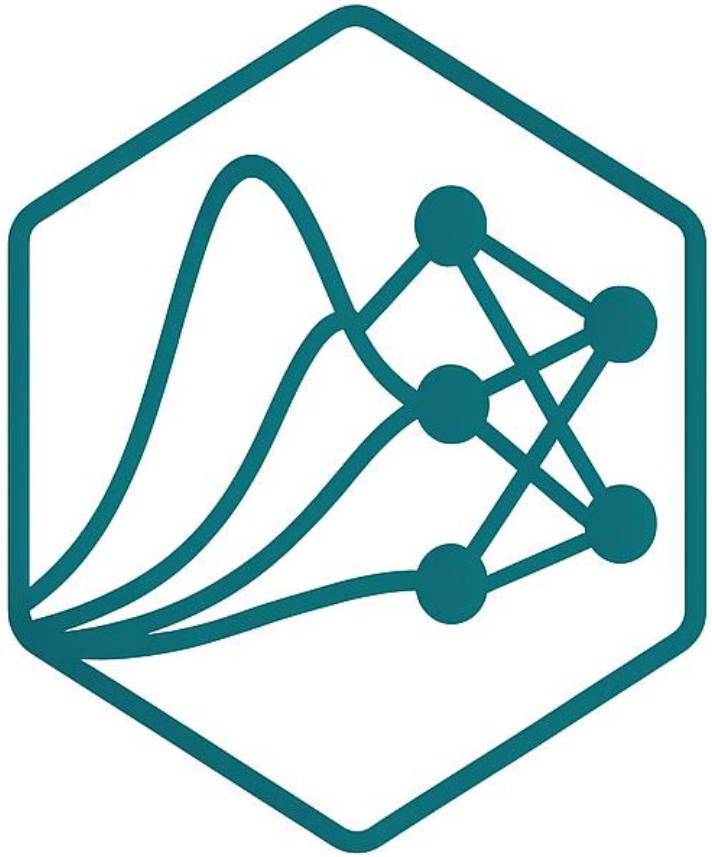


IPML

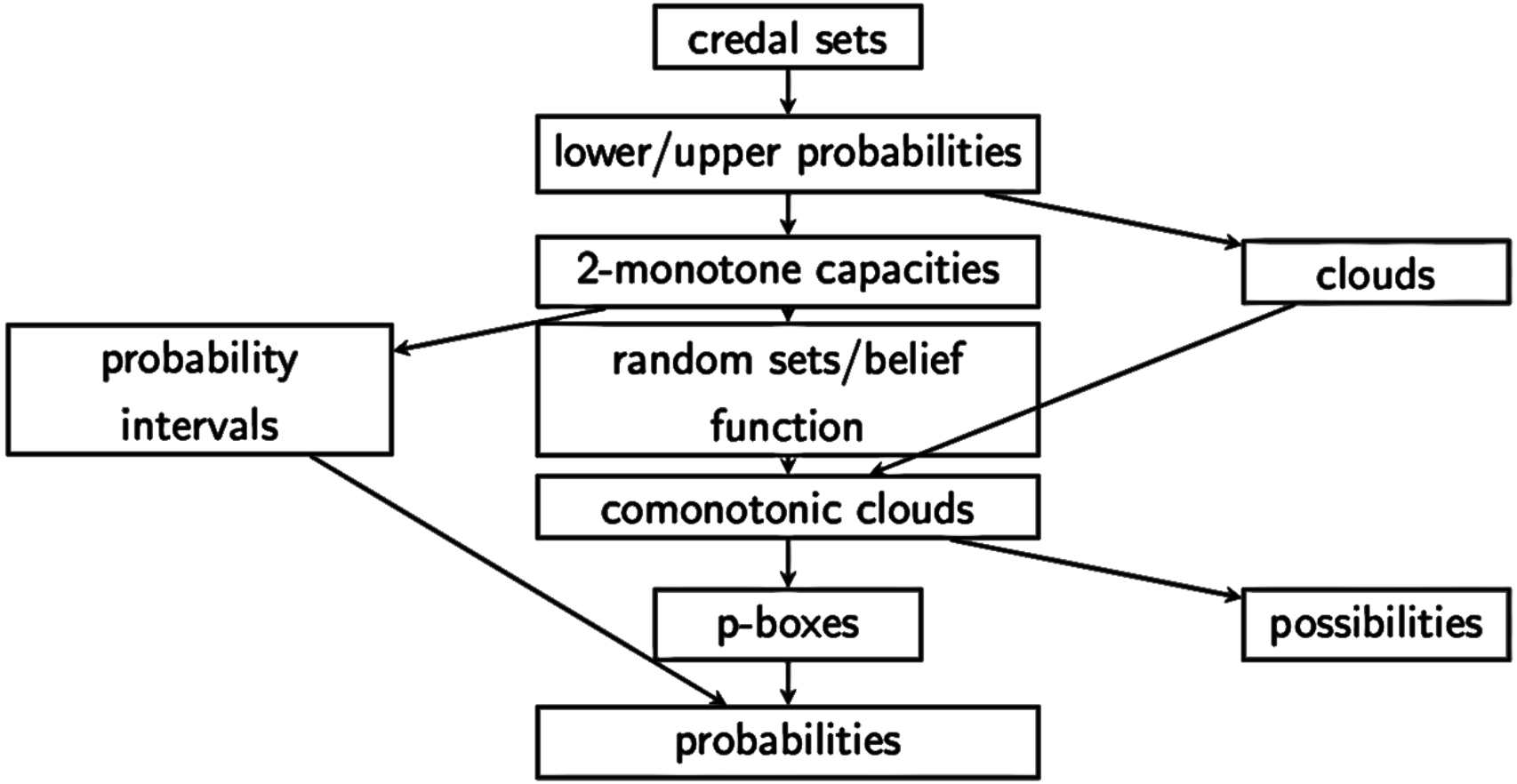
IMPRECISE
PROBABILISTIC
MACHINE LEARNING

Lecture 7: Imprecise Classification and Regression

Krikamol Muandet

12 December 2025

Overview


credal sets
lower/upper probabilities
2-monotone capacities
clouds
probability intervals
random sets/belief function
comonotonic clouds
p-boxes
possibilities
probabilities

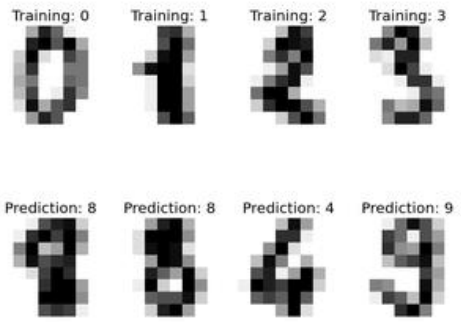
Outline

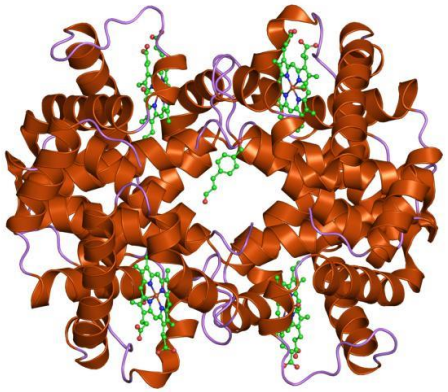
1. Naïve Credal Classifier
2. Computation of NCC
3. Real-world Example
4. Regression

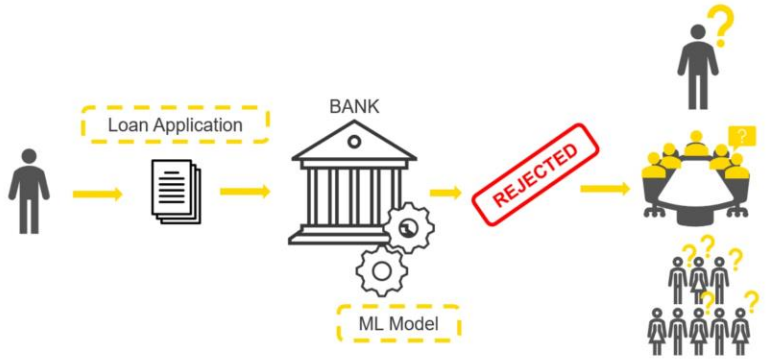
Naïve Credal Classifier

Classification

- Many real-world problems can be casted as a classification problem.







Problem Setup

- Y is the **target** variable taking values in $\mathcal{Y} = \{y_1, \ldots, y_m\}, m \geq 2$.
- $X_1, X_2 \ldots, X_d$ are d **features** taking values $x_1, x_2, \ldots, x_d$ in $\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_d$.
- The discrete joint probability distribution

$$P(Y, X_1, X_2 \ldots, X_d)$$

- The classification of a new pattern $(x_1, x_2, \ldots, x_d)$ is realised by selecting a class $y \in \mathcal{Y}$ that maximises

$$P(y \mid x_1, x_2, \ldots, x_d)$$

- This classification rule minimises the expected cost of misclassification.

Independence Assumption

- The number of probabilities grows **exponentially** with the number of attributes, i.e., k^d where k is the number of values X can take.
- Duda and Hart (1973) proposed an **independence** assumption:

$$P(X_1, X_2 \ldots, X_d \mid Y) = \prod_{i=1}^{d} P(X_i \mid Y), \qquad (k^d \rightarrow kd)$$

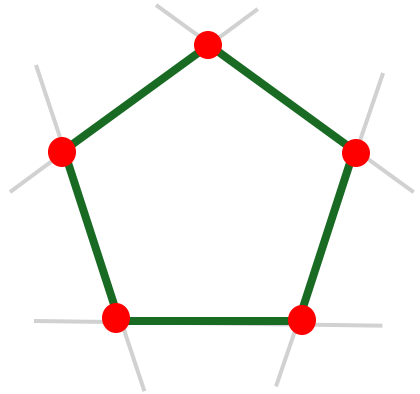
- This results in the **naïve Bayes classifier (NBC)**.
- While unrealistic, several work in the literature often claim that this assumption is not critical for classificatioin.

Naïve Credal Classifier

- A **naïve credal classifier (NCC)** is characterised by a credal set:

$$\mathcal{P} = \left\{ P(Y) \prod_{i=1}^{d} P(X_i|Y) \;\middle|\; P(Y) \in \mathcal{P}_Y, P(X_i \mid y) \in \mathcal{P}_{X_i}^{y} \right\}$$

- Here, $\mathcal{P}_Y$ is a **local** credal set of the probability distributions $P(Y)$ and $\mathcal{P}_{X_i}^{y}$ is a **local** credal set of the conditional distributions $P(X_i \mid y)$.
- How to specify the local credal sets $\mathcal{P}_Y$ and $\mathcal{P}_{X_i}^{y}$?
  1. A finitely generated credal set (FGCS)
  2. Linear constraints on the unknown probabilities



Naïve Credal Classifier

- A **naïve credal classifier (NCC)** is characterised by a credal set:

$$\mathcal{P} = \left\{ P(Y) \prod_{i=1}^{d} P(X_i|Y) \;\middle|\; P(Y) \in \mathcal{P}_y, P(X_i \mid y) \in \mathcal{P}_{X_i}^{y} \right\}$$

- From the credal sets, we can compute the lower and upper probabilities:

$$\underline{P}(y) = \min_{P \in \mathcal{P}_y} P(y), \qquad \overline{P}(y) = \max_{P \in \mathcal{P}_y} P(y)$$
$$\underline{P}(x_i \mid y) = \min_{P \in \mathcal{P}_{X_i}^{y}} P(x_i \mid y), \quad \overline{P}(x_i \mid y) = \max_{P \in \mathcal{P}_{X_i}^{y}} P(x_i \mid y)$$

- NCC assumes that the local credal sets can be specified **separately**.

Credal Classification Rules

- NBC classifies a pattern $(x_1, x_2, \ldots, x_d)$ by selecting the class $y \in \mathcal{Y}$ of **maximum** posterior probability $P(y \mid x_1, x_2, \ldots, x_d)$.
- To compare intervals, consider **strong dominance** (Luce and Raiffa, 1957):

$$[a, b] \text{ dominates } [c, d] \Leftrightarrow \ a > d$$

- For each class $y \in \mathcal{Y}$, we only need to compute:

$$\underline{P}(y \mid x_1, x_2, \ldots, x_d) = \min_{P(y, x_1, x_2, \ldots, x_d) \in \mathcal{P}} P(y \mid x_1, x_2, \ldots, x_d)$$

$$\overline{P}(y \mid x_1, x_2, \ldots, x_d) = \max_{P(y, x_1, x_2, \ldots, x_d) \in \mathcal{P}} P(y \mid x_1, x_2, \ldots, x_d)$$

Computation

Computation

- Let's focus on $\underline{P}(y \mid x_1, x_2, \ldots, x_d) = \min_{P(y,x_1,x_2,\ldots,x_d)\in\mathcal{P}} P(y \mid x_1, x_2, \ldots, x_d)$

$$P(y \mid x_1, x_2, \ldots, x_d) = \frac{P(y,x_1,x_2,\ldots,x_d)}{\sum_{\acute{y}} P(\acute{y},x_1,x_2,\ldots,x_d)} = \left(1 + \frac{\sum_{\acute{y}\neq y} P(\acute{y},x_1,x_2,\ldots,x_d)}{P(y,x_1,x_2,\ldots,x_d)}\right)^{-1}$$

$$= \left(1 + \frac{\sum_{\acute{y}\neq y} P(\acute{y}) \prod_{i=1}^{d} P(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} P(x_i \mid y)}\right)^{-1}$$

- The optimisation becomes

$$\min_{P(Y)\in\mathcal{P}_Y} \min_{P(X_i \mid y)\in\mathcal{P}_{X_i}^{y}} \left(1 + \frac{\sum_{\acute{y}\neq y} P(\acute{y}) \prod_{i=1}^{d} P(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} P(x_i \mid y)}\right)^{-1}$$

Computation

- The optimisation becomes

$$\min_{P(Y)\in\mathcal{P}_Y}\min_{P(X_i\mid y)\in\mathcal{P}_{X_i}^{y}}\left(1+\frac{\sum_{\acute{y}\neq y}P(\acute{y})\prod_{i=1}^{d}P(x_i\mid\acute{y})}{P(y)\prod_{i=1}^{d}P(x_i\mid y)}\right)^{-1}$$

$$=\min_{P(Y)\in\mathcal{P}_Y}\left(1+\frac{\sum_{\acute{y}\neq y}P(\acute{y})\prod_{i=1}^{d}\overline{P}(x_i\mid\acute{y})}{P(y)\prod_{i=1}^{d}\underline{P}(x_i\mid y)}\right)^{-1}$$

- Similar formular for the upper probability:

$$\overline{P}(y\mid x_1,x_2,\dots,x_d)=\max_{P(Y)\in\mathcal{P}_Y}\left(1+\frac{\sum_{\acute{y}\neq y}P(\acute{y})\prod_{i=1}^{d}\underline{P}(x_i\mid\acute{y})}{P(y)\prod_{i=1}^{d}\overline{P}(x_i\mid y)}\right)^{-1}$$

Interpretation

- The lower and upper posterior probabilities:

$$\underline{P}(y \mid x_1, x_2, \ldots, x_d) = \min_{P(Y) \in \mathcal{P}_Y} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \overline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \underline{P}(x_i \mid y)}\right)^{-1}$$

$$\overline{P}(y \mid x_1, x_2, \ldots, x_d) = \max_{P(Y) \in \mathcal{P}_Y} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \underline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \overline{P}(x_i \mid y)}\right)^{-1}$$

- We can think of $P(y)$ as **prior probabilities** and $\prod_{i=1}^{d} \overline{P}(x_i \mid y)$ and $\prod_{i=1}^{d} \underline{P}(x_i \mid y)$ as upper and lower **likelihood functions**.

Combinatorial Procedure

- Based on the **extreme points** of $\mathcal{P}_y$, we can rewrite the problems as

$$\underline{P}(y \mid x_1, x_2, \ldots, x_d) = \min_{P(Y) \in \text{ext}[\mathcal{P}_y]} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \overline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \underline{P}(x_i \mid y)}\right)^{-1}$$

$$\overline{P}(y \mid x_1, x_2, \ldots, x_d) = \max_{P(Y) \in \text{ext}[\mathcal{P}_y]} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \underline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \overline{P}(x_i \mid y)}\right)^{-1}$$

- To compute the lower and upper probabilities of $P(x_i \mid y)$ for $y \in \mathcal{Y}$:

$$\underline{P}(x_i \mid y) = \min_{P(x_i \mid y) \in \text{ext}[\mathcal{P}_{X_i}^{y}]} P(x_i \mid y)$$

$$\overline{P}(x_i \mid y) = \max_{P(x_i \mid y) \in \text{ext}[\mathcal{P}_{X_i}^{y}]} P(x_i \mid y)$$

Computational Complexity

- Let K be the **maximum of the number of extreme distributions**, taken over all the local credal sets.
- Computing the extremes of $P(x_i \mid y)$ takes $O(K)$ time in the worst case, which must be repeated for all classes and features, yielding $O(dK|\mathcal{Y}|)$.
- When the values of the conditional upper and lower probabilities are known, the final steps require $O(d|\mathcal{Y}|)$ time. Repearing this for each extreme distribution in $\mathrm{ext}[\mathcal{P}_y]$ yields $O(dK|\mathcal{Y}|)$.
- The overall worst-case complexity is

$$O(dK|\mathcal{Y}|).$$

Linear Programming

- The number of extreme points K can grow **exponentially**.
- Alternatively, consider a set of probability intervals

$$I_X = \{[l_i, u_i] | 0 \leq l_i \leq u_i \leq 1,\ i = 1, \ldots, T\}$$

- Let L be the worst-case complexity to solve a linear program. Then, the overall worst-case complexity is

$$O(dL|\mathcal{Y}|).$$

- L only grows **polynomially** even when K grows exponentially.

Real-World Example

Risk Assessment

- An insurance company wants to assess the risk (R) about the car insurance for a *new* customer:

$$R \in \{\text{low}, \text{medium}, \text{high}\}$$

- The company infers the risk on the basis of two attributes: the age (A) and the city (C) where the customer lives:
  - $A \in \{\text{Young}, \text{Middle}-\text{aged}, \text{Old}\}$
  - $C \in \{\text{Tübingen}, \text{Saarbrücken}, \text{Berlin}\}$

Risk Assessment

Recall that the lower and upper probabilities can be defined as

$$\underline{P}(S) := \max\left\{\sum_{x\in S}\underline{p}_x, 1-\sum_{x\in S^c}\overline{p}_x\right\}, \qquad \overline{P}(S) := \min\left\{\sum_{x\in S}\overline{p}_x, 1-\sum_{x\in S^c}\underline{p}_x\right\}$$

$P(R)$

R	Intervals
Low	[0.77, 0.85]
Medium	[0.10, 0.15]
High	[0.05, 0.08]

$P(A \mid R)$

A	R		
	Low	Medium	High
Young	[0.15,0.22]	[0.27,0.32]	[0.60,0.70]
Middle-aged	[0.50,0.55]	[0.33,0.38]	[0.05,0.15]
Old	[0.28,0.34]	[0.34,0.38]	[0.20,0.30]

$P(C \mid R)$

C	R		
	Low	Medium	High
Tü	[0.70,0.72]	[0.15,0.20]	[0.02,0.06]
Saar	[0.18,0.20]	[0.60,0.65]	[0.22,0.28]
Berlin	[0.08,0.10]	[0.20,0.25]	[0.66,0.72]

Risk Assessment

$P(R \mid A = \text{old}, C = \text{Tü})$

R	Intervals
Low	[0.922, 0.975]
Medium	[0.024, 0.070]
High	[0.001, 0.009]

$P(R \mid A = \text{young}, C = \text{Ber})$

R	Intervals
Low	[0.150, 0.426]
Medium	[0.085, 0.290]
High	[0.435, 0.693]

$P(R \mid A = \text{young}, C = \text{Saar})$

R	Intervals
Low	[0.307, 0.621]
Medium	[0.238, 0.525]
High	[0.100, 0.267]

$$\underline{P}(y \mid x_1, x_2, \ldots, x_d) = \min_{P(Y) \in \text{ext}[\mathcal{P}_y]} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \overline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \underline{P}(x_i \mid y)}\right)^{-1}$$

$$\overline{P}(y \mid x_1, x_2, \ldots, x_d) = \max_{P(Y) \in \text{ext}[\mathcal{P}_y]} \left(1 + \frac{\sum_{\acute{y} \neq y} P(\acute{y}) \prod_{i=1}^{d} \underline{P}(x_i \mid \acute{y})}{P(y) \prod_{i=1}^{d} \overline{P}(x_i \mid y)}\right)^{-1}$$

Summary

- Unlike standard naïve Bayes classifier, naïve credal classifier may produce a set of prediction $y \in \mathcal{Y}_{\text{pred}} \subset \mathcal{Y}$.
- Interval dominance as a classification rule:

$$\left[\underline{P}(y \mid x_1, x_2, \ldots, x_d),\ \overline{P}(y \mid x_1, x_2, \ldots, x_d)\right]$$

- But the credal sets typically contain more information than the intervals.
  - Other classification rules such as **credal dominance** can be applied.
  - If there is a utility/loss function associated with the prediction $y \in \mathcal{Y}_{\text{pred}}$, we can apply imprecise decision rules to obtain final prediction.
- A fundamental issue is **how to learn** the local credal sets from data.

Other credal classifiers

- Credal decision trees
- Imprecise Dirichlet model (IDM)
- Evidential k-NN
- Evidential Neural Network (ENN)

Regression

Regression

- In regression, the target is a **real value** $y \in \mathbb{R}$.
- To capture the uncertainty, we can instead output an interval $[y^{\min}, y^{\max}]$:

$$[y] = [\underline{y}, \overline{y}] = \left\{ y \in \mathbb{R} \,\middle|\, \underline{y} \leq y \leq \overline{y} \right\}$$

- For intervals $[a] = [\underline{a}, \overline{a}]$ and $[b] = [\underline{b}, \overline{b}]$, **basic operations** are:

Addition: $[\mathrm{a}] + [\mathrm{b}] = [\underline{a} + \underline{b}, \overline{a} + \overline{b}]$

Subtraction: $[\mathrm{a}] - [\mathrm{b}] = [\underline{a} - \overline{b}, \overline{a} - \underline{b}]$

- The second operation is problematic! Assume $a = b = [1,2]$. Then, we have

$$[\mathrm{a}] - [\mathrm{b}] = [1,2] - [1,2] = [-1,1].$$

dependency problem leads to unnecessarily wide intervals.

Precise Regression

- Assume a linear dependence between input and output parametrised by $\boldsymbol{w}_\star$ with an *additive* Gaussian noise:

$$y = \boldsymbol{w}_\star{}^{\mathrm{T}} \boldsymbol{x} + \epsilon$$

- Here, $x \in \mathbb{R}^d$ is a d-dimensional **input feature** drawn i.i.d. from some unknown distribution $\mathcal{P}_X$ and $y \in \mathbb{R}$ is the **true label**.
- The MSE loss over n data pair $\{(x_i, y_i)\}_{\mathrm{i}=1,\ldots,n}$ and its minimiser:

$$\hat{\boldsymbol{w}} = \operatorname{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} \frac{1}{2n} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|^2$$

- Assuming $\boldsymbol{X}$ is full rank, we have the unique solution: $\hat{\boldsymbol{w}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$

Imprecise Regression

- Assume that the labels $y_i s$ are intervals $y_i = [\underline{y_i}, \overline{y_i}]$. The model's parameters then become an interval $\boldsymbol{w} = [\underline{\boldsymbol{w}}, \overline{\boldsymbol{w}}]$.
- For a predictor $\boldsymbol{h} = X[\boldsymbol{w}]$ and interval labels $[\boldsymbol{y}]$, there are two main approaches:

$$L_{\text{INN}} = \frac{1}{2} \| [\boldsymbol{h}] - [\boldsymbol{y}] \|^2$$

$$L_{\text{ISH}} = \frac{1}{2} \left\| \underline{\boldsymbol{h}} - \underline{\boldsymbol{y}} \right\|^2 + \frac{1}{2} \left\| \overline{\boldsymbol{h}} - \overline{\boldsymbol{y}} \right\|^2$$

- The first approach is prone to the **dependency problem**.

Dependency Problem

- Let's take a closer look:

$$L_{\mathrm{INN}} = \frac{1}{2}\|[\boldsymbol{h}] - [\boldsymbol{y}]\|^2 = \frac{1}{2}\left\|\begin{pmatrix} [\boldsymbol{w}^{\mathrm{T}}]\,\boldsymbol{x}_1 - [y_1] \\ [\boldsymbol{w}^{\mathrm{T}}]\,\boldsymbol{x}_2 - [y_2] \\ \vdots \\ [\boldsymbol{w}^{\mathrm{T}}]\,\boldsymbol{x}_n - [y_n] \end{pmatrix}\right\|^2 = \frac{1}{2}\sum_{i=1}^{n}\left([\boldsymbol{w}^{\mathrm{T}}]\,\boldsymbol{x}_i - [y_i]\right)^2$$

- Due to the interval subtraction, we face the ***dependency*** problem.
- Suppose the model correctly predict all intervals: $[\boldsymbol{w}^{\mathrm{T}}]\boldsymbol{x}_i = [y_i], \forall i \in [n]$,

$$L_{\mathrm{INN}} = \frac{1}{2}\sum_{i=1}^{n}([\hat{y}_i] - [y_i])^2 \gg 0$$

- The parameters get updated unnecessarily!

Previous Results

1. Lynne Billard and Edwin Diday. **"Regression analysis for interval-valued data"**. In: Data analysis, classification, and related methods. Springer, 2000, pp. 369–374.
2. Francisco de AT de Carvalho, Eufrasio de A Lima Neto, and Camilo P Tenorio. **"A new method to fit a linear regression model for interval-valued data"**. In: Annual conference on artificial intelligence. Springer. 2004, pp. 295–306.
3. Eufrasio de A Lima Neto and Francisco de AT De Carvalho. **"Constrained linear regression models for symbolic interval-valued variables"**. In: Computational Statistics & Data Analysis 54.2 (2010), pp. 333–347.

- **Center Method (CM)** [1]
  - Features and labels are both interval.
  - Apply standard regressors on the mid-points of the intervals.
  - Efficient and scalable (e.g., can be used with neural networks).
  - Allow for interval input.
  - Discard useful information contained within the intervals.
- **Center and Range Method (CRM and CCRM)** [2,3]
  - Train two independent models for lower and upper bounds of intervals
  - CCRM adds a constraint to guarantee that the upper bound is always larger than or equal to the lower bound.
  - Allow for interval input, efficient and scalable.

Sharp Collection Region (SCR)

- The **SCR** is the set of all ***precise*** *model* h such that:

$$h(\boldsymbol{x}_i) \in [\underline{y}_i, \overline{y}_i],\ \forall i \in \{1, \dots, n\}$$

- It captures the full range of parameter values compatible with the interval-valued data.

  Example:

$$x = 2, [y] = [3,5], h(x) = w \cdot x \Longrightarrow h(2) = 2w \in [3,5] \Longrightarrow w \in [1.5,2.5]$$

  All models $h_w(x) = w \cdot x$ where $w \in [1.5,2.5]$

Georg Schollmeyer and Thomas Augustin. "**Statistical modeling under partial identification: Distinguishing three types of identification regions in regression analysis with interval data**". In: International Journal of Approximate Reasoning 56 (2015), pp. 224–248.

Gradient Descent

- Recall the loss function

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{2}\|X[\boldsymbol{w}] - [\boldsymbol{y}]\|^2$$

- Gradient descent (GD) with step size γ is:

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \gamma\nabla\mathcal{L}(\boldsymbol{w}_k) = \boldsymbol{w}_k - \gamma X^{\mathrm{T}}X[\boldsymbol{w}_k] + \gamma X^{\mathrm{T}}[\boldsymbol{y}]$$

  which follows from

$$\nabla\mathcal{L}(\boldsymbol{w}_k) = X^{\mathrm{T}}(X[\boldsymbol{w}_k] - [\boldsymbol{y}]).$$

- Hence, $\boldsymbol{w}_k$ is a **linear** function of $\boldsymbol{w}_0$ and $\boldsymbol{y}$.

The Methodology

- **Mean Value Theorem (MVT):** Let $f\colon \mathbb{R}^d \to \mathbb{R}$ be differentiable, and let $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d$. Then, there exists $\mathbf{c} \in (\boldsymbol{a}, \boldsymbol{b})$ such that:

$$f(\boldsymbol{b}) = f(\boldsymbol{a}) + \nabla f(\boldsymbol{c})^{\mathrm{T}}(\boldsymbol{b} - \boldsymbol{a})$$

- We can now extend the MVT to the interval-valued function:

$$f([\boldsymbol{x}]) = f(\mathrm{mid}[\boldsymbol{x}]) + \mathcal{J}_{\boldsymbol{w}}([\boldsymbol{x}])^{\mathrm{T}}([\boldsymbol{x}] - \mathrm{mid}[\boldsymbol{x}])$$

- Then, this can be used to *linearly* approximate the radius of error of

$$\boldsymbol{w}_{\mathrm{k}} = f(\boldsymbol{w}_0, \boldsymbol{y})$$

The Methodology

- For linear regression problems with interval labels:

$$[\boldsymbol{w}_{k+1}(\boldsymbol{w}_0, \boldsymbol{y})] = \boldsymbol{w}_{k+1}(\mathrm{mid}(\boldsymbol{w}_0), \mathrm{mid}(\boldsymbol{y})) + \begin{pmatrix} \frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{w}_0} \\ \frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{y}} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} [\boldsymbol{w}_0] - \mathrm{mid}(\boldsymbol{w}_0) \\ [\boldsymbol{y}] - \mathrm{mid}(\boldsymbol{y}) \end{pmatrix}$$

$$\frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{w}_0} = \frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{w}_k} \cdot \frac{\partial \boldsymbol{w}_k}{\partial \boldsymbol{w}_{k-1}} \cdots \frac{\partial \boldsymbol{w}_1}{\partial \boldsymbol{w}_0} = \left(I - \gamma X^{\mathrm{T}} X\right)^k, \qquad \frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{y}} = \gamma X^{\mathrm{T}}$$

- We obtain $\boldsymbol{w}_{k+1}$ by running GD on mid-point labels.
- For $\boldsymbol{y} = \mathrm{mid}([\boldsymbol{y}])$, $\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \gamma X^{\mathrm{T}} X \boldsymbol{w}_k + \gamma X^{\mathrm{T}} \boldsymbol{y}$ and $[\boldsymbol{y}] - \mathrm{mid}([\boldsymbol{y}])$ gives a symmetric interval centered at zero.

Intuition

- The GD is executed on mid-points and error is computed once at the end.
- $\frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{y}}$ measures how our parameter changes with a slight change in $\boldsymbol{y}$.
- Since model is linear, the error scales linearly and is constant over time since $\frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{y}} = \gamma X^{\mathrm{T}}$.
- One can initialise $\boldsymbol{w}_0$ at zero and the only uncertainty comes from $\boldsymbol{y}$.

$$\frac{\partial \boldsymbol{w}_{k+1}}{\partial \boldsymbol{y}}([\boldsymbol{y}] - \mathrm{mid}(\boldsymbol{y})) = \gamma X^{\mathrm{T}}([\boldsymbol{y}] - \mathrm{mid}(\boldsymbol{y}))$$
$$\Longrightarrow [\boldsymbol{w}_{k+1}(\boldsymbol{w}_0, \boldsymbol{y})] = \boldsymbol{w}_{k+1}(\mathrm{mid}(\boldsymbol{w}_0), \mathrm{mid}(\boldsymbol{y})) \pm \tfrac{1}{2}\gamma X^{\mathrm{T}}([\boldsymbol{y}] - \mathrm{mid}(\boldsymbol{y}))$$

Wine Quality Dataset

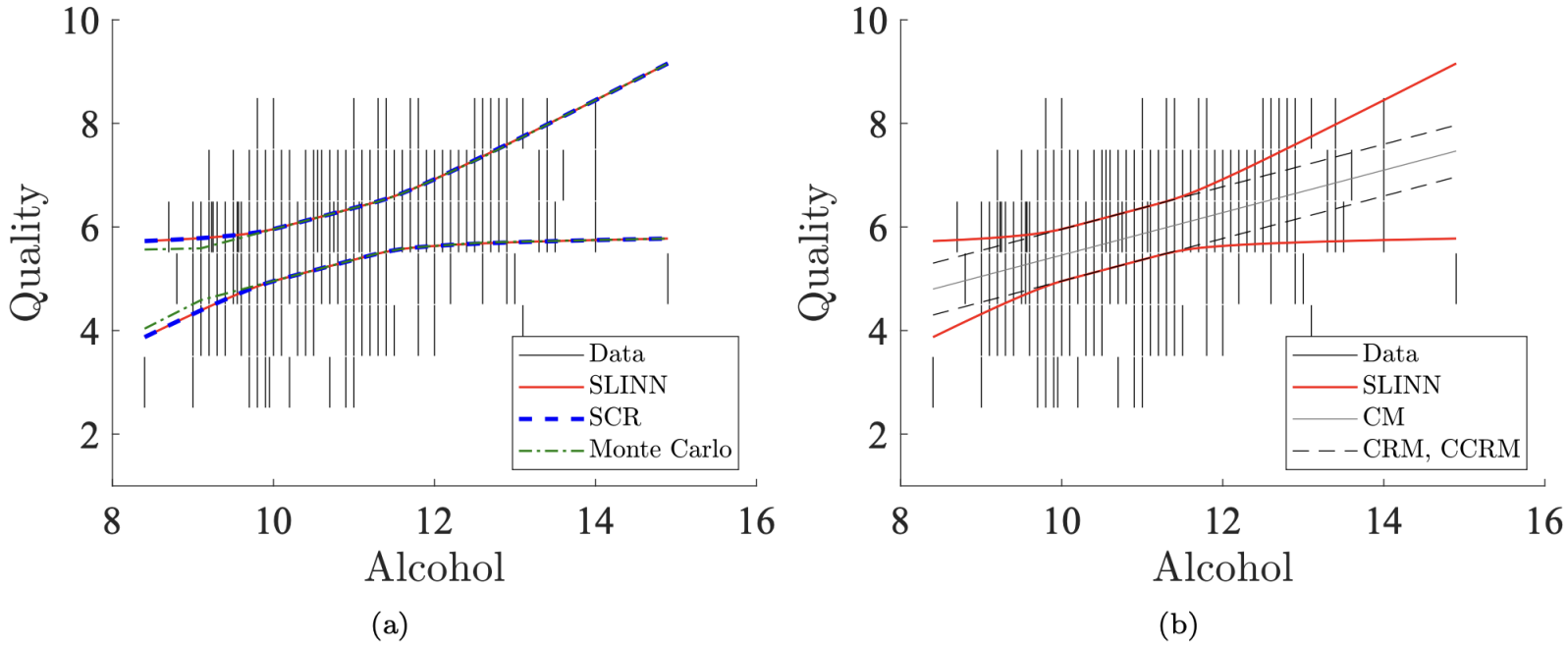


Figure 3: Linear regression example: (a) expectation band obtained from the SLINN (bounded in red lines), from the SCR method (dashed blue lines), Monte Carlo exploration (green dash-dotted lines); (b) the CRM and CCRM methods (dashed black lines) and the CM method produced the solid gray line.

Wine Quality Dataset

Table 2: Summary of results from the selected model. The values are interval coefficients from the linear regression.

	SLINN	SCR	Monte Carlo
(intercept)	[-2.61, 7.48]	[-2.61 ,7.48]	[-1.68, 6.56]
`Volatile acidity`	[-56.27, 20.55]	[-56.27, 20.56]	[-46.56, 10.85]
`Citric acid`	[-37.93, 40.21]	[-37.93, 40.21]	[-29.85, 32.12]
`Chlorides`	[-138.01, 73.44]	[-138.18, 73.53]	[-107.86, 43.21]
`Sulphates`	[-18.78, 50.20]	[-18.78, 50.20]	[-14.63, 46.06]
`Alcohol`	[-1.30, 10.73]	[-1.30, 10.73]	[-0.96,10.39]

Housing Price Dataset

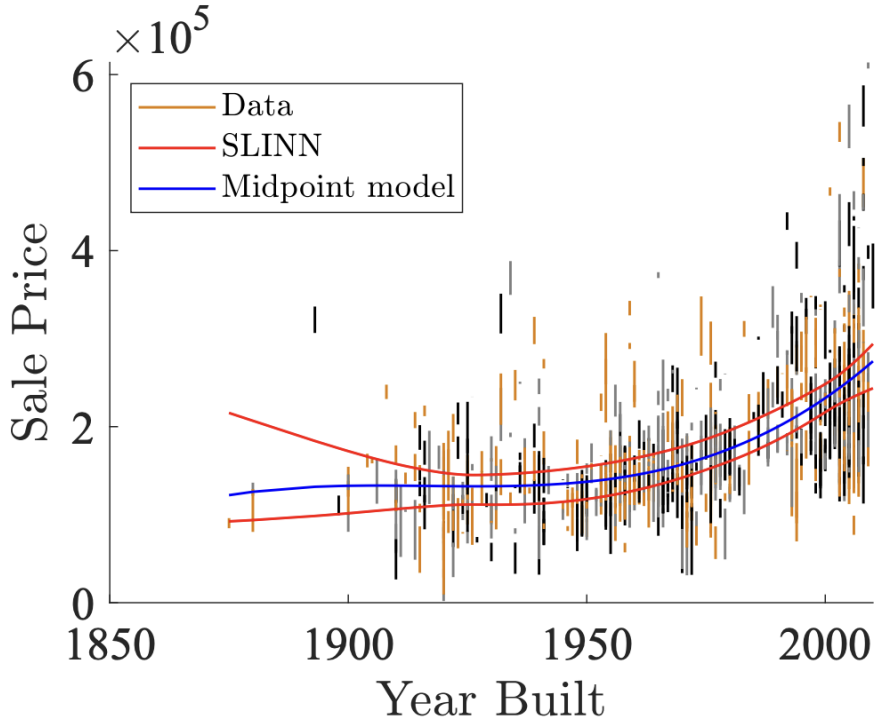


Figure 4: Nonlinear regression example. Expectation band (in red) obtained by fitting a 3rd-order polynomial to interval data and a 3rd-order polynomial fitted to midpoints of the data set.

Recommended Reading

- The Naïve Credal Classifier by Marco Zaffalon
- Statistical modeling under partial identification: Distinguishing three types of identification regions in regression analysis with interval data by Georg Schollmeyer and Thomas Augustin (IJAR 2015)
- Neural network model for imprecise regression with interval dependent variables by Krasymyr Tretiak, Georg Schollmeyer, and Scott Ferson (Neural Networks 2023)

This lecture is based in part on the presentation of **Ali Zindari** submitted for the Imprecise Probabilistic Machine Learning (IPML) seminar at Saarland University.