Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer

David Madras, Toniann Pitassi, Richard Zemel

VECTOR INSTITUTE | INSTITUT VECTEUR

Motivation

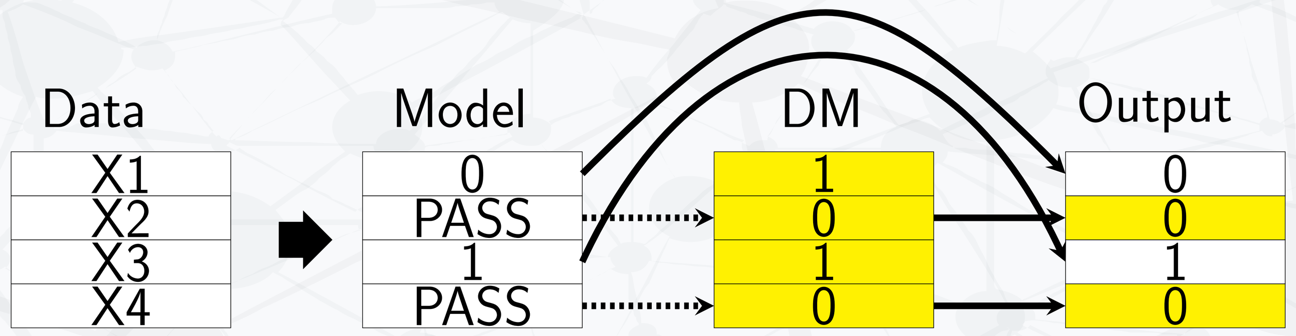
Can humans and machines make decisions jointly?

- Frequently, machine learning models are intended to be used as part of interactive systems, jointly with another decision-maker (DM)

Main Idea

We propose **learning to defer**, or **adaptive rejection learning**, which lets us optimize a model which will be used as one component of a larger system containing multiple decision-making agents.

A Joint Decision-Making Framework



- Real-world decision systems are interactive processes with many agents
- Our framework: ML model + decision-maker (DM) e.g. human user
- Two-stage decision cascade: model can PASS, in which case DM chooses final output
- In *rejection learning* (Chow, 1957; Cortes et al., 2016), we also have pass/reject option, but model is considered to be the final stage
- The purpose of PASSing can vary by application (culling a large pool, auditing DM, flagging cases for review, etc)

Example: Model is trained to detect melanoma, and if it PASSes, a human doctor can run an extra suite of medical tests. Model learns that it is very inaccurate at detecting amelanocytic (non-pigmented) melanoma, PASSes if this might be the case. However, if the doctor is even *less* accurate at detecting amelanocytic melanoma than the model is, we may prefer the model to make a prediction despite its uncertainty.

Model

Given data X, auxiliary data Z, and labels Y, define system output $\hat{Y}$, model predictions $\hat{Y}_M$, model PASS decisions s, and DM predictions $\hat{Y}_D$:

$$\hat{Y} = (1-s)\hat{Y}_M + s\hat{Y}_D$$

$$\hat{Y}_M = P_M(Y=1|X); \quad s = g_s(X); \quad \hat{Y}_D = P_D(Y=1|X,Z)$$

We describe the joint probability P_{defer} and negative log-likelihood $\mathcal{L}_{defer}$ of the system (with ℓ as example-wise cross-entropy):

$$P_{defer}(Y|X,Z) = \prod_i [\hat{Y}_{M,i}^{Y_i}(1-\hat{Y}_{M,i})^{1-Y_i}]^{1-s_i}[\hat{Y}_{D,i}^{Y_i}(1-\hat{Y}_{D,i})^{1-Y_i}]^{s_i}$$

$$\mathcal{L}_{defer}(Y,\hat{Y}_M,\hat{Y}_D,s) = -\sum_i [(1-s_i)\ell(Y_i,\hat{Y}_{M,i}) + s_i\ell(Y_i,\hat{Y}_{D,i})]$$

This is **learning to defer**. When optimizing $\mathcal{L}_{defer}$, we optimize the output of the *system as a whole*.

We can think of learning to defer as **adaptive rejection learning**. Rejection learning (Cortes et al., 2016) is equivalent to learning to defer to a DM with loss γ_{reject} on each example (e.g. an oracle for $\gamma_{reject} = 0$).

$$\mathcal{L}_{reject}(Y,\hat{Y}_M,s) = -\sum_i [(1-s_i)\ell(Y_i,\hat{Y}_{M,i}) + s_i\gamma_{reject}]$$

Fair Regularization: Suppose some sensitive attribute A (e.g. race). We want equalized odds ($\hat{Y} \perp A|Y$) (Hardt et al., 2016). We add a term $\alpha \cdot \mathcal{R}(Y,\hat{Y})$ ($\alpha \in \mathbb{R}$) to the loss $\mathcal{L}$:

$$\mathcal{R}(Y,\hat{Y}) = \frac{1}{2}\sum_{y=0,1} |\mathbb{E}(\hat{Y} \neq Y|A=0,Y=y) - \mathbb{E}(\hat{Y} \neq Y|A=1,Y=y)|$$

Learning Adaptively within Decision Systems

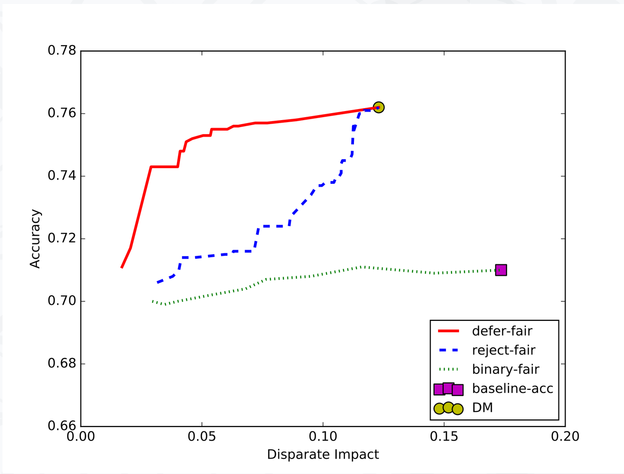
Idea: The system is a mixture-of-experts (Jacobs et al., 1991) between model and DM, with gating variable $s \sim Ber(\pi)$. We optimize $\hat{Y}_M, \pi$.

Post-hoc Thresholding ($\pi, \hat{Y}_M \in \{0,1\}$)**:** Here, $\pi = g_\pi(\hat{Y}_M) = g_\pi(f_M(X))$. Learn two thresholds t_0, t_1. Use trained classifier which outputs score β. If $t_0 < \beta < t_1$, set $\pi = 1$; else $\pi = 0$.

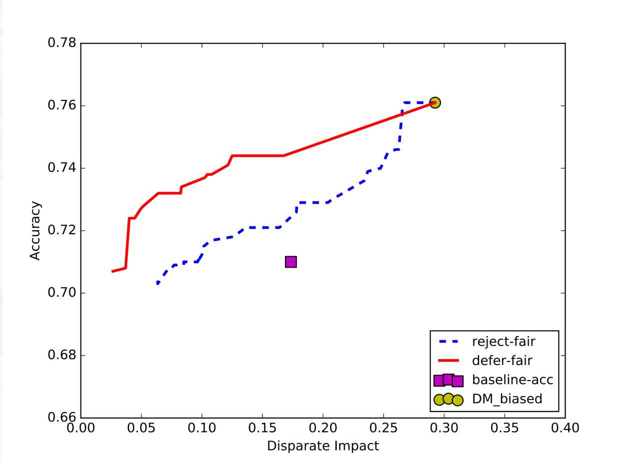
Differentiable Model ($\pi, \hat{Y}_M \in [0,1]$)**:** Here, $\pi = g_\pi(\hat{Y}_M, X) = g_\pi(f_M(X), X)$. More flexible; a DM's output may depend heterogenously on the data. Parametrize $\pi, \hat{Y}_M$ with neural networks, threshold at 0.5 at test time. Use a gradient estimator for discrete sampling $s \sim Ber(\pi)$ at training time (Maddison et al., 2016; Jang et al., 2016).

Experiments - Three types of DMs

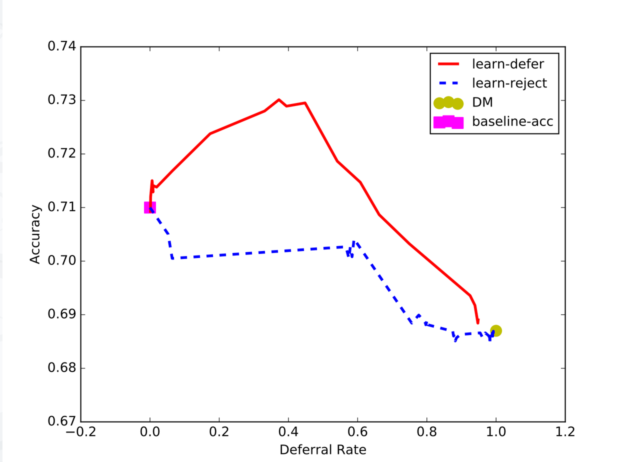
- Datasets: COMPAS (recidivism/race), Health (co-morbidity/age)
- Learning to defer improves tradeoffs between accuracy and fairness/deferral rate over learning to reject (results shown for COMPAS)
- A **high-accuracy** DM may have access to useful auxiliary information Z. Simulated a DM by training a classifier to predict Y from data X and Z, yielding a DM with higher accuracy than the model.



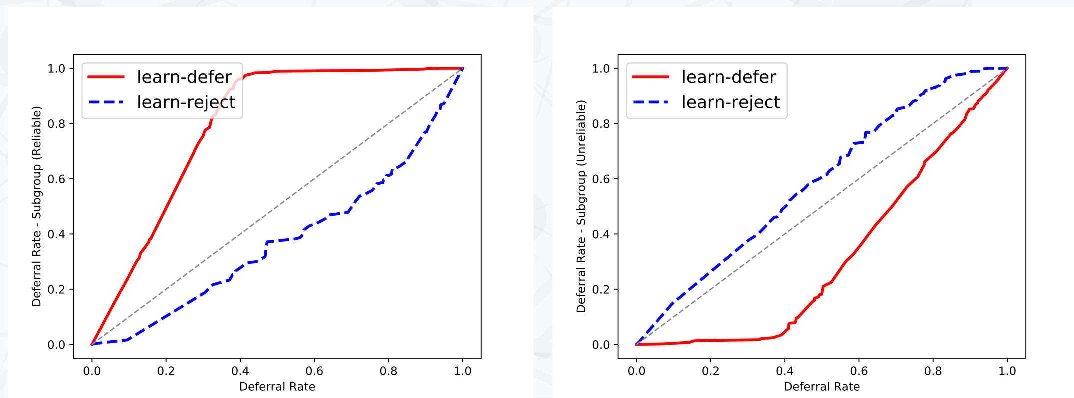
- A **highly-biased** DM may have internal biases against some subgroups. Simulated these biases by training DM with a fairness regularization coefficient $\alpha < 0$.



- An **inconsistent** DM may have low accuracy, despite having auxiliary information Z (Dawes et al., 1989). Simulated this by post-hoc flipping DM's predictions on some "unreliable" subgroup.



- With inconsistent DM, deferring models PASS less on the unreliable subgroup (figure on right) than rejecting models



Takeaways

- Many ML models will be used as part of larger systems
- This should affect the way we train these models
- **Learning to defer** is a generalization of rejection learning; allows us to better optimize the behaviour of a system as a whole, for a wide range of objectives