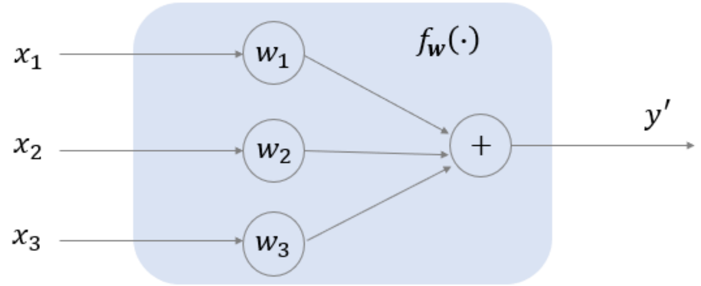
CS 4501/6501 Interpretable Machine Learning

Interpretable Generalized Additive Models

Hanjie Chen, Yangfeng Ji
Department of Computer Science
University of Virginia
{hc9mx, yangfeng}@virginia.edu

Interpretability

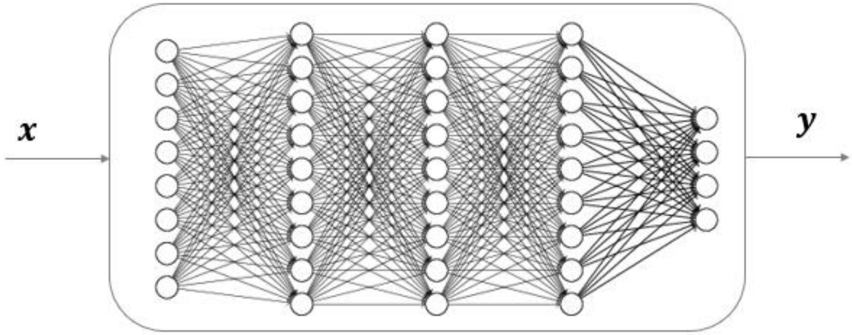
Bad performance
Good interpretability



- Three parameters (w_1, w_2, w_3)
- $y' = w_1x_1 + w_2x_2 + w_3x_3$
- Contributions:

$$x_1 : w_1x_1$$
$$x_2 : w_2x_2$$
$$x_3 : w_3x_3$$

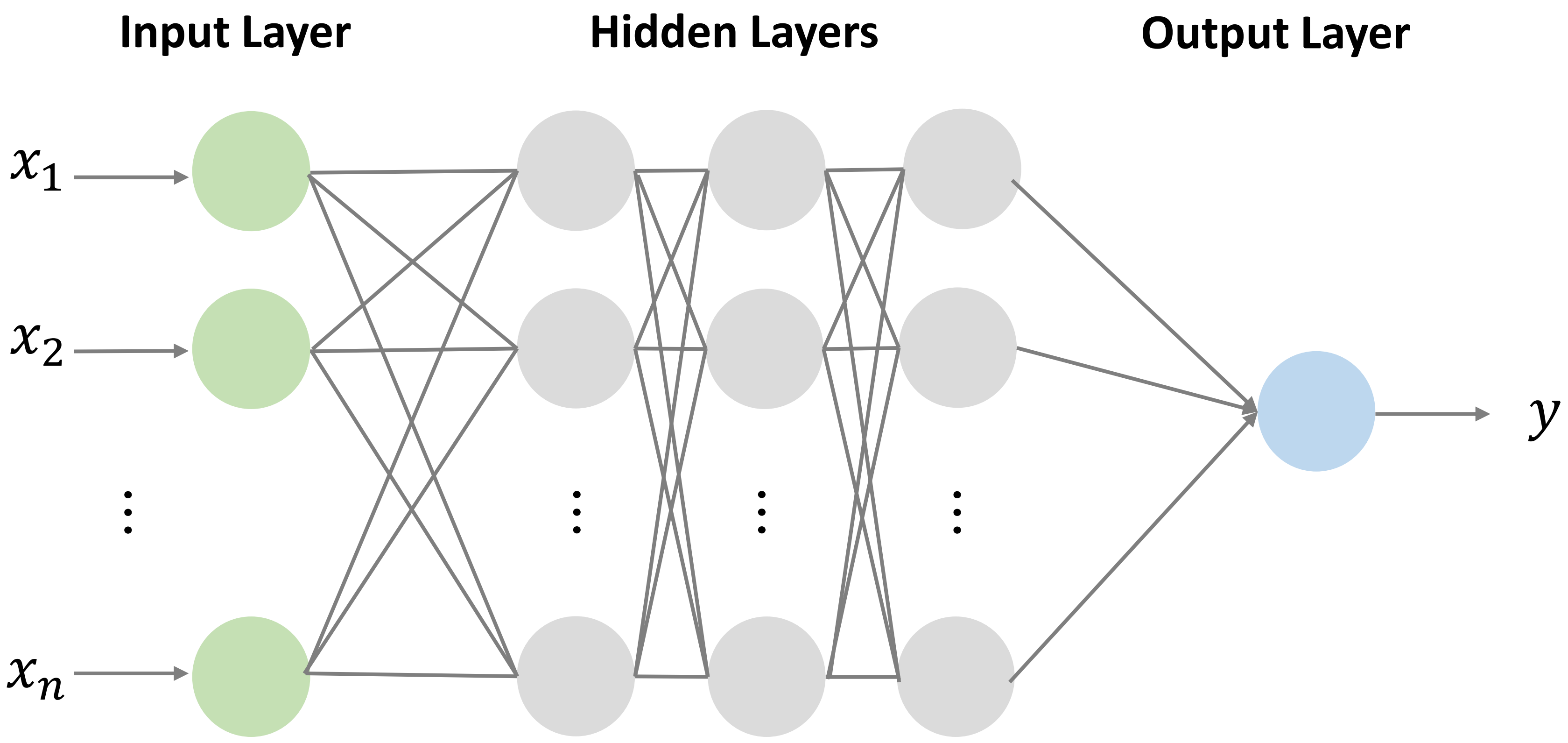
Good performance
Bad interpretability



- Millions of parameters
- $\boldsymbol{y}' = f_{\boldsymbol{w}}(\boldsymbol{x})$ (complex transformations)
- Model decision-making and feature attributions are unclear

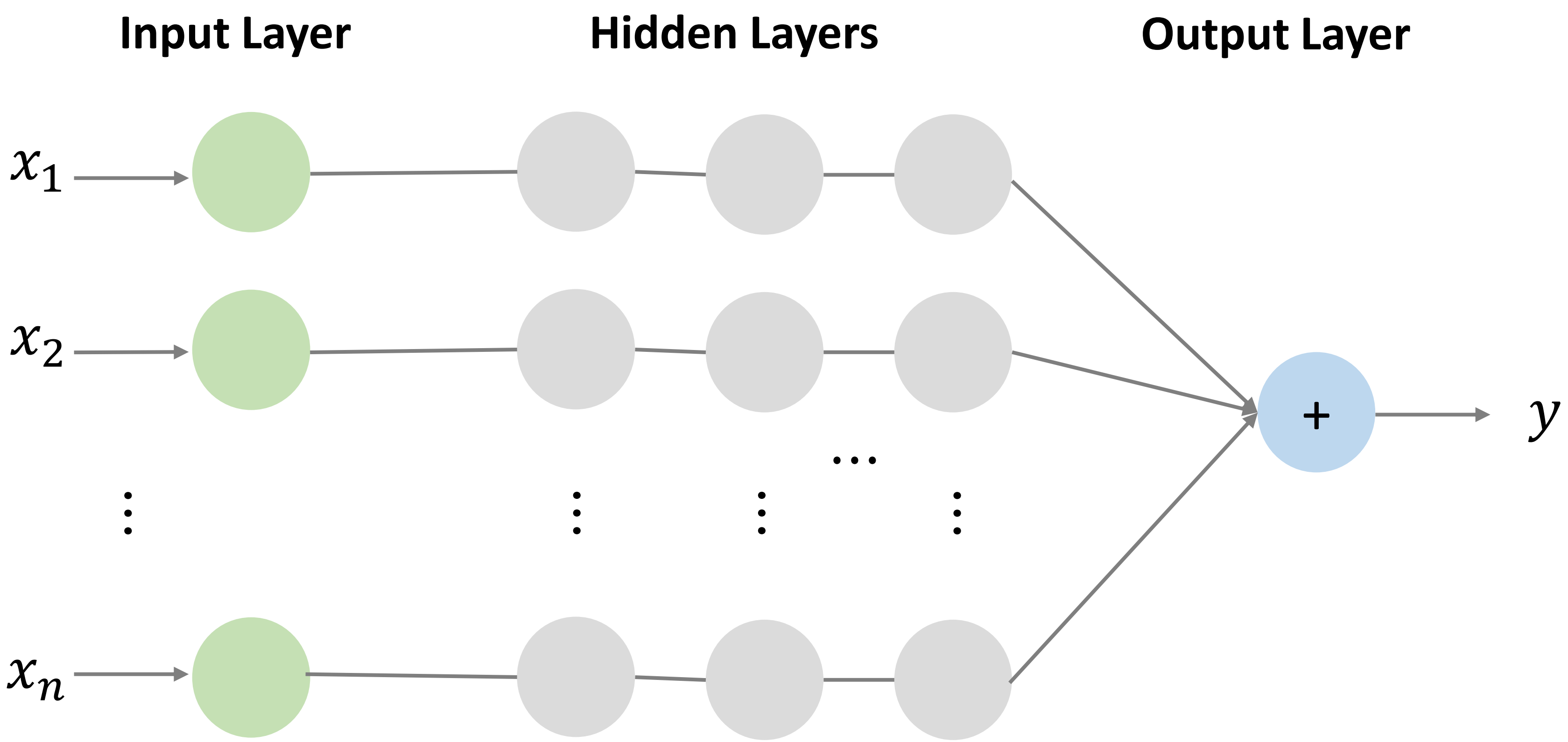
Trade-off

The information of input features is mixed



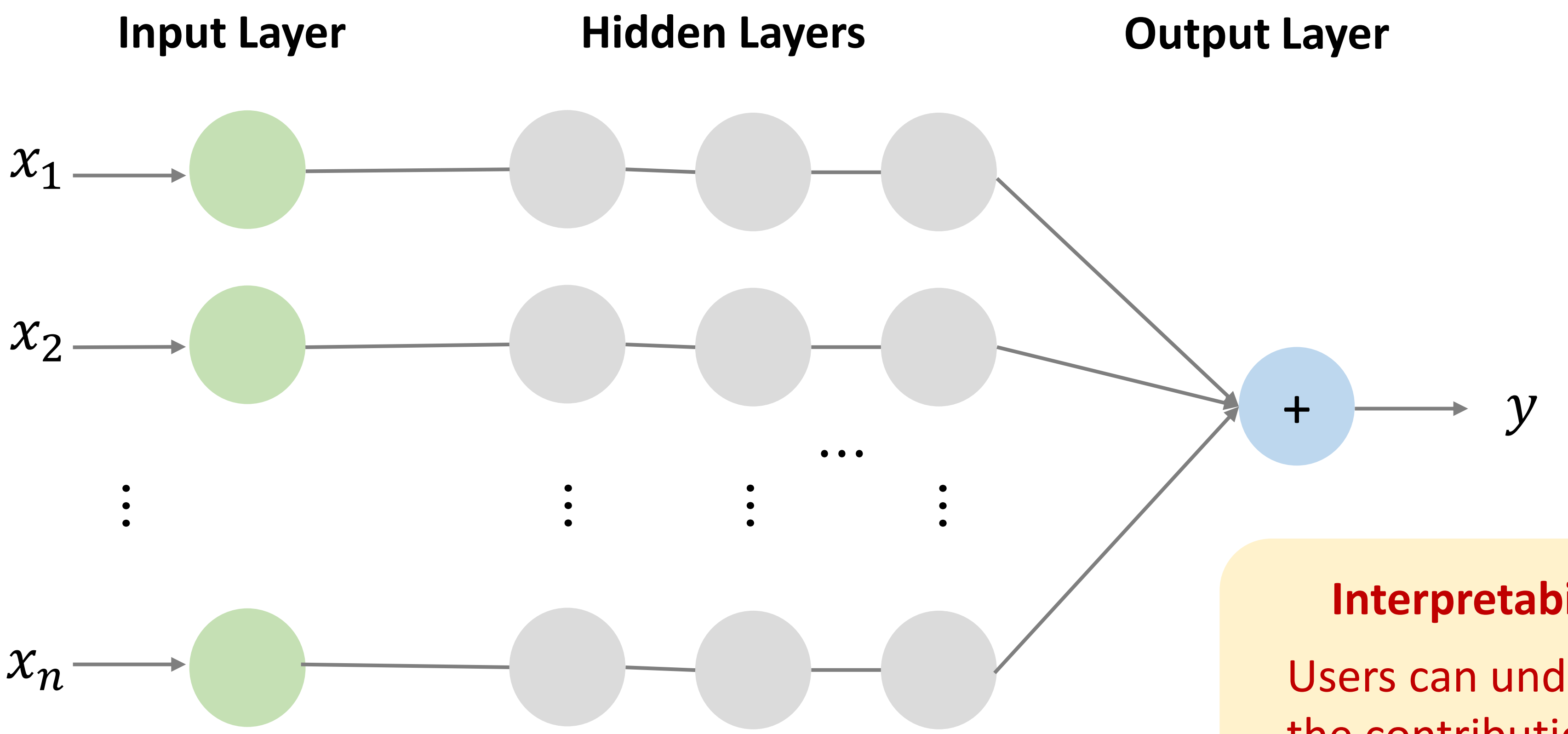
Trade-off

Keep the information of individual features “locally”



Trade-off

Keep the information of individual features "locally"



Interpretability

Users can understand the contributions of individual features

Trade-off

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- Permit complex relationships between individual features (x_i) and the target ($g(y)$)
- Exclude complex interactions between features

GAM

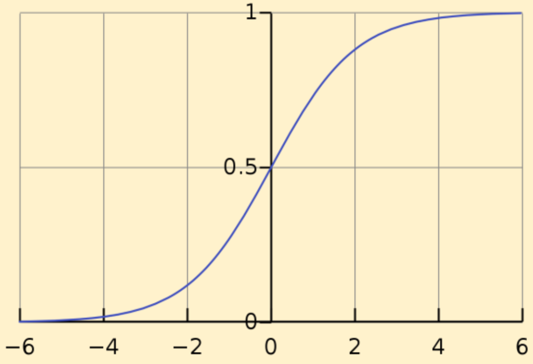
Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $g(\cdot)$: link function
  - Identity: $g(y) = y$ ⟶ Regression
  - Logistic function: $g(y)$ represents the probability on a class ⟶ Classification

$$\frac{L}{1 + e^{-k(x-x_0)}}$$

$$(L = 1, k = 1, x_0 = 0)$$

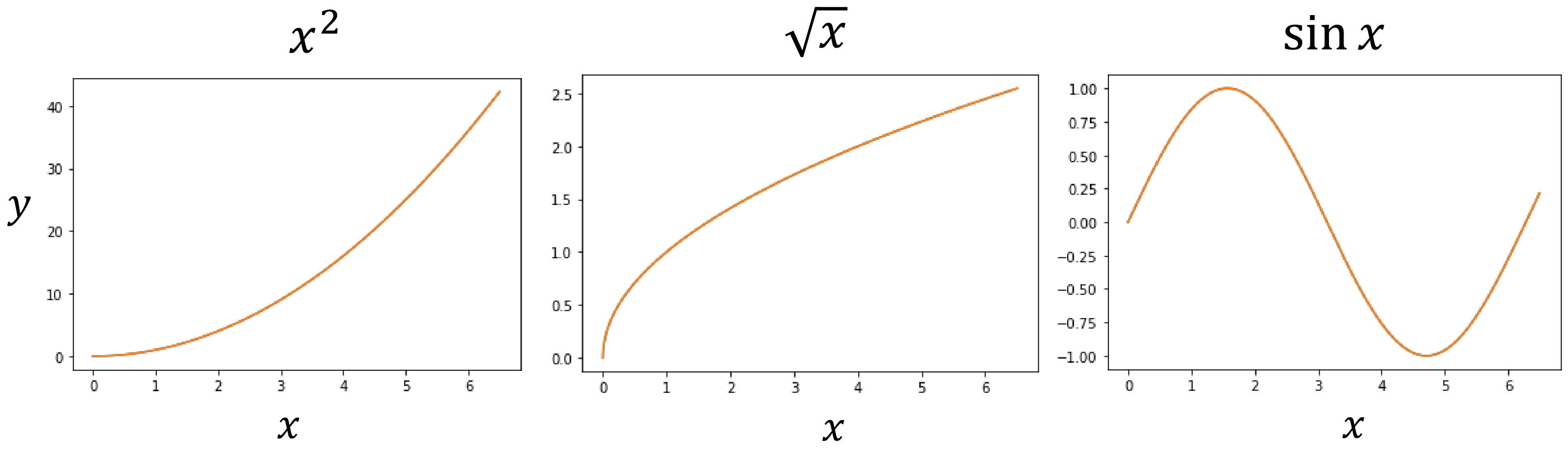


GAM

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $f_i(\cdot)$: shape function
  - Splines

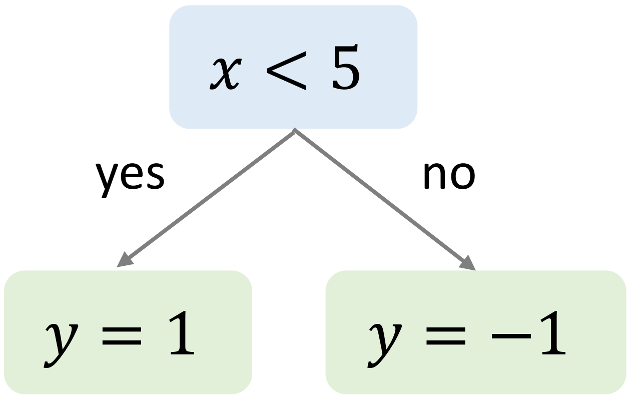


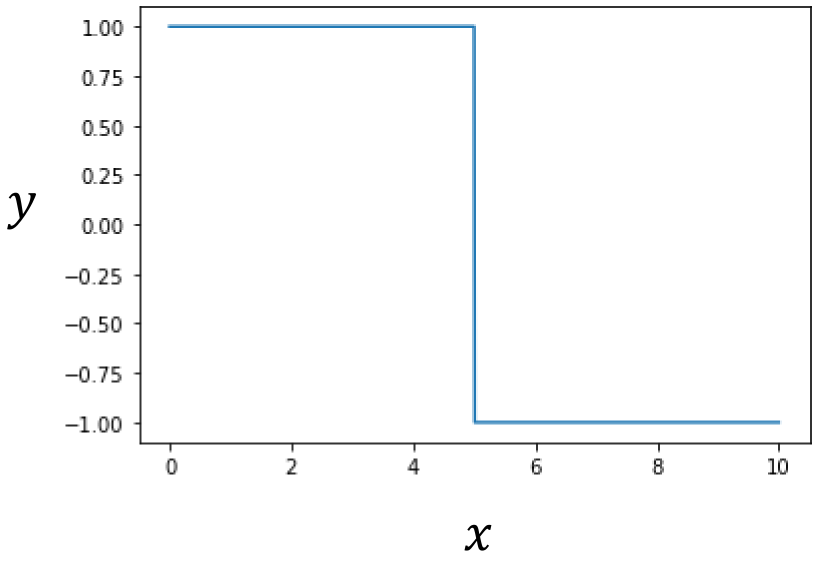
GAM

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $f_i(\cdot)$: shape function
  - Binary Trees



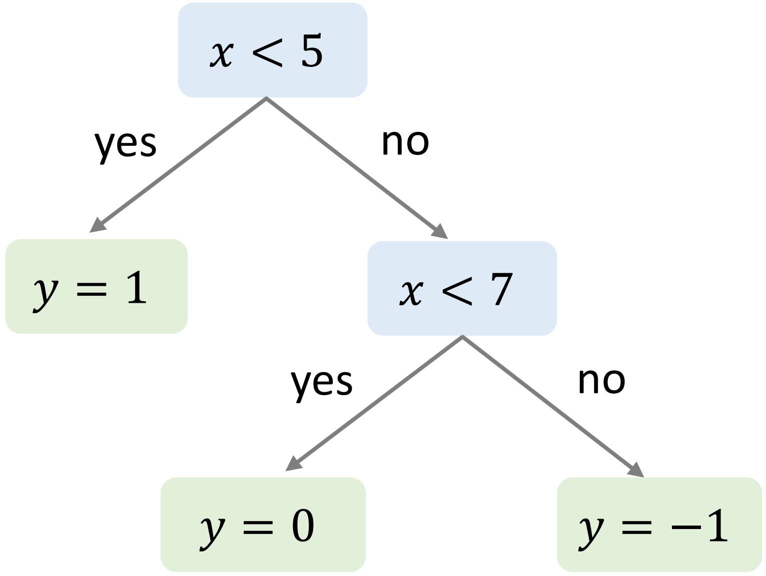


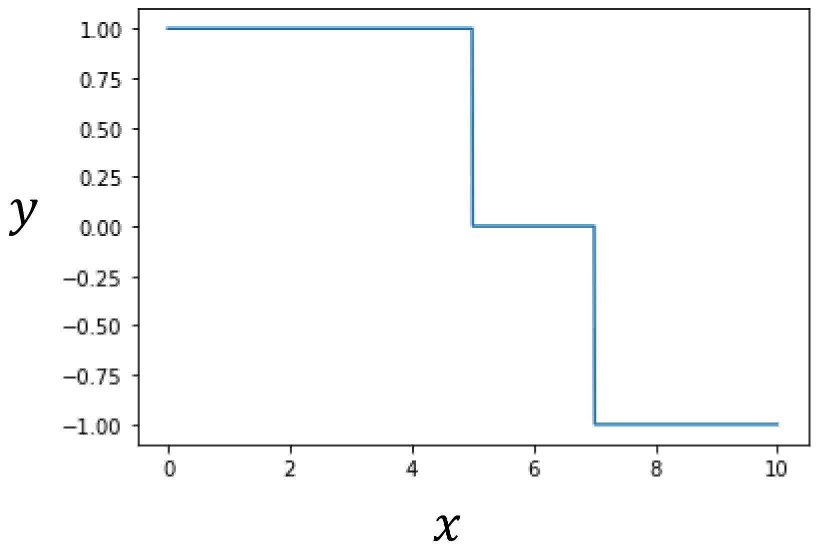
GAM

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $f_i(\cdot)$: shape function
  - Binary Trees





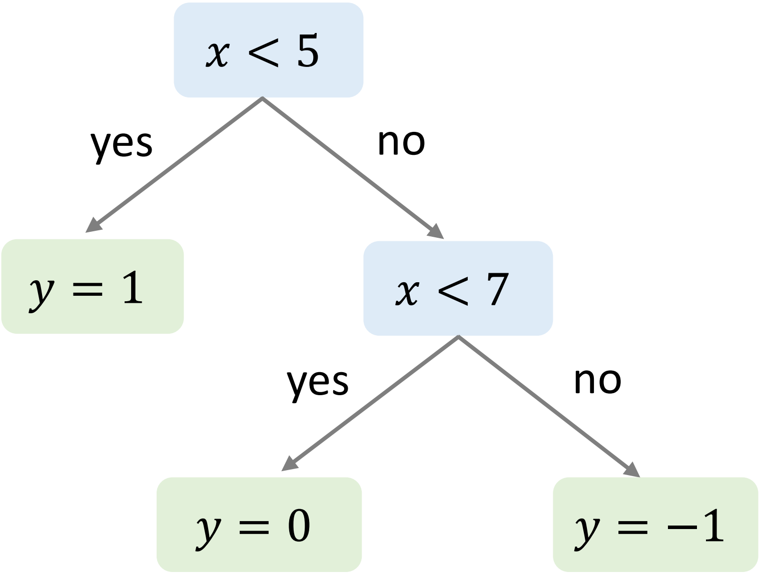
GAM

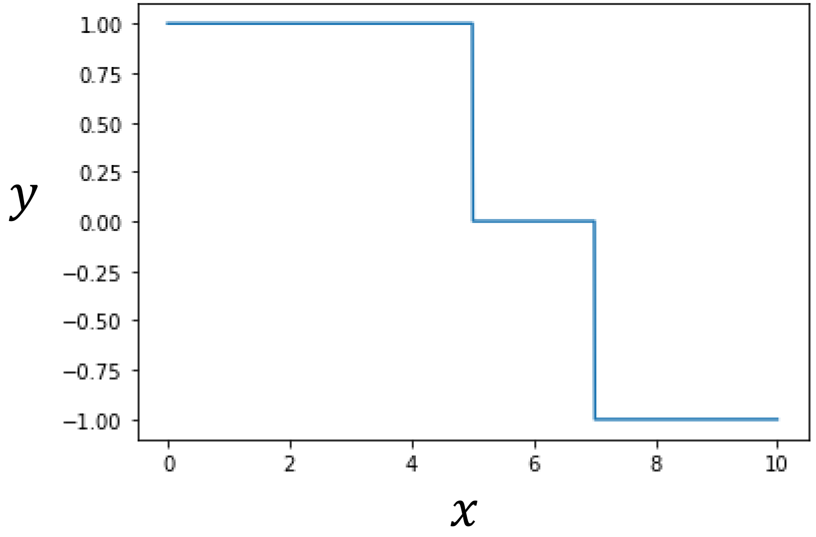
Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $f_i(\cdot)$: shape function
  - Binary Trees

For interpretability, we control tree complexity (nodes, leaves, depth)



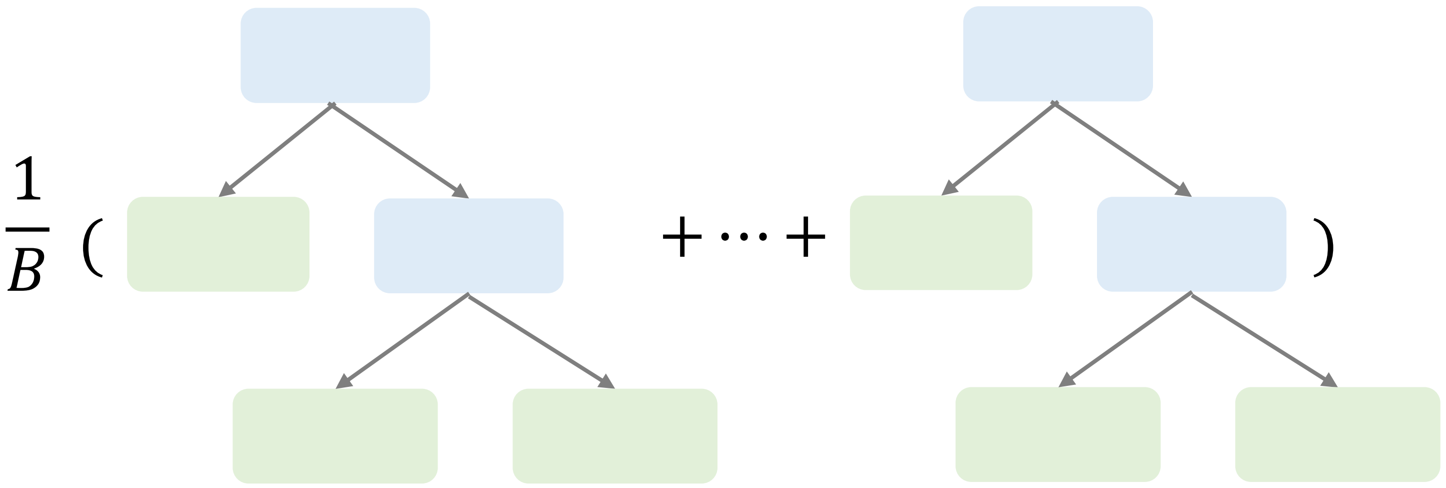


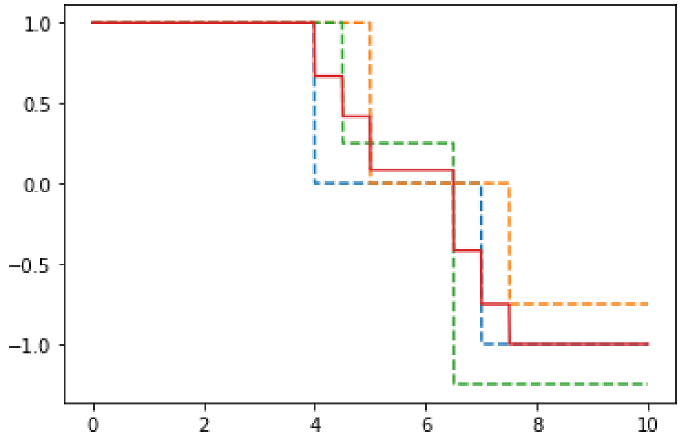
GAM

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- $f_i(\cdot)$: shape function
  - Bagged Trees (reduce the variance)





GAM

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- Training
  - Shape functions for individual features
  - Learning methods

Learning GAM

- Gradient Boosting
  - Learning tree or tree ensemble shape functions

Algorithm Gradient Boosting for GAM

1. $f_j \leftarrow 0, j = 1, \cdots, n$ Initialize all shape functions as zero
2. **for** $m = 1, \cdots, M$ **do** Loop over M iterations
3. **for** $j = 1, \cdots, n$ **do** Loop over all features
4. $\mathcal{R} \leftarrow \left\{x_{ij}, y_i - \sum_k f_k\right\}_{i=1}^{N}$ Calculate residuals
5. Learning shape function $S: x_j \rightarrow y$ using $\mathcal{R}$ as training data Learn the one-dimensional function to predict the residuals
6. $f_j \leftarrow f_j + S$ Update the shape function

Learning GAM

- Gradient Boosting
  - Learning tree or tree ensemble shape functions

Algorithm Gradient Boosting for GAM

1. $f_j \leftarrow 0, j = 1, \cdots, n$ Initialize all shape functions as zero
2. **for** $m = 1, \cdots, M$ **do** Loop over M iterations
3. **for** $j = 1, \cdots, n$ **do** Loop over all features
4. $\mathcal{R} \leftarrow \left\{x_{ij}, y_i - \sum_k f_k\right\}_{i=1}^{N}$ Calculate residuals
5. Learning shape function S: $x_j \rightarrow y$ using $\mathcal{R}$ as training data Learn the one-dimensional function to predict the residuals
6. $f_j \leftarrow f_j + S$ Update the shape function

Learning GAM

- Gradient Boosting

Training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$

i	x_1	x_2	$\cdots$	x_j	$\cdots$	x_n	y
1	x_{11}	x_{12}	$\cdots$	x_{1j}	$\cdots$	x_{1n}	y_1
$\boldsymbol{x}_2$ 2	x_{21}	x_{22}	$\cdots$	x_{2j}	$\cdots$	x_{2n}	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	x_{N1}	x_{N2}	$\cdots$	x_{Nj}	$\cdots$	x_{Nn}	y_N

Learning GAM

- Gradient Boosting

Training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$

i	x_1	x_2	$\cdots$	x_j	$\cdots$	x_n	y
1	x_{11}	x_{12}	$\cdots$	x_{1j}	$\cdots$	x_{1n}	y_1
2	x_{21}	x_{22}	$\cdots$	x_{2j}	$\cdots$	x_{2n}	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	x_{N1}	x_{N2}	$\cdots$	x_{Nj}	$\cdots$	x_{Nn}	y_N

Learning GAM

- Gradient Boosting

Training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$

i	x_1	x_2	$\cdots$	x_j	$\cdots$	x_n	y
1	x_{11}	x_{12}	$\cdots$	x_{1j}	$\cdots$	x_{1n}	y_1
2	x_{21}	x_{22}	$\cdots$	x_{2j}	$\cdots$	x_{2n}	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	x_{N1}	x_{N2}	$\cdots$	x_{Nj}	$\cdots$	x_{Nn}	y_N

Learning GAM

- Gradient Boosting

Training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$

f_j

i	x_1	x_2	$\cdots$	x_j	$\cdots$	x_n	y
1	x_{11}	x_{12}	$\cdots$	x_{1j}	$\cdots$	x_{1n}	y_1
2	x_{21}	x_{22}	$\cdots$	x_{2j}	$\cdots$	x_{2n}	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	x_{N1}	x_{N2}	$\cdots$	x_{Nj}	$\cdots$	x_{Nn}	y_N

Residuals

$\rightarrow y_1 - \sum_k f_k$

$\rightarrow y_2 - \sum_k f_k$

$\vdots$

$\rightarrow y_N - \sum_k f_k$

(errors made by the current model)

Learning GAM

- Gradient Boosting

Update f_j based on $\{(\underbrace{x_{ij}}_{x}, \underbrace{y_i - \sum_k f_k}_{y})\}_{i=1}^{N}$

- Learn a shape function S that fits: $x \to y$
- Update $f_j \leftarrow f_j + S$

Learning GAM

- Gradient Boosting

Example

x	y
1	8
5	5
3	8
9	7

Residuals

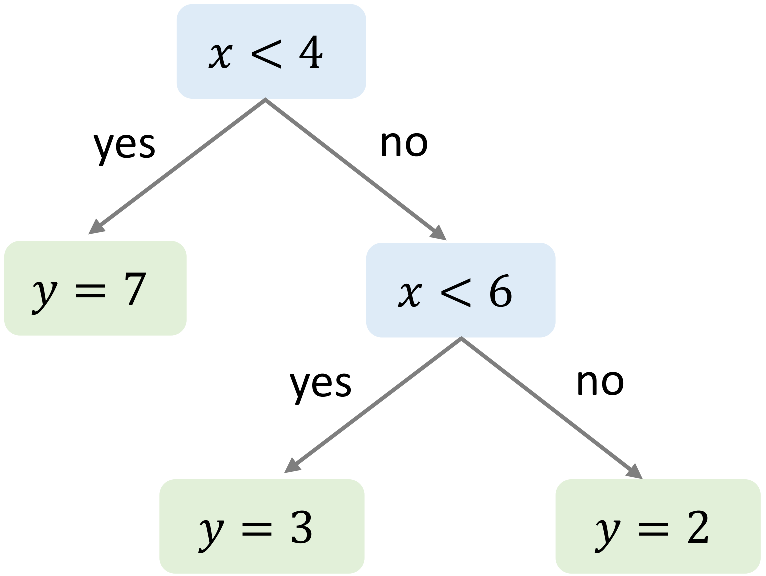
$8 - 7 = 1$

$5 - 3 = 2$

$8 - 7 = 1$

$7 - 2 = 5$

f_j



Learning GAM

- Gradient Boosting

Example

x	y
1	8
5	5
3	8
9	7

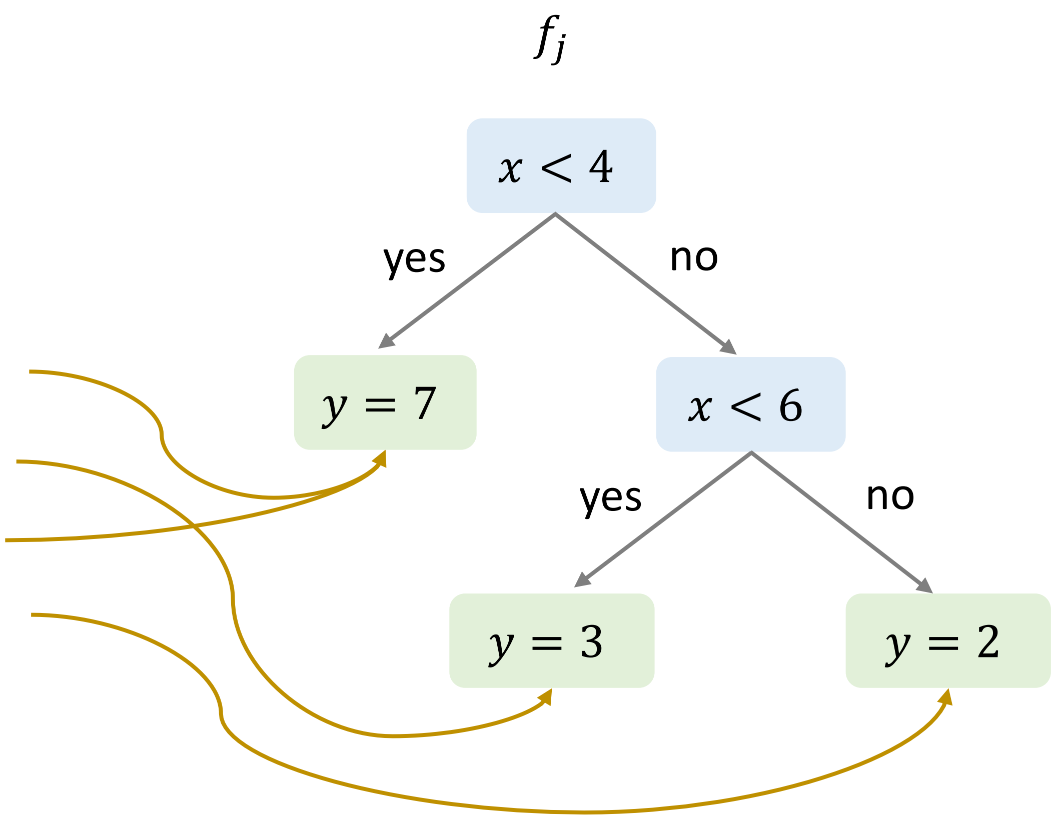
Residuals

$8 - 7 = 1$

$5 - 3 = 2$

$8 - 7 = 1$

$7 - 2 = 5$



Learning GAM

- Gradient Boosting

Example

x	y
1	8
5	5
3	8
9	7

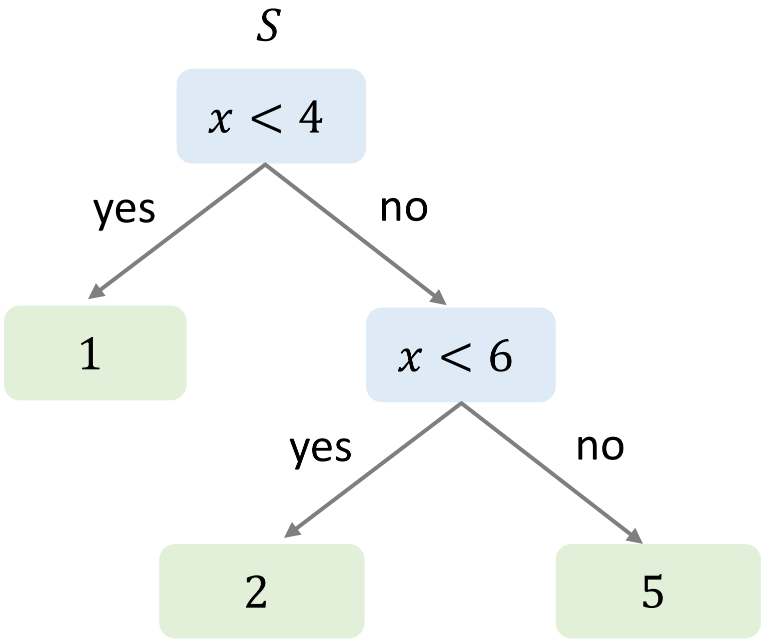
Residuals

$8 - 7 = 1$

$5 - 3 = 2$

$8 - 7 = 1$

$7 - 2 = 5$



Learning GAM

- Gradient Boosting

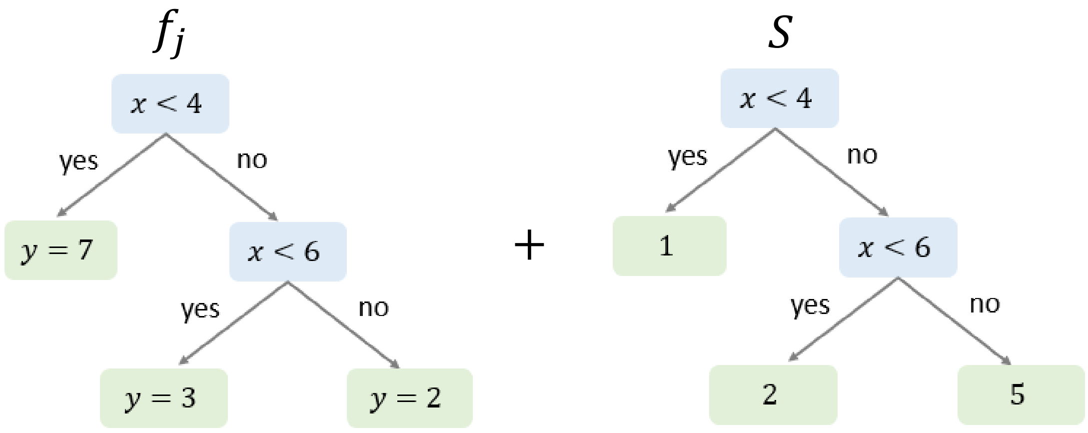
Example

x	y
1	8
5	5
3	8
9	7

Residuals

$$8-(7+1)=0$$
$$5-(3+2)=0$$
$$8-(7+1)=0$$
$$7-(2+5)=0$$

Update $f_j \leftarrow f_j + S$



Learning GAM

- Gradient Boosting

Example

x	y
1	8
5	5
3	8
9	7

Residuals

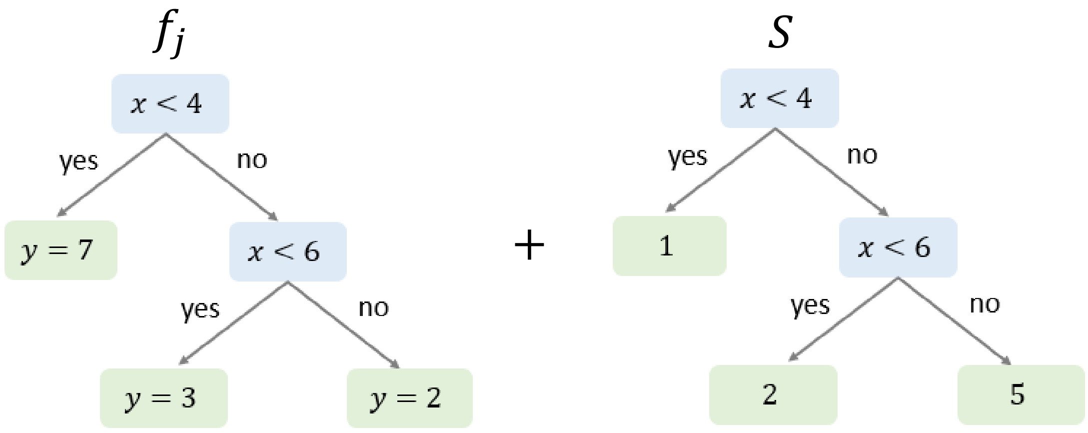
$8 - (7 + 1) = \boxed{0}$

$5 - (3 + 2) = \boxed{0}$

$8 - (7 + 1) = \boxed{0}$

$7 - (2 + 5) = \boxed{0}$

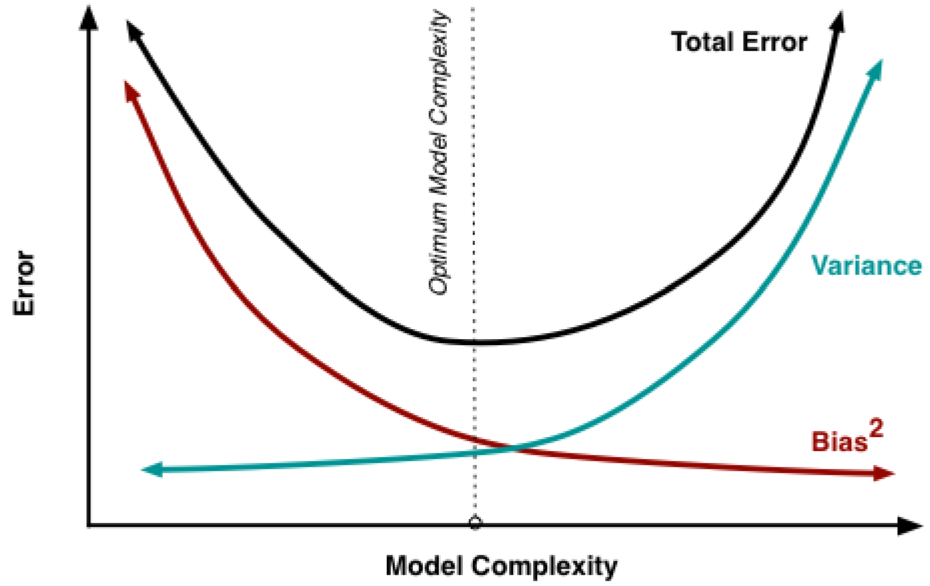
Update $f_j \leftarrow f_j + S$



Do we learn a perfect model?

Learning GAM

The model fits training data too well



We have low bias, but probably have high variance

Source: http://scott.fortmann-roe.com/docs/BiasVariance.html

Learning GAM

- Gradient Boosting

Update $f_j \leftarrow f_j + \gamma \times S$

Example

x	y
1	8
5	5
3	8
9	7

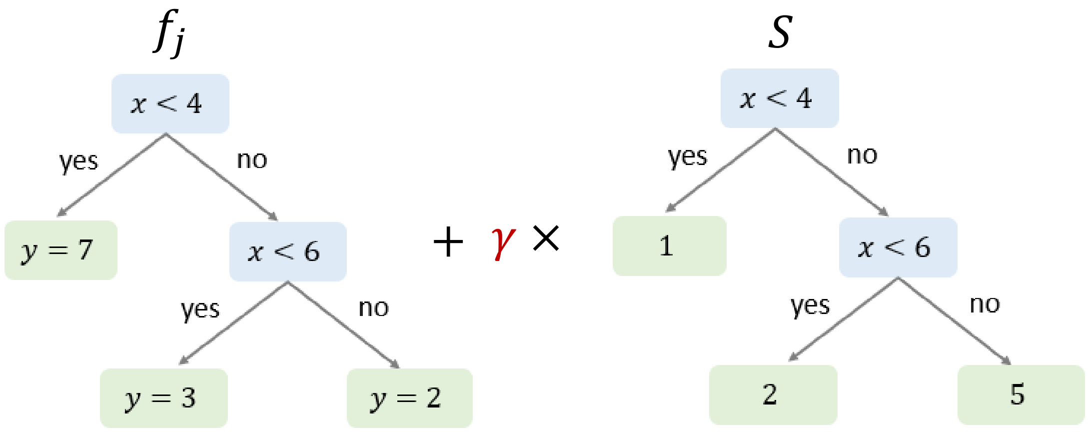
Residuals

$8 - (7 + 0.1 \times 1) = 0.9$

$5 - (3 + 0.1 \times 2) = 1.8$

$8 - (7 + 0.1 \times 1) = 0.9$

$7 - (2 + 0.1 \times 5) = 4.5$



Add a learning rate to scale the contribution of the new tree

Learning GAM

- Gradient Boosting
  - Learning tree or tree ensemble shape functions

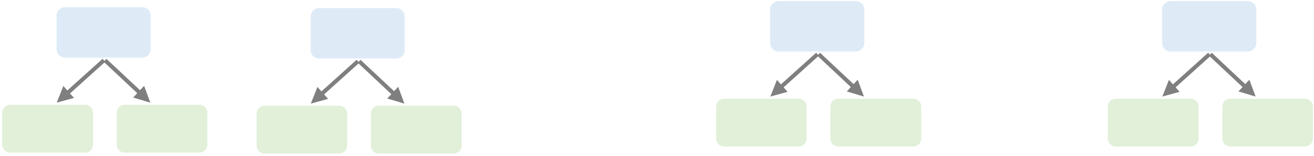
Algorithm Gradient Boosting for GAM

1. $f_j \leftarrow 0, j = 1, \cdots, n$ Initialize all shape functions as zero
2. **for** $m = 1, \cdots, M$ **do** Loop over M iterations
3. **for** $j = 1, \cdots, n$ **do** Loop over all features
4. $\mathcal{R} \leftarrow \left\{x_{ij}, y_i - \sum_k f_k\right\}_{i=1}^{N}$ Calculate residuals
5. Learning shape function $\mathrm{S}: x_j \rightarrow y$ using $\mathcal{R}$ as training data Learn the one-dimensional function to predict the residuals
6. $f_j \leftarrow f_j + S$ Update the shape function

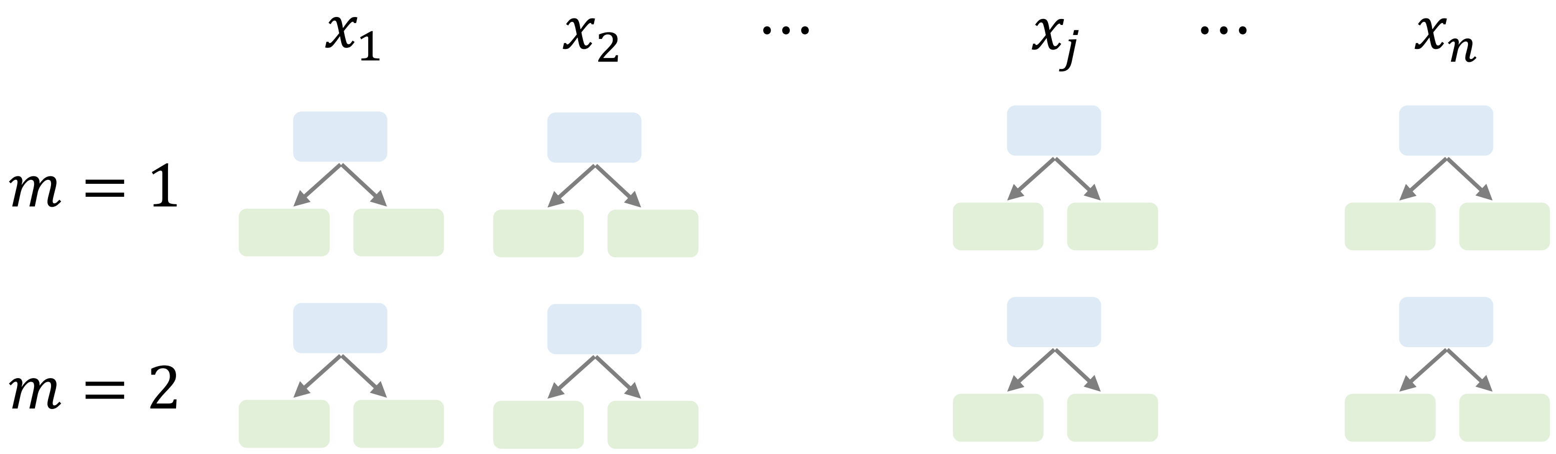
Learning GAM

$x_1 \quad x_2 \quad \cdots \quad x_j \quad \cdots \quad x_n$

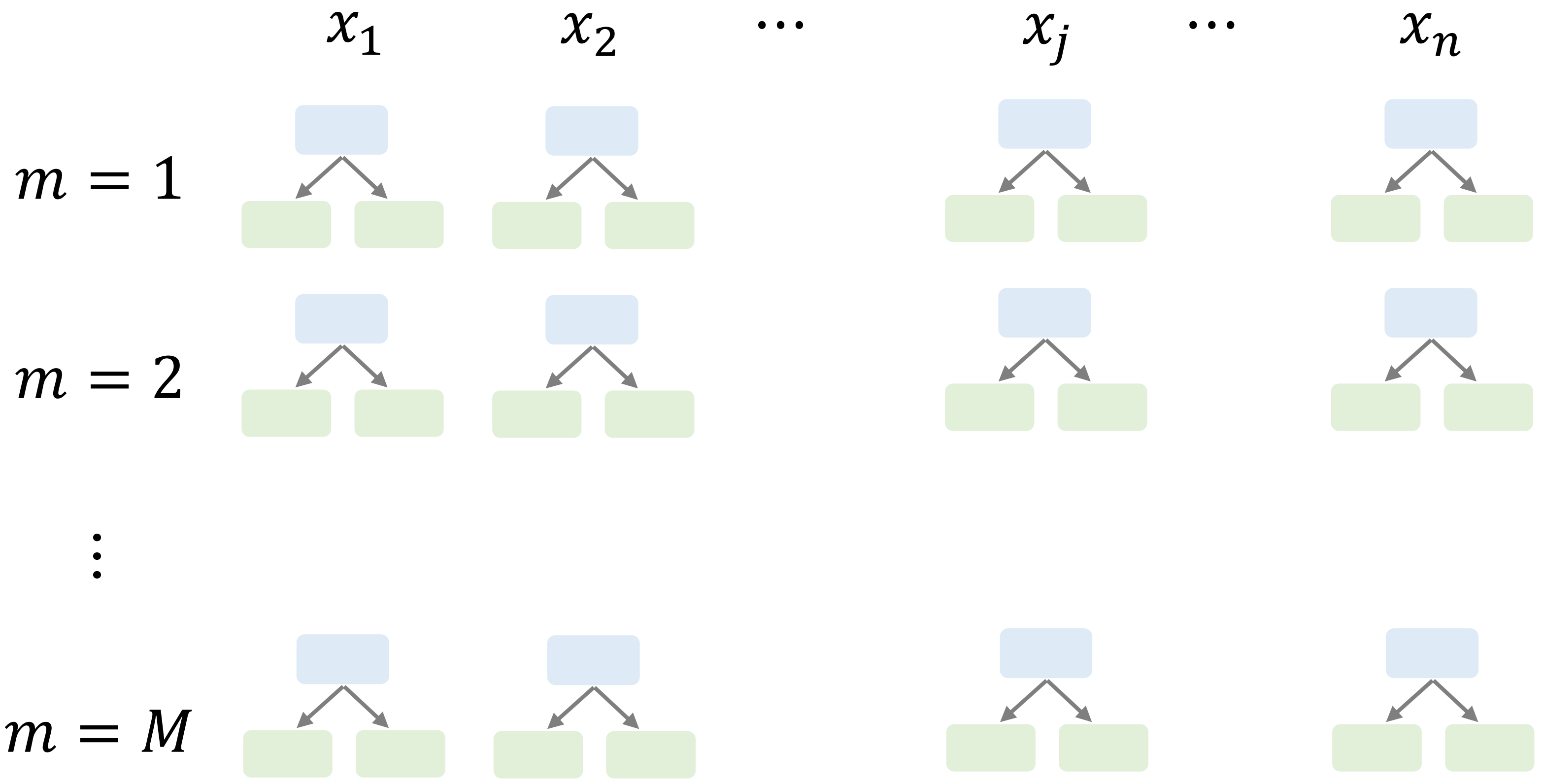
$m = 1$



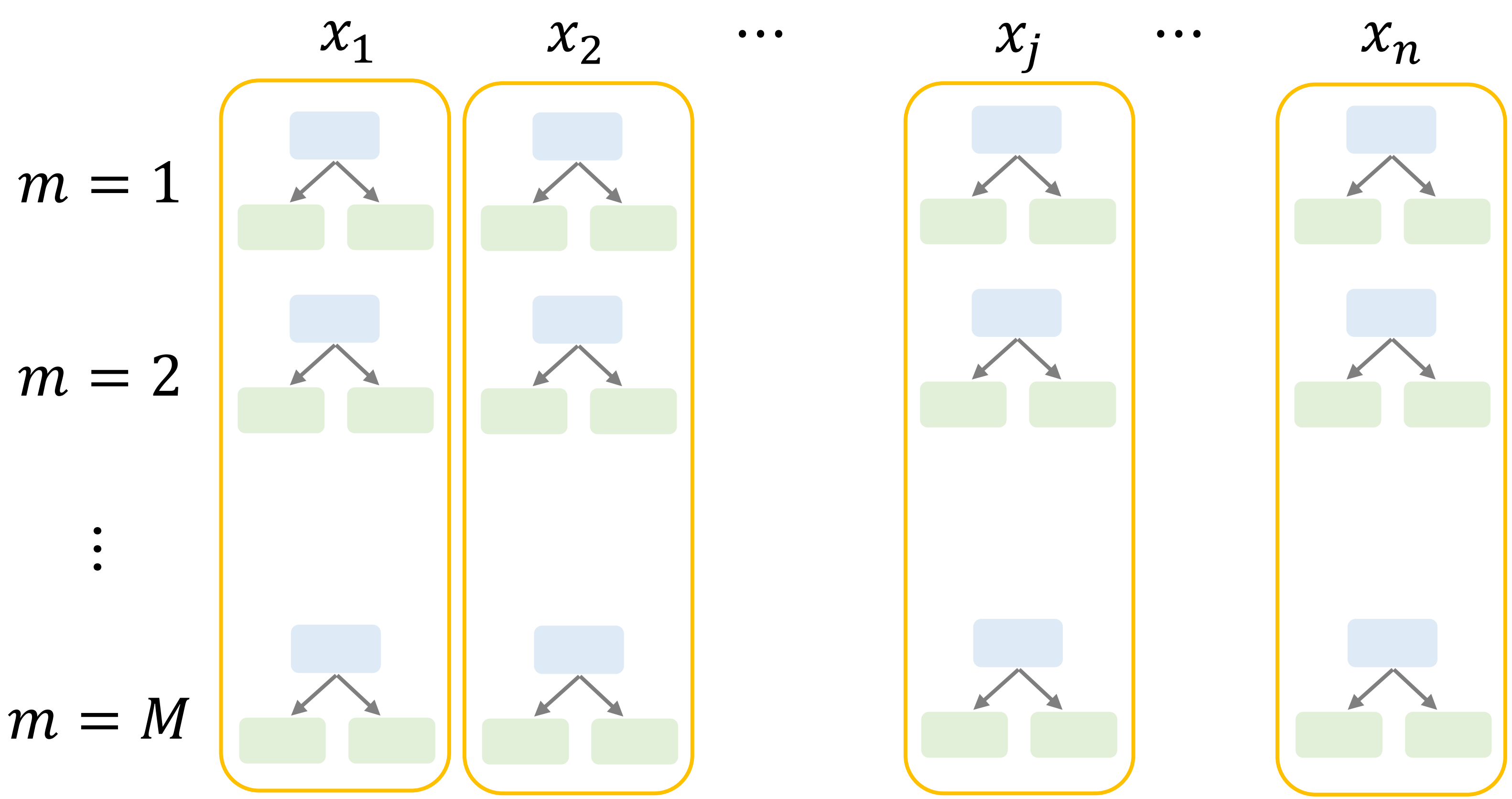
Learning GAM



Learning GAM



Learning GAM



Question?

Learning GAM

- Backfitting
  - Learning tree or tree ensemble shape functions

Algorithm Backfitting for GAM

---

1. $f_j \leftarrow 0, j = 1, \cdots, n$ Initialize all shape functions as zero
2. Learn f_1 using the training set $\{(x_{i1}, y_i)\}_{i=1}^{N}$
3. **for** $j = 2, \cdots, n$ **do** Loop over rest features
4. $\mathcal{R} \leftarrow \left\{x_{ij}, y_i - \sum_{k=1}^{j-1} f_k\right\}_{i=1}^{N}$ Calculate residuals
5. Learning shape function $S: x_j \rightarrow y$ using $\mathcal{R}$ as training data Learn the one-dimensional function to predict the residuals
6. $f_j \leftarrow S$ Update the shape function
7. Retrain f_1 based on the residuals of other $n - 1$ shape functions

Learning GAM

- Least Squares
  - Learning spline shape functions $g(y) = \beta_1 {x_1}^2 + \beta_2\sqrt{x_2} + \cdots + \beta_n \sin x_n$
  - Reducing to fitting a linear model

$$\boldsymbol{y} = \boldsymbol{X\beta}$$

$$\underline{\boldsymbol{X_i}} = [{x_{i1}}^2, \sqrt{x_{i2}}, \cdots, \sin x_{in}]$$

i^{th} example

$$\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_n]^T$$

Objective

$\min \|\boldsymbol{y} - \boldsymbol{X\beta}\|_2$

Learning GAM

- Least Squares
  - Learning spline shape functions $\quad g(y) = \beta_1 {x_1}^2 + \beta_2\sqrt{x_2} + \cdots + \beta_n \sin x_n$
  - Reducing to fitting a linear model

$$\boldsymbol{y} = \boldsymbol{X\beta}$$

$$\underline{\boldsymbol{X_i}} = [{x_{i1}}^2, \sqrt{x_{i2}}, \cdots, \sin x_{in}]$$

i^{th} example

$$\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_n]^T$$

Objective

$\min \|\boldsymbol{y} - \boldsymbol{X\beta}\|_2$

Simple, but not flexible

Summary

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- Training
  - Shape functions for individual features: splines, trees, ensembles of trees
  - Learning methods: Least Squares, Gradient Boosting, Backfitting

Summary

Generalized additive models (GAMs)

$$g(y) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

- Training
  - Shape functions for individual features: splines, trees, ensembles of trees
  - Learning methods: Least Squares, Gradient Boosting, Backfitting
- Interpretability

$$\begin{tikzcd}
x_1 \arrow[r, "\text{contributions}"] & f_1(x_1) \\
\vdots & \vdots \\
x_n \arrow[r] & f_n(x_n)
\end{tikzcd}$$

Application

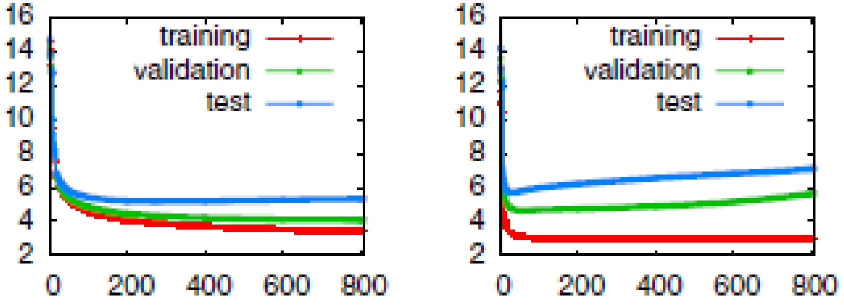
- Dataset: “Concrete” (Blast Furnace Slag, Fly Ash, Superplasticizer…)
- Task: predicting the compressive strength of concrete
- Models:

Shape Function	Least Squares	Gradient Boosting	Backfitting
Splines	P-LS/P-IRLS	BST-SP	BF-SP
Single Tree	N/A	BST-TRx	BF-TR
Bagged Trees	N/A	BST-bagTRx	BF-bagTR
Boosted Trees	N/A	BST-TRx	BF-bstTRx
Boosted Bagged Trees	N/A	BST-bagTRx	BF-bbTRx

Lou, Yin, Rich Caruana, and Johannes Gehrke. "Intelligible models for classification and regression." *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2012.

Empirical Results

- GAMs perform better than linear or logistic regression (without feature shaping)
- Tree-based shaping methods are more accurate than spline-based methods
- Bagged-trees with 2-4 leaves as shape functions in combination with gradient boosting as learning method perform better
- Controlling the complexity of trees can avoid overfitting



(a) BST-bagTR2 (2 leaves)

(b) BST-bagTR16 (16 leaves)

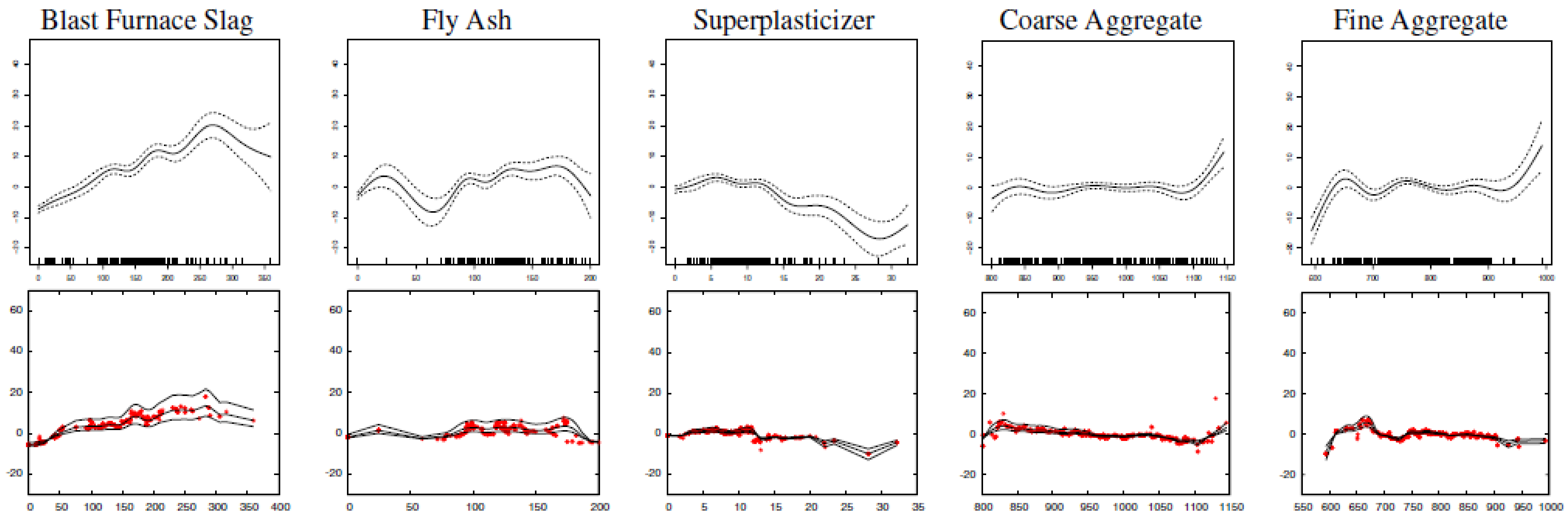
Lou, Yin, Rich Caruana, and Johannes Gehrke. "Intelligible models for classification and regression." *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2012.

Interpretation

Shapes of features for the “Concrete” dataset (versus the compressive strength of concrete)

(Splines)

(Bagged trees)



Lou, Yin, Rich Caruana, and Johannes Gehrke. "Intelligible models for classification and regression." *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2012.

Question?

GA^2M

Limitation: GAMs do not consider feature dependency

Model		**Goal**
$f_1(x_1) + f_2(x_2)$	$\xleftrightarrow{\text{Gap}}$	$f_1(x_1) + f_2(x_2) + x_1 x_2 + \cdots$

GA^2M

Limitation: GAMs do not consider feature dependency

Model		**Goal**
$f_1(x_1) + f_2(x_2)$	$\xleftrightarrow{\text{Gap}}$	$f_1(x_1) + f_2(x_2) + x_1 x_2 + \cdots$

Add two-dimensional interactions

GA^2M $$\sum f_i(x_i) + \sum f_{i,j}(x_i, x_j)$$

GA^{2}M

Definitions

- Dataset $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$
- $\boldsymbol{x}_i = [x_{i1}, \cdots, x_{in}]$ with n features
- y_i is the response
- $\boldsymbol{x} = (x_1, \cdots, x_n)$ denote the features in the dataset
- $U_1 = \{\{i\} | 1 \leq i \leq n\}$,$U_2 = \{\{i,j\} | 1 \leq i < j \leq n\}$, $U = U_1 \cup U_2$, i.e., U contains all indices for all features and pairs of features
- For any $u \in U$, let H_u denote the Hilbert space of $f_u(x_u)$
- $H = \sum_{u \in U} H_u$, $H_1 = \sum_{u \in U_1} H_u$, $H_2 = \sum_{u \in U_2} H_u$

GA^{2}M

GA^{2}M $\quad F(\boldsymbol{x}) = \sum_{u \in U} f_u(x_u)$

Objective $\quad \min_{F \in H} E[L(y, F(\boldsymbol{x}))]$

L: non-negative convex loss function

regression

classification

Squared loss

$(y - F(\boldsymbol{x}))^2$

Cross-entropy loss

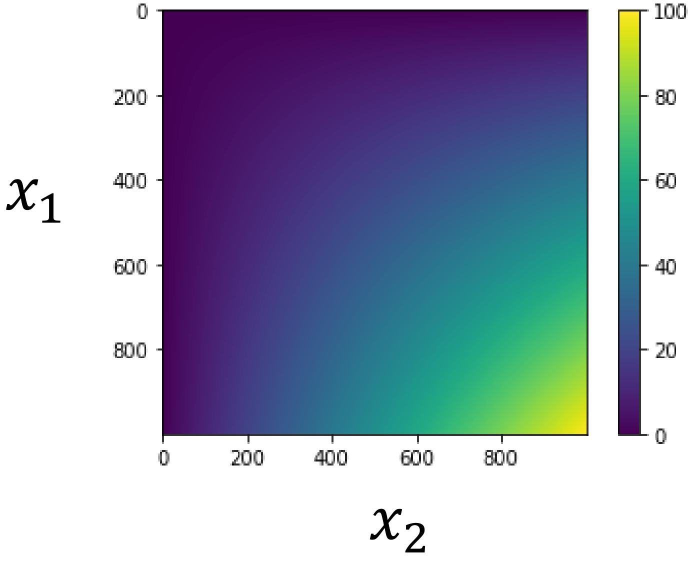
$-y \log F(\boldsymbol{x}) - (1 - y) \log(1 - F(\boldsymbol{x}))$

GA^{2}M

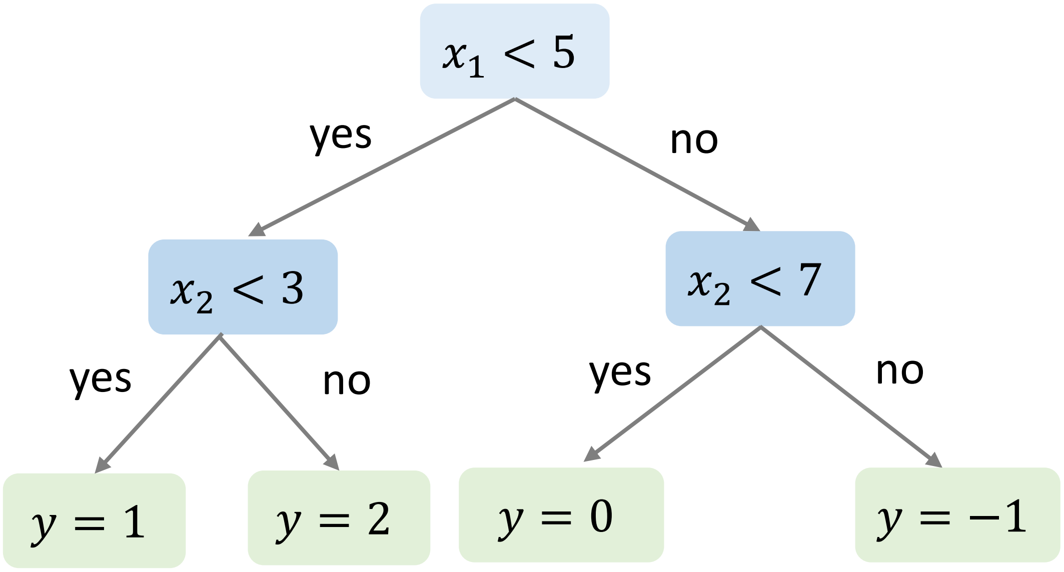
- We have known how to learn shape functions for GAMs
- Applicable to two-dimensional shape functions $f_u, u = \{i, j\}$

Splines

$f_{1,2} = x_1 x_2$



Trees



GA^{2}M

Challenge

$$\sum f_i(x_i) + \sum f_{i,j}(x_i, x_j)$$

n features $\longrightarrow$ $O(n^2)$ features interactions

How to find true feature interactions?

GA^{2}M

Algorithm GA^{2}M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg \min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$
5. $R \leftarrow y - F(\boldsymbol{x})$
6. **for** all $u \in Z$ **do**
7. $F_u \leftarrow E[R \mid x_u]$
8. $u^* \leftarrow \arg \min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$
9. $S \leftarrow S \cup \{u^*\}$
10. $Z \leftarrow Z - \{u^*\}$

GA^{2}M

Algorithm GA^{2}M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg \min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$
6. **for** all $u \in Z$ **do**
7. $F_u \leftarrow E[R \mid x_u]$
8. $u^* \leftarrow \arg\min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$
9. $S \leftarrow S \cup \{u^*\}$
10. $Z \leftarrow Z - \{u^*\}$

Learning shape functions for all single features ($f_i(x_i)$) and the selected feature pairs ($f_{i,j}(x_i, x_j)$). When $S = \emptyset$, F is the GAM.

GA^{2}M

Algorithm GA^{2}M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg \min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$ Calculate residual
6. **for** all $u \in Z$ **do**
7. $F_u \leftarrow E\left[R \mid x_u\right]$
8. $u^* \leftarrow \arg \min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$
9. $S \leftarrow S \cup \{u^*\}$
10. $Z \leftarrow Z - \{u^*\}$

GA^{2}M

Algorithm GA^{2}M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg\min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$ Calculate residual
6. **for** all $u \in Z$ **do** Loop over all remaining feature pairs
7. $F_u \leftarrow E[R \mid x_u]$ Build an interaction model on the residual
8. $u^* \leftarrow \arg\min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$
9. $S \leftarrow S \cup \{u^*\}$
10. $Z \leftarrow Z - \{u^*\}$

Learning a shape function for each feature pair

GA^2M

Algorithm GA^2M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg\min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$ Calculate residual
6. **for all** $u \in Z$ **do** Loop over all remaining feature pairs
7. $F_u \leftarrow E\left[R \mid x_u\right]$ Build an interaction model on the residual
8. $u^* \leftarrow \arg\min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$ Select the best feature pair
9. $S \leftarrow S \cup \{u^*\}$
10. $Z \leftarrow Z - \{u^*\}$

GA^2M

Algorithm GA^2M

1. $S \leftarrow \emptyset$ The set of the selected pairs
2. $Z \leftarrow U_2$ The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg \min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$ Calculate residual
6. **for all** $u \in Z$ **do** Loop over all remaining feature pairs
7. $F_u \leftarrow E\left[R \mid x_u\right]$ Build an interaction model on the residual
8. $u^* \leftarrow \arg \min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$ Select the best feature pair
9. $S \leftarrow S \cup \{u^*\}$ Put the best feature pair in S
10. $Z \leftarrow Z - \{u^*\}$ Remove that from Z

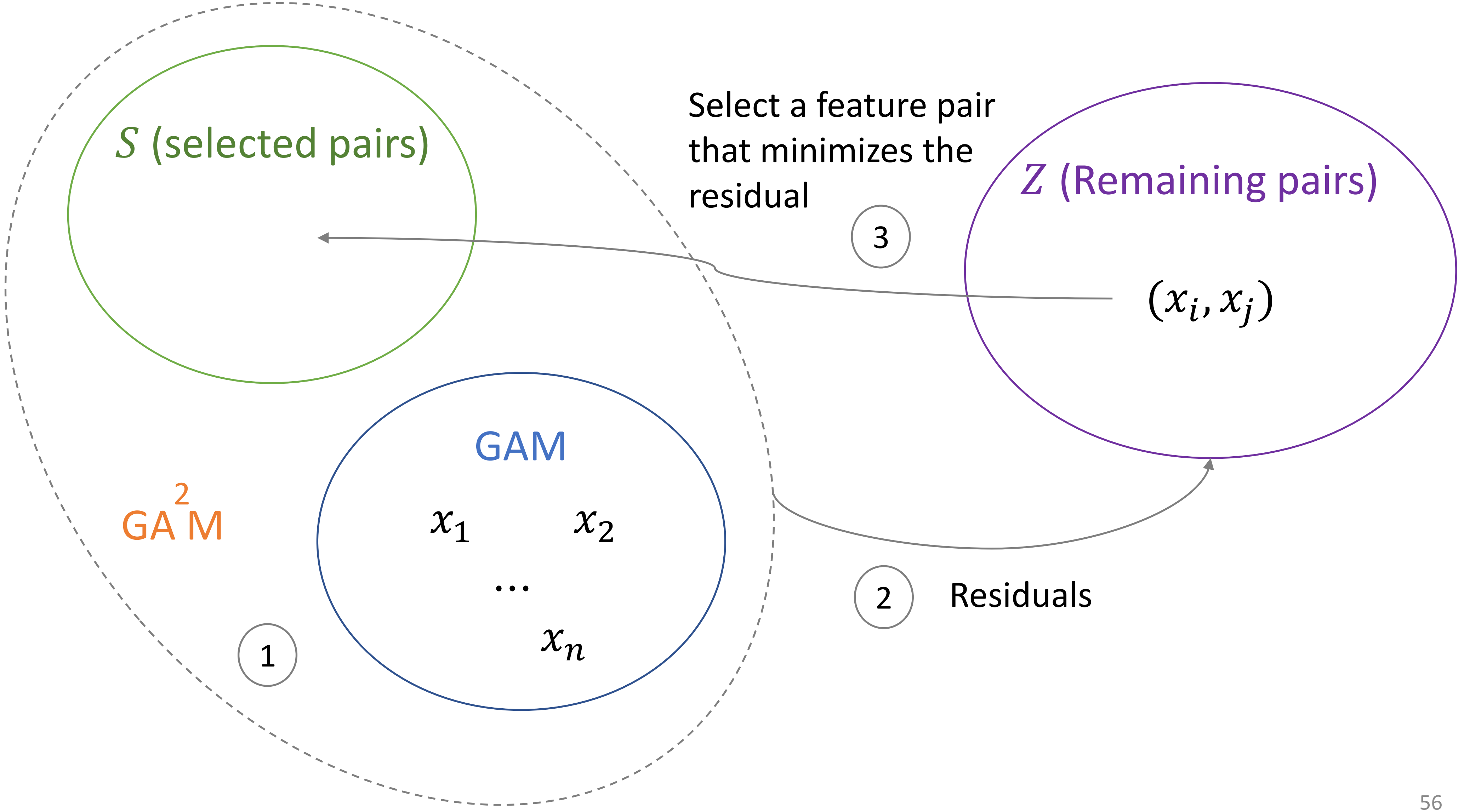
GA^2M

Algorithm GA^2M

1. $S \leftarrow \emptyset$ — The set of the selected pairs
2. $Z \leftarrow U_2$ — The set of the remaining pairs
3. **While** not converge **do**
4. $F \leftarrow \arg\min_{F \in H_1 + \sum_{u \in S} H_u} \frac{1}{2} E\left[\left(y - F(\boldsymbol{x})\right)^2\right]$ — The best additive model F so far in Hilbert space $H_1 + \sum_{u \in S} H_u$
5. $R \leftarrow y - F(\boldsymbol{x})$ — Calculate residual
6. **for all** $u \in Z$ **do** — Loop over all remaining feature pairs $\longrightarrow O(n^2)$
7. $F_u \leftarrow E\left[R \mid x_u\right]$ — Build an interaction model on the residual
8. $u^* \leftarrow \arg\min_{u \in Z} \frac{1}{2} E[(R - f_u(x_u))^2]$ — Select the best feature pair
9. $S \leftarrow S \cup \{u^*\}$ — Put the best feature pair in S
10. $Z \leftarrow Z - \{u^*\}$ — Remove that from Z

See a fast interaction detection algorithm in [Lou et al., 2013]

GA²M



S (selected pairs)
Select a feature pair that minimizes the residual
3
Z (Remaining pairs)
(x_i, x_j)
GAM
GA²M
x_1
x_2
$\cdots$
x_n
1
2
Residuals

Question?

Application

“Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission”

Caruana et al., KDD 2015

Background

- In the mid 90's, a project was funded by Cost-Effective HealthCare (CEHC) to evaluate the application of machine learning to important problems in healthcare such as predicting pneumonia risk
- **Goal**: predict the probability of death (POD) for patients with pneumonia
- High-risk: patients could be admitted to the hospital
- Low-risk: patients were treated as outpatients

Background

- In the mid 90's, a project was funded by Cost-Effective HealthCare (CEHC) to evaluate the application of machine learning to important problems in healthcare such as predicting pneumonia risk
- **Goal**: predict the probability of death (POD) for patients with pneumonia
- High-risk: patients could be admitted to the hospital
- Low-risk: patients were treated as outpatients

Models

Logistic regression

Rule-based learning

k-nearest neighbor

Neural networks

…

Background

- In the mid 90’s, a project was funded by Cost-Effective HealthCare (CEHC) to evaluate the application of machine learning to important problems in healthcare such as predicting pneumonia risk
- **Goal**: predict the probability of death (POD) for patients with pneumonia
- High-risk: patients could be admitted to the hospital
- Low-risk: patients were treated as outpatients

Models

Logistic regression

Rule-based learning

k-nearest neighbor

Neural networks   AUC=0.86 (Best performance)

…

Background

- In the mid 90's, a project was funded by Cost-Effective HealthCare (CEHC) to evaluate the application of machine learning to important problems in healthcare such as predicting pneumonia risk
- **Goal**: predict the probability of death (POD) for patients with pneumonia
- High-risk: patients could be admitted to the hospital
- Low-risk: patients were treated as outpatients

Models

Logistic regression AUC=0.77 (safer to use on patients)

Rule-based learning

k-nearest neighbor

~~Neural networks~~ ~~AUC=0.86 (Best performance)~~

...

Problem

The rule-based model learned a rule:

$$\text{HasAsthama}(x) \implies \text{LowerRisk}(x)$$

Rule-based models are interpretable

```
if (chance_of_rain > 0.75)
{ umbrella <- "yes" }
else { umbrella <- "no" }
```

Problem

The rule-based model learned a rule:

$$\text{HasAsthama}(x) \implies \text{LowerRisk}(x)$$

counterintuitive

Problem

The rule-based model learned a rule:

$$\text{HasAsthama}(x) \implies \text{LowerRisk}(x)$$

counterintuitive

The model captures a true pattern in the training data:

Patients:
asthma +
pneumonia

Hospital (ICU)

The aggressive care lowered the risk of dying from pneumonia

Problem

The rule-based model learned a rule:

$\cancel{\text{HasAsthama}(x) \implies \text{LowerRisk}(x)}$

Asthmatics have much higher risk!

It would be dangerous if the model predicts low risk on patients who have not been hospitalized

Problem

How about other potential patterns?

$$\text{Pregnancy} \implies \text{Lower Risk ?}$$

Problem

How about other potential patterns?

$$\text{Pregnancy} \implies \text{Lower Risk ?}$$

MUST understand ML models in healthcare.
Otherwise, models may hurt patients
because of true patterns in data!

Generalized Additive Models

- Better prediction performance than logistic regression (capture more data patterns)
- Interpretable

Case Study: Pneumonia Risk

- There are 46 features describing each patient
- Bagged trees with gradient boosting

Patient-history findings			
chronic lung disease	-	age	C
re-admission to hospital	-	gender	-
admitted through ER	-	diabetes mellitus	-
admitted from nursing home	-	asthma	-
congestive heart failure	-	cancer	-
ischemic heart disease	-	number of diseases	C
cerebrovascular disease	-	history of seizures	-
chronic liver disease	-	renal failure	-
history of chest pain	-		
Physical examination findings			
diastolic blood pressure	C	wheezing	-
gastrointestinal bleeding	-	stridor	-
respiration rate	C	heart murmur	-
altered mental status	-	temperature	C
heart rate	C		
Laboratory findings			
liver function tests	-	BUN level	C
glucose level	C	creatinine level	C
potassium level	C	albumin level	C
hematocrit	C	WBC count	C
percentage bands	C	pH	C
pO2	C	pCO2	C
sodium level	C		
Chest X-ray findings			
positive chest x-ray	-	lung infiltrate	-
pleural effusion	-	pneumothorax	-
cavitation/empyema	-	chest mass	-
lobe or lung collapse	-		

Prediction Performance

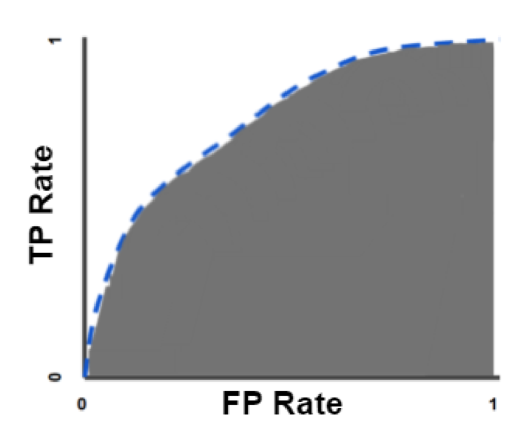
AUC for different learning methods

Model	Pneumonia
Logistic Regression	0.8432
GAM	0.8542
GA^2M	0.8576
Random Forests	0.8460
LogitBoost	0.8493

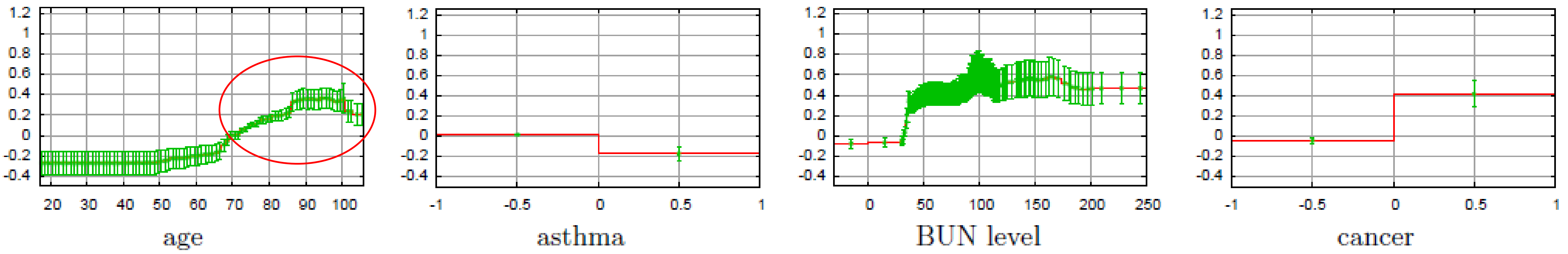
AUC: Area Under the ROC Curve

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$



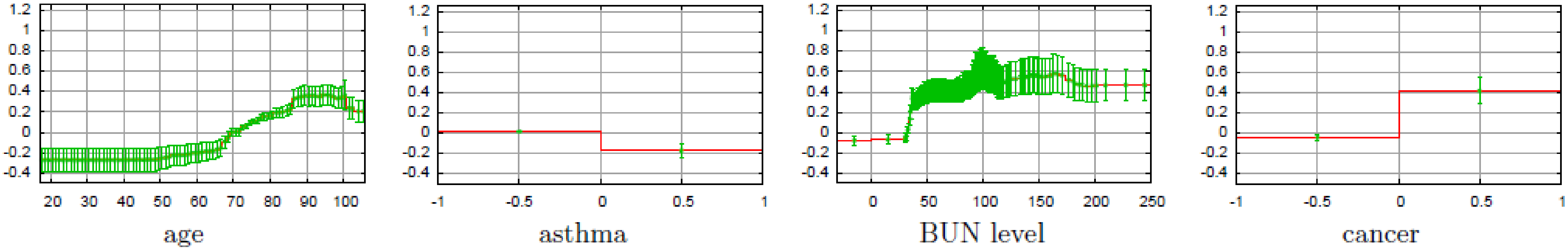
Interpretation



(Blood Urea Nitrogen)

Older people have higher risk

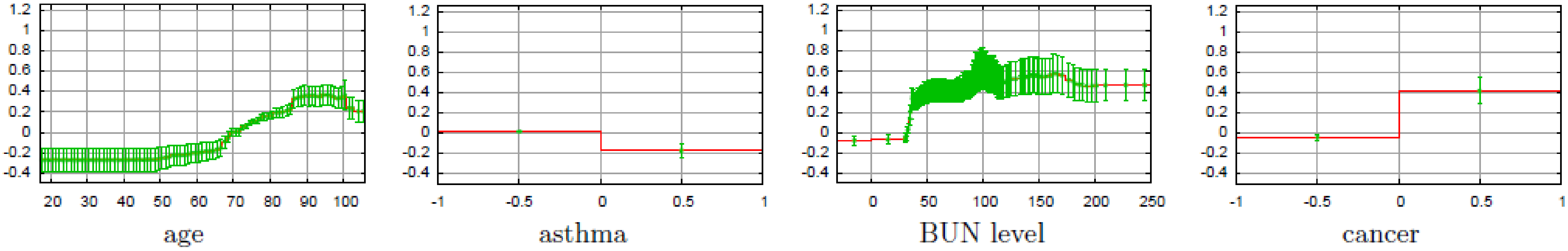
Interpretation



(Blood Urea Nitrogen)

GAMs also found the pattern: asthma lowers the risk

Interpretation



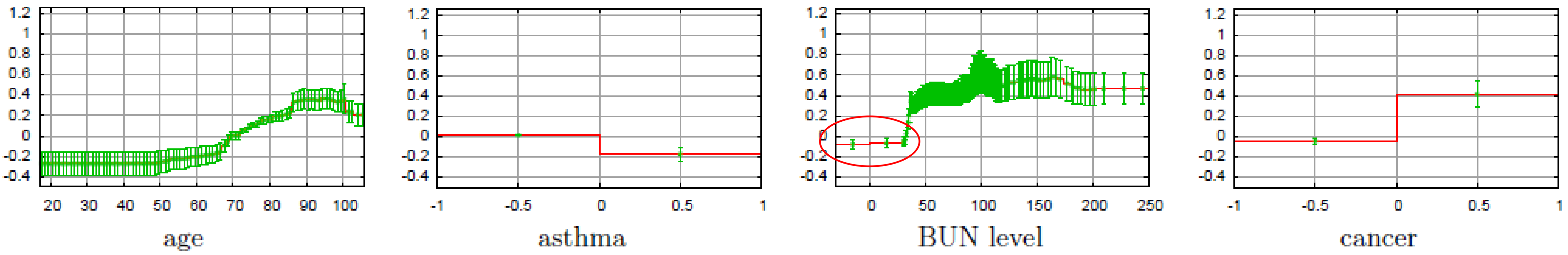
(Blood Urea Nitrogen)

GAMs also found the pattern: asthma lowers the risk

↓

Repair: eliminate this term

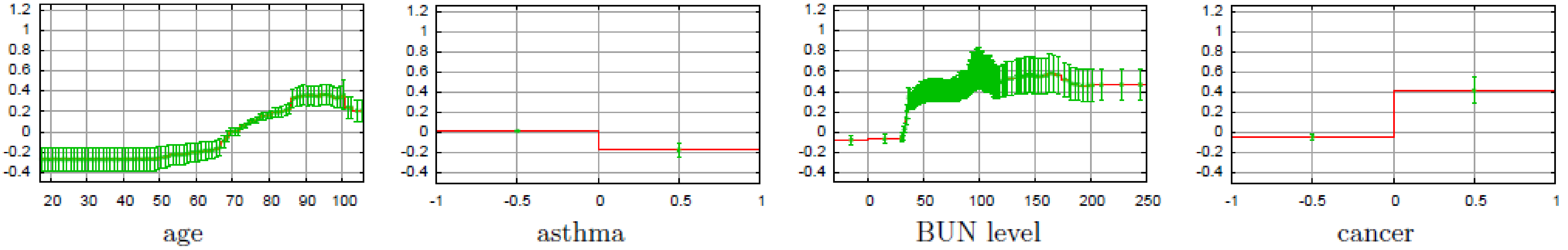
Interpretation



(Blood Urea Nitrogen)

Most patients have BUN=0 because, as in many medical datasets, if the variable is not measured or assumed normal it is coded as 0

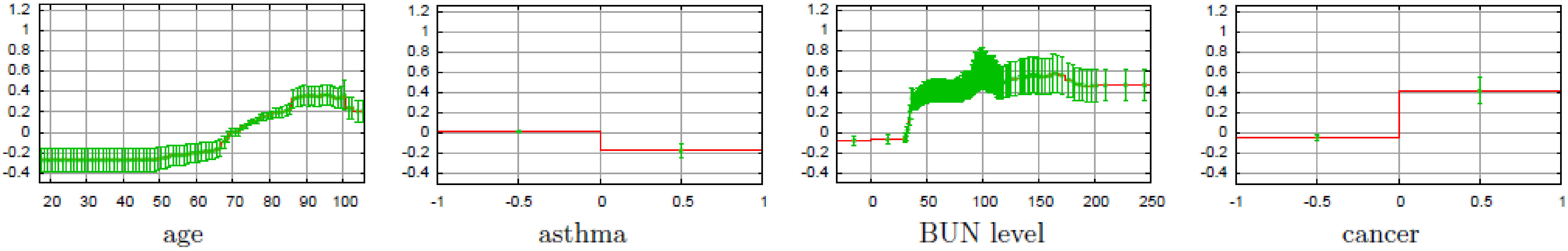
Interpretation



(Blood Urea Nitrogen)

BUN levels below 30 appear to be low risk, while levels from 50-200 indicate higher risk

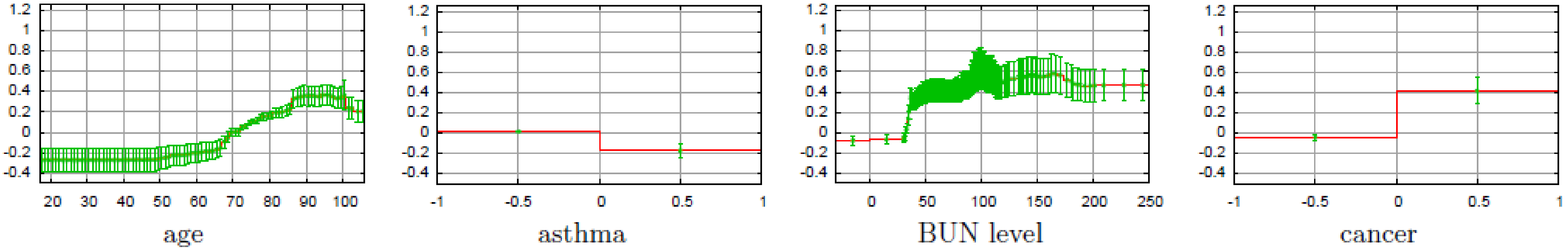
Interpretation



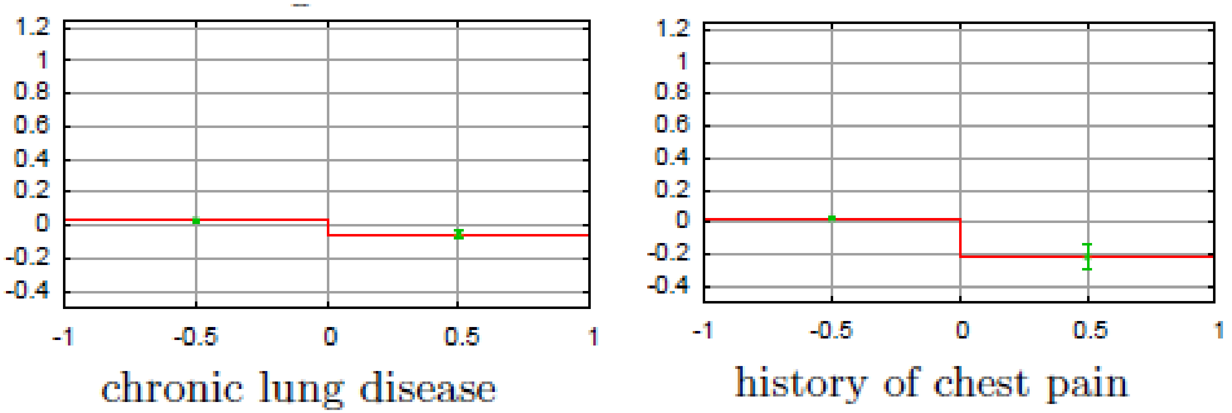
(Blood Urea Nitrogen)

Having cancer significantly increases the risk of dying from pneumonia

Interpretation

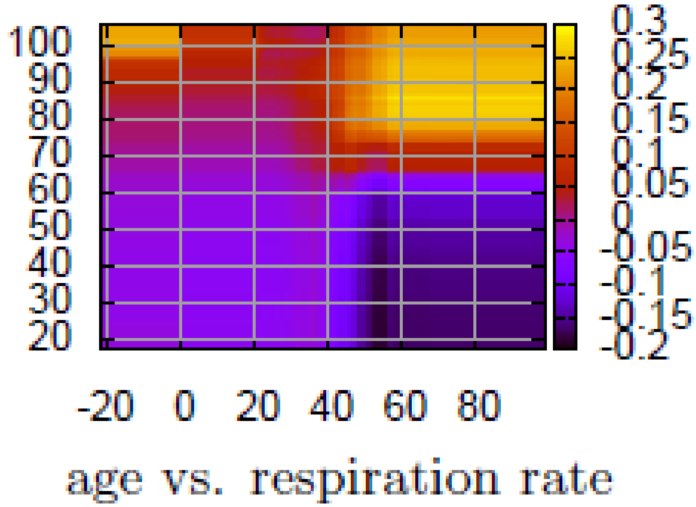


(Blood Urea Nitrogen)

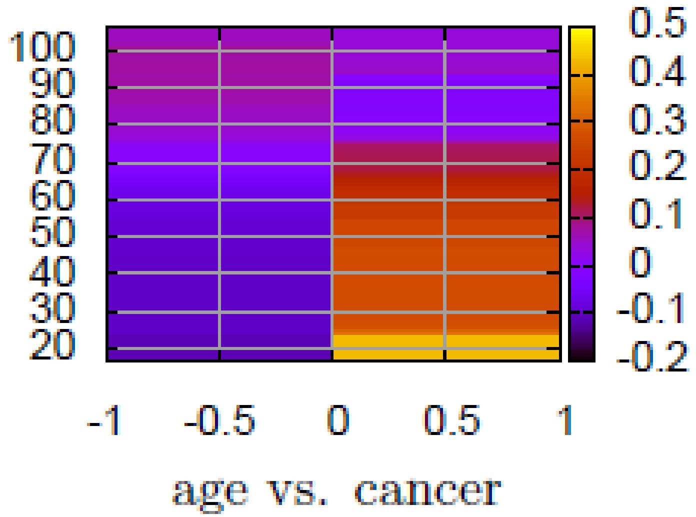


Chronic lung disease and a history of chest pain both lower risk (similar problem as asthma)

Interpretation



Old people with high respiration rate have the highest risk



- Risk is highest for the youngest patients
- It declines for patients who acquire cancer later in life
- For patients without cancer, risk rises as expected with age

Takeaway

- If a model contains a modest number of terms (e.g., less than 50), it is best to show terms in the model to experts in the order they are most familiar with

- When the number of terms grows large, it is best to provide a well-defined ordering of the terms for a patient (from terms that increase risk most to terms that decrease risk most)

Question?

Reference

- Lou, Yin, Rich Caruana, and Johannes Gehrke. "Intelligible models for classification and regression." *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2012.
- Lou, Yin, et al. "Accurate intelligible models with pairwise interactions." *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2013.
- Caruana, Rich, et al. "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission." *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015.