Metric Elicitation

Bidipta Sarkar, Tanvi Deshpande

October 16, 2023

Introduction[1]

- In binary classification problems, want to select appropriate performance metric
- Metric elicitation: goal is to discover the performance metric of a practitioner, reflects rewards/costs for correct/incorrect classification

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi Koyejo. Performance Metric Elicitation from Pairwise Classifier Comparisons. In AISTATS, 2019.

Introduction[1]

- In binary classification problems, want to select appropriate performance metric
- Metric elicitation: goal is to discover the performance metric of a practitioner, reflects rewards/costs for correct/incorrect classification

Example

Tradeoffs made in medical contexts such as diagnoses or treatment

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi Koyejo. Performance Metric Elicitation from Pairwise Classifier Comparisons. In AISTATS, 2019.

Introduction[1]

- In binary classification problems, want to select appropriate performance metric
- Metric elicitation: goal is to discover the performance metric of a practitioner, reflects rewards/costs for correct/incorrect classification

Example

Tradeoffs made in medical contexts such as diagnoses or treatment

- Rather than menu of a few default choices, devise a metric that best matches the preferences of the practitioners/users with pairwise comparisons
- Minimize amount of feedback needed from "oracle"

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi Koyejo. Performance Metric Elicitation from Pairwise Classifier Comparisons. In AISTATS, 2019.

Motivation

- Humans are bad at providing absolute feedback → pairwise comparison
- Confusion matrices
  - Accurately capture binary metrics such as accuracy, F_β, Jaccard similarity
- Use binary-search procedures to come close to the oracle's performance metric
- Linear performance metrics, linear-fractional performance metrics

Notation

- $X \in \mathcal{X}$ is input RV
- $Y \in \{0, 1\}$ is output RV
- Dataset of size n denoted by $\{(x, y)_i\}_{i=1}^{n}$ generated iid from $\mathbb{P}(X, Y)$
- $\eta(\vec{x}) = \mathbb{P}(Y = 1|X = x)$
- $\zeta = \mathbb{P}(Y = 1)$
- Set of all classifiers: $\mathcal{H} = \{h : \mathcal{X} \to \{0, 1\}\}$
- Confusion matrix for h is $\mathrm{C}(h, \mathbb{P}) \in \mathbb{R}^{2\times 2}$
    - $C_{ij}(h, \mathbb{P}) = \mathbb{P}(Y = i, h = j)$ for $i, j \in \{0, 1\}$
    - TN, FN, TP, FP

Notation

- $X \in \mathcal{X}$ is input RV
- $Y \in \{0, 1\}$ is output RV
- Dataset of size n denoted by $\{(x, y)_i\}_{i=1}^n$ generated iid from $\mathbb{P}(X, Y)$
- $\eta(\vec{x}) = \mathbb{P}(Y = 1|X = x)$
- $\zeta = \mathbb{P}(Y = 1)$
- Set of all classifiers: $\mathcal{H} = \{h : \mathcal{X} \to \{0, 1\}\}$
- Confusion matrix for h is $\mathrm{C}(h, \mathbb{P}) \in \mathbb{R}^{2\times 2}$
  - $C_{ij}(h, \mathbb{P}) = \mathbb{P}(Y = i, h = j)$ for $i, j \in \{0, 1\}$
  - TN, FN, TP, FP
- $FN(h, \mathbb{P}) = \zeta - TP(h, \mathbb{P})$ and $FP(h, \mathbb{P}) = 1 - \zeta - TN(h, \mathbb{P})$
  - Reduces 4-dimensional space to 2-dimensional space

Notation

- $X \in \mathcal{X}$ is input RV
- $Y \in \{0, 1\}$ is output RV
- Dataset of size n denoted by $\{(x, y)_i\}_{i=1}^{n}$ generated iid from $\mathbb{P}(X, Y)$
- $\eta(\vec{x}) = \mathbb{P}(Y = 1|X = x)$
- $\zeta = \mathbb{P}(Y = 1)$
- Set of all classifiers: $\mathcal{H} = \{h : \mathcal{X} \to \{0, 1\}\}$
- Confusion matrix for h is $\mathrm{C}(h, \mathbb{P}) \in \mathbb{R}^{2\times 2}$
  - $C_{ij}(h, \mathbb{P}) = \mathbb{P}(Y = i, h = j)$ for $i, j \in \{0, 1\}$
  - TN, FN, TP, FP
- $FN(h, \mathbb{P}) = \zeta - TP(h, \mathbb{P})$ and $FP(h, \mathbb{P}) = 1 - \zeta - TN(h, \mathbb{P})$
  - Reduces 4-dimensional space to 2-dimensional space
- Any hyperplane (line) in (tp, tn) given by $\ell := a \cdot tp + b \cdot tn = c; a, b, c \in \mathbb{R}$
- $\phi : [0, 1]^{2\times 2} \to \mathbb{R}$ is the performance metric for a classifier h, determined by $C(h)$

Linear Performance Metric (LPM): φ_{LPM}

Given constants $\{a_{11}, a_{01}, a_{10}, a_{00}\} \in \mathbb{R}^4$, we define ϕ as:

$$\begin{aligned} \phi(C) &= a_{11}\,TP + a_{01}\,FP + a_{10}\,FN + a_{00}\,TN \\ &= m_{11}\,TP + m_{00}\,TN + m_0 \end{aligned}$$

where $m_{11} = (a_{11} - a_{10})$, $m_{00} = (a_{00} - a_{01})$, and $m_0 = a_{10}\zeta + a_{01}(1 - \zeta)$.

Performance Metrics

Linear Performance Metric (LPM): φ_{LPM}

Given constants $\{a_{11}, a_{01}, a_{10}, a_{00}\} \in \mathbb{R}^4$, we define ϕ as:

$$\begin{aligned}\phi(C) &= a_{11}\,TP + a_{01}\,FP + a_{10}\,FN + a_{00}\,TN \\ &= m_{11}\,TP + m_{00}\,TN + m_0\end{aligned}$$

where $m_{11} = (a_{11} - a_{10})$, $m_{00} = (a_{00} - a_{01})$, and $m_0 = a_{10}\zeta + a_{01}(1 - \zeta)$.

- Ex. Weighted accuracy: $WA = a_1\,TP + a_2\,TN$

Performance Metrics

Linear-Fractional Performance Metric (LFPM): : φ_{LFPM}

Given constants $\{a_{11}, a_{01}, a_{10}, a_{00}\,,\ b_{11}, b_{01}, b_{10}, b_{00}\} \in \mathbb{R}^8$, we define ϕ as:

$$\phi(C) = \frac{a_{11}\,TP + a_{01}\,FP + a_{10}\,FN + a_{00}\,TN}{b_{11}\,TP + b_{01}\,FP + b_{10}\,FN + b_{00}\,TN} = \frac{p_{11}\,TP + p_{00}\,TN + p_0}{q_{11}\,TP + q_{00}\,TN + q_0}$$

where $p_{11} = (a_{11} - a_{10})\,, p_{00} = (a_{00} - a_{01})\,, q_{11} = (b_{11} - b_{10})\,, q_{00} = (b_{00} - b_{01})\,, p_0 = a_{10}\zeta + a_{01}(1 - \zeta)$, $q_0 = b_{10}\zeta + b_{01}(1 - \zeta)$.

Performance Metrics

Linear-Fractional Performance Metric (LFPM): : φ_{LFPM}

Given constants $\{a_{11}, a_{01}, a_{10}, a_{00}\,,\ b_{11}, b_{01}, b_{10}, b_{00}\} \in \mathbb{R}^8$, we define ϕ as:

$$\phi(C) = \frac{a_{11}\,TP + a_{01}\,FP + a_{10}\,FN + a_{00}\,TN}{b_{11}\,TP + b_{01}\,FP + b_{10}\,FN + b_{00}\,TN} = \frac{p_{11}\,TP + p_{00}\,TN + p_0}{q_{11}\,TP + q_{00}\,TN + q_0}$$

where $p_{11} = (a_{11} - a_{10})\,, p_{00} = (a_{00} - a_{01})\,, q_{11} = (b_{11} - b_{10})\,, q_{00} = (b_{00} - b_{01})\,, p_0 = a_{10}\zeta + a_{01}(1-\zeta)$, $q_0 = b_{10}\zeta + b_{01}(1-\zeta)$.

- Ex. $F_\beta = \frac{TP}{\frac{TP}{1+\beta^2} - \frac{TN}{1+\beta^2} + \frac{\beta^2\zeta + 1 - \zeta}{1+\beta^2}}, JAC = \frac{TP}{1-TN}$

Bayes Optimal/Inverse-Optimal Classifiers

- Given performance metric ϕ:
  - Bayes utility $\bar{\tau}$: $\bar{\tau} = \sup_{h \in \mathcal{H}} \phi(C(h)) = \sup_{C \in \mathcal{C}} \phi(C)$
  - Bayes classifier $\bar{h}$ (when it exists): $\bar{h} = \arg\max_{h \in \mathcal{H}} \phi(C(h))$.
  - Bayes confusion matrix is given by $\bar{C} = \arg\max_{C \in \mathcal{C}} \phi(C)$
- Inverse Bayes utility/classifier/CM: replace all sup with inf

Proposition 1

Let $\phi \in \varphi_{LPM}$. Then

$$\bar{h}(x) = \begin{cases} \mathbb{1}\left[\eta(x) \geq \frac{m_{00}}{m_{11}+m_{00}}\right], & m_{11} + m_{00} \geq 0 \\ \mathbb{1}\left[\frac{m_{00}}{m_{11}+m_{00}} \geq \eta(x)\right], & \text{o.w.} \end{cases}$$

is a Bayes optimal classifier w.r.t ϕ. The inverse Bayes classifier is given by $\underline{h} = 1 - \bar{h}$.

Metric Elicitation Setup

- Oracle queries: Given two classifiers (CMs) h, h' (C, C'), we have $\Gamma(h, h') = \Omega(C, C') = \mathbb{1}[\phi(C) > \phi(C')] =: \mathbb{1}[C \succ C']$

Metric Elicitation (Population)

Suppose the true (oracle) performance metric is ϕ. Recover a metric $\hat{\phi}$ by querying the oracle for as few pairwise comparisons of the form $\Omega(C, C')$, such that $\|\phi - \hat{\phi}\|_{--} < \kappa$ for sufficiently small $\mathbb{R} \ni \kappa > 0$ and for any suitable norm $\|\cdot\|_{--}$.

Metric Elicitation Setup

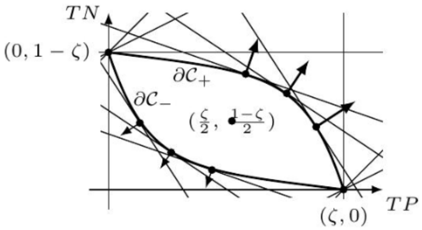
- Oracle queries: Given two classifiers (CMs) h, h' (C, C'), we have $\Gamma(h, h') = \Omega(C, C') = \mathbb{1}[\phi(C) > \phi(C')] =: \mathbb{1}[C \succ C']$

Metric Elicitation (Samples: $\{(x_i, y_i)\}_{i=1}^n$))

Suppose the true (oracle) performance metric is ϕ. Recover a metric $\hat{\phi}$ by querying the oracle for as few pairwise comparisons of the form $\Omega\left(\hat{C}, \hat{C}'\right)$, such that $\|\phi - \hat{\phi}\|_{--} < \kappa$ for sufficiently small $\mathbb{R} \ni \kappa > 0$ and for any suitable norm $\| \cdot \|_{-}$.

Confusion Matrices

- Assume $g(t) = \mathbb{P}[\eta(X) \geq t]$ is continuous and strictly decreasing for $t \in [0, 1]$
- $\mathcal{C}$ is convex, closed, contained in the rectangle $[0, \zeta] \times [0, 1 - \zeta]$ (bounded), and 180° rotationally symmetric around the center-point $\left(\frac{\zeta}{2}, \frac{1-\zeta}{2}\right)$. Under our assumption, $(0, 1 - \zeta)$ and $(\zeta, 0)$ are the only vertices of $\mathcal{C}$; $\mathcal{C}$ is strictly convex. Thus, any supporting hyperplane of $\mathcal{C}$ is tangent at only one point.
- Unique Bayes CM on boundary $\partial\mathcal{C}$, independent of bias term

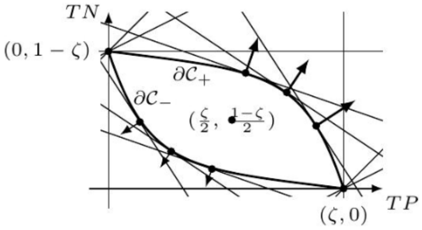


Confusion Matrices

- Vary tradeoffs $\mathrm{m} = (m_{00}, m_{11})$ s.t. $||\mathrm{m}|| = 1$; $\varphi_{LPM} = \{\mathrm{m} = (\cos\theta, \sin\theta) : \theta \in [0, 2\pi]\}$
- Given m, can recover Bayes classifier/CM; unique (convexity of $\mathcal{C}$).
- Also, supporting hyperplane

$$\bar{\ell}_{\mathrm{m}} := m_{11} \cdot tp + m_{00} \cdot tn = m_{11}\overline{TP}_{\mathrm{m}} + m_{00}\overline{TN}_{\mathrm{m}}$$

- Note: If m_{00}, m_{11} are of opposite signs, $\bar{h}_m$ is the trivial classifier predicting all 1's or all 0's. (positive slope)
- Note: Can split $\partial\mathcal{C}$ into upper/lower boundary $(\partial\mathcal{C}_+, \partial\mathcal{C}_-)$



LPM Elicitation

Main idea: For a metric ψ (quasiconvex and monotone increasing in TP/TN) or ϕ (quasiconcave and monotone increasing), and parametrization ρ^+/ρ^- of upper/lower boundary, composition $\psi \circ \rho^-$ is quasiconvex and unimodal on [0, 1], and $\phi \circ \rho^+$ is quasiconcave and unimodal on [0, 1], and.

- Therefore, binary-search type algorithm is possible for maximizer $\bar{C}$, minimizer $\underline{C}$ & first-order approximation of ϕ at these points (supporting hyperplane)

LPM Elicitation Algorithm

Algorithm 1 Quasiconcave Metric Maximization

Input: $\epsilon > 0$ and oracle Ω.
Initialize: $\theta_a = 0$, $\theta_b = \frac{\pi}{2}$.
while $|\theta_b - \theta_a| > \epsilon$ **do**
  Set $\theta_c = \frac{3\theta_a + \theta_b}{4}$, $\theta_d = \frac{\theta_a + \theta_b}{2}$, and $\theta_e = \frac{\theta_a + 3\theta_b}{4}$. Set corresponding slopes ($\mathbf{m}$'s) using (6).
  Obtain $\overline{h}_{\theta_a}, \overline{h}_{\theta_c}, \overline{h}_{\theta_d}, \overline{h}_{\theta_e}, \overline{h}_{\theta_b}$ using Proposition 1. Compute $\overline{C}_{\theta_a}, \overline{C}_{\theta_c}, \overline{C}_{\theta_d}, \overline{C}_{\theta_e}, \overline{C}_{\theta_b}$ using (1).
  Query $\Omega(\overline{C}_{\theta_c}, \overline{C}_{\theta_a}), \Omega(\overline{C}_{\theta_d}, \overline{C}_{\theta_c}), \Omega(\overline{C}_{\theta_e}, \overline{C}_{\theta_d})$, and $\Omega(\overline{C}_{\theta_b}, \overline{C}_{\theta_e})$.
  If $\overline{C}_{\theta} \succ \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$ for consecutive $\theta < \theta' < \theta''$, assume the default order $\overline{C}_{\theta} \prec \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$.
  if ($\overline{C}_{\theta_a} \succ \overline{C}_{\theta_c}$) Set $\theta_b = \theta_d$.
  elseif ($\overline{C}_{\theta_a} \prec \overline{C}_{\theta_c} \succ \overline{C}_{\theta_d}$) Set $\theta_b = \theta_d$.
  elseif ($\overline{C}_{\theta_c} \prec \overline{C}_{\theta_d} \succ \overline{C}_{\theta_e}$) Set $\theta_a = \theta_c$, $\theta_b = \theta_e$.
  elseif ($\overline{C}_{\theta_d} \prec \overline{C}_{\theta_e} \succ \overline{C}_{\theta_b}$) Set $\theta_a = \theta_d$.
  else Set $\theta_a = \theta_d$.
Output: $\overline{\mathbf{m}}, \overline{C}$, and $\overline{\ell}$, where $\overline{\mathbf{m}} = \mathbf{m}_d\ (\theta_d)$, $\overline{C} = \overline{C}_{\theta_d}$, and $\overline{\ell} := \langle \overline{\mathbf{m}}, (tp, tn) \rangle = \langle \overline{\mathbf{m}}, \overline{C} \rangle$.

LPM Elicitation Algorithm

Algorithm 1 Quasiconcave Metric Maximization

1: **Input:** $\epsilon > 0$ and oracle Ω.
2: **Initialize:** $\theta_a = 0$, $\theta_b = \frac{\pi}{2}$.
3: **while** $|\theta_b - \theta_a| > \epsilon$ **do**
4: Set $\theta_c = \frac{3\theta_a + \theta_b}{4}$, $\theta_d = \frac{\theta_a + \theta_b}{2}$, and $\theta_e = \frac{\theta_a + 3\theta_b}{4}$. Set corresponding slopes ($\mathbf{m}$'s) using (6).
5: Obtain $\overline{h}_{\theta_a}, \overline{h}_{\theta_c}, \overline{h}_{\theta_d}, \overline{h}_{\theta_e}, \overline{h}_{\theta_b}$ using Proposition 1. Compute $\overline{C}_{\theta_a}, \overline{C}_{\theta_c}, \overline{C}_{\theta_d}, \overline{C}_{\theta_e}, \overline{C}_{\theta_b}$ using (1).
6: Query $\Omega(\overline{C}_{\theta_c}, \overline{C}_{\theta_a}), \Omega(\overline{C}_{\theta_d}, \overline{C}_{\theta_c}), \Omega(\overline{C}_{\theta_e}, \overline{C}_{\theta_d})$, and $\Omega(\overline{C}_{\theta_b}, \overline{C}_{\theta_e})$.
7: If $\overline{C}_{\theta} \succ \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$ for consecutive $\theta < \theta' < \theta''$, assume the default order $\overline{C}_{\theta} \prec \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$.
8: **if** ($\overline{C}_{\theta_a} \succ \overline{C}_{\theta_c}$) Set $\theta_b = \theta_d$.
9: **elseif** ($\overline{C}_{\theta_a} \prec \overline{C}_{\theta_c} \succ \overline{C}_{\theta_d}$) Set $\theta_b = \theta_d$.
10: **elseif** ($\overline{C}_{\theta_c} \prec \overline{C}_{\theta_d} \succ \overline{C}_{\theta_e}$) Set $\theta_a = \theta_c$, $\theta_b = \theta_e$.
11: **elseif** ($\overline{C}_{\theta_d} \prec \overline{C}_{\theta_e} \succ \overline{C}_{\theta_b}$) Set $\theta_a = \theta_d$.
12: **else** Set $\theta_a = \theta_d$.
13: **Output:** $\overline{\mathbf{m}}, \overline{C}$, and $\overline{\ell}$, where $\overline{\mathbf{m}} = \mathbf{m}_d\ (\theta_d)$, $\overline{C} = \overline{C}_{\theta_d}$, and $\overline{\ell} := \langle \overline{\mathbf{m}}, (tp, tn) \rangle = \langle \overline{\mathbf{m}}, \overline{C} \rangle$.

- Quasiconvex: Start with $\theta \in \left[\pi, \frac{3}{2}\pi\right]$ and flip all $\prec$ and $\succ$.

LPM Elicitation Algorithm

Algorithm 1 Quasiconcave Metric Maximization

Input: $\epsilon > 0$ and oracle Ω.
Initialize: $\theta_a = 0$, $\theta_b = \frac{\pi}{2}$.
while $|\theta_b - \theta_a| > \epsilon$ **do**
  Set $\theta_c = \frac{3\theta_a + \theta_b}{4}$, $\theta_d = \frac{\theta_a + \theta_b}{2}$, and $\theta_e = \frac{\theta_a + 3\theta_b}{4}$. Set corresponding slopes ($\mathbf{m}$'s) using (6).
  Obtain $\overline{h}_{\theta_a}, \overline{h}_{\theta_c}, \overline{h}_{\theta_d}, \overline{h}_{\theta_e}, \overline{h}_{\theta_b}$ using Proposition 1. Compute $\overline{C}_{\theta_a}, \overline{C}_{\theta_c}, \overline{C}_{\theta_d}, \overline{C}_{\theta_e}, \overline{C}_{\theta_b}$ using (1).
  Query $\Omega(\overline{C}_{\theta_c}, \overline{C}_{\theta_a}), \Omega(\overline{C}_{\theta_d}, \overline{C}_{\theta_c}), \Omega(\overline{C}_{\theta_e}, \overline{C}_{\theta_d})$, and $\Omega(\overline{C}_{\theta_b}, \overline{C}_{\theta_e})$.
  If $\overline{C}_{\theta} \succ \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$ for consecutive $\theta < \theta' < \theta''$, assume the default order $\overline{C}_{\theta} \prec \overline{C}_{\theta'} \prec \overline{C}_{\theta''}$.
  if $(\overline{C}_{\theta_a} \succ \overline{C}_{\theta_c})$ Set $\theta_b = \theta_d$.
  elseif $(\overline{C}_{\theta_a} \prec \overline{C}_{\theta_c} \succ \overline{C}_{\theta_d})$ Set $\theta_b = \theta_d$.
  elseif $(\overline{C}_{\theta_c} \prec \overline{C}_{\theta_d} \succ \overline{C}_{\theta_e})$ Set $\theta_a = \theta_c$, $\theta_b = \theta_e$.
  elseif $(\overline{C}_{\theta_d} \prec \overline{C}_{\theta_e} \succ \overline{C}_{\theta_b})$ Set $\theta_a = \theta_d$.
  else Set $\theta_a = \theta_d$.
Output: $\overline{\mathbf{m}}, \overline{C}$, and $\overline{\ell}$, where $\overline{\mathbf{m}} = \mathbf{m}_d\ (\theta_d)$, $\overline{C} = \overline{C}_{\theta_d}$, and $\overline{\ell} := \langle \overline{\mathbf{m}}, (tp, tn) \rangle = \langle \overline{\mathbf{m}}, \overline{C} \rangle$.

- Quasiconvex: Start with $\theta \in \left[\pi, \frac{3}{2}\pi\right]$ and flip all $\prec$ and $\succ$.
- LPM elicitation: Run Algorithm 1, querying oracle, and take elicited metric $\hat{m}$ (maximizer) to be the slope of the resulting hyperplane.

LFPM Elicitation Algorithm

Assumption

Let $\phi \in \varphi_{LFPM}$. We assume $p_{11}, p_{00} \geq 0, p_{11} \geq q_{11}, p_{00} \geq q_{00}$, $p_0 = 0, q_0 = (p_{11} - q_{11})\zeta + (p_{00} - q_{00})(1 - \zeta)$, and $p_{11} + p_{00} = 1$.

- With this assumption, ϕ is bounded in $[0, 1]$ and monotonically increasing in TP and TN.

LFPM Elicitation Algorithm

Assumption

Let $\phi \in \varphi_{LFPM}$. We assume $p_{11}, p_{00} \geq 0, p_{11} \geq q_{11}, p_{00} \geq q_{00}$, $p_0 = 0, q_0 = (p_{11} - q_{11})\zeta + (p_{00} - q_{00})(1 - \zeta)$, and $p_{11} + p_{00} = 1$.

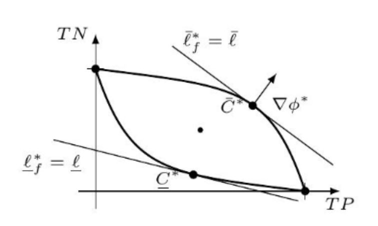
- With this assumption, ϕ is bounded in $[0, 1]$ and monotonically increasing in TP and TN.

Algorithm Overview

- Obtain maximizer and minimizer using Algorithm 1
- Results in two systems of equations w/ 1 degree of freedom
- The true metric is where solutions to the systems match pointwise on the CMs.

Algorithm Overview

- Obtain maximizer and minimizer using Algorithm 1
  - Suppose the true metric is $\phi^*(C) = \frac{p^*_{11} TP + p^*_{00} TN}{q^*_{11} TP + q^*_{00} TN + q^*_0}$, and let $\bar{\tau}, \underline{\tau}$ be the maximizer/minimizer of ϕ over $\mathcal{C}$.
  - There exists a hyperplane $\bar{\ell}^*_f := (p^*_{11} - \bar{\tau}^* q^*_{11})\, tp + (p^*_{00} - \bar{\tau}^* q^*_{00})\, tn = \bar{\tau}^* q^*_0$ which touches $\mathcal{C}$ at $\left(\overline{TP}^*, \overline{TN}^*\right)$ on $\partial\mathcal{C}_+$.
  - There also exists a hyperplane $\underline{\ell}^*_f := (p^*_{11} - \underline{\tau}^* q^*_{11})\, tp + (p^*_{00} - \underline{\tau}^* q^*_{00})\, \mathrm{tn} = \underline{\tau}^* q^*_0$, which touches $\mathcal{C}$ at $(\underline{TP}^*, \underline{TN}^*)$ on $\partial\mathcal{C}_-$.



LFPM Elicitation Algorithm

Algorithm Overview

- Results in two systems of equations w/ 1 degree of freedom
  - From Algorithm 1, we can get a hyperplane $\bar{\ell} := \bar{m}_{11} tp + \bar{m}_{00} tn = \bar{C}_0$,
    - where $\bar{C}_0 = \bar{m}_{11} \overline{TP}^* + \bar{m}_{00} \overline{TN}^*$
    - equivalent to $\bar{\ell}_f^*$ up to a constant multiple
  - SoE: $p_{11}^* - \bar{\tau}^* q_{11}^* = \alpha \bar{m}_{11}, p_{00}^* - \bar{\tau}^* q_{00}^* = \alpha \bar{m}_{00}, \bar{\tau}^* q_0^* = \alpha \bar{C}_0$
  - $p_{11}' - \bar{\tau}^* q_{11}' = \bar{m}_{11}, p_{00}' - \bar{\tau}^* q_{00}' = \bar{m}_{00}, \bar{\tau}^* q_0' = \bar{C}_0$.
  - From Algorithm 1, we also get the hyperplane $\underline{\ell} := \underline{m}_{11} tp + \underline{m}_{00} tn = \underline{C}_0$,
    - $\underline{C}_0 = \underline{m}_{11} \underline{TP}^* + \underline{m}_{00} \underline{TN}^*$.
    - equivalent to $\underline{\ell}_f^*$ up to a constant multiple
  - SoE: $p_{11}^* - \underline{\tau}^* q_{11}^* = \gamma \underline{m}_{11}, p_{00}^* - \underline{\tau}^* q_{00}^* = \gamma \underline{m}_{00}, \underline{\tau}^* q_0^* = \gamma \underline{C}_0$
  - $p_{11}'' - \underline{\tau}^* q_{11}'' = \underline{m}_{11}, p_{00}'' - \underline{\tau}^* q_{00}'' = \underline{m}_{00}, \underline{\tau}^* q_0'' = \underline{C}_0$.

LFPM Elicitation Algorithm

Algorithm Overview

- Results in two systems of equations w/ 1 degree of freedom
  - From Algorithm 1, we can get a hyperplane $\bar{\ell} := \bar{m}_{11} tp + \bar{m}_{00} tn = \bar{C}_0$,
    - where $\bar{C}_0 = \bar{m}_{11}\overline{TP}^* + \bar{m}_{00}\overline{TN}^*$
    - equivalent to $\bar{\ell}_f^*$ up to a constant multiple
  - SoE: $p_{11}^* - \bar{\tau}^* q_{11}^* = \alpha \bar{m}_{11}, p_{00}^* - \bar{\tau}^* q_{00}^* = \alpha \bar{m}_{00}, \bar{\tau}^* q_0^* = \alpha \bar{C}_0$
  - $p_{11}' - \bar{\tau}^* q_{11}' = \bar{m}_{11}, p_{00}' - \bar{\tau}^* q_{00}' = \bar{m}_{00}, \bar{\tau}^* q_0' = \bar{C}_0$.
  - From Algorithm 1, we also get the hyperplane $\underline{\ell} := \underline{m}_{11} tp + \underline{m}_{00} tn = \underline{C}_0$,
    - $\underline{C}_0 = \underline{m}_{11}\underline{TP}^* + \underline{m}_{00}\underline{TN}^*$.
    - equivalent to $\underline{\ell}_f^*$ up to a constant multiple
  - SoE: $p_{11}^* - \underline{\tau}^* q_{11}^* = \gamma \underline{m}_{11}, p_{00}^* - \underline{\tau}^* q_{00}^* = \gamma \underline{m}_{00}, \underline{\tau}^* q_0^* = \gamma \underline{C}_0$
  - $p_{11}'' - \underline{\tau}^* q_{11}'' = \underline{m}_{11}, p_{00}'' - \underline{\tau}^* q_{00}'' = \underline{m}_{00}, \underline{\tau}^* q_0'' = \underline{C}_0$.

Algorithm Overview

- Both of these SoE's have only one degree of freedom; knowing p'_{11} solves it as follows:
  - $p'_{00} = 1 - p'_{11}, q'_0 = \bar{C}_0 \frac{P'}{Q'}$
  - $q'_{11} = (p'_{11} - \bar{m}_{11}) \frac{P'}{Q'}, q'_{00} = (p'_{00} - \bar{m}_{00}) \frac{P'}{Q'}$
  - where $P' = p'_{11}\zeta + p'_{00}(1-\zeta)$ and $Q' = P' + \bar{C}_0 - \bar{m}_{11}\zeta - \bar{m}_{00}(1-\zeta)$
- Say we know p'_{11}. Then, we can solve the above SoE and obtain a metric ϕ'.
- Similarly, if we know p''_{11}, we can solve an analogous SoE and obtain a metric ϕ''.

Algorithm Overview

- Both of these SoE's have only one degree of freedom; knowing p'_{11} solves it as follows:
  - $p'_{00} = 1 - p'_{11}, q'_0 = \bar{C}_0 \frac{P'}{Q'}$
  - $q'_{11} = (p'_{11} - \bar{m}_{11}) \frac{P'}{Q'}, q'_{00} = (p'_{00} - \bar{m}_{00}) \frac{P'}{Q'}$
  - where $P' = p'_{11}\zeta + p'_{00}(1-\zeta)$ and $Q' = P' + \bar{C}_0 - \bar{m}_{11}\zeta - \bar{m}_{00}(1-\zeta)$
- Say we know p'_{11}. Then, we can solve the above SoE and obtain a metric ϕ'.
- Similarly, if we know p''_{11}, we can solve an analogous SoE and obtain a metric ϕ''.
- When $p^*_{11}/p^*_{00} = p'_{11}/p'_{00} = p''_{11}/p''_{00}$, then $\phi^*(C) = \phi'(C)/\alpha = -\phi''(C)/\gamma$.

Algorithm Overview

- Both of these SoE's have only one degree of freedom; knowing p'_{11} solves it as follows:
  - $p'_{00} = 1 - p'_{11}, q'_0 = \bar{C}_0 \frac{P'}{Q'}$
  - $q'_{11} = (p'_{11} - \bar{m}_{11}) \frac{P'}{Q'}, q'_{00} = (p'_{00} - \bar{m}_{00}) \frac{P'}{Q'}$
  - where $P' = p'_{11}\zeta + p'_{00}(1-\zeta)$ and $Q' = P' + \bar{C}_0 - \bar{m}_{11}\zeta - \bar{m}_{00}(1-\zeta)$
- Say we know p'_{11}. Then, we can solve the above SoE and obtain a metric ϕ'.
- Similarly, if we know p''_{11}, we can solve an analogous SoE and obtain a metric ϕ''.
- When $p^*_{11}/p^*_{00} = p'_{11}/p'_{00} = p''_{11}/p''_{00}$, then $\phi^*(C) = \phi'(C)/\alpha = -\phi''(C)/\gamma$.
- **We will grid search for p'_{11} on $[0,1]$, and compute ϕ', ϕ''. We will check a number of confusion matrices on the boundaries and select the value of p'_{11} for which the ratio ϕ''/ϕ' is the closest to constant**

LFPM Elicitation Algorithm

LFPM Elicitation (True metric ϕ^*)

1. Run Algorithm 1 to get $\overline{C}^*$, a hyperplane $\bar{\ell}$, and SoE (9).
2. Run Algorithm 2 to get $\underline{C}^*$, a hyperplane $\underline{\ell}$, and SoE (10).
3. Run the oracle-query independent Algorithm 3 to get the elicited metric, which satisfies both the SoEs.

Algorithm 3 Grid Search for Best Ratio

Input: k, Δ.
Initialize: $\sigma_{opt} = \infty, p'_{11,opt} = 0$.
Generate $C_1, ..., C_k$ on ∂C_+ and ∂C_- (Section 3).
for $(p'_{11} = 0; p'_{11} \leq 1; p'_{11} = p'_{11} + \Delta)$ **do**
    Compute ϕ', ϕ'' using Proposition 4. Compute array $r = [\frac{\phi'(C_1)}{\phi''(C_1)}, ..., \frac{\phi'(C_k)}{\phi''(C_k)}]$. Set $\sigma = \text{std}(r)$.
    if $(\sigma < \sigma_{opt})$ Set $\sigma_{opt} = \sigma$ and $p'_{11,opt} = p'_{11}$.
Output: $p'_{11,opt}$.

Guarantees

Theorem 1

Given $\epsilon, \epsilon_\Omega \geq 0$ and a metric ϕ satisfying our assumptions. Algorithm 1/2 finds an approximate maximizer/minimizer and supporting hyperplane. Also, the value of ϕ at that point is within $O\left(\sqrt{\epsilon_\Omega} + \epsilon\right)$ of the optimum, and the number of queries is $O\left(\log \frac{1}{\epsilon}\right)$.

Guarantees

Theorem 1

Given $\epsilon, \epsilon_\Omega \geq 0$ and a metric ϕ satisfying our assumptions. Algorithm 1/2 finds an approximate maximizer/minimizer and supporting hyperplane. Also, the value of ϕ at that point is within $O\left(\sqrt{\epsilon_\Omega} + \epsilon\right)$ of the optimum, and the number of queries is $O\left(\log \frac{1}{\epsilon}\right)$.

Theorem 2

Let m^* be the true performance metric. Given $\epsilon > 0$, *LPM* elicitation outputs a performance metric $\hat{m}$, s.t. $\|m^* - \hat{m}\|_\infty \leq \sqrt{2}\epsilon + \frac{2}{k_0}\sqrt{2k_1\epsilon_\Omega}$.

Guarantees

Theorem 1

Given $\epsilon, \epsilon_\Omega \geq 0$ and a metric ϕ satisfying our assumptions. Algorithm 1/2 finds an approximate maximizer/minimizer and supporting hyperplane. Also, the value of ϕ at that point is within $O\left(\sqrt{\epsilon_\Omega} + \epsilon\right)$ of the optimum, and the number of queries is $O\left(\log \frac{1}{\epsilon}\right)$.

Theorem 2

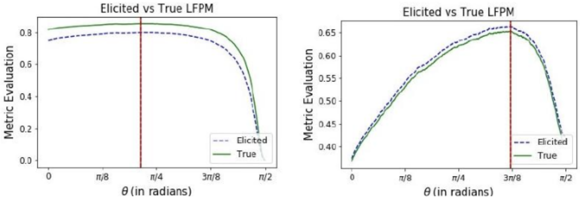
Let m^* be the true performance metric. Given $\epsilon > 0$, *LPM* elicitation outputs a performance metric $\hat{\mathrm{m}}$, s.t. $\|\mathrm{m}^* - \hat{\mathrm{m}}\|_\infty \leq \sqrt{2}\epsilon + \frac{2}{k_0}\sqrt{2k_1\epsilon_\Omega}$.

Lemma 3

Let h_θ and $\hat{h}_\theta$ be two classifiers estimated using η and $\hat{\eta}$, respectively. Further, let $\bar{\theta}$ be such that $h_{\bar{\theta}} = \arg\max_\theta \phi\left(h_\theta\right)$. Then $\|C(\hat{h}_{\bar{\theta}}) - C\left(h_{\bar{\theta}}\right)\|_\infty = O\left(\|\hat{\eta}_n - \eta\|_\infty\right)$.

Real-World Experiments

- Used Breast Cancer Diagnostic dataset (569 samples) and Magic (M) dataset (19020 samples)
- Both LPM/LFPM: Improved elicitation for dataset M: ME improves with larger datasets
- LPM: Recover all 28 for $\epsilon = 0.11$, $\epsilon = 0.02$ is too tight; algorithm is stuck at the closest achievable confusion matrix from finite samples, need not be optimal overall
- LFPM: Results for dataset M; elicited metrics equivalent to true metrics up to a constant



Further reading: Fair Performance Metric Elicitation[2]

- Select tradeoff between overall performance and discrepancy between performance on certain protected groups

Definition 1. *Fair Performance Metric: Let ϕ and φ be monotonically increasing linear functions of overall rates and group discrepancies, respectively. The fair metric Ψ is a trade-off between ϕ and φ. In particular, given $\mathbf{a} \in \mathbb{R}^q, \mathbf{a} \geq 0$ (misclassification weights), a set of vectors $\mathbf{B} := \{\mathbf{b}^{uv} \in \mathbb{R}^q, \mathbf{b}^{uv} \geq 0\}_{u,v=1,v>u}^{m}$ (fairness violation weights), and a scalar λ (trade-off) with*

$$\|\mathbf{a}\|_2 = 1, \qquad \sum\nolimits_{u,v=1,v>u}^{m} \|\mathbf{b}^{uv}\|_2 = 1, \qquad 0 \leq \lambda \leq 1, \tag{5}$$

(wlog., due to scale invariance), we define the metric Ψ as:

$$\Psi(\mathbf{r}^{1:m}; \mathbf{a}, \mathbf{B}, \lambda) := \underbrace{(1-\lambda)}_{\textit{trade-off}} \underbrace{\langle \mathbf{a}, \mathbf{r} \rangle}_{\phi(\mathbf{r})} + \lambda \underbrace{\left(\sum\nolimits_{u,v=1,v>u}^{m} \langle \mathbf{b}^{uv}, \mathbf{d}^{uv} \rangle \right)}_{\varphi(\mathbf{r}^{1:m})}. \tag{6}$$

[2] Gaurush Hiranandani, Harikrishna Narasimhan, and Oluwasanmi Koyejo. Fair Performance Metric Elicitation. In NeurIPS, 2020.

Multiclass Performance Metric Elicitation

Bidipta Sarkar, Tanvi Deshpande

October 16, 2023

Outline

Topic

Motivation[1]

Similar motivation to binary classification performance metrics, but extended to multiclass classification.

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi O Koyejo. Multiclass performance metric elicitation. In NeurIPS, 2019.

Motivation[1]

Similar motivation to binary classification performance metrics, but extended to multiclass classification.

Example

Test to determine which subtype of leukemia is present in a patient

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi O Koyejo. Multiclass performance metric elicitation. In NeurIPS, 2019.

Motivation[1]

Similar motivation to binary classification performance metrics, but extended to multiclass classification.

Example

Test to determine which subtype of leukemia is present in a patient

- It may be possible that some treatment options are worse than others during misclassification

[1] Gaurush Hiranandani, Shant Boodaghians, Ruta Mehta, and Oluwasanmi O Koyejo. Multiclass performance metric elicitation. In NeurIPS, 2019.

Notation

- $X \in \mathcal{X}$ is input RV
- $Y \in [k]$ is output RV
  - $[k]$ is the index set {1, 2, . . . , k}
- Dataset of size n denoted by $\{(\vec{x}, y)_i\}_{i=1}^{n}$ generated iid from $\mathbb{P}(X, Y)$
- $\eta_i(\vec{x}) = \mathbb{P}(Y = i | X = \vec{x})$
- $\xi_i = \mathbb{P}(Y = i)$
- Set of all classifiers: $\mathcal{H} = \{h : \mathcal{X} \to \Delta_k\}$
  - (Δ_k is (k-1) dimensional simplex)
- Confusion matrix for h is $C(h, \mathbb{P}) \in \mathbb{R}^{k \times k}$
  - $C_{ij}(h, \mathbb{P}) = \mathbb{P}(Y = i, h = j)$ for $i, j \in [k]$

Notation

- $X \in \mathcal{X}$ is input RV
- $Y \in [k]$ is output RV
  - $[k]$ is the index set {1, 2, . . . , k}
- Dataset of size n denoted by $\{(\vec{x}, y)_i\}_{i=1}^{n}$ generated iid from $\mathbb{P}(X, Y)$
- $\eta_i(\vec{x}) = \mathbb{P}(Y = i|X = \vec{x})$
- $\xi_i = \mathbb{P}(Y = i)$
- Set of all classifiers: $\mathcal{H} = \{h : \mathcal{X} \to \Delta_k\}$
  - (Δ_k is (k-1) dimensional simplex)
- Confusion matrix for h is $C(h, \mathbb{P}) \in \mathbb{R}^{k \times k}$
  - $C_{ij}(h, \mathbb{P}) = \mathbb{P}(Y = i, h = j)$ for $i, j \in [k]$

Usage of Confusion Matrix

Off-Diagonal Confusions

$\mathrm{C}(h, \mathbb{P})$ is uniquely determined by off-diagonal elements

- Number of off-diagonal elements is $q := k^2 - k$
- $\vec{c}(h, \mathbb{P}) = \textit{off-diag}\ (\mathrm{C}(h, \mathbb{P}))$
- Space of off-diagonal confusions is $\mathcal{C} = \{\vec{c}(h, \mathbb{P}) : h \in \mathcal{H}\}$

Diagonal Confusions

Some metrics only care about misclassification, not its type

- $\vec{d}(h, \mathbb{P}) = \textit{diag}\ (\mathrm{C}(h, \mathbb{P}))$
- Space of diagonal confusions is $\mathcal{D} = \{\vec{d}(h, \mathbb{P}) : h \in \mathcal{H}\}$

Topic

Metric Elicitation Reminder

Identical to binary classification, just using the new multiclass notation

Oracle Query

Given two classifiers h, h', a query to the oracle with metric ϕ is represented by:

$$\Gamma(h, h') = \Omega(\vec{c}, \vec{c}') = 1[\phi(\vec{c}) > \phi(\vec{c'})] =: 1[\vec{c} \succ \vec{c'}] \quad (1)$$

Metric Elicitation with Pairwise Queries

Let the oracle's performance metric be ϕ. Using oracle queries of the form $\Omega(\hat{c}, \hat{c}')$ (estimated confusions from samples), we want to recover metric $\hat{\phi}$ such that $||\phi - \hat{\phi}|| < \kappa$ under suitable norm for sufficiently small error tolerance.

Performance Metrics

Diagonal Linear Performance Metric (DLPM)

Family denoted by φ_{DLPM}. Given $\vec{a} \in \mathbb{R}^k$ such that $||\vec{a}||_1 = 1$, metric is $\psi(\vec{d}) :=< \vec{a}, \vec{d} >$

- Also known as weighted accuracy

Linear Performance Metric (LPM)

Family denoted by φ_{LPM}. Given $\vec{a} \in \mathbb{R}^q$ such that $||\vec{a}||_2 = 1$, metric is $\phi(\vec{c}) :=< \vec{a}, \vec{c} >$

Diagonal Linear Performance Metric (DLPM)

Family denoted by φ_{DLPM}. Given $\vec{a} \in \mathbb{R}^k$ such that $||\vec{a}||_1 = 1$, metric is $\psi(\vec{d}) :=< \vec{a}, \vec{d} >$

- Also known as weighted accuracy
- Focus on eliciting monotonically increasing DLPMs

Linear Performance Metric (LPM)

Family denoted by φ_{LPM}. Given $\vec{a} \in \mathbb{R}^q$ such that $||\vec{a}||_2 = 1$, metric is $\phi(\vec{c}) :=< \vec{a}, \vec{c} >$

- Focus on eliciting monotonically decreasing LPMs

Bayes Optimal Confusions and Classifiers

BO Confusion $\bar{c}$ over subset $\mathcal{S} \subseteq \mathcal{C}$

$\bar{c} := \text{argmax}_{\vec{c} \in \mathcal{S}} \phi(\vec{c})$

- analogous definition for diagonal confusions

Restricted Bayes Optimal (RBO) diagonal Confusion $\bar{d}_{k_1,k_2}$

$\bar{d}_{k_1,k_2} := \text{argmax}_{\vec{d} \in \mathcal{D}_{k_1,k_2}} \psi(\vec{d})$

- Setting where classifiers are restricted to only predict classes k_1 and k_2

Topic

Geometry of Diagonal Confusions

Vectors of trivial classifiers

$\vec{v}_i \in \mathbb{R}^k$ for $i \in [k]$ are vectors with ξ_i at the i-th index and zero elsewhere

Geometry of Diagonal Confusions

Vectors of trivial classifiers

$\vec{v}_i \in \mathbb{R}^k$ for $i \in [k]$ are vectors with ξ_i at the i-th index and zero elsewhere

- Represent diagonal confusions when only predicting class i

Geometry of Diagonal Confusions

Vectors of trivial classifiers

$\vec{v}_i \in \mathbb{R}^k$ for $i \in [k]$ are vectors with ξ_i at the i-th index and zero elsewhere

- Represent diagonal confusions when only predicting class i

Geometry of $\mathcal{D}$

Strictly convex, closed, and contained in the box $[0, \xi_1] \times \cdots \times [0, \xi_k]$

- $\vec{v}_i$ are the only vertices
- For any $k_1, k_2 \in [k]$, the 2D axis-aligned face of $\mathcal{D}$ is $\mathcal{D}_{k_1,k_2}$, equivalent to binary classification confusion matrices

Boundary of Diagonal Confusions

RBO classifier

Let $\psi \in \varphi_{DLPM}$ be parameterized by $\vec{a}$, then

$$\bar{h}_{k_1,k_2}(\vec{x}) = \begin{cases} k_1, & \text{if } a_{k_1}\eta_{k_1}(\vec{x}) \geq a_{k_2}\eta_{k_2}(\vec{x}) \\ k_2, & \text{o.w.} \end{cases} \tag{2}$$

is the RBO classifier with respect to ψ.

Upper boundary of $\mathcal{D}_{k_1,k_2}$, $\partial\mathcal{D}^{+}_{k_1,k_2}$

The RBO diagonal confusions confined to classes k_1 and k_2 for monotonically increasing DLPMs, such that $a_{k_1} + a_{k_2} > 0$.

- Parameterize by choosing $m \in [0, 1]$ and constructing DLPM as $a_{k_1} = m, a_{k_2} = 1 - m$
  - Parameterization denoted as $\nu(m; k_1, k_2)$

DLPM Elicitation

Main idea: For a metric ψ (quasiconcave and monotone increasing), and parameterization ρ^+ of upper boundary, composition $\psi \circ \rho^+$ is concave and unimodal on [0, 1].

- Therefore, binary-search type algorithm is possible!
  - Can estimate a_i^*/a_1^* for each i independently using binary search.

DLPM Elicitation Algorithm

Algorithm 1: DLPM Elicitation

Input: $\epsilon > 0$, oracle Ω, $\hat{a}_1 = 1$

For $i = 2, \cdots, k$ **do**

Initialize: $m^a = 0, m^b = 1.$

While $\left| m^b - m^a \right| > \epsilon$ **do**

- Set $m^c = \frac{3m^a + m^b}{4}$, $m^d = \frac{m^a + m^b}{2}$, and $m^e = \frac{m^a + 3m^b}{4}$.
- Set $\overline{\mathbf{d}}^a_{1,i} = \nu(m^a; 1, i)$ (i.e. parametrization of $\partial\mathcal{D}^+_{1,i}$ in Section 3.1). Similarly, set $\overline{\mathbf{d}}^c_{1,i}, \overline{\mathbf{d}}^d_{1,i}, \overline{\mathbf{d}}^e_{1,i}, \overline{\mathbf{d}}^b_{1,i}$.
- Query $\Omega(\overline{\mathbf{d}}^c_{1,i}, \overline{\mathbf{d}}^a_{1,i}), \Omega(\overline{\mathbf{d}}^d_{1,i}, \overline{\mathbf{d}}^c_{1,i})$, $\Omega(\overline{\mathbf{d}}^e_{1,i}, \overline{\mathbf{d}}^d_{1,i})$, and $\Omega(\overline{\mathbf{d}}^b_{1,i}, \overline{\mathbf{d}}^e_{1,i})$.
- $[m^a, m^b] \leftarrow$ *ShrinkInterval-1* (responses).

Set $m^d = \frac{m^a + m^b}{2}$. Then set $\hat{a}_i = \frac{1 - m^d}{m^d}\hat{a}_1$.

Output: $\hat{\mathbf{a}} = \left(\frac{\hat{a}_1}{\|\hat{\mathbf{a}}\|_1}, \cdots, \frac{\hat{a}_k}{\|\hat{\mathbf{a}}\|_1} \right)$.

Topic

Geometry of Off-Diagonal Confusions

Vectors of trivial classifiers

$\vec{u}_i \in \mathcal{C}$ for $i \in [k]$ represent diagonal confusions when only predicting class i

Geometry of $\mathcal{C}$

Convex (not strictly!), and contained in box $[0, \xi_1]^{(k-1)} \times \cdots \times [0, \xi_k]^{(k-1)}$

- $\{\vec{u}_i\}_{i=1}^k$ belongs to the set of vertices of $\mathcal{C}$.
- $\mathcal{C}$ always contains $\vec{o} = \frac{1}{k} \sum_{i=1}^k \vec{u}_i$: off-diagonal confusions for trivial classifier that randomly predicts each class with equal probability on the entire space $\mathcal{X}$.

Assumption: existence of sphere $S_\lambda \subset \mathcal{C}$ centered at $\vec{o}$

Essentially assumes that there is some signal for non-trivial classification

- Ensures that unique optimal off-diagonal confusion $\bar{c}$ over sphere S_λ is point on boundary of S_λ given by $\bar{c} = \lambda \vec{a} + \vec{o}$.

Boundary of Off-Diagonal Confusions

Lower boundary of S_λ, ∂S_λ^-

Set of optimal off-diagonal confusions over sphere S_λ for LPMs with $a_i \leq 0$ (monotonically decreasing condition)

Parameterizing ∂S_λ^-

Standard method of sphere parameterization by angles.

- $\vec{\theta}$ is (q-1) dimensional vector where primary angle is in the third quadrant and all others are in the second quadrant
  - Choice of quadrant ensures monotonically decreasing condition
- LMP construction:
  - for $i \in [q-1]$ set $a_i = \prod_{j=1}^{i-1} \sin(\theta_j)\cos(\theta_i)$
  - $a_q = \prod_{j=1}^{q-1} \sin(\theta_j)$

Main idea: $\partial\mathcal{C}$ may have flat regions (not strictly convex), but we can instead use query space from sphere $S_\lambda \subset \mathcal{C}$

- Can do coordinate-wise binary-search
  - Update single angle, keeping all others fixed using binary search
  - Convergence is assured via a dual interpretation: minimizing smooth, strongly convex function

LPM Elicitation Algorithm

Algorithm 2: LPM Elicitation

Input: $\epsilon > 0$, oracle Ω, λ, and $\boldsymbol{\theta} = \boldsymbol{\theta}^{(1)}$

For $t = 1, 2, \cdots, T$ **do**

Set $\boldsymbol{\theta}^a = \boldsymbol{\theta}^c = \boldsymbol{\theta}^d = \boldsymbol{\theta}^e = \boldsymbol{\theta}^b = \boldsymbol{\theta}^{(t)}$.

if $(t\%(q-1))$ Set $j = t\%(q-1)$; **else** $j = q-1$.

if $(j == q-1)$ **Initialize:** $\theta_j^a = \pi$, $\theta_j^b = 3\pi/2$.

else Initialize: $\theta_j^a = \pi/2$, $\theta_j^b = \pi$.

While $\left|\theta_j^b - \theta_j^a\right| > \epsilon$ **do**

- Set $\theta_j^c = \frac{3\theta_j^a + \theta_j^b}{4}$, $\theta_j^d = \frac{\theta_j^a + \theta_j^b}{2}$, and $\theta_j^e = \frac{\theta_j^a + 3\theta_j^b}{4}$.
- Set $\bar{\mathbf{c}}^a = \mu(\boldsymbol{\theta}^a)$ (i.e. parametrization of $\partial\mathcal{S}_\lambda^-$ in Section 3.2). Similarly, set $\bar{\mathbf{c}}^c, \bar{\mathbf{c}}^d, \bar{\mathbf{c}}^e, \bar{\mathbf{c}}^b$.
- Query $\Omega(\bar{\mathbf{c}}^c, \bar{\mathbf{c}}^a), \Omega(\bar{\mathbf{c}}^d, \bar{\mathbf{c}}^c), \Omega(\bar{\mathbf{c}}^e, \bar{\mathbf{c}}^d), \Omega(\bar{\mathbf{c}}^b, \bar{\mathbf{c}}^e)$
- $[\theta_j^a, \theta_j^b] \leftarrow$ *ShrinkInterval-2* (responses).

Set $\theta_j^d = \frac{1}{2}(\theta_j^a + \theta_j^b)$ and then set $\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^d$.

Output: $\hat{a}_i = \Pi_{j=1}^{i-1} \sin\theta_j^{(T)} \cos\theta_i{}^{(T)}\ \forall i \in [q-1]$,

$\hat{a}_q = \Pi_{j=1}^{q-1} \sin\theta_j^{(T)}$.

Topic

Guarantees

Robust under Oracle Feedback Noise ($\epsilon_\Omega \geq 0$): Oracle responds correctly as long as $|\phi(\vec{c}) - \phi(\vec{c'})| > \epsilon_\Omega$

DLPM

After $O((k-1)\log(\frac{1}{\epsilon}))$ queries to oracle, $||a^* - \hat{a}||_\infty \leq O(\epsilon + \sqrt{\epsilon_\Omega})$

- Equivalent to $||a^* - \hat{a}||_2 \leq O(\sqrt{k}(\epsilon + \sqrt{\epsilon_\Omega}))$

LPM

Assuming $\lambda >> \epsilon_\Omega$, after $O(z_1 \log(z_2/(q\epsilon^2))(q-1)log(\frac{\pi}{2\epsilon}))$ queries to the oracle, $||a^* - \hat{a}||_2 \leq O(\sqrt{q}(\epsilon + \sqrt{\epsilon_\Omega/\lambda}))$, where z_1 and z_2 are constants independent of ϵ and q

Experiments

Real World data from SensIT dataset (78823 instances, 3 classes) and Vehicle dataset (846 instances, 4 classes), and tested on datasets with 50% and 75% of datapoints

- Standardize features and split each dataset into 2 parts: S_1, S_2
  - S_1 used to learn $\{\hat{\eta}_i(x)\}_{i=1}^{k}$ using regularized softmax regression model
  - S_2 used for making prediction and computing sample confusion

Randomly select 100 DLPMs, and use $\epsilon = 0.01$ to recover estimates.

- Check proportion of times $||a^* - \hat{a}||_\infty \leq \omega$ for different values of ω

Experiments

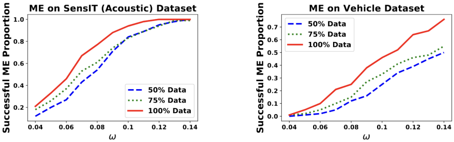


Figure 3: DLPM elicitation on real data for $\epsilon = 0.01$. For randomly chosen hundred $\mathbf{a}^*$, we show the proportion of times our estimates $\hat{\mathbf{a}}$ obtained with $4(k-1)\lceil\log(1/\epsilon)\rceil$ queries satisfy $\|\mathbf{a}^* - \hat{\mathbf{a}}\|_\infty \leq \omega$.