Master Program in *Artificial Intelligent Systems*

Advanced Topics in Machine Learning

Machine Learning Models That Know What They Do Not Know

Andrea Pugnana

DISI
University of Trento, Italy
andrea.pugnana@unitn.it

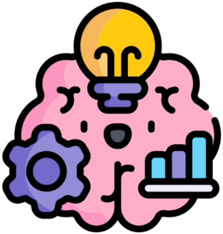
Agenda

- Motivation
- Three valuable options
  - Abstain (I do not know)
  - Defer (You know better)
  - Inform (I am not confident)
- Wrap up

Why Do We Care?

- Machine Learning (ML) models can make mistakes
- Mistakes can be costly
- How can we reduce mistakes?

A healthcare application



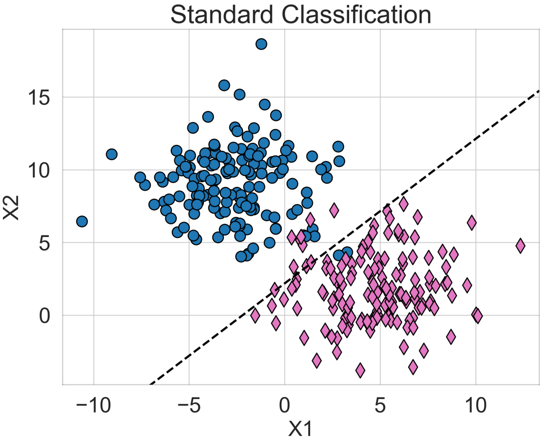
- A ML model always predicts!

A healthcare application

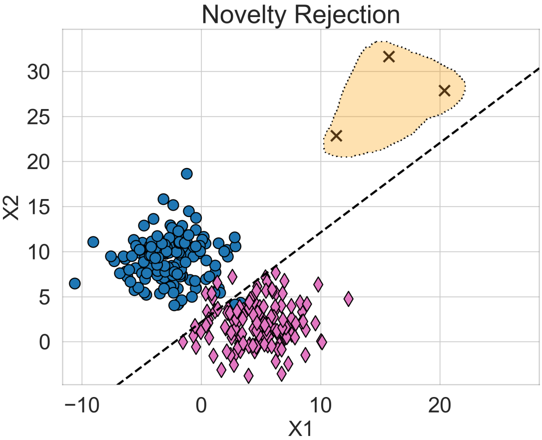
- A ML model always predicts!
- A doctor can abstain!

Abstaining Systems: “I do not know”

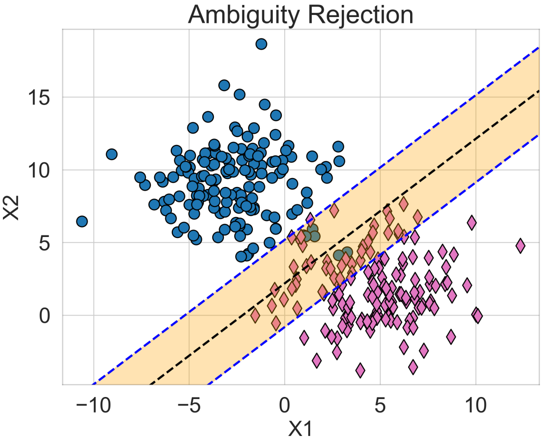
Canonical classifier



Have you ever seen such an instance?



Are you really sure about it?



Abstaining Systems

- Predictor $f : \mathcal{X} \to \mathcal{Y} \cup \{\text{abstain}\}$
- Two conditions for abstention [Hendrickx et al., 2024]
  - Ambiguity Rejection (are you really sure?)
  - Novelty Rejection (have you seen it?)

Ambiguity Rejection

- Abstaining on instances close to the decision boundary
- Two main frameworks [Ruggieri and Pugnana, 2025]:
    - **Selective Prediction**
    - Learning to Reject

Selective Prediction [El-Yaniv and Wiener, 2010]

- Predictor $f : \mathcal{X} \to \mathcal{Y}$
- Selection function $g : \mathcal{X} \to \{0, 1\}$
  - decide whether to accept ($g(\mathbf{x}) = 1$) or to reject ($g(\mathbf{x}) = 0$)
- Selective predictor is the pair

$$(f, g)(\mathbf{x}) = \begin{cases} f(\mathbf{x}) & \text{if } g(\mathbf{x}) = 1 \\ \text{abstain} & \text{otherwise.} \end{cases} \tag{1}$$

Selective Prediction

- Natural trade-off between the fraction of accepted instances and the risk over accepted instances:

Selective Prediction

- Natural trade-off between the fraction of accepted instances and the risk over accepted instances:
  - *Coverage*

$$\phi(g) = \mathbb{E}[g(\mathbf{x})]$$

Selective Prediction

- Natural trade-off between the fraction of accepted instances and the risk over accepted instances:
  - *Coverage*
  $$\phi(g) = \mathbb{E}[g(\mathbf{x})]$$
  - *Selective Risk*
  $$R(f, g) = \frac{\mathbb{E}[l(f(\mathbf{x}), y)g(\mathbf{x})]}{\phi(g)},$$

  where $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_0^+$ is some loss function.

The Selective Prediction Framework

Two problem formulations [Franc et al., 2023]:

The Selective Prediction Framework

Two problem formulations [Franc et al., 2023]:

- *Bounded-abstention* problem:

$$\arg\min_{\theta,\psi} R(f_\theta, g_\psi) \quad \text{s.t.} \quad \phi(g_\psi) \geq c \tag{2}$$

  given c *target coverage*

The Selective Prediction Framework

Two problem formulations [Franc et al., 2023]:

- *Bounded-abstention* problem:

$$\arg\min_{\theta,\psi} R(f_\theta, g_\psi) \quad \text{s.t.} \quad \phi(g_\psi) \geq c \tag{2}$$

  given c *target coverage*

- *Bounded-improvement* problem:

$$\arg\max_{\theta,\psi} \phi(g_\psi) \quad \text{s.t.} \quad R(f_\theta, g_\psi) \leq \epsilon \tag{3}$$

  given ϵ *target risk*

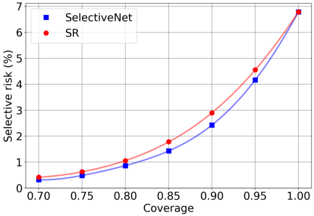
The Selective Prediction Framework

Two problem formulations [Franc et al., 2023]:

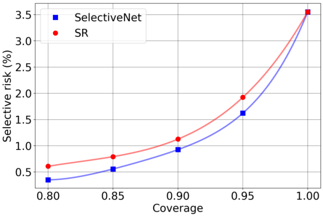
- *Bounded-abstention* problem:

$$\arg\min_{\theta,\psi} R(f_\theta, g_\psi) \quad \text{s.t.} \quad \phi(g_\psi) \geq c \tag{4}$$

given c *target coverage*

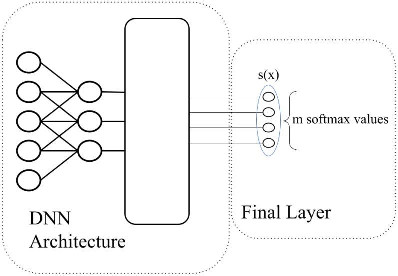


(a) Cifar-10



(b) Cats vs. dogs

Selection functions: Score-based



- **Intuition**: use functions of the final scores

Selection functions: Score-based

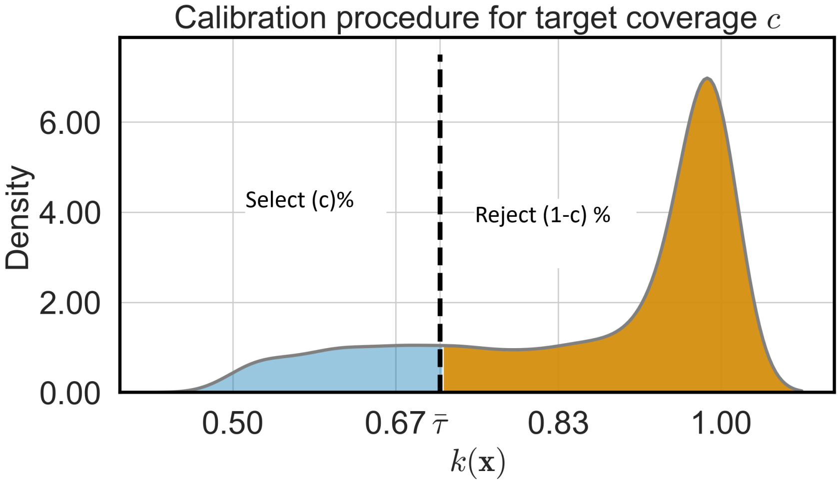
- **Example:** standard Neural Network (with parameters θ) $f : \mathcal{X} \to \mathcal{Y}$

$$f(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} s_y(\mathbf{x}),$$

  where $s_y(x)$ are the softmax values of the final logits
- Use $k(\mathbf{x}) = 1 - \max_y s_y(x)$ [Geifman and El-Yaniv, 2017] as "confidence"
- The selection function becomes

$$g(\mathbf{x}) = \begin{cases} 0 \text{ if } k(\mathbf{x}) > \tau \\ 1 \text{ if } k(\mathbf{x}) \leq \tau \end{cases}$$

The calibration procedure for coverage c



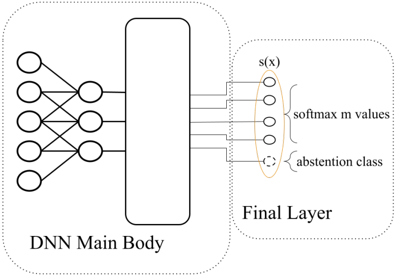
Selection functions: Score-based

- **Intuition:** Binary case
- Model very uncertain on labels will output $s_0(\mathbf{x}) \approx .50, s_1(\mathbf{x}) \approx .50$
- So $k(\mathbf{x}) \approx .50$

Selection functions: Score-based

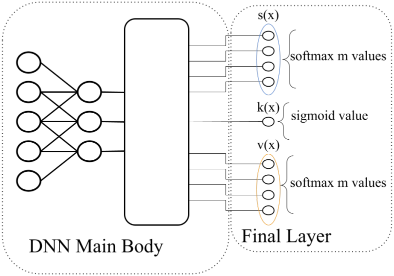
- **Intuition:** Binary case
- Model very uncertain on labels will output $s_0(\mathbf{x}) \approx .50, s_1(\mathbf{x}) \approx .50$
- So $k(\mathbf{x}) \approx .50$
- Instead if model very confident in one of its own prediction, $k(\mathbf{x}) << .50$, hence we will predict

Selection functions: Learning to Abstain



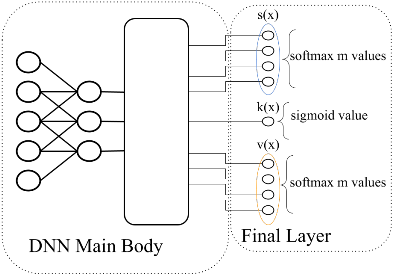
- **Intuition:** add a class representing abstention
- Use specific losses to train a model with the extra class, e.g., Liu et al. [2019], Huang et al. [2020]
- Use functions of the extra class logit as a measure for confidence

Selection functions: Learning to Select



- **Intuition:** use extra head to weight instances according to their correctness;
- Directly model the coverage constraint in the loss
- Example: SelNet by Geifman and El-Yaniv [2019]

SelNet Loss



- SelNet minimizes the following loss:

$$\mathcal{L}(s,k,v,\mathbf{x},y,\alpha) = \alpha \overbrace{\left((l(s(\mathbf{x}),y)k(\mathbf{x})) + \lambda(\max\{0,(c-k(\mathbf{x})\})^2\right)}^{\text{Main loss that weights were the network makes mistakes}} + \underbrace{(1-\alpha)l(v(\mathbf{x}),y)}_{\text{Auxiliary loss}} \quad (5)$$

Ambiguity Rejection

- Abstaining on instances close to the decision boundary
- Two main frameworks
  - Selective Prediction
  - **Learning to Reject**

The Learning to Reject Framework [Chow, 1970]

- Abstention cost, rather than target coverage/risk parameter
- Risk:

$$R(f, g, a) = \mathbb{E}[l\,(f(\mathbf{x}), y)g(\mathbf{x}) + a\,(1 - g(\mathbf{x}))]$$

  where a is the *cost of abstention*.
- LtR problem formulation:

$$\arg\min_{\theta,\psi} R(f_\theta, g_\psi, a) \tag{6}$$

Novelty Rejection [Cordella et al., 1995]

- Abstaining on instances far away from the training distribution
- Selection function $g : \mathcal{X} \to \{0, 1\}$
- Out-of-distribution ($g(\mathbf{x}) = 0$) or in-sample ($g(\mathbf{x}) = 1$)

Novelty Rejection

- Estimate how likely an instance belong to the training distribution
- $\hat{F}(\mathbf{x}) \approx P(\mathbf{X} > \mathbf{x})$ (marginal density estimator)
- $g(\mathbf{x}) = \mathbb{I}_{\{\hat{F}(\mathbf{x}) > \tau\}}$

Novelty Rejection

- Estimate how likely an instance belong to the training distribution
- $\hat{F}(\mathbf{x}) \approx P(\mathbf{X} > \mathbf{x})$ (marginal density estimator)
- $g(\mathbf{x}) = \mathbb{I}_{\{\hat{F}(\mathbf{x}) > \tau\}}$
- To estimate $\hat{F}$: e.g.,

Novelty Rejection

- Estimate how likely an instance belong to the training distribution
- $\hat{F}(\mathbf{x}) \approx P(\mathbf{X} > \mathbf{x})$ (marginal density estimator)
- $g(\mathbf{x}) = \mathbb{I}_{\{\hat{F}(\mathbf{x}) > \tau\}}$
- To estimate $\hat{F}$: e.g.,
  - Gaussian Mixtures Models (GMM) [Landgrebe et al., 2004],

Novelty Rejection

- Estimate how likely an instance belong to the training distribution
- $\hat{F}(\mathbf{x}) \approx P(\mathbf{X} > \mathbf{x})$ (marginal density estimator)
- $g(\mathbf{x}) = \mathbb{I}_{\{\hat{F}(\mathbf{x}) > \tau\}}$
- To estimate $\hat{F}$: e.g.,
  - Gaussian Mixtures Models (GMM) [Landgrebe et al., 2004],
  - Normalizing Flows [Nalisnick et al., 2019],

Novelty Rejection

- Estimate how likely an instance belong to the training distribution
- $\hat{F}(\mathbf{x}) \approx P(\mathbf{X} > \mathbf{x})$ (marginal density estimator)
- $g(\mathbf{x}) = \mathbb{I}_{\{\hat{F}(\mathbf{x}) > \tau\}}$
- To estimate $\hat{F}$: e.g.,
  - Gaussian Mixtures Models (GMM) [Landgrebe et al., 2004],
  - Normalizing Flows [Nalisnick et al., 2019],
  - Variational Autoencoders [Wang and Yiu, 2020]

Abstaining Systems: further directions

- Non-distributive loss functions [Pugnana and Ruggieri, 2023];
- Benchmarking current approaches [Pugnana et al., 2024];
- Novelty and ambiguity at the same time [Narasimhan et al., 2024, Franc et al., 2024];

Deferring Systems: “You know better”

Learning to Defer Framework

- ML Predictor $f : \mathcal{X} \rightarrow \mathcal{Y}$
- Human Expert h
- Deferring System (Human-AI Team):

$$(f, h, g) = \begin{cases} h(\mathbf{x}) & \text{if} \quad g(\mathbf{x}) = 1 \\ f(\mathbf{x}) & \text{if} \quad g(\mathbf{x}) = 0 \end{cases} \tag{7}$$

Learning to Defer Framework [Madras et al., 2018]

- The LtD Risk can generally be written as:

$$\mathcal{E}_{0-1}(f, g, h) = \mathbb{E}_{(\mathbf{x},y,h)\sim P}[\mathbb{I}_{f(\mathbf{x})\neq y}\ (1 - g(\mathbf{x})) + \mathbb{I}_{h\neq y} g(\mathbf{x})]$$

where we consider the zero-one loss for both human and

- Problem formulation:

$$\arg\min_{\theta,\phi} \mathcal{E}_{0-1}(f_\theta, g_\phi, h) \ \text{ s.t. } \ \mathbb{E}[g(\mathbf{x})] \leq (1 - c) \tag{8}$$

given c *target coverage* of the ML predictor

- Intuition: learn to defer to the human only those cases where the human is better than the ML predictor!

Main Concern

- The loss is not directly tractable;

Main Concern

- The loss is not directly tractable;
- Instead of using $\mathcal{E}_{0-1}$ use some "surrogate" loss ℓ such that:
  - $\ell(f(\mathbf{x}), g(\mathbf{x}), h, y)$ substitutes $\mathbb{I}_{f(\mathbf{x}) \neq y}\ (1 - g(\mathbf{x})) + \mathbb{I}_{h \neq y} g(\mathbf{x})$

Main Concern

- The loss is not directly tractable;
- Instead of using $\mathcal{E}_{0-1}$ use some "surrogate" loss ℓ such that:
  - $\ell(f(\mathbf{x}), g(\mathbf{x}), h, y)$ substitutes $\mathbb{I}_{f(\mathbf{x})\neq y}\,(1 - g(\mathbf{x})) + \mathbb{I}_{h\neq y} g(\mathbf{x})$
- Score-model approach: use a single model $f : \mathcal{X} \to \bar{\mathcal{Y}}$, where $\bar{\mathcal{Y}} = \mathcal{Y} \cup \{\perp\}$

Main Concern

- The loss is not directly tractable;
- Instead of using $\mathcal{E}_{0-1}$ use some "surrogate" loss ℓ such that:
  - $\ell(f(\mathbf{x}), g(\mathbf{x}), h, y)$ substitutes $\mathbb{I}_{f(\mathbf{x}) \neq y}\, (1 - g(\mathbf{x})) + \mathbb{I}_{h \neq y} g(\mathbf{x})$
- Score-model approach: use a single model $f : \mathcal{X} \to \bar{\mathcal{Y}}$, where $\bar{\mathcal{Y}} = \mathcal{Y} \cup \{\perp\}$
- Guarantee some properties;

Bayes Consistency

Bayes-Consistency

A surrogate loss L_{surr} is Bayes-consistent for a loss L_{orig} if, minimising it over the entire hypothesis class $\mathcal{Q}$ (*e.g.*, f and s in LtD) guarantees the minimisation of the original intractable loss function L_{orig} over the same class [Mao et al., 2025], i.e.:

$$\lim_{n\to\infty} \mathcal{E}_{L_{surr}}(q_n) - \mathcal{E}^*_{L_{surr}} = 0 \implies \lim_{n\to\infty} \mathcal{E}_{L_{orig}}(q_n) - \mathcal{E}^*_{L_{orig}} = 0, \tag{9}$$

where q_n is a sequence of hypotheses, each learned from n samples.

- **Intuition:** if I minimize this loss over **all measurable** functions, with enough data, I am sure that I also minimize the 0-1 loss

Realizable $\mathcal{Q}$-Consistency

Realizable -$\mathcal{Q}$ consistency

A surrogate loss is *relizable $\mathcal{Q}$-consistent* if, whenever there exists a hypothesis $q^* \in \mathcal{Q}$ such that $\mathcal{E}_{L_{orig}}(q^*) = 0$, we have that if $q' = \arg\min_{q \in \mathcal{Q}} \mathcal{E}_{L_{surr}}(q)$, then $\mathcal{E}_{L_{surr}}(q') = 0$.

- **Intuition:** if a zero error predictor exists, when I minimize this loss over **my chosen** $\mathcal{Q}$ set of functions, with enough data, I am sure that I reach that zero error solution

Example of surrogate loss

- Consider a predictor $f : \mathcal{X} \to \mathcal{Y} \cup \perp$
- Use the following loss:

$$\ell_{RS}(f, g, \mathbf{x}, y, h) = -c(\mathbf{x}, y) \log \left(\frac{\exp \tilde{f}_y(\mathbf{x})}{\sum_{y' \in \mathcal{Y} \cup \{\perp\}} \exp \tilde{f}'_y(\mathbf{x})} \right) + \\ (c(\mathbf{x}, y) - 1) \log \left(\frac{\exp \tilde{f}_y(\mathbf{x}) + \exp \tilde{f}_\perp(\mathbf{x})}{\sum_{y' \in \mathcal{Y} \cup \{\perp\}} \exp \tilde{f}'_y(\mathbf{x})} \right),$$

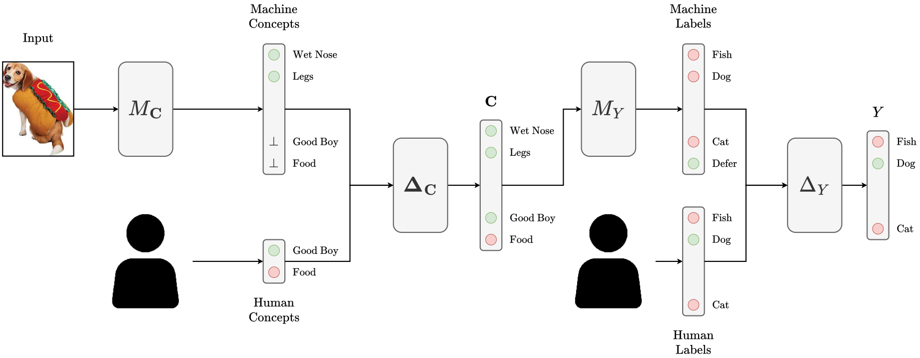
  where $c(\mathbf{x}, y) \in [0, 1]$ is a cost associated with querying the human
- Mao et al. [2024] show it is both Bayes-consistent and Realizable consistent

Further Directions

- New surrogate losses;
- Multi-expert and multi-task;
- How to evaluate;
- Ethical aspects and user studies

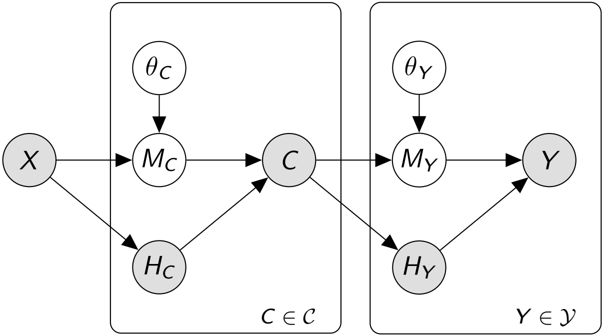
Bonus Track: Deferring Concept Bottleneck Models

Deferring Concept Bottleneck Models (DCBMs) [Pugnana et al., 2025] consider each concept and task predictor as a deferring system.



DCBMs are Probabilistic Models

We formalize DCBMs as **graphical probabilistic models** and train them by maximizing the **likelihood** of latent parameters Θ.



Training DCBMs

Some of our favourite properties on the Maximum Likelihood of DCBMs 🥳

Training DCBMs

Some of our favourite properties on the Maximum Likelihood of DCBMs 🥳

- It generalizes existing L2D losses to multi-variate,

Training DCBMs

Some of our favourite properties on the Maximum Likelihood of DCBMs 🥳

- It generalizes existing L2D losses to multi-variate,
- It allows for independent cost factors,

Training DCBMs

Some of our favourite properties on the Maximum Likelihood of DCBMs 🥳

- It generalizes existing L2D losses to multi-variate,
- It allows for independent cost factors,
- It is consistent with the intractable zero-one loss under the concept-independence assumption.

Uncertainty-aware Systems: “I am not confident”

Uncertainty Estimation [Hüllermeier and Waegeman, 2021]

- Abstaining or deferring not always possible
  - e.g., time-critical or mission-critical decisions
- Two approaches:
  - Set-valued Predictions: provide multiple predictions with some guarantees
  - Uncertainty Quantification: enrich the prediction with additional information

The Conformal Prediction Framework [Vovk et al., 2005, Angelopoulos and Bates, 2023]

- From Predictor $f : \mathcal{X} \to \mathcal{Y}$
- To Set-valued Predictor $\mathcal{C} : \mathcal{X} \to 2^{\mathcal{Y}}$
- For a given α, train $\mathcal{C}$ such that

$$P\left(Y \in \mathcal{C}(\mathbf{x})\right) \geq 1 - \alpha \tag{10}$$

Conformal Coverage Guarantee [Vovk et al., 2005, Angelopoulos and Bates, 2023]

- Conformal score $z : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$
  - larger scores encode worse agreement
  - e.g., $z(\mathbf{x}, y) = \sum_{i=1}^{k} s(\mathbf{x})_{\pi_i(\mathbf{x})}$
    where π is descending order, and $y = \pi_k(\mathbf{x})$
- Estimate distribution of s over (held-out) sample
- Compute $(1 - \alpha)$-quantile $\hat{q}$
- Build $\mathcal{C}(\mathbf{x})$ by including all the y's such that $s(\mathbf{x}, y) \geq \hat{q}$

Uncertainty Quantification

- Intuition: enrich the prediction with additional information
- Example 1: regression with prediction intervals
- Example 2: classification with probabilistic models

Uncertainty Quantification

- Calibration of probabilistic scores [Silva Filho et al., 2023]
- Most-methods do not take into account uncertainty naturally
- Frequentist vs Bayesian
- Model-agnostic vs Model-specific

Further Directions

- Uncertainty in Large Language Models [Yin et al., 2023]
- Cognitively-robust communication [Kompa et al., 2021]
- Explanation of uncertainty [Zukerman and Maruf, 2024]

Conclusions & Contacts

- Knowing what one does not know is a form of intelligence [Grant, 2023]
- Towards ML models with the ability to know what they do not know
- Many open challenges for further research
- mail: andrea[dot]pugnana[at]unitn.it
- X: @andrepugni

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Found. Trends Mach. Learn.*, 16(4):494–591, 2023.

C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Trans. Inf. Theory*, 16(1):41–46, 1970.

Luigi P. Cordella, Claudio De Stefano, Carlo Sansone, and Mario Vento. An adaptive reject option for LVQ classifiers. In *ICIAP*, volume 974 of *Lecture Notes in Computer Science*, pages 68–73. Springer, 1995.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *J. Mach. Learn. Res.*, 11:1605–1641, 2010.

Vojtech Franc, Daniel Průša, and Václav Vorácek. Optimal strategies for reject option classifiers. *J. Mach. Learn. Res.*, 24:11:1–11:49, 2023.

Vojtech Franc, Jakub Paplhám, and Daniel Prruvsa. SCOD: from heuristics to theory. In *ECCV (84)*, volume 15142 of *Lecture Notes in Computer Science*, pages 424–441. Springer, 2024.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *NIPS*, pages 4878–4887, 2017.

Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 2151–2159. PMLR, 2019.

Adam Grant. *Think Again: The Power of Knowing What You Don't Know*. Penguin Publishing Group, 2023.

Kilian Hendrickx, Lorenzo Perini, Dries Van der Plas, Wannes Meert, and Jesse Davis. Machine learning with a reject option: a survey. *Mach. Learn.*, 113(5):3073–3110, 2024.

Lang Huang, Chao Zhang, and Hongyang Zhang. Self-adaptive training: beyond empirical risk minimization. In *NeurIPS*, 2020.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.*, 110(3):457–506, 2021.

Benjamin Kompa, Jasper Snoek, and Andrew L. Beam. Second opinion needed: communicating uncertainty in medical machine learning. *npj Digit. Medicine*, 4, 2021.

Thomas C.W . Landgrebe, David M. J. Tax, Pavel Paclík, Robert P. W. Duin, and Andrew Colin. A combining strategy for ill-defined problems. In *Fifteenth Annual Symposium of the Pattern Recognition Association of South Africa*, pages 57–62, 2004.

Ziyin Liu, Zhikang Wang, Paul Pu Liang, Ruslan Salakhutdinov, Louis-Philippe Morency, and Masahito Ueda. Deep gamblers: Learning to abstain with portfolio theory. In *NeurIPS*, pages 10622–10632, 2019.

David Madras, Toniann Pitassi, and Richard S. Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *NeurIPS*, pages 6150–6160, 2018.

Anqi Mao, Mehryar Mohri, and Yutao Zhong. Realizable h-consistent and bayes-consistent loss functions for learning to defer. In *NeurIPS*, 2024.

Anqi Mao, Mehryar Mohri, and Yutao Zhong. Enhanced h-consistency bounds. In *ALT*, volume 272 of *Proceedings of Machine Learning Research*, pages 772–813. PMLR, 2025.

Eric T. Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Görür, and Balaji Lakshminarayanan. Hybrid models with deep and invertible features. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 4723–4732. PMLR, 2019.

Harikrishna Narasimhan, Aditya Krishna Menon, Wittawat Jitkrittum, and Sanjiv Kumar. Plugin estimators for selective classification with out-of-distribution detection. In *ICLR*. OpenReview.net, 2024.

Andrea Pugnana and Salvatore Ruggieri. AUC-based selective classification. In *AISTATS*, volume 206 of *Proceedings of Machine Learning Research*, pages 2494–2514. PMLR, 2023.

Andrea Pugnana, Lorenzo Perini, Jesse Davis, and Salvatore Ruggieri. Deep neural network benchmarks for selective classification. *J. Data-centric Mach. Learn. Res.*, 1 (17):1–58, 2024.

Andrea Pugnana, Riccardo Massidda, Francesco Giannini, Pietro Barbiero, Mateo Espinosa Zarlenga, Roberto Pellungrini, Gabriele Dominici, Fosca Giannotti, and Davide Bacciu. Deferring concept bottleneck models: Learning to defer interventions to inaccurate experts. In *NeurIPS*, 2025.

Salvatore Ruggieri and Andrea Pugnana. Things machine learning models know that they don't know. In *AAAI*, pages 28684–28693. AAAI Press, 2025.

Telmo de Menezes Silva Filho, Hao Song, Miquel Perelló-Nieto, Raúl Santos-Rodríguez, Meelis Kull, and Peter A. Flach. Classifier calibration: a survey on how to assess and improve predicted class probabilities. *Mach. Learn.*, 112(9):3211–3260, 2023.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005.

Xin Wang and Siu-Ming Yiu. Classification with rejection: Scaling generative classifiers with supervised deep infomax. In *IJCAI*, pages 2980–2986. ijcai.org, 2020.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large language models know what they don't know? In *ACL (Findings)*, pages 8653–8665. Association for Computational Linguistics, 2023.

Ingrid Zukerman and Sameen Maruf. Communicating uncertainty in explanations of the outcomes of machine learning models. In *INLG*, pages 30–46. Association for Computational Linguistics, 2024.