

Classification Under Human Assistance

Abir De^{1*}, Nastaran Okati^{2*}, Ali Zarezade², Manuel Gomez Rodriguez²

¹IIT Bombay

²Max Planck Institute for Software Systems

Abstract

Most supervised learning models are trained for full automation. However, their predictions are sometimes worse than those by human experts on some specific instances. Motivated by this empirical observation, our goal is to design classifiers that are optimized to operate under different automation levels. More specifically, we focus on convex margin-based classifiers and first show that the problem is NP-hard. Then, we further show that, for support vector machines, the corresponding objective function can be expressed as the difference of two functions $f = g - c$, where g is monotone, non-negative and γ -weakly submodular, and c is non-negative and modular. This representation allows us to utilize a recently introduced deterministic greedy algorithm, as well as a more efficient randomized variant of the algorithm, which enjoy approximation guarantees at solving the problem. Experiments on synthetic and real-world data from several applications in medical diagnosis illustrate our theoretical findings and demonstrate that, under human assistance, supervised learning models trained to operate under different automation levels can outperform those trained for full automation as well as humans operating alone.

1 Introduction

In recent years, machine learning models have matched, or even surpassed, the average performance of human experts at tasks for which intelligence is required (Graves, Abdel-Rahman, and Hinton 2013; Krizhevsky, Sutskever, and Hinton 2012; Silver et al. 2016; Sutskever, Vinyals, and Le 2014). As a consequence, there is a widespread discussion on the possibility of letting machine learning models take high-stake decisions—the promise is that the timeliness and quality of the decisions would greatly improve. For example, in medical diagnosis, patients would not need to wait for months to be diagnosed by a specialist. In content moderation, online publishers could moderate toxic comments before they trigger incivility in their platforms. In software development, developers would easily find bugs in large software projects and would not need to spend long hours in code reviews.

Unfortunately, the decisions taken by machine learning models are still worse than those by human experts on

some instances, where they make far more errors than average (Raghu et al. 2019). Motivated by this observation, there has been a paucity of work on developing machine learning models that are optimized to operate under different automation levels (De et al. 2020a; Bordt and von Luxburg 2020; Mozannar and Sontag 2020; Raghu et al. 2019; Wilder, Horvitz, and Kamar 2020)—models that are optimized to take decisions for a given fraction of the instances and leave the remaining ones to humans. However, most of this work has developed heuristic algorithms that do not enjoy theoretical guarantees. One of the only exceptions is the work by De et al. (2020a), which has reduced the problem of ridge regression under different automation levels to the maximization of an α -submodular function (Gatmiry and Gomez-Rodriguez 2019). In our work, rather than (ridge) regression, we focus on classification under human assistance and show that, for support vector machines, the problem can be solved using algorithms with theoretical guarantees.

More specifically, we first show that, for convex margin-based classifiers, the problem of classification under human assistance is NP-hard. This is due to its combinatorial nature—for each potential meta-decision about which instances the classifier will decide upon, there is an optimal set of parameters for the classifier, however, the meta-decision is also something we seek to optimize. Then, for support vector machines, we derive an alternative representation of the objective function as a difference of two functions $f = g - c$, where g is monotone, non-negative, and γ -weakly submodular (Bian et al. 2017; Das and Kempe 2018) and c is non-negative and modular. Moreover, we further show that, in our problem, the submodularity ratio γ , which characterizes how close is the function g to being submodular, can be lower bounded. These properties allow a recently introduced deterministic greedy algorithm (Algorithm 1) as well as a more efficient randomized variant of the algorithm (Harshaw et al. 2019) to enjoy nontrivial approximation guarantees.

Finally, we experiment with synthetic and real-world data from several applications in medical diagnosis. Our experiments on synthetic data reveal that, by outsourcing samples to humans during training, the resulting support vector machine is able to reduce the number of training samples inside or on the wrong side of the margin, among those samples it

*Equal contributions.

needs to decide upon. Our experiments on real data demonstrate that, under human assistance, support vector machines trained to operate under different automation levels outperform those trained for full automation as well as humans operating alone¹.

Further related work. There is a rapidly increasing line of work devoted to designing classifiers that are able to defer decisions (Bartlett and Wegkamp 2008; Cortes, DeSalvo, and Mohri 2016; El-Yaniv et al. 2010; Geifman and El-Yaniv 2019; Geifman, Uziel, and El-Yaniv 2018; Hsu et al. 2020; Liu et al. 2019; Ramaswamy, Tewari, and Agarwal 2018; Thulasidasan et al. 2019; Wiener and El-Yaniv 2011; Ziyin et al. 2020). However, this line of work does not consider there is a human decision maker, with a human error model, who takes a decision whenever the classifiers defer it and the classifiers are trained to predict the labels of all samples in the training set, as in full automation.

Our work also relates to the area of active learning (Chen and Price 2017; Cohn, Ghahramani, and Jordan 1995; Guo and Schuurmans 2008; Hashemi et al. 2019; Hoi et al. 2006; Sabato and Munos 2014; Sugiyama 2006; Willett, Nowak, and Castro 2006), where the goal is to determine which subset of training samples one should label so that a supervised machine learning model, trained on these samples, generalizes well across the entire feature space during test. However, there is a fundamental difference between our work and active learning. In our work, the trained model only needs to accurately predict samples which are close to the samples assigned to the machine during training time. In contrast, in active learning, the trained model needs to predict well *any* sample during test time.

Finally, our work advances the state of the art on human-machine collaboration (Ghosh et al. 2019; Grover et al. 2018; Hadfield-Menell et al. 2016; Haug, Tschitschek, and Singla 2018; Kamalaruban et al. 2019; Macindoe, Kaelbling, and Lozano-Pérez 2012; Meresht et al. 2020; Nikolaidis et al. 2017, 2015; Radanovic et al. 2019; Tschitschek et al. 2019; Wilson and Daugherty 2018). However, rather than considering a setting in which the machine and the human interact with each other as most previous work, we develop algorithms that learn to distribute decisions between humans and machines.

2 Problem Formulation

In this section, we formally introduce the problem of designing convex margin-based classifiers that are optimized to operate under different automation levels. Then, we show that, for convex margin-based classifiers, the problem is NP-hard. For simplicity, we will consider binary classification, however, our ideas can be extended to m-ary classification.

In binary classification, one needs to find a mapping function $f(\mathbf{x})$ between feature vectors $\mathbf{x} \in \mathbb{R}^m$, with $\mathbf{x} \sim p(\mathbf{x})$, and class labels $y \in \{-1, 1\}$, with $y \sim p(y | \mathbf{x})$. To this end, one utilizes a training set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i \in \mathcal{V}}$ to construct a mapping that works *well* on an *unseen* test set. For margin-based classifiers, finding this mapping usually reduces to

building a decision boundary defined by a set of parameters θ that separates feature vectors in the training set according to their class labels. One typically looks for the decision boundary that achieves the greatest classification accuracy in a test set by minimizing a convex loss function $\ell(h_\theta(\mathbf{x}), y)$ over a training set, *i.e.*, $\theta^* = \operatorname{argmin}_\theta \sum_{i \in \mathcal{V}} \ell(h_\theta(\mathbf{x}_i), y_i)$, where $h_\theta(\mathbf{x}_i)$ denotes the signed distance from the feature vector \mathbf{x} to the decision boundary. Then, given an *unseen* feature vector \mathbf{x} from the test set, the classifier predicts $f(\mathbf{x}) = 1$ if $h_{\theta^*}(\mathbf{x}_i) \geq 0$ and $f(\mathbf{x}) = -1$ otherwise.

In binary classification under human assistance, for every feature vector $\mathbf{x} \in \mathbb{R}^m$, the mapping function $f(\mathbf{x})$ can resort to either a classifier or a human expert. For margin-based classifiers, finding the mapping then reduces to picking the subset of training samples $\mathcal{S} \subseteq \mathcal{V}$ that are outsourced to human experts, with $|\mathcal{S}| \leq n$, and building a decision boundary that separates feature vectors in the subset of training samples $\mathcal{S}^c = \mathcal{V} \setminus \mathcal{S}$ according to their class labels. Using the same convex loss function $\ell(h_\theta(\mathbf{x}), y)$ as in the standard binary classification, our goal is then to solve the following minimization problem:

$$\begin{aligned} & \underset{\mathcal{S}, \theta}{\text{minimize}} && \sum_{i \in \mathcal{V} \setminus \mathcal{S}} \ell(h_\theta(\mathbf{x}_i), y_i) + \sum_{i \in \mathcal{S}} c(\mathbf{x}_i, y_i) \\ & \text{subject to} && |\mathcal{S}| \leq n, \end{aligned} \quad (1)$$

where $c(\mathbf{x}, y)$ denotes the human *error per sample*, which we will define more precisely in the next section. Here, we assume that human annotations are independent, which allows us to cast the total human error as the sum of human error per sample over all instances. Moreover, we use a linear constraint on the number of examples outsourced to humans because, in most practical scenarios, humans get paid every time they make a prediction—if they make n predictions, they get paid n times.

Given the optimal set \mathcal{S}^* , we can find the optimal parameter $\theta^* = \theta^*(\mathcal{V} \setminus \mathcal{S}^*)$ in polynomial time since, by assumption, the loss $\ell(h_\theta(\mathbf{x}_i), y_i)$ is convex. Unfortunately, the following Theorem tells us that, in general, we cannot expect to find both \mathcal{S}^* and θ^* in polynomial time (proven in the long version of our paper (De et al. 2020b)):

Theorem 1 *The problem of designing margin-based classifiers under human assistance defined in Eq. 1 is NP-Hard.*

Moreover, given the solution to the above minimization problem, we would still need to decide whether to outsource an *unseen* feature vector \mathbf{x} from the test set to a human expert even if $\mathbf{x} \neq \mathbf{x}_i$ for all $i \in \mathcal{V}$. To this end, we could train an additional model $\pi(d | \mathbf{x})$ to decide which samples to outsource to a human using the labeled set $\{(\mathbf{x}_i, d_i)\}_{i \in \mathcal{V}}$, where \mathbf{x}_i are the feature vectors in the training set and $d_i = +1$ if $i \in \mathcal{S}^*$ and $d_i = -1$ otherwise. Then, as long as this model does not make mistakes on the training set, one can readily conclude that the samples assigned to the classifier during training are as if they were *sampled* from the feature distribution $p(\mathbf{x})\pi(d = -1 | \mathbf{x})$ induced by π . As a direct consequence, if the model $\pi(d | \mathbf{x})$ is smooth with respect to \mathbf{x} , one can further conclude that the trained margin-based classifier will work *well* on the *unseen* samples it needs to decide upon at test time, *i.e.*, samples from $p(\mathbf{x})\pi(d = -1 | \mathbf{x})$.

¹Our code and data are available in <https://github.com/Networks-Learning/classification-under-assistance>

3 Algorithms with Approximation Guarantees for Support Vector Machines

In this section, we show that, for support vector machines (SVMs), the optimization problem defined in Eq. 1 can be rewritten as a maximization of the difference of two functions $g - c$, where g is monotone, non-negative, and γ -weakly submodular and c is non-negative modular. Moreover, we further show that the submodularity ratio γ can be lower bounded and, as a consequence, a recently introduced deterministic greedy algorithm (Harshaw et al. 2019) as well as a more efficient randomized variant of the algorithm enjoy approximation guarantees at solving the problem.

Monotonicity and weak submodularity. For (soft margin) SVMs, we can first rewrite the minimization problem defined in Eq. 1 as follows²:

$$\begin{aligned} \text{minimize}_{\mathcal{S}, \mathbf{w}, b} \quad & \sum_{i \in \mathcal{V} \setminus \mathcal{S}} \underbrace{[\lambda \|\mathbf{w}\|^2 + [1 - y_i(\mathbf{w}^\top \Phi(\mathbf{x}_i) + b)]_+]}_{\ell(h_{\mathbf{w}, b}(\mathbf{x}_i), y_i)} \\ & + \sum_{i \in \mathcal{S}} \underbrace{[1 - y_i h(\mathbf{x}_i)]_+}_{c(\mathbf{x}_i, y_i)} \end{aligned} \quad (2)$$

subject to $|\mathcal{S}| \leq n$,

where $\Phi(\cdot)$ denotes a given feature transformation, $h(\cdot) \in [-H, H]$ is the (normalized) score provided by the human experts, which is only known for the training samples, and $H > 0$ is a given constant. In the above, we measure the human error $c(\mathbf{x}, y)$ using a hinge loss $[1 - y \cdot h(\mathbf{x})]_+$ because the SVM formulation also uses a hinge loss $[1 - y \cdot (\mathbf{w}^\top \mathbf{x} + b)]_+$ to measure the machine error. This is necessary in order to compare human and machine performance meaningfully. However, our solution is agnostic to this specific choice—it is applicable to any human error model.

Now, for any given set \mathcal{S} , let $\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})$ and $b^*(\mathcal{V} \setminus \mathcal{S})$ be the parameters that minimize the objective function above, i.e., $\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S}), b^*(\mathcal{V} \setminus \mathcal{S}) = \operatorname{argmin}_{\mathbf{w}, b} \sum_{i \in \mathcal{V} \setminus \mathcal{S}} [\lambda \|\mathbf{w}\|^2 + (1 - y_i(\mathbf{w}^\top \Phi(\mathbf{x}_i) + b))]_+$. Here, note that these parameters can be found in polynomial time since the first two parameters in the objective function are convex. Then, we can rewrite the above minimization problem as a set function maximization problem:

$$\text{maximize}_{\mathcal{S}} \quad g(\mathcal{S}) - c(\mathcal{S}), \quad \text{subject to } |\mathcal{S}| \leq n, \quad (3)$$

where

$$\begin{aligned} g(\mathcal{S}) = & \lambda |\mathcal{V}| \|\mathbf{w}^*(\mathcal{V})\|^2 + \sum_{i \in \mathcal{V}} [1 - y_i(\mathbf{w}^*(\mathcal{V})^\top \Phi(\mathbf{x}_i) + b^*(\mathcal{V}))]_+ \\ & - \lambda |\mathcal{V} \setminus \mathcal{S}| \|\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})\|^2 \\ & - \sum_{i \in \mathcal{V} \setminus \mathcal{S}} [1 - y_i(\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})^\top \Phi(\mathbf{x}_i) + b^*(\mathcal{V} \setminus \mathcal{S}))]_+, \end{aligned} \quad (4)$$

and

$$c(\mathcal{S}) = \sum_{i \in \mathcal{S}} [1 - y_i h(\mathbf{x}_i)]_+. \quad (5)$$

²In the long version of our paper (De et al. 2020b), we also consider hard margin linear SVMs, which are relevant whenever the data is linearly separable.

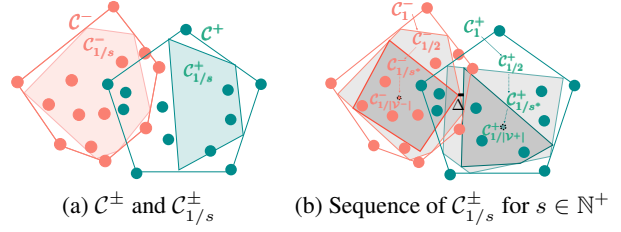


Figure 1: Convex hulls and reduced convex hulls. Panels (a) and (b) show the convex hulls \mathcal{C}^\pm and reduced convex hulls $\mathcal{C}_{1/s}^\pm$ of a training set whose feature vectors are non separable. In both panels, cyan and orange dots represent feature vectors \mathbf{x}_i with $y_i = 1$ and $y_i = -1$, respectively.

In the above, the first term $\lambda |\mathcal{V}| \|\mathbf{w}^*(\mathcal{V})\|^2 + \sum_{i \in \mathcal{V}} [1 - y_i(\mathbf{w}^*(\mathcal{V})^\top \Phi(\mathbf{x}_i) + b^*(\mathcal{V}))]_+$ ensures that the function $g(\mathcal{S})$ is non-negative and the function $c(\mathcal{S})$ is clearly non-negative and modular³. Moreover, we have the following proposition, which shows that $g(\mathcal{S})$ is a monotone function (proven in the long version of our paper (De et al. 2020b)):

Proposition 2 *The set function $g(\mathcal{S})$, defined in Eq. 4, is monotone, i.e., $g(\mathcal{S} \cup \{j\}) - g(\mathcal{S}) \geq 0$ for all $\mathcal{S} \subseteq \mathcal{V}$ and $j \in \mathcal{V}$.*

If the feature vectors in the training set are not separable according to their class labels, there exist instances of the problem in which the function $g(\mathcal{S})$ has submodularity ratio $\gamma = 0$ ⁴. However, we will now identify a general class of feature distributions for which the function $g(\mathcal{S})$ has a nonzero submodularity ratio and its value can be lower bounded. This lower bound will allow a recently introduced deterministic greedy algorithm as well as its randomized variant to enjoy approximation guarantees at solving the problem. In the remainder, we first consider linear SVMs, i.e., $\Phi(\mathbf{x}) = \mathbf{x}$, and then nonlinear SVMs.

Let \mathcal{V}^+ and \mathcal{V}^- be the set of training samples with positive and negative labels, respectively, \mathcal{C}^+ and \mathcal{C}^- be their corresponding convex hulls, i.e.,

$$\mathcal{C}^\pm = \left\{ \sum_{i \in \mathcal{V}^\pm} \mu_i \mathbf{x}_i \mid \sum_{i \in \mathcal{V}^\pm} \mu_i = 1, \mu_i \geq 0 \right\}, \quad (6)$$

and Δ be the minimum distance between them, i.e., $\Delta = \min_{\mathbf{x}^+ \in \mathcal{C}^+, \mathbf{x}^- \in \mathcal{C}^-} \|\mathbf{x}^+ - \mathbf{x}^-\|_2$. Then, note that, whenever the feature vectors in the training set are not separable according to their class labels, the above convex hulls overlap and thus $\Delta = 0$. However, under mild technical conditions, there will always exist subsets of feature vectors within these convex hulls that do not overlap, as shown in Figure 1. To characterize these subsets, we introduce the notion of re-

³A set function $f(\mathcal{S})$ is modular iff it satisfies that $f(\mathcal{S} \cup \{j\}) - f(\mathcal{S}) = f(\mathcal{L} \cup \{j\}) - f(\mathcal{L})$ for all $\mathcal{S} \subseteq \mathcal{L} \subseteq \mathcal{V}$ and $j \in \mathcal{V}$.

⁴A set function $f(\mathcal{S})$ is γ -weakly submodular iff we have $\sum_{j \in \mathcal{L} \setminus \mathcal{S}} [f(\mathcal{S} \cup \{j\}) - f(\mathcal{S})] \geq \gamma [f(\mathcal{S} \cup \mathcal{L}) - f(\mathcal{S})] \forall \mathcal{S}, \mathcal{L} \subseteq \mathcal{V}$ and $j \in \mathcal{V}$. The largest $\gamma \leq 1$ such that the inequality is true is called submodularity ratio.

duced convex hulls (Bennett and Bredensteiner 2000):

$$\mathcal{C}_{1/s}^\pm = \left\{ \sum_{i \in \mathcal{V}^\pm} \mu_i \mathbf{x}_i \mid \sum_{i \in \mathcal{V}^\pm} \mu_i = 1, 0 \leq \mu_i \leq \frac{1}{s} \right\}, \quad (7)$$

where $s \in \mathbb{N}^+$, with $s \leq \min\{|\mathcal{V}^+|, |\mathcal{V}^-|\}$ and, similarly as before, we denote the minimum distance between them as $\Delta_{1/s} = \min_{\mathbf{x}^+ \in \mathcal{C}_{1/s}^+, \mathbf{x}^- \in \mathcal{C}_{1/s}^-} \|\mathbf{x}^+ - \mathbf{x}^-\|$.

Now, consider the sequence of reduced convex hulls $\{(\mathcal{C}_{1/s}^+, \mathcal{C}_{1/s}^-)\}_{s=1}^{\mathcal{V}_{\min}}$, where $\mathcal{V}_{\min} = \min\{|\mathcal{V}^+|, |\mathcal{V}^-|\}$, illustrated in Figure 1(b), and note that the corresponding minimum distances, by definition, satisfy that $\Delta_{1/s} \leq \Delta_{1/s'}$ for all $s' > s$. Then, we measure to what extent feature vectors with positive and negative labels overlap using the distance

$$\Delta^* = \min_{s \in \{1, \dots, \mathcal{V}_{\min}\}} \{\Delta_{1/s} \mid \Delta_{1/s} > 0\}, \quad (8)$$

where the higher the value of $s^* = \operatorname{argmin}_s \{\Delta_{1/s} \mid \Delta_{1/s} > 0\}$, the higher the overlap between feature vectors with positive and negative labels. Moreover, if there are no elements in the sequence with positive distance, we set $\Delta^* = 0$ and $s^* = 0$.

Given the above, we are now ready to present and prove one of our key results, which characterizes the submodularity ratio of the function $g(\mathcal{S})$ in terms of the amount of overlap between feature vectors with positive and negative labels, as measured by the distance Δ^* (proven in the long version of our paper (De et al. 2020b)):

Theorem 3 Let $\Phi(\mathbf{x}) = \mathbf{x}$, $\rho^* = \mathcal{V}_{\min}/|\mathcal{V}|$, $\sigma^* = s^*/|\mathcal{V}|$, $\eta = (2\sqrt{\lambda} + \max_{i \in \mathcal{V}} \|\mathbf{x}_i\|)/\sqrt{\lambda}$. Then, the submodularity ratio γ of the function $g(\mathcal{S})$ (defined in Eq. 4) satisfies that

$$\gamma \geq \gamma^* = \frac{\min \left\{ \frac{[\Delta^* \sigma^*]^2}{4\lambda}, \frac{1}{(\eta - 2)^2} \right\}}{\eta + \frac{\eta^2}{2} \left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{4|\mathcal{V}|(\eta - 1)}{\eta^2}} \right) + (\eta - 1)|\mathcal{V}|}$$

as long as the number of samples outsourced to humans $n \leq (\rho^* - \sigma^*)|\mathcal{V}|$.

For nonlinear SVMs, rather than characterizing the submodularity ratio in terms of Δ^* and s^* , we resort to the spectral properties of the kernel matrix $\mathbb{K} = [K(\mathbf{x}_i, \mathbf{x}_j)]_{i,j \in \mathcal{V}}$. In particular, our key result is the following Theorem (proven in the long version of our paper (De et al. 2020b)):

Theorem 4 Let $\eta = (2\sqrt{\lambda} + \max_{i \in \mathcal{V}} \sqrt{K(\mathbf{x}_i, \mathbf{x}_i)})/\sqrt{\lambda}$, $\mathbf{Y} = \operatorname{diag}(\{y_i\}_{i \in \mathcal{V}})$, $\rho^* = \mathcal{V}_{\min}/|\mathcal{V}|$, and

$$\zeta = \min_{\substack{\boldsymbol{\mu} \geq 0, \sum_{i \in \mathcal{V}^+} \mu_i = 1, \sum_{i \in \mathcal{V}^-} \mu_i = 1}} \boldsymbol{\mu}^\top \mathbf{Y}^\top \mathbb{K} \mathbf{Y} \boldsymbol{\mu},$$

If the kernel matrix is full rank, i.e., $\operatorname{rank}(\mathbb{K}) = |\mathcal{V}|$, then the submodularity ratio γ of the function $g(\mathcal{S})$ satisfies that

$$\gamma \geq \gamma^* = \frac{\min \left\{ \frac{\zeta \sigma^{*2}}{4\lambda}, \frac{1}{(\eta - 2)^2} \right\}}{\eta + \frac{\eta^2}{2} \left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{4|\mathcal{V}|(\eta - 1)}{\eta^2}} \right) + (\eta - 1)|\mathcal{V}|}$$

as long as $n \leq (\rho^* - \sigma^*)|\mathcal{V}|$ for some $\sigma^* \in (0, \rho^*]$.

Algorithm 1: Distorted greedy algorithm

Input: Ground set \mathcal{V} , functions g and c , parameters n and γ .
Initialize: $\mathcal{S} \leftarrow \emptyset$
for $i = 0, \dots, n - 1$ **do**
 $\omega_i \leftarrow (1 - \frac{\gamma}{n})^{n-(i+1)}$
 $k^* \leftarrow \operatorname{argmax}_{k \in \mathcal{V} \setminus \mathcal{S}} \{\omega_i \cdot [g(\mathcal{S} \cup \{k\}) - g(\mathcal{S})] - c(\{k\})\}$
 if $\omega_i \cdot [g(\mathcal{S} \cup \{k^*\}) - g(\mathcal{S})] - c(\{k^*\}) > 0$ **then**
 $\mathcal{S} \leftarrow \mathcal{S} \cup \{k^*\}$
 end
end
Return \mathcal{S}

Finally, for the particular case of (soft margin) SVMs without offset, i.e., $b = 0$ in Eq. 2, we can derive a stronger lower bound, which does not depend on Δ^* and s^* , by exploiting their greater stability properties (proven in the long version of our paper (De et al. 2020b)):

Theorem 5 If $\eta = (2\sqrt{\lambda} + \max_{i \in \mathcal{V}} \sqrt{K(\mathbf{x}_i, \mathbf{x}_i)})/\sqrt{\lambda}$, then for SVMs without offset, ($b = 0$ in Eq. 2), the submodularity ratio of the function $g(\mathcal{S})$ is given by

$$\gamma \geq \gamma^* = \min \left\{ \frac{1}{(\eta - 2)^2}, \frac{1}{2} \right\} / \left[\eta + \frac{\eta^2}{2} \right].$$

Proof sketch of our key technical results. The proofs of Theorems 3, 4 and 5 consist of two steps. In the first step, they show that $g(\mathcal{S} \cup \{j\}) - g(\mathcal{S}) \geq \lambda \|\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})\|^2$ and use the dual formulation of SVM as well as the properties of the corresponding SVM model to derive a lower bound of $\|\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})\|$. In the second step, they use the stability property of SVM (Bousquet and Elisseeff 2002) to derive the upper bound on $g(\mathcal{S} \cup \mathcal{L}) - g(\mathcal{S})$. These two steps together lead to the bound on γ^* .

While the bounds in all the above theorems are tight in terms of the size of the dataset, there are notable differences between the different model classes. For SVMs with offsets, Theorems 3 and 4 suggest that the submodularity ratio bound decreases proportionally to $1/|\mathcal{V}|$ and this is due to their poor stability properties (Hush, Scovel, and Stewart 2007). For SVMs without offsets, Theorem 5 tells us that the submodularity ratio bound is independent of $|\mathcal{V}|$ and this is due to their greater stability properties (Bousquet and Elisseeff 2002). More specifically, for any $\mathcal{L} \subseteq \mathcal{V} \setminus \mathcal{S}$, the marginal gain $g(\mathcal{S} \cup \mathcal{L}) - g(\mathcal{S})$ can be upper bounded by a smaller quantity than in the case of SVMs with offsets and this results into a stronger lower bound.

Distorted greedy algorithm. The distorted greedy algorithm (Harshaw et al. 2019) proceeds iteratively and, at each iteration, it assigns to the humans the sample (\mathbf{x}_k, y_k) that provides the highest marginal *distorted* gain among the remaining training samples $\mathcal{V} \setminus \mathcal{S}$. Algorithm 1 summarizes the algorithm, which requires the value of the submodularity ratio γ as an input. Since we only have data dependant bounds on γ , in our experiments, we use the meta algorithm proposed in Harshaw et al. (2019) to *guess* the value of γ .

Since we have shown that the objective function in Eq. 3 can be expressed as the difference of two functions $g - c$,

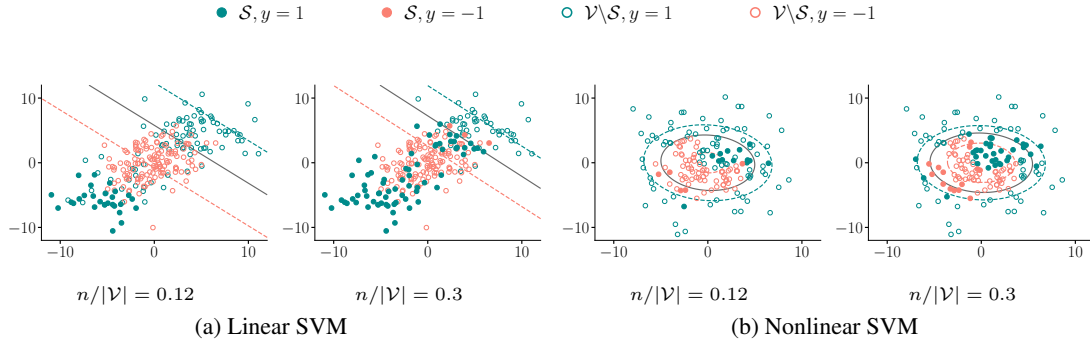


Figure 2: Linear and non linear support vector machines with offset under human assistance trained using the stochastic greedy algorithm (Alg. 1). In each panel, circles represent the feature vectors in the training set, filled and empty circles are assigned to humans and machines, respectively, the solid line indicates $\mathbf{w}^{*\top}(\mathcal{V} \setminus \mathcal{S}^*)\phi(\mathbf{x}) + b(\mathcal{V} \setminus \mathcal{S}^*) = 0$, and the cyan and orange dashed lines indicate $\mathbf{w}^{*\top}(\mathcal{V} \setminus \mathcal{S}^*)\phi(\mathbf{x}) + b(\mathcal{V} \setminus \mathcal{S}^*) = 1$ and $\mathbf{w}^{*\top}(\mathcal{V} \setminus \mathcal{S}^*)\phi(\mathbf{x}) + b(\mathcal{V} \setminus \mathcal{S}^*) = -1$, respectively. For both linear and nonlinear SVMs, we used $\lambda = 1$, $\Delta H = 0.2$ and, for nonlinear SVMs, we used a quadratic kernel $K(\mathbf{x}_i, \mathbf{x}_j) = (\frac{1}{2}\langle \mathbf{x}_i, \mathbf{x}_j \rangle)^2$.

where g is monotone, non-negative and γ -weakly submodular and c is non-negative modular, it readily follows from Theorem 3 in Harshaw et al. (2019) that the distorted greedy algorithm enjoys approximation guarantees. More specifically, the distorted greedy algorithm is guaranteed to return a set \mathcal{S} such that

$$g(\mathcal{S}) - c(\mathcal{S}) \geq (1 - e^{-\gamma})g(\text{OPT}) - c(\text{OPT}), \quad (9)$$

where OPT is the optimal set and $\gamma \leq \gamma^*$ with γ^* defined in Theorem 3 (linear SVMs with offset), Theorem 4 (nonlinear SVMs with offset), or Theorem 5 (SVMs without offset).

In addition to the distorted greedy algorithm, which needs to make $O(n|\mathcal{V}|)$ evaluations of the function $g(\cdot)$, Harshaw et al. (2019) has also proposed a randomized variant of the algorithm, which enjoys an asymptotically faster run time due to the use of sampling techniques and is also applicable to our problem. Instead of optimizing over the entire ground set \mathcal{V} at each iteration, it optimizes over a random sample $\mathcal{B}_i \subseteq \mathcal{V}$ of size $O(\frac{|\mathcal{V}|}{n} \log \frac{1}{\epsilon})$. Hence, it only needs to make $O(|\mathcal{V}| \log(\frac{1}{\epsilon}))$ evaluations of $g(\cdot)$ and it returns a set \mathcal{S} such that

$$\mathbb{E}[g(\mathcal{S}) - c(\mathcal{S})] \geq (1 - e^{-\gamma} - \epsilon)g(\text{OPT}) - c(\text{OPT}). \quad (10)$$

In the next sections, we will demonstrate that, in addition to enjoying the above approximation guarantees, the distorted greedy algorithm as well as its randomized variant perform better in practice than several competitive baselines.

4 Experiments on Synthetic Data

In this section, we first look into the solutions provided by the distorted greedy algorithm (Alg. 1) in a variety of synthetic examples. Then, we compare the performance of the distorted greedy algorithm and its randomized variant with several competitive baselines. Finally, we investigate the influence that the amount of human error has on the number of samples outsourced to humans by the distorted greedy algorithm.

Experimental setup. In all our experiments, we generate $|\mathcal{V}| = 400$ samples using a generative process under which the corresponding SVM under full automation

is unable to perform well. For linear SVMs with offset, we draw the class labels uniformly at random, *i.e.*, $y \sim \text{Bernoulli}(0.5)$, and then draw the features from two different distributions, $p(\mathbf{x}|y = -1) = \mathcal{N}([0, 0], [6, 1; 1, 6])$ and $p(\mathbf{x}|y = 1) = \beta\mathcal{N}([5, 5], [6, 1; 1, 6]) + (1 - \beta)\mathcal{N}([-5, -5], [6, 1; 1, 6])$, where $\beta \sim \text{Bernoulli}(0.5)$. For nonlinear SVMs, we draw the features from a single distribution $p(\mathbf{x}) \sim \mathcal{N}([0, 0], [12, 1; 1, 14])$ and, for each sampled feature \mathbf{x} , we set $y = +1$ if $\|\mathbf{x} - [1, 1]\| \leq 2$ or $\|\mathbf{x} + [1, 1]\| \geq 5$ and $y = -1$, otherwise. Here, we use a quadratic kernel $K(\mathbf{x}_i, \mathbf{x}_j) = (\frac{1}{2}\langle \mathbf{x}_i, \mathbf{x}_j \rangle)^2$. Moreover, we generate the scores provided by human experts $h(\mathbf{x})$ per label by drawing samples from two uniform distributions, *i.e.*, $h(\mathbf{x}) \sim \text{Unif}[-\Delta H, -\Delta H + 1]$ if $y = 1$ and $h(\mathbf{x}) \sim \text{Unif}[-1 + \Delta H, \Delta H]$ otherwise. The exact choice of ΔH varies across experiments and is mentioned therein. Finally, in each experiment, we use 60% of samples for training and 40% of samples for testing, set $\lambda = 1$ and we follow the procedure described in Section 2 to train a multi-layer perceptron $\pi(\cdot|\mathbf{x})$ that decides which samples to outsource to humans at test time.

Baseline methods. We compare the performance of the distorted greedy algorithm and its randomized variant with four competitive baselines:

- Triage based on algorithmic uncertainty (Raghu et al. 2019): it first trains a support vector machine for full automation, *i.e.*, $\mathcal{S} = \emptyset$. Then, at test time, it outsources to humans the top n testing samples sorted in decreasing order of the classification uncertainty of this support vector machine, defined as $1/|\mathbf{w}^*(\mathcal{V})\phi(\mathbf{x}) + b^*(\mathcal{V})|$.

- Triage based on predicted error (Raghu et al. 2019): it trains a support vector machine for full automation, *i.e.*, $\mathcal{S} = \emptyset$, and two additional supervised models that predict the human error and the support vector machine error. Then, at test time, it outsources to humans the top n testing samples sorted in decreasing order of the difference between the predicted machine error and the predicted human error.

- Full automation: it trains a support vector machine for full automation and, at test time, the trained support vector

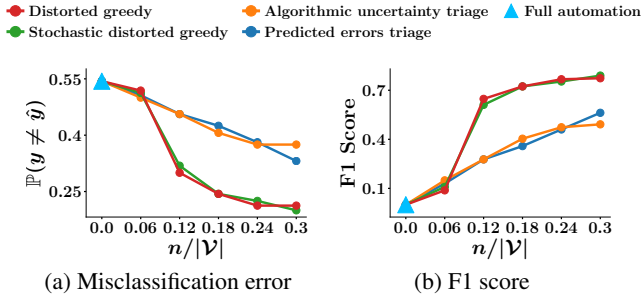


Figure 3: Performance of all methods on synthetic data with linear SVM with offset. For all methods, we set $\Delta H = 0.2$. If humans predict all testing samples (No automation), we have that $\mathbb{P}(\hat{y} \neq y) = 0.19$ and F1-score = 0.81.

machine predicts all testing samples.

— No automation: humans predict all testing samples.

We evaluate the performance of all the methods in terms of the misclassification test error $\mathbb{P}(y \neq \hat{y})$ and F1 score on positive test samples. Here, our F1 score is a useful metric in medical diagnosis, since it measures the ability of a model to detect the specific disease.

Results. First, we look into the solutions $(\mathcal{S}^*, \mathbf{w}^*(\mathcal{V} \setminus \mathcal{S}^*), b^*(\mathcal{V} \setminus \mathcal{S}^*))$ provided by the distorted greedy algorithm. Figure 2 summarizes the results, which shows that: (i) the distorted greedy algorithm outsources to humans those samples that are more prone to be misclassified by the machine, *i.e.*, $\{(x, y) \mid y[\mathbf{w}^{*\top}(\mathcal{V} \setminus \mathcal{S}^*)\phi(x) + b(\mathcal{V} \setminus \mathcal{S}^*)] < 0\}$; and, (ii) the higher the samples n outsourced to humans, the lower the number of training samples inside the margin, *i.e.*, $\{(x, y) \mid |\mathbf{w}^{*\top}(\mathcal{V} \setminus \mathcal{S}^*)\phi(x) + b(\mathcal{V} \setminus \mathcal{S}^*)| \leq 1\}$, among those samples the SVM needs to decide upon.

Next, we compare the performance of the distorted greedy algorithm and its randomized variant with all the baselines—algorithmic uncertainty triage, predicted error triage, full automation and no automation—in terms of the misclassification test error $\mathbb{P}(y \neq \hat{y})$ and F1 score on positive test samples. Figure 3 summarizes the results, which shows that both algorithms consistently outperform all the baselines by large margins. Finally, we investigate the influence that the amount of human error has on the number of samples $|\mathcal{S}| \leq n$ outsourced to humans by the distorted greedy algorithm and its randomized variant. Figure 4 summarizes the results for the linear SVM with offset. We find that, as long as the amount of human error is small, both algorithms outsource $|\mathcal{S}| = n$ samples to humans, however, for higher levels of human error, the algorithm outsources fewer samples $|\mathcal{S}| < n$ since it is more beneficial for the minimization of the overall error.

5 Experiments on Real Data

We experiment with three real-world medical datasets and first show that, under human assistance, support vector machines trained using the distorted greedy algorithm as well as its randomized variant outperform those trained for full automation as well as humans operating alone. Then, we in-

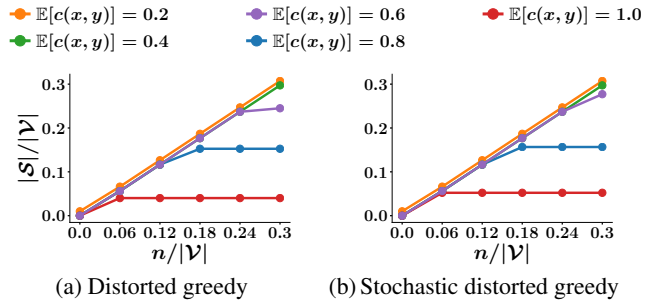


Figure 4: Number of samples $|\mathcal{S}|$ outsourced to humans by the distorted greedy algorithm and its randomized variant against the maximum number of outsourced samples n for different amount of human errors.

vestigate the influence that the amount of human error has on the performance of the distorted greedy algorithm.

Experimental setup. We experiment with three publicly available datasets (Decencière et al. 2014; Hoover, Kouznetsova, and Goldbaum 2000), each of them from a different application in medical diagnosis:

- (i) **Messidor:** It consists of $|\mathcal{V}| = 400$ eye images. Each image is assigned a score by one single medical expert, on a four-point scale, which measures the severity of a retinopathy.
- (ii) **Stare:** It consists of $|\mathcal{V}| = 373$ retinal images. Each image is assigned a score by one single medical expert, on a five-point scale, which measures the severity of a retinal hemorrhage.
- (iii) **Aptos:** It consists of $|\mathcal{V}| = 705$ retinal images. Each image is given a score by one single clinician, on a five-point scale, which measures the severity of diabetic retinopathy.

Following previous work, we first generate a 1000 dimensional feature vector using Resnet (He et al. 2016) for each sample in the Stare dataset and a 4096-dimensional feature vector using VGG16 (Simonyan and Zisserman 2014) for each sample in the Messidor and the Aptos datasets. Then, we use the top 50 features, as identified by PCA, as \mathbf{x} in our experiments. Moreover, for each sample, we set $y = -1$ if its severity score q corresponds to one of the two lowest grades of the associated disease and $y = +1$ otherwise.

For each sample, we only have access to the score q given by a single human expert. Therefore, we sample the scores given by humans from a categorical distribution $h(\mathbf{x}) = \hat{q} \sim \text{Cat}(\mathbf{p}_{\mathbf{x},q})$, where $\mathbf{p}_{\mathbf{x},q} = \text{Dirichlet}(\chi_{\mathbf{x},q})$ are the probabilities of each potential score \hat{q} for a sample (\mathbf{x}, y) and $\chi_{\mathbf{x},q}$ is a vector parameter that controls the human accuracy, and then scale these scores so that $h(\cdot) \in [-1, 1]$ and compute the human error as $c(\mathbf{x}, y) = \mathbb{E}[(1 - yh(\mathbf{x}))_+]$. For each sample (\mathbf{x}, y) , the element of $\chi_{\mathbf{x},q}$ corresponding to the score $\hat{q} = q$ given by the human expert has the highest value.

Finally, for each dataset, we use 60% of samples for training and 40% of samples for testing, set the value of λ using cross validation under full automation, and followed the procedure described in Section 2 to train a logistic regression model $\pi(d \mid \mathbf{x})$ that decides which samples to outsource to

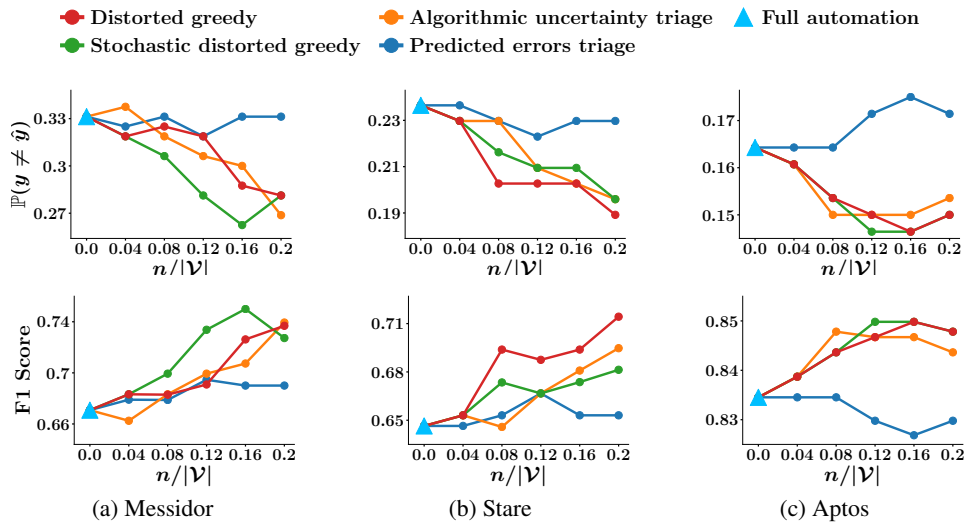


Figure 5: Performance of all methods on three medical datasets. If humans predict all testing samples (No automation), we have that $\mathbb{P}(\hat{y} \neq y) = 0.15, 0.14, 0.20$ and F1-score = $0.87, 0.78, 0.8$ for Messidor, Stare and Aptos datasets respectively.

humans at test time. Here, the logistic regression model uses just $|\mathbf{w}^*(\mathcal{V} \setminus \mathcal{S})\phi(\mathbf{x}) + b^*(\mathcal{V} \setminus \mathcal{S})|$ as the only single feature. The long version of our paper (De et al. 2020b) contains additional details about the generation of human scores, the additional model π and the baseline algorithms.

Result. First, we compare the performance of all methods in Figure 5. The results show that the distorted greedy algorithm as well as its randomized variant: (i) outperform the triage baselines in most of the automation levels and (ii) benefit from human assistance both in cases when humans on their own (No automation) are better than machines on their own (Full automation) and vice versa (Panels (a) and (b) vs Panel (c)); and, (iii) perform comparably, sometimes one beating the other and vice versa (refer to Section 3.4. in Harshaw et al. (2019) to better understand the reasons). Here, note that we intentionally experimented with different values of $\chi_{\bullet, \bullet}$ across different datasets so that the performance of the no automation baseline varies. Moreover, since in our datasets, the parameter $\rho^* = \min\{|\mathcal{V}^+|, |\mathcal{V}^-|\}/|\mathcal{V}| \in (0.33, 0.45)$, then to satisfy the conditions of Theorem 3, we only experiment with $n/|\mathcal{V}| \leq 0.2 < \rho^*$.

Next, we assess the influence that the human error $c(\mathbf{x}, y)$ has on the performance of the greedy algorithm. Figure 6 summarizes the results for the Aptos dataset, which show that, as long as the human error is small, the performance of the distorted greedy algorithm improves with respect to the maximum number of samples n . However, for higher levels of human error, outsourcing samples does not help.

6 Conclusions

In this paper, we have shown that, for support vector machines, we can solve the problem of classification under human assistance using algorithms with approximation guarantees. Moreover, we have further shown that, under human assistance, support vector machines trained to operate under different automation levels can provide superior empirical

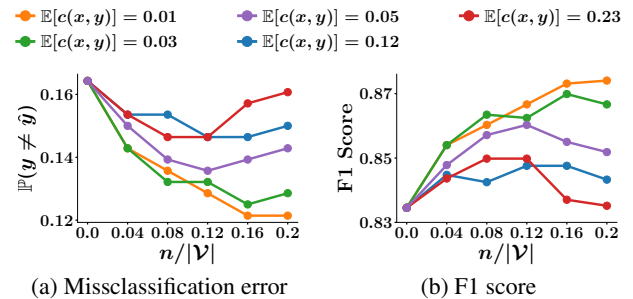


Figure 6: Performance of the stochastic distorted greedy algorithm against the maximum number of outsourced samples n for different levels of human error $c(\mathbf{x}, y)$ on the Aptos dataset.

performance than those trained for full automation.

Our work also opens many interesting venues for future work. For example, we have assumed that the human error is known, however, in practice, the spectrum of human abilities spans a broad range. It would be interesting to develop algorithms that, over time, adapt to the particular human(s) they are dealing with. Moreover, we have assumed that the human annotations are independent, however, there exist scenarios in which annotations are correlated, *e.g.*, when a single user sequentially reviews a set of items in a session. It would be interesting to lift the independence assumption and design algorithms that account for the correlation between annotations. It would also be valuable to find tighter lower bounds on the parameter γ , which better characterize the good empirical performance. Finally, it would be worth to extend our analysis to other convex margin-based classifiers as well as design machine learning algorithms operating under different automation levels in sequential decision-making tasks, *e.g.*, semi-autonomous driving (Meresht et al. 2020).

Broader Impact

There are several technical, societal, and legal challenges that prevents the use of fully autonomous machine learning systems for high-stake decision making in safety-critical applications such as medical diagnosis. In our work, we argue that it is possible to address such challenges by letting humans and machines to assist and complement each other. Moreover, our theoretical and experimental results suggest that humans and machines working together may achieve better performance than what they would achieve on their own.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 945719).

References

- Bartlett, P.; and Wegkamp, M. 2008. Classification with a reject option using a hinge loss. *JMLR* 9(Aug): 1823–1840.
- Bennett, K. P.; and Bredensteiner, E. J. 2000. Duality and geometry in SVM classifiers. In *ICML*, volume 2000, 57–64.
- Bian, A. A.; Buhmann, J. M.; Krause, A.; and Tschitschek, S. 2017. Guarantees for greedy maximization of non-submodular functions with applications. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 498–507.
- Bordt, S.; and von Luxburg, U. 2020. When Humans and Machines Make Joint Decisions: A Non-Symmetric Bandit Model. *arXiv preprint arXiv:2007.04800*.
- Bousquet, O.; and Elisseeff, A. 2002. Stability and generalization. *Journal of machine learning research* 2(Mar): 499–526.
- Chen, X.; and Price, E. 2017. Active regression via linear-sample sparsification. *arXiv preprint arXiv:1711.10051*.
- Cohn, D.; Ghahramani, Z.; and Jordan, M. 1995. Active learning with statistical models. In *NeurIPS*, 705–712.
- Cortes, C.; DeSalvo, G.; and Mohri, M. 2016. Learning with rejection. In *ALT*, 67–82. Springer.
- Das, A.; and Kempe, D. 2018. Approximate submodularity and its applications: subset selection, sparse approximation and dictionary selection. *The Journal of Machine Learning Research* 19(1): 74–107.
- De, A.; Koley, P.; Ganguly, N.; and Gomez-Rodriguez, M. 2020a. Regression under human assistance. In *AAAI*.
- De, A.; Okati, N.; Zarezade, A.; and Gomez-Rodriguez, M. 2020b. Classification Under Human Assistance. *arXiv preprint arXiv:2006.11845*.
- Decencière, E.; Zhang, X.; Cazuguel, G.; Lay, B.; Cochener, B.; Trone, C.; Gain, P.; Ordonez, R.; Massin, P.; Erginay, A.; Charton, B.; and Klein, J.-C. 2014. Feedback on a publicly distributed database: the Messidor database. *Image Analysis & Stereology* 33(3): 231–234.
- El-Yaniv, R.; et al. 2010. On the Foundations of Noise-free Selective Classification. *Journal of Machine Learning Research* 11(5).
- Gatmiry, K.; and Gomez-Rodriguez, M. 2019. Non-submodular Function Maximization subject to a Matroid Constraint, with Applications. *arXiv preprint arXiv:1811.07863*.
- Geifman, Y.; and El-Yaniv, R. 2019. SelectiveNet: A Deep Neural Network with an Integrated Reject Option. *arXiv preprint arXiv:1901.09192*.
- Geifman, Y.; Uziel, G.; and El-Yaniv, R. 2018. Bias-Reduced Uncertainty Estimation for Deep Neural Classifiers .
- Ghosh, A.; Tschitschek, S.; Mahdavi, H.; and Singla, A. 2019. Towards deployment of robust AI agents for human-machine partnerships. In *AAMAS*.
- Graves, A.; Abdel-Rahman, M.; and Hinton, G. E. 2013. Speech recognition with deep recurrent neural networks. In *ICASSP*.
- Grover, A.; Al-Shedivat, M.; Gupta, J. K.; Burda, Y.; and Edwards, H. 2018. Learning policy representations in multi-agent systems. In *ICML*.
- Guo, Y.; and Schuurmans, D. 2008. Discriminative batch mode active learning. In *NeurIPS*, 593–600.
- Hadfield-Menell, D.; Russell, S. J.; Abbeel, P.; and Dragan, A. 2016. Cooperative inverse reinforcement learning. In *Advances in neural information processing systems*, 3909–3917.
- Harshaw, C.; Feldman, M.; Ward, J.; and Karbasi, A. 2019. Submodular Maximization Beyond Non-negativity: Guarantees, Fast Algorithms, and Applications. In *ICML*.
- Hashemi, A.; Ghasemi, M.; Vikalo, H.; and Topcu, U. 2019. Submodular Observation Selection and Information Gathering for Quadratic Models. *arXiv preprint arXiv:1905.09919*.
- Haug, L.; Tschitschek, S.; and Singla, A. 2018. Teaching inverse reinforcement learners via features and demonstrations. In *Advances in Neural Information Processing Systems*, 8464–8473.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.
- Hoi, S.; Jin, R.; Zhu, J.; and Lyu, M. R. 2006. Batch mode active learning and its application to medical image classification. In *ICML*.
- Hoover, A.; Kouznetsova, V.; and Goldbaum, M. 2000. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical imaging* 19(3): 203–210.
- Hsu, Y.-C.; Shen, Y.; Jin, H.; and Kira, Z. 2020. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10951–10960.

- Hush, D.; Scovel, C.; and Steinwart, I. 2007. Stability of unstable learning algorithms. *Machine learning* 67(3): 197–206.
- Kamalaruban, P.; Devidze, R.; Cevher, V.; and Singla, A. 2019. Interactive teaching algorithms for inverse reinforcement learning. In *IJCAI*.
- Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*.
- Liu, Z.; Wang, Z.; Liang, P. P.; Salakhutdinov, R. R.; Morency, L.-P.; and Ueda, M. 2019. Deep Gamblers: Learning to Abstain with Portfolio Theory. In *NeurIPS*.
- Macindoe, O.; Kaelbling, L. P.; and Lozano-Pérez, T. 2012. Pomcop: Belief space planning for sidekicks in cooperative games. In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*.
- Meresht, V. B.; De, A.; Singla, A.; and Gomez-Rodriguez, M. 2020. Learning to Switch Between Machines and Humans. *arXiv preprint arXiv:2002.04258* .
- Mozannar, H.; and Sontag, D. 2020. Consistent Estimators for Learning to Defer to an Expert. *arXiv preprint arXiv:2006.01862* .
- Nikolaidis, S.; Forlizzi, J.; Hsu, D.; Shah, J.; and Srinivasa, S. 2017. Mathematical models of adaptation in human-robot collaboration. *arXiv preprint arXiv:1707.02586* .
- Nikolaidis, S.; Ramakrishnan, R.; Gu, K.; and Shah, J. 2015. Efficient model learning from joint-action demonstrations for human-robot collaborative tasks. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 189–196. IEEE.
- Radanovic, G.; Devidze, R.; Parkes, D.; and Singla, A. 2019. Learning to collaborate in markov decision processes. In *ICML*.
- Raghu, M.; Blumer, K.; Corrado, G.; Kleinberg, J.; Obermeyer, Z.; and Mullainathan, S. 2019. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220* .
- Ramaswamy, H.; Tewari, A.; and Agarwal, S. 2018. Consistent algorithms for multiclass classification with an abstain option. *Electronic J. of Statistics* 12(1): 530–554.
- Sabato, S.; and Munos, R. 2014. Active regression by stratification. In *NeurIPS*, 469–477.
- Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* .
- Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .
- Sugiyama, M. 2006. Active learning in approximately linear regression based on conditional expectation of generalization error. *JMLR* 7(Jan): 141–166.
- Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *NIPS*.
- Thulasidasan, S.; Bhattacharya, T.; Bilmes, J.; Chennupati, G.; and Mohd-Yusof, J. 2019. Combating Label Noise in Deep Learning Using Abstention. *arXiv preprint arXiv:1905.10964* .
- Tschiatschek, S.; Ghosh, A.; Haug, L.; Devidze, R.; and Singla, A. 2019. Learner-aware teaching: Inverse reinforcement learning with preferences and constraints. In *Advances in Neural Information Processing Systems*, 4147–4157.
- Wiener, Y.; and El-Yaniv, R. 2011. Agnostic selective classification. In *Advances in neural information processing systems*, 1665–1673.
- Wilder, B.; Horvitz, E.; and Kamar, E. 2020. Learning to Complement Humans. *arXiv preprint arXiv:2005.00582* .
- Willett, R.; Nowak, R.; and Castro, R. M. 2006. Faster rates in regression via active learning. In *NeurIPS*.
- Wilson, H. J.; and Daugherty, P. R. 2018. Collaborative intelligence: humans and AI are joining forces. *Harvard Business Review* 96(4): 114–123.
- Ziyin, L.; Chen, B.; Wang, R.; Liang, P. P.; Salakhutdinov, R.; Morency, L.-P.; and Ueda, M. 2020. Learning Not to Learn in the Presence of Noisy Labels. *arXiv preprint arXiv:2002.06541* .