lambda^c}^\top (a(\lambda) \odot X_{\mathcal{A}_\lambda}) \psi_{\mathcal{A}_\lambda}(\lambda). \tag{7}$$

For simplicity, we will define the step size $\Delta = (\lambda_1 - \lambda_2) > 0$. Therefore, according to Proposition 1 it is possible to design an algorithm to compute the entire regularization path of Huberized SHIM exactly using a homotopy algorithm that exploits the linearity of the path between each two consecutive transition points of direction $(\psi, \gamma)$ changes.

## The Entire Regularization Path of Robust-SHIM

A homotopy algorithm of robust-SHIM sequentially tracks and updates the sign and the active set of the optimal solutions, and the parameter vectors $s(r)$ and $a$ depending on the signs and values of each component of the residual vector $r$. At any two consecutive steps represented by $\lambda_t$ and $\lambda_{t+1}$, where $t$ is an index of the transition points (kinks) of the $\lambda$-path, either of the following three events occurs:

- (Addition): a zero variable becomes non-zero, that is,

$$\exists \ell \in \mathcal{A}_{\lambda_t}^c, \quad \text{s.t.} \quad |x_\ell^\top h(\lambda_{t+1})| = \lambda_{t+1}, \quad \text{or,}$$

- (Deletion): a non-zero variable becomes zero, that is,

$$\exists \ell \in \mathcal{A}_{\lambda_t}, \ \text{s.t.} \ \beta_\ell(\lambda_t) \neq 0, \ \text{but} \ \beta_\ell(\lambda_{t+1}) = 0, \ \text{or,}$$

- (Knot-crossing): a residual $r_i$ hits a Huberized knot point and the value of $a_i$ changes, that is,

$$\exists i \in [n], \quad \text{s.t.} \quad |y_i - X_{i,\mathcal{A}(\lambda_t)} \beta_{\mathcal{A}_{\lambda_t}}(\lambda_{t+1})| = \delta.$$

Overall, the next change in the direction vectors occur at $\lambda_{t+1} = \lambda_t + \Delta$, such that

$$\Delta = \min\Big(\Delta_1(\ell_1^*), \Delta_2(\ell_2^*), \Delta_3(i^*)\Big), \tag{8}$$

$$\text{where} \quad \ell_1^* = \arg\min_{\ell \in \mathcal{A}_{\lambda_t}^c} \Delta_1(\ell),$$
$$\ell_2^* = \arg\min_{\ell \in \mathcal{A}_{\lambda_t}} \Delta_2(\ell),$$
$$i^* = \arg\min_{i \in [n]} \Delta_3(i), \quad \text{and}$$

$$\Delta_1(\ell) = \left( \frac{(x_\ell \mp x_k)^\top h(\lambda_t) \pm \alpha \beta_k(\lambda_t)}{(x_\ell \mp x_k)^\top \big(a(\lambda_t) \odot v(\lambda_t)\big) \mp \alpha \psi_k(\lambda_t)} \right)_{++},$$
$$\Delta_2(\ell) = \left( -\frac{\beta_\ell(\lambda_t)}{\psi_\ell(\lambda_t)} \right)_{++},$$
$$\Delta_3(i) = \left\{ \min\left( \left(\frac{r_i(\lambda_t) - \delta}{v_i(\lambda_t)}\right)_{++}, \left(\frac{r_i(\lambda_t) + \delta}{v_i(\lambda_t)}\right)_{++} \right) \right\},$$

for any $k \in \mathcal{A}_{\lambda_t}$, and we defined $v(\lambda) = X_{\mathcal{A}_\lambda} \psi_{\mathcal{A}_\lambda}(\lambda)$. Here, we use the convention that for any $g \in \mathbb{R}, (g)_{++} = g$, if $g > 0$ and $\infty$ otherwise. Therefore, similar to classical LARS algorithm (Efron et al. 2004), we can construct the entire regularization path $(\lambda \mapsto \beta(\lambda))$ of robust-SHIM "exactly" using the homotopy method (Rosset and Zhu 2007) by keeping track of direction changes of a piecewise linear path and computing the step-size of next event ('addition' or 'deletion' or 'knot-crossing'). The entire regularization paths of robust-SHIM and SHIM for a target true model have been shown in Figure 1. The main difference between a SHIM and a robust-SHIM is that a third event "knot-crossing" appears in the regularization path of robust-SHIM when the residual of an instance cross the knot of Huber loss (Rosset and Zhu 2007).

**Branch and bound tree pruning condition.** However, naively (by simply minimizing over all possible interaction terms) determining the step size of inclusion $(\Delta_1(\ell_1^*))$ will be intractable for the SHIM type problem. In SHIM, the search space grows exponentially due to the combinatorial

effect of high-order interaction terms. Therefore, fitting of a SHIM is non-trivial and a SHIM model will have a significantly large number of parameters to be considered unless both number of features $(m)$ and the order of interactions $(d)$ are very small. Several algorithms for fitting a sparse high-order interaction model have been proposed in the literature (Tsuda 2007; Saigo et al. 2009; Nakagawa et al. 2016; Das et al. 2022; Das, Ndiaye, and Takeuchi 2024). A common approach adopted in these existing works is to exploit the hierarchical structure of high-order interaction features. In other words, a tree structure (Figure 2) of interaction terms (patterns) is constructed progressively where each node represents a single feature or an interaction term, and a branch and bound tree pruning strategy is employed using tree anti-monotonicity property (Definition 1) in order to avoid handling all the exponentially increasing number of high-order interaction features.

**Definition 1** (Pattern Tree)**.** *A pattern tree is constructed in such a way that for any pair of nodes $(\ell, \ell')$, where $\ell$ is the ancestor of $\ell'$, i.e., $\ell \subset \ell'$,*

$$x_{i\ell} \geq x_{i\ell'}, \quad \forall i \in [n].$$

The above anti-monotonicity property of tree patterns is always true for any pair of binary features or for any pair of real features scaled between 0 and 1 or for any mixed pair of a binary and a scaled real features. Because, $\forall i \in [n]$, and $\forall \ell, \ell' : \ell \subset \ell'$, if $x_{i\ell}, x_{i\ell'} \in \{0, 1\}$, then

$$x_{i\ell'} = 1 \implies x_{i\ell} = 1 \quad \text{and} \quad x_{i\ell} = 0 \implies x_{i\ell'} = 0,$$

and if $x_{i\ell}, x_{i\ell'} \in [0, 1]$, then $x_{i\ell} \geq x_{i\ell'}$. Hence, anti-monotonicity also holds true for a mixed pair of features. This anti-monotonicity property of tree patterns can be used to derive a tree pruning condition to ignore a large number of unnecessary high-order interaction terms (or patterns). Therefore, we can design an efficient branch

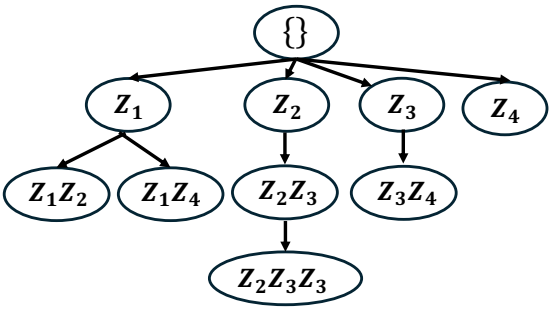


Figure 2: An illustration of a pattern tree, built by leveraging the inherent hierarchy found in high-order interaction terms. Due to the branch and bound tree pruning strategy, some patterns were excluded and do not appear in the final structure.

and bound algorithm using this tree anti-monotonicity property to make the computation practically feasible for fitting a robust-SHIM. In the following section, we present an efficient tree pruning strategy where each node of the tree represents an interaction term. The basic idea of tree pruning is that we construct a tree of interaction terms in a 'progressive manner' as shown in Figure 2. That is, we keep track of the current minimum step size of inclusion up to the construction of $\ell^{th}$ pattern as we construct the tree progressively, and prune a large part of the tree if some bound condition fails (Lemma 1).

Algorithm 1: Exact $\lambda$-path of Huberized SHIM

**Input**: $\mathcal{D}_n = \{(X_i, y_i)\}_{i=1}^n$
Initialize $t = 0$, $\lambda_0 = \lambda_{max}$ using (10), $\mathcal{A}_{\lambda_0} = \{\ell^*\}$, $a_i(\lambda_0)$ and $s(r_i(\lambda_0))$ using (3), $\mathbb{A} = \mathcal{A}_{\lambda_0}, \mathbb{B} = \{\mathbf{0}\}$.
**while** $(\lambda > 0)$ **do**
  Compute $\Delta$ using (8).
  Update: $\lambda_{t+1} \leftarrow \lambda_t + \Delta$, $\beta_{\mathcal{A}_{\lambda_t}}(\lambda_{t+1}) \leftarrow \beta_{\mathcal{A}_{\lambda_t}}(\lambda_t) + \Delta \cdot \psi_{\mathcal{A}_{\lambda_t}}(\lambda_t)$, $\beta_{t+1} \leftarrow [\beta_{\mathcal{A}_{\lambda_t}}(\lambda_{t+1}), \mathbf{0}]$.
  **if** $\Delta = \Delta_{\lambda_1}$ **then**
    add $\ell$ into $\mathcal{A}_{\lambda_t}$.
  **else if** $\Delta = \Delta_{\lambda_2}$ **then**
    remove $\ell$ from $\mathcal{A}_{\lambda_t}$.
  **else if** $\Delta = \Delta_{\lambda_3}$ **then**
    update $a_i(\lambda), \forall i \in [n]$ using (3).
  **end if**
  $\mathbb{A} = \mathbb{A} \cup \mathcal{A}(\lambda_{t+1}), \mathbb{B} = \mathbb{B} \cup \{\beta_{t+1}\}$.
  Update $\psi_{\mathcal{A}_{\lambda_t}}(\lambda_t)$ using (7).
  t = t +1.
**end while**
**Output:** $\mathbb{A}, \mathbb{B}$

**Lemma 1.** *For any given node $\ell$, if $\Delta_1(\ell_1^\dagger)$ is the current minimum step size, that is,*

$$\ell_1^\dagger = \underset{j \in \{1,2,\ldots,\ell\} \cap \mathcal{A}_{\lambda_t^c}}{\arg\min} \Delta_1(j),$$

*then $\forall \ell' \supset \ell, \Delta_1(\ell') \geq \Delta_1(\ell_1^\dagger)$ if*

$$b_\ell(w(\lambda_t)) + \Delta_1(\ell_1^\dagger) b_\ell(u(\lambda_t)) + b_\ell(\kappa(\lambda_t)) < |\bar{\rho}_k(\lambda_t)| - \Delta_1(\ell_1^\dagger)|\bar{\eta}_k(\lambda_t)| - |\theta_k(\lambda_t)|, \tag{9}$$

where $w(\lambda_t) = a(\lambda_t) \odot r(\lambda_t)$, $u(\lambda_t) = a(\lambda_t) \odot v(\lambda_t)$, $\kappa(\lambda_t) = \delta(1 - a(\lambda_t)) \odot s\big(r(\lambda_t)\big)$, $\bar{\rho}_k(\lambda_t) = \rho_k(\lambda_t) - \alpha\beta_k(\lambda_t)$, $\bar{\eta}_k(\lambda_t) = \eta_k(\lambda_t) + \alpha\psi_k(\lambda_t)$, $\rho_k(\lambda_t) = x_k^\top w(\lambda_t)$, $\eta_k(\lambda_t) = x_k^\top u(\lambda_t)$, $\theta_k(\lambda_t) = x_k^\top \kappa(\lambda_t)$, and for a vector $g \in \mathbb{R}^n$ we defined

$$b_\ell(g) := \max\left\{ \sum_{g_i > 0} |g_i| x_{i\ell}, \sum_{g_i < 0} |g_i| x_{i\ell} \right\}.$$

The Lemma 1 essentially states that if for any node $\ell$ the condition in (9) is satisfied, then one can safely ignore the subtree with $\ell$ as the root node, thereby dramatically improving the computational efficiency.

**Algorithm of Huberized-SHIM.** The complete algorithm to compute the entire exact regularization path of Huberized-SHIM has been provided in Algorithm 1.

**Derivation of first $\ell^*$ and $\lambda_{max}$ :** Let's define

$$G(\ell) = |X_\ell^\top h(\lambda)|, \quad \text{then}$$
$$\ell^* = \underset{\ell \in [p]}{\arg\max}\, G(\ell), \quad \lambda_{max} = G(\ell^*). \tag{10}$$

## 4 Conformal Prediction

A single point estimate is inadequate for automated decision-making in high-risk domains (Angelino et al. 2018; Rudin 2019; Das et al. 2019; Das 2019). In such critical scenarios, equipping estimators with coverage information enhances decision-makers' confidence, enabling more informed and reliable choices when stakes are high. In this study we consider inductive (or split) conformal prediction which is widely used for its low computational burden. Given a calibration dataset $\{(x_i, y_i)\}_{i=1}^{n}$, a coverage level $\alpha \in [0, 1]$ and a new observation $x_{n+1}$, the objective of the inductive conformal prediction framework is to generate a statistically valid prediction set $\mathcal{C}(x_{n+1})$ for the unknown response $y_{n+1}$, ensuring coverage guarantees(Vovk, Gammerman, and Shafer 2005; Shafer and Vovk 2008), i.e.,

$$\mathbb{P}(y_{n+1} \in \mathcal{C}(x_{n+1})) \geq 1 - \alpha. \tag{11}$$

## 5 Results and Discussion

We evaluated our proposed method using both synthetic and real-world data. To demonstrate the statistical efficiency we used inductive conformal prediction (Papadopoulos et al. 2002; Angelopoulos and Bates 2022) and reported the mean and standard deviation of the conformal prediction (CP) set lengths ('length') and coverage ('cov') along with coefficient of determination ($R^2$). For all experiments, we considered a coverage guarantee of 90%, that is, significance level = $0.1$. To demonstrate the effectiveness of proposed robust-SHIM, we first artificially injected outliers to both synthetic and two real-world dataset. Later, we conducted experiments with two other real-world dataset, believed to contain natural outliers due to several factors.

### Experiments Using Artificially Injected Outliers

We consider 'clean' data and then gradually added 'outliers' to demonstrate the difference. The results (in Table 1, 2, 3, 4) show the 'mean (standard deviation)' of 10 independent runs in the order of Huberized-SHIM / SHIM. To simulate the effect of outliers we used the following strategy.

```
1 outlier_indices = np.random.choice(range
      (y_train.shape[0]), n_out, replace=
      True),
2 y_train[outlier_indices] += 2*(y_train.
      max() - y_train.min()),
```

where n_out represents the number of outliers. We set the $\ell_1$ regularization hyperparameter $\lambda = 1.0$, $\ell_2$ regularization hyperparameter $\alpha = 0.001$, and huber hyperparameter $\delta = 1.0$. We have chosen a random sample of size $n = 300$ which is first (randomly) split into a training set and test test using standard scikit-learn' train_test_split method considering a split-ratio of $= 0.25$. The training set is further split into a proper training set and a calibration set using the same train_test_split method considering a split-ratio of $= 0.5$. All experiments are repeated for 10 independent runs and the mean and standard deviations of all 10 independent runs have been reported. In all three experiments, it can be observed that Huberized-SHIM can mitigate the negative impact of data outliers by producing a compact (shorter length) CP set and high accuracy (larger $R^2$-score) while maintaining the desired finite sample coverage guarantee (cov $\geq 90\%$) for a chosen significance of $0.1$.

| | Clean | One outlier | Five outliers | Ten outliers |
|---|---|---|---|---|
| length | 5.53 (1.77) / 7.72 (1.28) | 5.72 (1.81) / 35.84 (11.41) | 6.09 (2.17) 94.38 (21.18) | 6.49 (2.17) / 137.26 (23.57) |
| cov | 0.89 (0.05) / 0.90 (0.04) | 0.91 (0.03) / 0.92 (0.03) | 0.91 (0.03) 0.89 (0.04) | 0.89 (0.06) 0.89 (0.03) |
| $R^2$ | 0.94 (0.02) / 0.95 (0.02) | 0.94 (0.04) / 0.01 (0.54) | 0.93 (0.04) -5.78 (2.29) | 0.93 (0.04) / -12.11 (3.63) |

Table 1: Results using synthetic SHIM data.

| | Clean | One outlier | Five outliers | Ten outliers |
|---|---|---|---|---|
| length | 0.58 (0.08) / 0.58 (0.08) | 0.62 (0.07) / 0.85 (0.15) | 1.17 (0.48) 1.68 (0.34) | 1.85 (0.39) / 2.30 (0.42) |
| cov | 0.91 (0.03) / 0.91 (0.03) | 0.91 (0.02) / 0.90 (0.04) | 0.91 (0.06) 0.92 (0.05) | 0.92 (0.04) 0.91 (0.06) |
| $R^2$ | 0.65 (0.10) / 0.65 (0.10) | 0.58 (0.10) / 0.26 (0.30) | -0.24 (0.88) -1.37 (0.94) | -2.35 (1.85) / -4.39 (2.70) |

Table 2: Results using Fluorescence data (fitness='red').

**Synthetic data.** We randomly generated i.i.d. samples $(Z_i, y_i) \in \{0, 1\}^m \times \mathbb{R}$, where $i \in [n]$, ensuring that, on average, $100m(1 - \zeta)\%$ of the features in $Z_i \in \mathbb{R}^m$ take a value of 1. The parameter $\zeta \in [0, 1]$ controls the sparsity of the design matrix, while the regularization parameter $\lambda$ governs the sparsity of the model coefficients. The effectiveness of the tree pruning condition relies on the sparsity of the design matrix, leveraging the tree's anti-monotonicity property (Definition 1). Since high-dimensional real-world data tend to be sparse, the choice of $\zeta$ in our experiments serves purely a demonstration purpose and the proposed method should work for any choice of sparsity parameter $\zeta$. The response variable $y_i \in \mathbb{R}$ is sampled from the normal distribution $\mathcal{N}(\mu(Z_i), \sigma^2)$. For demonstration, we adopt a true model incorporating up to fourth-order interactions, defined as: $\mu(Z_i) = -z_{i2} + z_{i3} + 20z_{i5} - 7z_{i2}z_{i3}z_{i4} - 20z_{i1}z_{i2}z_{i3}z_{i4}$, where $\sigma = 1$. We set $\zeta = 0.2, m = 10$ for the statistical results in Table 1. To demonstrate the efficacy of tree pruning (Table 7 and Figure 3) we used the same true model and $m = 30$, but varied the sparsity level $\zeta \in \{0.4, 0.6, 0.8\}$. This model is used solely for illustration purposes, and the proposed method is applicable to any chosen model. To generate the results in Table 1, we fit a SHIM considering interaction terms up to $3^{rd}$-order for both SHIM and Huberized-SHIM.

**Real data.** For the real data (Table 2 and Table 3) we considered Entacmaea quadricolor fluorescent protein $eqFP611$, two variant of which namely one bright deep-red ($mKate2, \lambda_{ex} = 590nm, \lambda_{em} = 635nm$) and one bright blue ($mTagBFP2, \lambda_{ex} = 405nm, \lambda_{em} = 460nm$) are separated by thirteen mutations (Poelwijk, Socolich, and Ranganathan 2019). Form biological perspective it is important to identify the crucial mutations and their pattern of epistasis (high-order interactions among mutations) that relate to the phenotypes (e.g., brightness). We also evaluated our approach using ProPublica's COMPAS recidivism dataset (Table 4), which includes seven categorical and integer-valued features along with continuous recidivism scores (Larson et al. 2016). An equivalent set of 14 binary

| | Clean | One outlier | Five outliers | Ten outliers |
|---|---|---|---|---|
| length | 1.17 (0.20) / 1.15 (0.19) | 1.14 (0.18) / 1.24 (0.27) | 1.68 (0.45) 1.99 (0.33) | 2.26 (0.51) / 2.61 (0.45) |
| cov | 0.90 (0.05) / 0.90 (0.05) | 0.90 (0.06) / 0.88 (0.08) | 0.91 (0.05) 0.91 (0.03) | 0.91 (0.03) 0.90 (0.05) |
| $R^2$ | 0.18 (0.13) / 0.18 (0.13) | 0.15 (0.14) / 0.00 (0.17) | -0.53 (0.80) -1.19 (0.70) | -2.02 (1.43) / -3.21 (1.78) |

Table 3: Results using Fluorescence data (fitness='blue').

| | Clean | One outlier | Five outliers | Ten outliers |
|---|---|---|---|---|
| length | 9.15 (0.63) / 9.48 (0.77) | 9.14 (0.64) / 9.50 (0.72) | 9.14 (0.68) 12.13 (1.68) | 9.33 (0.67) / 16.11 (2.84) |
| cov | 0.93 (0.03) / 0.93(0.02) | 0.93 (0.03) / 0.91 (0.03) | 0.93 (0.03) 0.93(0.02) | 0.93 (0.03) 0.94 (0.03) |
| $R^2$ | 0.19 (0.14) / 0.12 (0.06) | 0.17 (0.15) / 0.03 (0.18) | 0.19 (0.14) -0.46 (0.46) | 0.18 (0.15) / -1.41 (0.48) |

Table 4: Results using Compas data.

features and continuous response was obtained from the CORELS GitHub repository (Angelino et al. 2017). Model interpretabilty is crucial for the analysis of such high-stake decision making problems where an algorithm derived predictions are associated with the life of a human being or critical biological analysis. A SHIM which is interpretable by design and capable of generating flexible non-linear model due to the incorporation of high-order interaction terms can be a good fit in such settings. Furthermore, robust-SHIM (the proposed method) augments SHIM to mitigate the negative impact of possible data outliers on model fitting.

### Results of Real-World Data with Natural Outliers

In this experiment we considered other two real-world dataset ("HCV" and "Air Quality") believed to contain natural outliers. Both the dataset are first (randomly) split into a training set and test set using Scikit-learn's "train_test_split " function considering a split-ratio of $= 0.25$. The training set is further split into a proper training set and a calibration set using the same "train_test_split" method considering a split-ratio of $= 0.5$. We generated results for 3 independent runs and the mean and standard deviations of all 3 independent runs have been reported. For each run, the optimum hyperparameters are chosen using five-folds cross validation methods where the range of hyperparameters are $\lambda \in \{0.001, 0.01, 0.1, 1.0, 10.0\}, \alpha \in \{0.0001, 0.001, 0.01\}$, and $\delta \in \{1.0, 1.5, 2.0, 2.5, 3.0\}$. We used the minimum mean squared error as the metric for the selection of best hyperparameter combinations. The results are reported in Table 5 and Table 6 in the order of Huberized-SHIM / SHIM. The standard deviations are reported in brackets. We varied the maximum order of interaction from one to five and it can be observed that a high-order interaction model is statistically more efficient (shorter CP set length and high $R^2$-score). From the results one can clearly observe that Huberized-SHIM can mitigate the presence of natural data outliers in real-world data by producing a shorter CP set length and high $R^2$-score compared to that of a SHIM.

**HCV data.** Here, we considered clinical laboratory data, namely the "HCV (Hepatitis C Virus) blood test dataset" from UCI ML repository (Lichtinghagen, Klawonn, and

| max-depth | 1st | 2nd | 3rd | 4th |
|---|---|---|---|---|
| length | 39.21 (4.62) / 43.35 (3.96) | 36.91 (2.77) / 41.15 (5.47) | 36.74 (3.95) 42.99 (5.22) | 37.33 (3.49) / 42.10 (5.15) |
| cov | 0.91 (0.03) / 0.90(0.02) | 0.91 (0.03) / 0.90 (0.03) | 0.92 (0.03) 0.92(0.02) | 0.92 (0.02) 0.92 (0.02) |
| $R^2$ | 0.25 (0.06) / -0.37 (0.34) | 0.48 (0.11) / -0.02 (0.57) | 0.49 (0.06) 0.36 (0.04) | 0.50 (0.07) / 0.31 (0.06) |

Table 5: Results using HCV data.

| max-depth | 1st | 2nd | 3rd | 4th | 5th |
|---|---|---|---|---|---|
| length | 2.39 (0.12) / 2.47 (0.08) | 0.16 (0.01) / 0.17 (0.01) | 0.15 (0.00) 0.16 (0.01) | 0.15 (0.01) / 0.16 (0.01) | 0.15 (0.01) / 0.16 (0.01) |
| cov | 0.92 (0.01) / 0.91(0.00) | 0.90 (0.03) / 0.91 (0.02) | 0.92 (0.01) 0.92(0.01) | 0.92 (0.02) 0.92 (0.02) | 0.91 (0.01) 0.91 (0.00) |
| $R^2$ | 0.99 (0.00) / 0.99 (0.00) | 1.0 (0.00) / 1.0 (0.00) | 1.0 (0.00) 1.0 (0.00) | 1.00 (0.00) / 1.00 (0.00) | 1.0 (0.00) / 1.0 (0.00) |

Table 6: Results using air quality data.

Hoffmann 2020). This dataset includes 13 clinical lab features (e.g. ALT, AST, Bilirubin, Albumin, etc.) and it is believed to contain outliers due to biological variability and measurement noise. This data includes both real, binary, integer and categorical values. Among the continuous features, ALT, AST, or Bilirubin are routinely used to assess liver inflammation and damage, and they often show extreme values in hepatitis, fibrosis, or cirrhosis cases (Hoffmann et al. 2018). In our studies, we consider ALT (Alanine Aminotransferase) as the response variable, a liver enzyme that's highly sensitive to hepatocellular injury and known to exhibit natural outliers due to disease progression, alcohol use, or medication effects. Elevated ALT levels indicate liver inflammation or damage and are used routinely in liver function panels. This dataset contains $n = 615$ instances, however after removing the missing entries there are $n = 589$ instances. The categorical columns, namely 'Category' and 'Sex' are converted into binary features using "OneHotEncoder" and the remaining numeric features are scaled between 0 and 1 using "MinMaxScaler" functions of Scikit-learn. Therefore, the final dataset consists of a mix of both binary and continuous features, totaling 18 features.

**Air Quality data.** The air quality dataset was obtained from the UCI machine learning dataset (Vito 2008; Vito et al. 2008). This dataset contains hourly averaged readings from 5 metal oxide chemical sensors deployed in a heavily polluted Italian city. The full dataset consists of 9,358 hourly entries across 13 variables (excluding date/time), featuring integer, categorical, and real-valued data. However, many records contain missing values (denoted by -200), and after cleaning, only 827 instances remain. It is reported that the evidence of cross-sensitivity and drift (both conceptual and technical) are present in this data, which may impact the accuracy of gas concentration estimates, making this data ideal for the analysis of robust regression methods. In this study, we considered benzene concentration "C6H6(GT)" as the target variable, with the remaining 12 features as predictors.

## 6 Computational Efficiency Analysis

To demonstrate the computational efficiency of the proposed pruning strategy for the $\lambda$-path, we generated a synthetic dataset of $n = 100$ and $m = 30$ for three different sparsity levels of the design matrix ($\zeta = 0.4, 0.6, 0.8$) using the same

| $d$ | Search space (# nodes) | $\lambda = 1$ | | | | | | $\lambda = 0.1$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | With pruning | | | Without pruning | | | With pruning | | | Without pruning | | |
| | | $\zeta = 0.4$ | $\zeta = 0.6$ | $\zeta = 0.8$ | $\zeta = 0.4$ | $\zeta = 0.6$ | $\zeta = 0.8$ | $\zeta = 0.4$ | $\zeta = 0.6$ | $\zeta = 0.8$ | $\zeta = 0.4$ | $\zeta = 0.6$ | $\zeta = 0.8$ |
| 2 | 465 | 0.142 | 0.108 | 0.089 | 0.122 | 0.084 | 0.128 | 0.439 | 0.140 | 0.103 | 0.406 | 0.136 | 0.117 |
| 3 | 4525 | 0.896 | 0.559 | 0.227 | 0.944 | 0.850 | 1.195 | 1.332 | 0.688 | 0.138 | 1.551 | 1.020 | 0.784 |
| 4 | 31930 | 2.989 | 1.614 | 0.317 | 6.360 | 5.980 | 6.596 | 2.976 | 1.583 | 0.259 | 6.850 | 6.345 | 5.988 |
| 5 | 174436 | 5.918 | 2.686 | 0.334 | 39.711 | 33.361 | 30.664 | 5.026 | 2.169 | 0.213 | 48.511 | 38.381 | 38.965 |
| 10 | 53009101 | 48.366 | 5.226 | 0.368 | $> 1$ day | $> 1$ day | $> 1$ day | 28.558 | 6.229 | 0.230 | $> 1$ day | $> 1$ day | $> 1$ day |
| 15 | 614429671 | 50.747 | 4.981 | 0.345 | $> 1$ day | $> 1$ day | $> 1$ day | 34.200 | 6.192 | 0.283 | $> 1$ day | $> 1$ day | $> 1$ day |
| 20 | 1050777736 | 50.874 | 4.678 | 0.258 | $> 1$ day | $> 1$ day | $> 1$ day | 31.658 | 6.309 | 0.162 | $> 1$ day | $> 1$ day | $> 1$ day |
| 25 | 1073709892 | 50.533 | 3.949 | 0.364 | $> 1$ day | $> 1$ day | $> 1$ day | 32.800 | 3.552 | 0.173 | $> 1$ day | $> 1$ day | $> 1$ day |

Table 7: Average computation time (in sec) with and without pruning using two different $\lambda$ values for three different sparsity levels ($\zeta$). All computation times were measured on Intel(R) Xeon(R) E5-2690 CPU @ 2.60GHz, RAM 256 GB, CentOS.

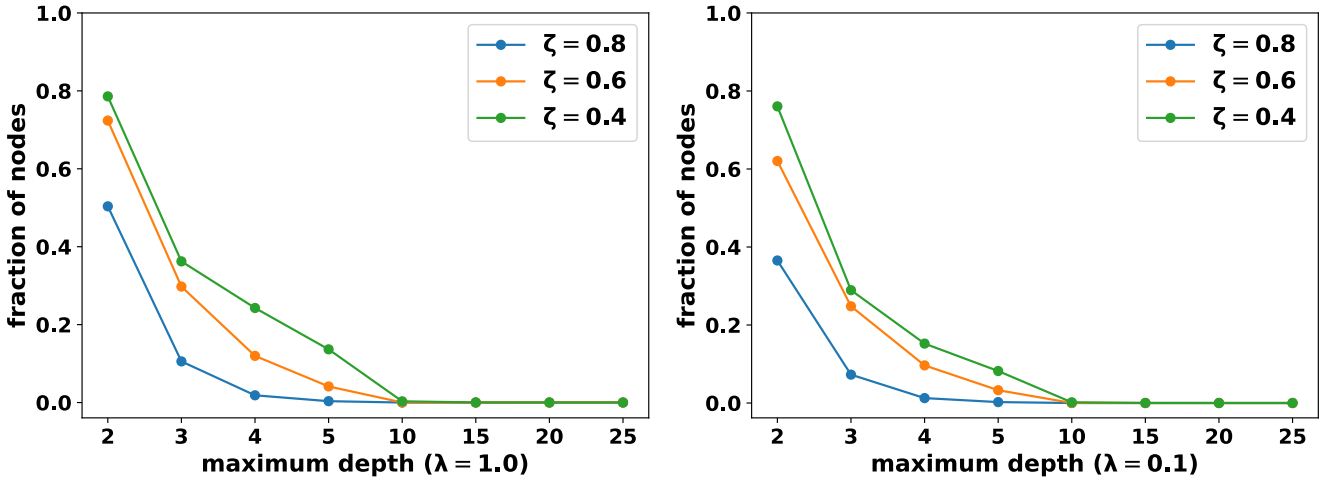


Figure 3: This figure illustrates how the proportion of nodes explored ("fraction of nodes") varies with different values of maximum depth. For each maximum depth, the fraction is calculated as the number of nodes traversed divided by the total number of possible combinations of interaction terms. Results are presented for two different $\lambda$ values ($\lambda = 0.1$ and 1), across three levels of sparsity: $\zeta = 0.4, 0.6,$ and $0.8$. Notably, pruning becomes more effective as the dataset becomes sparser.

$4^{th}$ order model as used to demonstrate the statistical efficiency in Tables 1. We compared both the fraction of nodes traversed (Figure 3) and the time taken (Table 7) during the step-size of inclusion $\Delta_1(\ell_1^*)$ calculation in (8) against a different maximum interaction order $d$ for three different sparsity levels ($\zeta = 0.4, 0.6, 0.8$) using two different $\lambda$ values ($\lambda = 1, 0.1$). The results in Table 7 and Figure 3 show the average computation time (in sec) and average fraction of node counts, averaged over all the kinks of the regularization path with and without tree pruning. It can be observed that the tree pruning is more effective at the deeper nodes of the tree and saturates after a certain depth of the tree. This is evident as the sparsity of the data increases at the deeper nodes, and the pruning exploits the anti-monotonicity of high-order interaction terms constructed as tree of patterns. In the case of the homotopy method without pruning, we stopped the execution of the program if the $\lambda$-path was not finished in one day. From Table 7, it can be observed that without the tree pruning, the construction of the $\lambda$-path is not practical as we progress to the deeper nodes of the tree because of the generation of an exponential number of high-order interaction terms. Figure 3 shows the variation of node counts ('fraction of node counts') for different maximum depth during the construction of $\lambda$-path. One can observe that our pruning condition is more effective when data is highly sparse and also at the deeper nodes of the tree.

## 7 Conclusion

This paper introduces Huberized-SHIM, an extension of the Sparse High-order Interaction Model (SHIM) that enhances robustness against outliers while maintaining interpretability in high-stakes applications. The proposed homotopy-based regularization path algorithm and tree-pruning criterion efficiently manage computational complexity, making SHIM scalable for real-world datasets. Additionally, the incorporation of conformal prediction provides statistical coverage guarantees, reinforcing model reliability. Our experiments demonstrate that Huberized-SHIM surpasses standard SHIM in robustness and predictive accuracy, offering a powerful tool for transparent, data-driven decision-making.

## Acknowledgments

Diptesh Das is supported by JSPS KAKENHI 23K16942. Koji Tsuda is supported by JST MIRAI JPMJMI24H2, JST ERATO JPMJER1903 and JST CREST JPMJCR21O2.

## References

Angelino, E.; Larus-Stone, N.; Alabi, D.; Seltzer, M.; and Rudin, C. 2017. Learning certifiably optimal rule lists. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 35–44.

Angelino, E.; Larus-Stone, N.; Alabi, D.; Seltzer, M.; and Rudin, C. 2018. Learning Certifiably Optimal Rule Lists for Categorical Data. *Journal of Machine Learning Research*, 18(234): 1–78.

Angelopoulos, A. N.; and Bates, S. 2022. A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. arXiv:2107.07511.

Das, D. 2019. *Interpretable Machine Learning Models for Medical Data*. Ph.D. diss., Department of Computational Biology and Medical Sciences, The University of Tokyo., Kashiwa, Japan.

Das, D.; Ito, J.; Kadowaki, T.; and Tsuda, K. 2019. An interpretable machine learning model for diagnosis of Alzheimer's disease. *PeerJ*, 7: e6543.

Das, D.; Le Duy, V. N.; Hanada, H.; Tsuda, K.; and Takeuchi, I. 2022. Fast and more powerful selective inference for sparse high-order interaction model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 9999–10007.

Das, D.; Ndiaye, E.; and Takeuchi, I. 2024. A confidence machine for sparse high-order interaction model. *Stat*, 13(1): e633.

Das, J. 2023. *Comparison of Different Robust Methods in Linear Regression and Applications in Cardiovascular Data*. Master's thesis, The University of Texas at El Paso.

Deng, W.; Zhang, K.; He, C.; Liu, S.; and Wei, H. 2021. HB-PLS: A statistical method for identifying biological process or pathway regulators by integrating Huber loss and Berhu penalty with partial least squares regression. *Forestry Research*, 1: 6.

Deutelmoser, H.; Scherer, D.; Brenner, H.; Waldenberger, M.; study, I.; Suhre, K.; Kastenmüller, G.; and Lorenzo Bermejo, J. 2021. Robust Huber-LASSO for improved prediction of protein, metabolite and gene expression levels relying on individual genotype data. *Briefings in bioinformatics*, 22(4): bbaa230.

Efron, B.; Hastie, T.; Johnstone, I.; and Tibshirani, R. 2004. Least Angle Regression. *The Annals of Statistics*, 32: 407–451.

Fannjiang, C.; Bates, S.; Angelopoulos, A. N.; Listgarten, J.; and Jordan, M. I. 2022. Conformal prediction under feedback covariate shift for biomolecular design. *Proceedings of the National Academy of Sciences*, 119(43): e2204569119.

He, M.; Hao, X.; Zhang, Y.; and Meng, F. 2021. Forecasting stock return volatility using a robust regression model. *Journal of forecasting*, 40(8): 1463–1478.

Hoffmann, G.; Bietenbeck, A.; Lichtinghagen, R.; and Klawonn, F. 2018. Using machine learning techniques to generate laboratory diagnostic pathways—a case study. *Journal of Laboratory and Precision Medicine*, 3(6).

Huang, C.; Das, D.; and Tsuda, K. 2023. Feature Importance Measurement based on Decision Tree Sampling. arXiv:2307.13333.

Huber, P. J. 1964. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1): 73–101.

Huber, P. J.; and Ronchetti, E. M. 2011. *Robust statistics*. John Wiley & Sons.

Iversen, A.; Shafer, R. W.; Wehrly, K.; Winters, M. A.; Mullins, J. I.; Chesebro, B.; and Merigan, T. C. 1996. Multidrug-resistant human immunodeficiency virus type 1 strains resulting from combination antiretroviral therapy. *Journal of virology*, 70(2): 1086–1090.

Karim, S.; Iqbal, M. S.; Ahmad, N.; Ansari, M. S.; Mirza, Z.; Merdad, A.; Jastaniah, S. D.; and Kumar, S. 2023. Gene expression study of breast cancer using Welch Satterthwaite t-test, Kaplan-Meier estimator plot and Huber loss robust regression model. *Journal of King Saud University-Science*, 35(1): 102447.

Korgialas, C.; and Kotropoulos, C. 2023. On Robust Electric Network Frequency Detection Using Huber Regression. In *Proceedings of the 27th Pan-Hellenic Conference on Progress in Computing and Informatics*, 249–253.

Larson, J.; Mattu, S.; Kirchner, L.; and Angwin, J. 2016. How We Analyzed the COMPAS Recidivism Algorithm. https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm. Accessed: 2025-11-09.

Lichtinghagen, R.; Klawonn, F.; and Hoffmann, G. 2020. HCV data. https://archive.ics.uci.edu/dataset/571/hcv+data. Accessed: 2025-11-09.

Nakagawa, K.; Suzumura, S.; Karasuyama, M.; Tsuda, K.; and Takeuchi, I. 2016. Safe pattern pruning: An efficient approach for predictive pattern mining. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 1785–1794.

Normolle, D. P. 1993. AN algorithm for robust non-linear analysis of radioimmunoassays and other bioassays. *Statistics in medicine*, 12(21): 2025–2042.

Owen, A. B. 2007. A robust hybrid of lasso and ridge regression. *Contemporary Mathematics*, 443(7): 59–72.

Papadopoulos, H.; Proedrou, K.; Vovk, V.; and Gammerman, A. 2002. Inductive confidence machines for regression. In *Proceedings of the 13th European Conference on Machine Learning*, 345–356. Springer-Verlag.

Pervez, A.; and Ali, I. 2024. Robust regression analysis in analyzing financial performance of public sector banks: A case study of India. *Annals of Data Science*, 11(2): 677–691.

Poelwijk, F. J.; Socolich, M.; and Ranganathan, R. 2019. Learning the pattern of epistasis linking genotype and phenotype in a protein. *Nature communications*, 10(1): 4213.

Reichel, F. 2025. Statistically Significant Linear Regression Coefficients Solely Driven By Outliers In Finite-sample Inference. arXiv:2505.10738.

Rhee, S.-Y.; Taylor, J.; Wadhera, G.; Ben-Hur, A.; Brutlag, D. L.; and Shafer, R. W. 2006. Genotypic predictors of human immunodeficiency virus type 1 drug resistance. *Proceedings of the National Academy of Sciences*, 103(46): 17355–17360.

Rosset, S.; and Zhu, J. 2007. Piecewise linear regularized solution paths. *The Annals of Statistics*, 35(3): 1012–1030.

Rudin, C. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5): 206–215.

Saigo, H.; Nowozin, S.; Kadowaki, T.; Kudo, T.; and Tsuda, K. 2009. gBoost: a mathematical programming approach to graph classification and regression. *Machine Learning*, 75(1): 69–89.

Saigo, H.; Uno, T.; and Tsuda, K. 2007. Mining complex genotypic features for predicting HIV-1 drug resistance. *Bioinformatics*, 23(18): 2455–2462.

Shafer, G.; and Vovk, V. 2008. A Tutorial on Conformal Prediction. *Journal of Machine Learning Research*, 9: 371–421.

Suzumura, S.; Nakagawa, K.; Umezu, Y.; Tsuda, K.; and Takeuchi, I. 2017. Selective Inference for Sparse High-Order Interaction Models. In *Proceedings of the 34th International Conference on Machine Learning*, 3338–3347. PMLR.

Tsuda, K. 2007. Entire Regularization Paths for Graph Data. In *Proceedings of the 24th International Conference on Machine Learning*, 919–926. Association for Computing Machinery.

Tsukurimichi, T.; Inatsu, Y.; Duy, V. N. L.; and Takeuchi, I. 2022. Conditional selective inference for robust regression and outlier detection using piecewise-linear homotopy continuation. *Annals of the Institute of Statistical Mathematics*, 74(6): 1197–1228.

Vito, S. 2008. Air Quality. https://archive.ics.uci.edu/dataset/360/air+quality. Accessed: 2025-07-30.

Vito, S. D.; Massera, E.; Piga, M.; Martinotto, L.; and Francia, G. D. 2008. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. *Sensors and Actuators B-chemical*, 129: 750–757.

Vivet-Boudou, V.; Didierjean, J.; Isel, C.; and Marquet, R. 2006. Nucleoside and nucleotide inhibitors of HIV-1 replication. *Cellular and Molecular Life Sciences CMLS*, 63: 163–186.

Vovk, V.; Gammerman, A.; and Shafer, G. 2005. *Algorithmic learning in a random world*. Springer-Verlag.

Wilcox, R. R. 1996. A review of some recent developments in robust regression. *British Journal of Mathematical and Statistical Psychology*, 49(2): 253–274.