Improve learning combining crowdsourced labels by weighting Areas Under the Margin

Tanguy Lefort
IMAG, Univ Montpellier, CNRS
Inria, LIRMM, Univ Montpellier, CNRS

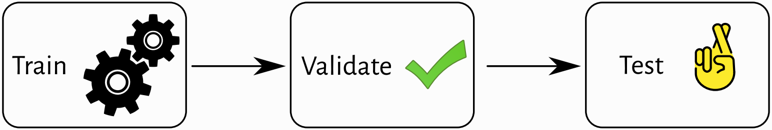
Ongoing joint work with...

- Benjamin Charlier (CNRS, IMAG, Univ Montpellier)
- Alexis Joly (Inria, LIRMM, Univ Montpellier CNRS)
- Joseph Salmon (CNRS, IMAG, Univ Montpellier, IUF)

Improve learning combining crowdsourced labels by weighting Areas Under the Margin

`https://arxiv.org/abs/2209.15380`

Problem: Can we trust our data

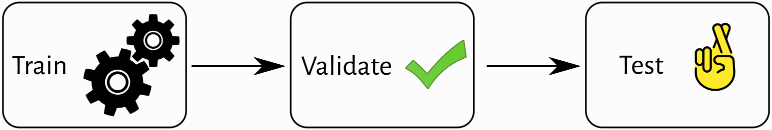


(1) A. Krizhevsky and G. Hinton (2009). "Learning multiple layers of features from tiny images". In.

(2) (N.d.). https://github.com/googlecreativelab/quickdraw-dataset.

(3) Y. LeCun et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

Problem: Can we trust our data



Inside the dataset during training …

$y^\star = \text{cat}$

CIFAR-10[1]

$y^\star = \text{T-shirt}$

Quickdraw[2]

$y^\star = 6$

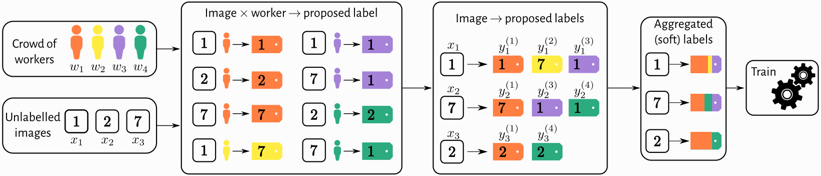
MNIST[3]

[1] A. Krizhevsky and G. Hinton (2009). "Learning multiple layers of features from tiny images". In.

[2] (N.d.). https://github.com/googlecreativelab/quickdraw-dataset.

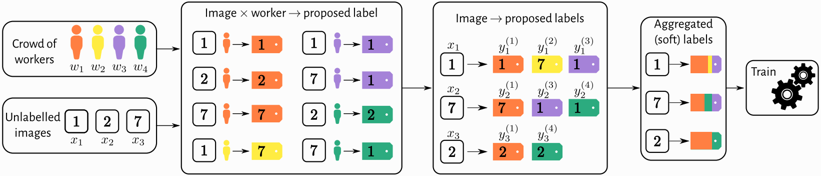
[3] Y. LeCun et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$ pairs of tasks $\times$ labels $\in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Where do the labels come from? **Crowdsourcing**



How can we **identify too ambiguous** tasks in a **crowdsourcing setting**?

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  pairs of tasks $\times$ labels $\in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Where do the labels come from? **Crowdsourcing**



How can we **identify too ambiguous** tasks in a **crowdsourcing setting**?

Why not look at label distribution entropy?
Not reliable (numbers of labels, biases, psychology mechanisms, spammers)

Where are our usual labels coming from?

Simple strategy.

- Most of the time, a majority vote
  (naive and highly unreliable outside of asymptotic framework)

(4) R. Snow et al. (2008). “Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks”. In: *Conference on Empirical Methods in Natural Language Processing*. EMNLP 2008. Association for Computational Linguistics, pp. 254–263.

Where are our usual labels coming from?

Simple strategy.

- Most of the time, a majority vote
(naive and highly unreliable outside of asymptotic framework)

Other common strategies.

- y_i is the first label that reaches a consensus of p people (often $p = 5$)[(4)]
$\rightarrow$ arbitrary choice that is not theoretically supported
- y_i is the arg max of the aggregated soft labels (better, but not enough…)

(4) R. Snow et al. (2008). "Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks". In: *Conference on Empirical Methods in Natural Language Processing*. EMNLP 2008. Association for Computational Linguistics, pp. 254–263.

A first solution: classify the quality
ImageNet oddities

- curated set of probes[5] in the training data (OOD=Out Of Distribution)
  e.g.: ImageNet[6] $+14$ millions tasks, $K = 1000$ classes

$$(\texttt{task}_i, \texttt{label}_i, \texttt{metadata}_i) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{M}$$

Black bear | Dishwasher | School bus | Mud turtle | Jeep | Loafer

(a) Typical | (b) Atypical | (c) Corrupted | (d) Rand Label | (e) OOD | (f) Rand Input

[5] S. A. Siddiqui et al. (2022). *Metadata Archaeology: Unearthing Data Subsets by Leveraging Training Dynamics*.

[6] O. Russakovsky et al. (2015). “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252.

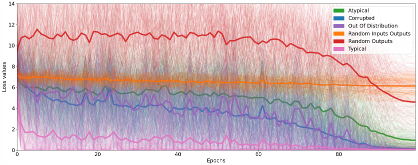
A FIRST SOLUTION: CLASSIFY THE QUALITY
IMAGENET ODDITIES

- curated set of probes[5] in the training data (OOD=Out Of Distribution)
  e.g.: ImageNet[6] +14 millions tasks, $K = 1000$ classes

$$(\texttt{task}_i, \texttt{label}_i, \texttt{metadata}_i) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{M}$$

Black bear | Dishwasher | School bus | Mud turtle | Jeep | Loafer

(a) Typical | (b) Atypical | (c) Corrupted | (d) Rand Label | (e) OOD | (f) Rand Input



- 1 metadata = 1 dynamic
- Identify the ambiguity

[5] S. A. Siddiqui et al. (2022). *Metadata Archaeology: Unearthing Data Subsets by Leveraging Training Dynamics*.

[6] O. Russakovsky et al. (2015). “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252.

When was the last time you had a curated set of metadata up your sleeve?

(7) G. Pleiss et al. (2020). "Identifying mislabeled data using the area under the margin ranking". In: *NeurIPS*.

(8) C. Northcutt, L. Jiang, and I. Chuang (2021). "Confident learning: Estimating uncertainty in dataset labels". In: *J. Artif. Intell. Res.* 70, pp. 1373–1411.

(9) J. Han, P. Luo, and X. Wang (2019). "Deep self-learning from noisy labels". In: *ICCV*, pp. 5138–5147.

(10) K.-H. Lee et al. (2018). "Cleannet: Transfer learning for scalable image classifier training with label noise". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5447–5456.

When was the last time you had a curated set of metadata up your sleeve?
Never

Assuming we have a hard label($\in [K]$):

- Study the dynamics:
  - AUM[7]
- Confident learning[8]
- Self learning[9]
- Representative Sampling (`CleanNet`[10])
- ...

[7] G. Pleiss et al. (2020). "Identifying mislabeled data using the area under the margin ranking". In: *NeurIPS*.

[8] C. Northcutt, L. Jiang, and I. Chuang (2021). "Confident learning: Estimating uncertainty in dataset labels". In: *J. Artif. Intell. Res.* 70, pp. 1373–1411.

[9] J. Han, P. Luo, and X. Wang (2019). "Deep self-learning from noisy labels". In: *ICCV*, pp. 5138–5147.

[10] K.-H. Lee et al. (2018). "Cleannet: Transfer learning for scalable image classifier training with label noise". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5447–5456.

Area under the margins[11]

Setting. $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times [K]$. Let $\mathcal{C}$ an iterative classifier *s.t.* at epoch $t \leq T$ we have $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores

AUM

$$\mathrm{AUM}(x_i, y_i) = \frac{1}{T} \sum_{t=1}^{T} \left[\mathcal{C}^{(t)}(x_i)_{y_i} - \max_{\ell \neq y_i} \mathcal{C}^{(t)}(x_i)_\ell \right] \in \mathbb{R}$$

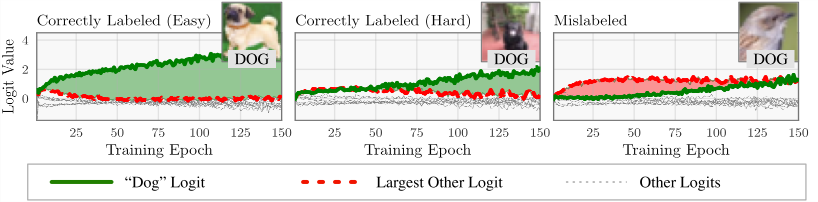
[11] G. Pleiss et al. (2020). "Identifying mislabeled data using the area under the margin ranking". In: *NeurIPS*.

Area under the margins[11]

Setting. $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times [K]$. Let $\mathcal{C}$ an iterative classifier *s.t.* at epoch $t \leq T$ we have $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores

AUM

$$\mathrm{AUM}(x_i, y_i) = \frac{1}{T} \sum_{t=1}^{T} \left[\mathcal{C}^{(t)}(x_i)_{y_i} - \max_{\ell \neq y_i} \mathcal{C}^{(t)}(x_i)_\ell \right] \in \mathbb{R}$$



[11] G. Pleiss et al. (2020). "Identifying mislabeled data using the area under the margin ranking". In: *NeurIPS*.

Dissecting the AUM
Building to the crowdsourced extension

$$\mathrm{AUM}(x_i, y_i) = \overbrace{\frac{1}{T}\sum_{t=1}^{T}}^{\text{Average = Stability}} \overbrace{\left[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i}}_{\text{Score of assigned label}} - \underbrace{\max_{\ell \neq y_i} \mathcal{C}^{(t)}(x_i)_{\ell}}_{\text{Other maximum score}} \right]}^{\text{Margin between scores: content of Hinge loss}}$$

Average = Stability

Margin between scores: content of Hinge loss

$$\mathrm{AUM}(x_i, y_i) = \underbrace{\frac{1}{T}\sum_{t=1}^{T}}_{\text{Average = Stability}} \overbrace{\left[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i}}_{\text{Score of assigned label}} - \underbrace{\max_{\ell \neq y_i} \mathcal{C}^{(t)}(x_i)_\ell}_{\text{Other maximum score}} \right]}^{\text{Margin between scores: content of Hinge loss}}$$

Score of assigned label

Other maximum score

Problem for crowdsourcing.

- We don't have a single y_i but multiple $y_i^{(j)}$ (one for each worker w_j answering task x_i)
  - ...so $\mathcal{C}^{(t)}(x_i)_{y_i}$ does not exist
  - ...and same issue with $\ell \neq y_i$.

$$\widetilde{\mathrm{AUM}}(x_i) = \underbrace{\frac{1}{|\mathcal{A}(x_i)|}\sum_{j\in\mathcal{A}(x_i)}}_{\text{Averaging workers AUM}} \frac{1}{T}\sum_{t=1}^{T} \overbrace{\Big[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i^{(j)}}}_{\text{Score of assigned label by worker } w_j} - \underbrace{\max_{\ell\neq y_i^{(j)}} \mathcal{C}^{(t)}(x_i)_\ell}_{\text{Other maximum score}}\Big]}^{\text{Margin between scores: content of Hinge loss}}$$

- Multiple answers $\implies$ average each AUM.
- Let $\mathcal{A}(x_i) := \{j \in [n_{\text{worker}}] :\ \text{worker } j \text{ answered task } i\}$.

Dissecting the AUM

Building to the crowdsourced extension

$$\widetilde{\mathrm{AUM}}(x_i) = \overbrace{\frac{1}{|\mathcal{A}(x_i)|}\sum_{j\in\mathcal{A}(x_i)}}^{\text{Averaging workers AUM}} \frac{1}{T}\sum_{t=1}^{T} \overbrace{\left[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i^{(j)}}}_{\text{Score of assigned label by worker } w_j} - \underbrace{\max_{\ell\neq y_i^{(j)}} \mathcal{C}^{(t)}(x_i)_\ell}_{\text{Other maximum score}}\right]}^{\text{Margin between scores: content of Hinge loss}}$$

- Multiple answers $\Longrightarrow$ average each AUM.
- Let $\mathcal{A}(x_i) := \{j \in [n_{\text{worker}}] : \text{ worker } j \text{ answered task } i\}$.

Problem of reliability.

- The AUM of an expert shouldn't count as much as anyone's
  - ▶ …so we need a weighting score for workers.

$$\widetilde{\widetilde{\mathrm{AUM}}}(x_i) = \overbrace{\frac{1}{S}\sum_{j\in\mathcal{A}(x_i)}}^{\text{Weighted average of AUM}} \overbrace{s^{(j)}(x_i)}^{\text{Trust score of } w_j \text{ for } x_i} \frac{1}{T}\sum_{t=1}^{T} \overbrace{\Big[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i^{(j)}}}_{\text{Score of assigned label by worker } w_j} - \underbrace{\max_{\ell\neq y_i^{(j)}} \mathcal{C}^{(t)}(x_i)_\ell}_{\text{Other maximum score}}\Big]}^{\text{Margin between scores: content of Hinge loss}}$$

- Introduce weights $s^{(j)}(x_i)$ as the trust score in worker j for task x_i
- Denote $S = \sum_{j\in\mathcal{A}(x_i)} s^{(j)}(x_i)$,

[12] C. Ju, A. Bibaut, and M. van der Laan (2018). "The relative performance of ensemble methods with deep convolutional neural networks for image classification". In: *J. Appl. Stat.* 45.15, pp. 2800–2818.

[13] M. Lapin, M. Hein, and B. Schiele (2016). "Loss functions for top-k error: Analysis and insights". In: *CVPR*, pp. 1468–1477; F. Yang and S. Koyejo (2020). "On the consistency of top-k surrogate losses". In: *ICML*, pp. 10727–10735.

Dissecting the AUM

Building to the crowdsourced extension

$$\widetilde{\mathrm{AUM}}(x_i) = \overbrace{\frac{1}{S}\sum_{j\in\mathcal{A}(x_i)}}^{\text{Weighted average of AUM}} \overbrace{s^{(j)}(x_i)}^{\text{Trust score of } w_j \text{ for } x_i} \frac{1}{T}\sum_{t=1}^{T} \overbrace{\left[\underbrace{\mathcal{C}^{(t)}(x_i)_{y_i^{(j)}}}_{\text{Score of assigned label by worker } w_j} - \underbrace{\max_{\ell\neq y_i^{(j)}} \mathcal{C}^{(t)}(x_i)_\ell}_{\text{Other maximum score}}\right]}^{\text{Margin between scores: content of Hinge loss}}$$

- Introduce weights $s^{(j)}(x_i)$ as the trust score in worker j for task x_i
- Denote $S = \sum_{j\in\mathcal{A}(x_i)} s^{(j)}(x_i)$,

Modifying the margin

- Scale effects in the scores, need to use a quantity that can be controlled[12]
- Use margin with better theoretical properties for top-k classification[13]

---

[12] C. Ju, A. Bibaut, and M. van der Laan (2018). "The relative performance of ensemble methods with deep convolutional neural networks for image classification". In: *J. Appl. Stat.* 45.15, pp. 2800–2818.

[13] M. Lapin, M. Hein, and B. Schiele (2016). "Loss functions for top-k error: Analysis and insights". In: *CVPR*, pp. 1468–1477; F. Yang and S. Koyejo (2020). "On the consistency of top-k surrogate losses". In: *ICML*, pp. 10727–10735.

$$\mathrm{WAUM}(x_i) := \overbrace{\frac{1}{S}\sum_{j\in\mathcal{A}(x_i)}}^{\text{Weighted average of AUM}} \overbrace{s^{(j)}(x_i)}^{\text{Trust score of } w_j \text{ for } x_i} \overbrace{\frac{1}{T}\sum_{t=1}^{T}}^{\text{Average = Stability}} \overbrace{\left[\underbrace{\mathrm{softmax}^{(t)}_{y_i^{(j)}}(x_i)}_{\text{Probability of assigned label by worker } w_j} - \underbrace{\mathrm{softmax}^{(t)}_{[2]}(x_i)}_{\text{Second maximum probability}}\right]}^{\text{Margin between scores: content of Hinge loss}}$$

- Denote $\mathrm{softmax}(x_i) = \mathrm{softmax}(\mathcal{C}(x_i)) \in \Delta_{K-1}$ (simplex of dim $K-1$)
- Softmax output ordered as $\mathrm{softmax}_{[1]}(x_i) \geq \cdots \geq \mathrm{softmax}_{[K]}(x_i) > 0$

Choosing $s^{(j)}(x_i)$:

- if $s^{(j)}(x_i) = 1$ all workers have the same weight
- if $s^{(j)}(x_i) = c_j$ the weights only depend on the worker

(14) A. Dawid and A. Skene (1979). "Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm". In: *J. R. Stat. Soc. Ser. C. Appl. Stat.* 28.1, pp. 20–28.

Choosing $s^{(j)}(x_i)$:

- if $s^{(j)}(x_i) = 1$ all workers have the same weight
- if $s^{(j)}(x_i) = c_j$ the weights only depend on the worker
- …there is already a literature on trusting workers !

Dawid and Skene[14]

Model each worker with a confusion matrix $\pi^{(j)}$.
Each worker answers independently as:

$$y_i^{(j)} \mid y_i^\star = \ell \sim \mathcal{M}\text{ultinomial}\left(\pi_{\ell\bullet}^{(j)}\right)$$

The diagonal of $\pi^{(j)}$ represents worker ability to be correct.

[14] A. Dawid and A. Skene (1979). “Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm”. In: *J. R. Stat. Soc. Ser. C. Appl. Stat.* 28.1, pp. 20–28.

Likelihood.

$$\prod_{k \in [K]} \pi_{\ell k}^{(j)}$$

- 1 task, 1 worker and 1 answer conditioned on $y_i^\star = \ell$

Dawid and Skene likelihood

Likelihood.

$$\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\pi_{\ell k}^{(j)}$$

- 1 task, 1 worker and 1 answer conditioned on $y_i^\star = \ell$
- Multiple workers answer independently

Likelihood.

$$\prod_{\ell \in [K]} \left[\mathbb{P}(y_i^\star = \ell) \prod_{j \in [n_{\texttt{worker}}]} \prod_{k \in [K]} \pi_{\ell k}^{(j)} \right]^{\mathbf{1}_{\{y_i^\star = \ell\}}}$$

- 1 task, 1 worker and 1 answer conditioned on $y_i^\star = \ell$
- Multiple workers answer independently
- Remove conditioning assumption on $y_i^\star$: $\mathbb{P}(y_i^\star = \ell) = \rho_\ell$

Likelihood.

$$\prod_{i\in[n_{\texttt{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\pi_{\ell k}^{(j)}\right]^{T_{i\ell}}$$

- 1 task, 1 worker and 1 answer conditioned on $y_i^\star = \ell$
- Multiple workers answer independently
- Remove conditioning assumption on $y_i^\star$: $\mathbb{P}(y_i^\star = \ell) = \rho_\ell$
- Each task is independent: $T_{i\ell} = 1$ if task i has label ℓ and 0 otherwise

Likelihood.

$$\prod_{i\in[n_{\texttt{task}}]}\prod_{\ell\in[K]}\left[\overbrace{\rho_\ell}^{\text{Prevalence of class } \ell}\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\left(\underbrace{\pi^{(j)}_{\ell k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\right)\right]^{\overbrace{T_{i\ell}}^{\text{Indicator of class } \ell \text{ for task } i}}$$

Dawid and Skene vanilla algorithm

Likelihood.

$$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\Bigg[\overbrace{\boxed{\rho_\ell}}^{\text{Prevalence of class } \ell}\prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\Big(\underbrace{\boxed{\pi^{(j)}_{\ell k}}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Big)\Bigg]^{\overbrace{\boxed{T_{i\ell}}}^{\text{Indicator of class } \ell \text{ for task } i}}$$

Initialization: $\forall i \in [n_{\mathtt{task}}], \forall \ell \in [K],\ \hat{T}_{i\ell} = \frac{1}{|\mathcal{A}(x_i)|}\sum_{j\in\mathcal{A}(x_i)} \mathbf{1}_{\{y_i^{(j)}=\ell\}}$

while *Convergence not achieved* **do**

`// M-step: Get` $\hat{\pi}$ `and` $\hat{\rho}$ `assuming` $\hat{T}$`s are known`

$$\forall(\ell,k)\in[K]^2,\ \hat{\pi}^{(j)}_{\ell k} \leftarrow \frac{\sum_{i\in[n_{\mathtt{task}}]}\hat{T}_{i\ell}}{\sum_{k\in[K]}\sum_{i'\in[n_{\mathtt{task}}]}\hat{T}_{i'\ell}}$$

$$\forall \ell\in[K],\ \hat{\rho}_\ell \leftarrow \frac{1}{n_{\mathtt{task}}}\sum_{i\in[n_{\mathtt{task}}]}\hat{T}_{i\ell}$$

`// E-step: Estimate` $\hat{T}$`s with current` $\hat{\pi}$ `and` $\hat{\rho}$

$$\forall i \in [n_{\mathtt{task}}], \forall \ell \in [K],\ \hat{T}_{i\ell} = \frac{\prod_{j\in\mathcal{A}(x_i)}\prod_{k\in[K]}\hat{\rho}_\ell\cdot\hat{\pi}^{(j)}_{\ell k}}{\sum_{\ell'\in[K]}\prod_{j'\in\mathcal{A}(x_i)}\prod_{k'\in[K]}\hat{\rho}_{\ell'}\cdot\hat{\pi}^{(j')}_{\ell' k'}}$$

Labels: $\forall i \in [n_{\mathtt{task}}],\ \hat{y}_i = \hat{T}_{i\bullet} \in \mathbb{R}^K$

GLAD[15]

Adding the task difficulty

- DS assumes the error comes only from workers
- …Is there a model that takes into account task difficulty?

[15] J. Whitehill et al. (2009). "Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise". In: *NeurIPS*. vol. 22.

GLAD [15]

Adding the task difficulty

- DS assumes the error comes only from workers
- …Is there a model that takes into account task difficulty?

GLAD

Model each worker with an ability $\alpha \in \mathbb{R}$ and each task with a difficulty score $\beta \in \mathbb{R}_+^\star$. Model workers answers as:

$$\mathbb{P}\left(y_i^{(j)} = y_i^\star | \alpha, \beta\right) = \frac{1}{1 + e^{-\alpha_j \beta_i}}$$

The trust score is a bilinear function in a worker term α_j and a task term β_i
Assumption. Error is uniform on other labels (not true in practice!)

[15] J. Whitehill et al. (2009). "Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise". In: *NeurIPS*. vol. 22.

Weights in the WAUM

Using the tasks and not just labels

- Keep the product of a worker term and a task term
- Use multidimensionality of DS confusion matrices
- Use a network as control agent[16]

$$s^{(j)}(x_i) = \langle \mathrm{diag}\hat{\pi}^{(j)} \,|\, \mathrm{softmax}^{(T)}(x_i) \rangle \in [0,1]$$

[16] M. Servajean et al. (2017). "Crowdsourcing thousands of specialized labels: A Bayesian active training approach". In: *IEEE Trans. Multimed.* 19.6, pp. 1376–1391.

- Estimate confusion matrices $\pi^{(j)}$

- Estimate confusion matrices $\pi^{(j)}$
- For each worker
  - Train a network on $\left\{\left(x_i, y_i^{(j)}\right);\ x_i \text{ is answered by } w_j\right\}$
  - Compute for the answered tasks:

$$\mathrm{AUM}(x_i, y_i^{(j)}) = \frac{1}{T}\sum_{t=1}^{T}\left[\mathrm{softmax}^{(t)}_{y_i^{(j)}}(x_i) - \mathrm{softmax}^{(t)}_{[2]}(x_i)\right]$$

  - Compute trust scores $s^{(j)}(x_i)$

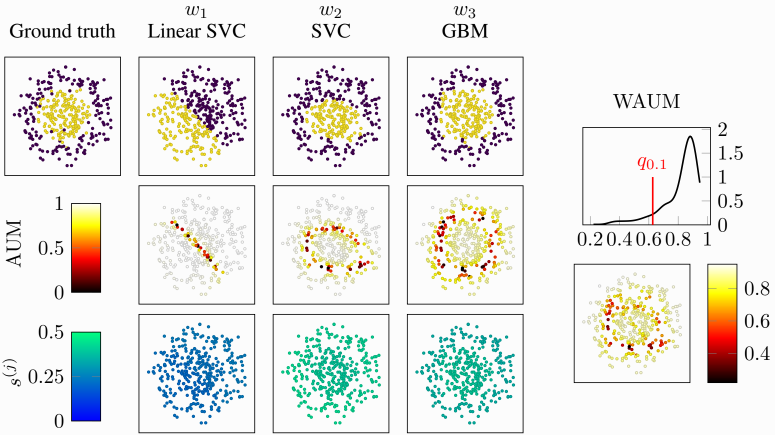
- Estimate confusion matrices $\pi^{(j)}$
- For each worker
  - Train a network on $\left\{ \left(x_i, y_i^{(j)}\right);\ x_i \text{ is answered by } w_j \right\}$
  - Compute for the answered tasks:

$$\mathrm{AUM}(x_i, y_i^{(j)}) = \frac{1}{T}\sum_{t=1}^{T}\left[\mathrm{softmax}^{(t)}_{y_i^{(j)}}(x_i) - \mathrm{softmax}^{(t)}_{[2]}(x_i)\right]$$

  - Compute trust scores $s^{(j)}(x_i)$
- For each task compute the WAUM as the weighted average of AUMs

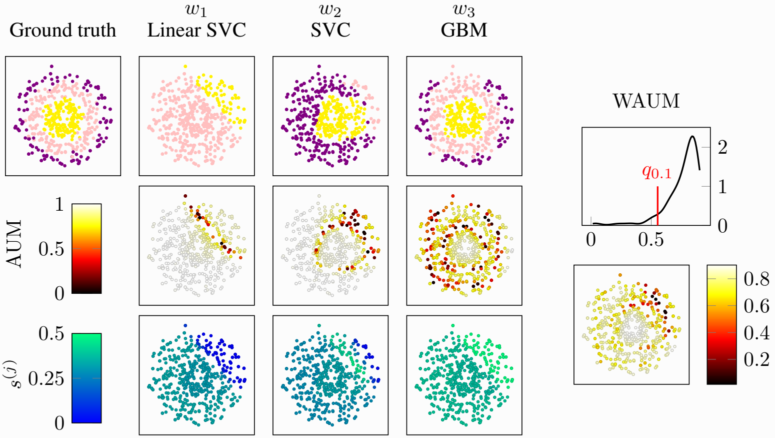
Simulation with circles

Binary setting



- Workers = simulated classifiers (answering 500 tasks)
- Normalized trust scores

Simulation with circles
Three classes

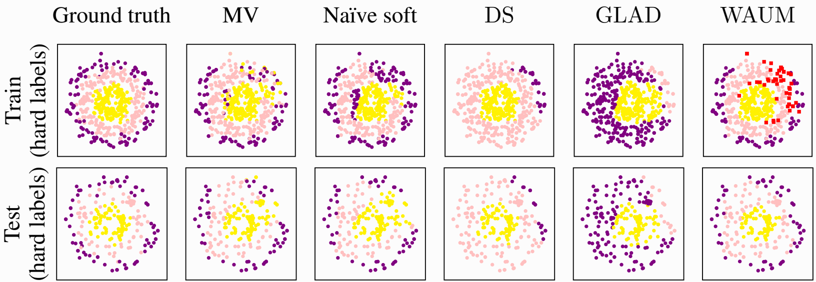


- 3 classes with 250 tasks per class
- Normalized trust scores

- Compute $(\mathrm{WAUM}(x_i))_i$
- Remove the data with WAUM below quantile q_α
- Estimate confusion matrices $\hat{\pi}^{(j)}$ on pruned training dataset

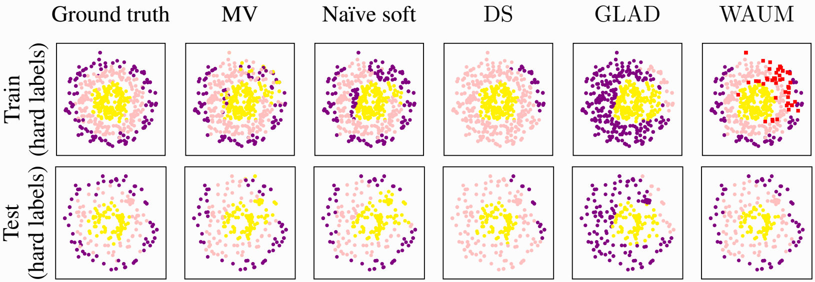
How can we use the WAUM?
Pruning to avoid learning of too ambiguous data

- Compute $(\mathrm{WAUM}(x_i))_i$
- Remove the data with WAUM below quantile q_α
- Estimate confusion matrices $\hat{\pi}^{(j)}$ on pruned training dataset
- $\hat{y}_i = \left(\sum_{j\in\mathcal{A}(x_i)} \pi^{(j)}_{k,k}\mathbf{1}_{\{y_i^{(j)}=k\}}\right)_{k\in[K]}$ normalized $\rightarrow$ our soft labels to learn



How can we use the WAUM?
Pruning to avoid learning of too ambiguous data

- Compute $(\mathrm{WAUM}(x_i))_i$
- Remove the data with WAUM below quantile q_α
- Estimate confusion matrices $\hat{\pi}^{(j)}$ on pruned training dataset
- $\hat{y}_i = \left(\sum_{j \in \mathcal{A}(x_i)} \pi^{(j)}_{k,k} \mathbf{1}_{\{y_i^{(j)}=k\}} \right)_{k \in [K]}$ normalized $\rightarrow$ our soft labels to learn



	MV	Naive soft	DS	GLAD	$\mathrm{WAUM}(\alpha = 0.1)$
Test accuracy	0.727	0.697	0.753	0.578	0.806

"3 answers per task is not enough!"

(17) C. Garcin et al. (2021). "Pl@ntNet-300K: a plant image dataset with high label ambiguity and a long-tailed distribution". In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

(18) F. Rodrigues and F. Pereira (2018). "Deep learning from crowds". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

"3 answers per task is not enough!"

- Yes ! It is not
- ...but it happens $\to$ Pl@ntNet[17] (future work), LabelMe[18]
- LabelMe 1000 images (subset of LabelMe image segmentation project)
- Each image was labelled by 1, 2 or 3 workers

[17] C. Garcin et al. (2021). “Pl@ntNet-300K: a plant image dataset with high label ambiguity and a long-tailed distribution”. In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

[18] F. Rodrigues and F. Pereira (2018). “Deep learning from crowds”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

"3 answers per task is not enough!"

- Yes ! It is not
- …but it happens → Pl@ntNet[17] (future work), LabelMe[18]
- LabelMe 1000 images (subset of LabelMe image segmentation project)
- Each image was labelled by 1, 2 or 3 workers

LabelMe and task difficulty

- Entropy is not reliable **at all**
- GLAD can't estimate a task difficulty for tasks with 1 label

---

[17] C. Garcin et al. (2021). "Pl@ntNet-300K: a plant image dataset with high label ambiguity and a long-tailed distribution". In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.

[18] F. Rodrigues and F. Pereira (2018). "Deep learning from crowds". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

"Can I use the WAUM in my framework?"

Most probably yes

- Most frameworks are built on DS model
  - the WAUM only needs a network and $\hat{\pi}^{(j)}$

The Benefits of a Model of Annotation

Rebecca J. Passonneau
Center for Computational Learning Systems
Columbia University
New York, NY USA
becky@ccls.columbia.edu

Bob Carpenter
Department of Statistics
Columbia University
New York, NY USA
carp@alias-i.com

Analysis of Minimax Error Rate for Crowdsourcing and Its Application to Worker Clustering Model

Hideaki Imamura [1,2] Issei Sato [1,2] Masashi Sugiyama [2,1]

The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)

Deep Learning from Crowds

Filipe Rodrigues, Francisco C. Pereira
Dept. of Management Engineering, Technical University of Denmark
Bygning 116B, 2800 Kgs. Lyngby, Denmark
rodr@dtu.dk, camara@dtu.dk

Learning from Crowds by Modeling Common Confusions

Zhendong Chu, Jing Ma, Hongning Wang
Department of Computer Science, University of Virginia
{zc9uy, jm3mr, hw5x}@virginia.edu

Learning From Noisy Labels By Regularized Estimation Of Annotator Confusion

Ryutaro Tanno[1] * Ardavan Saeedi[2] Swami Sankaranarayanan[2]
Daniel C. Alexander[1] Nathan Silberman[2]
[1]University College London, UK [2]Butterfly Network, New York, USA
[1] {r.tanno, d.alexander}@ucl.ac.uk [2]{asaeedi,swamiviv,nsilberman}@butterflynetinc.com

Conclusion

Take home message(s).

- Crowdsourcing is great
- …but if we judge workers, do it on tasks they can actually answer.

Conclusion

Take home message(s).

- Crowdsourcing is great
- …but if we judge workers, do it on tasks they can actually answer.
- Better data quality $\Rightarrow$ better performance (not new, but still…)
- Label uncertainty contains important information to learn!

For future you.

"I swear that, if I make a crowdsourcing experiment,
I will release both the tasks and labels"

Conclusion

Take home message(s).

- Crowdsourcing is great
- ...but if we judge workers, do it on tasks they can actually answer.
- Better data quality $\Rightarrow$ better performance (not new, but still...)
- Label uncertainty contains important information to learn!

For future you.

"I swear that, if I make a crowdsourcing experiment,
I will release both the tasks and labels"

Thank you!

References I

- (N.d.). `https://github.com/googlecreativelab/quickdraw-dataset`.
- Dawid, A. and A. Skene (1979). "Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm". In: *J. R. Stat. Soc. Ser. C. Appl. Stat.* 28.1, pp. 20–28.
- Garcin, C. et al. (2021). "Pl@ntNet-300K: a plant image dataset with high label ambiguity and a long-tailed distribution". In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.
- Han, J., P. Luo, and X. Wang (2019). "Deep self-learning from noisy labels". In: *ICCV*, pp. 5138–5147.
- Ju, C., A. Bibaut, and M. van der Laan (2018). "The relative performance of ensemble methods with deep convolutional neural networks for image classification". In: *J. Appl. Stat.* 45.15, pp. 2800–2818.
- Krizhevsky, A. and G. Hinton (2009). "Learning multiple layers of features from tiny images". In.
- Lapin, M., M. Hein, and B. Schiele (2016). "Loss functions for top-k error: Analysis and insights". In: *CVPR*, pp. 1468–1477.

- LeCun, Y. et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.
- Lee, K.-H. et al. (2018). "Cleannet: Transfer learning for scalable image classifier training with label noise". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5447–5456.
- Northcutt, C., L. Jiang, and I. Chuang (2021). "Confident learning: Estimating uncertainty in dataset labels". In: *J. Artif. Intell. Res.* 70, pp. 1373–1411.
- Pleiss, G. et al. (2020). "Identifying mislabeled data using the area under the margin ranking". In: *NeurIPS*.
- Rodrigues, F. and F. Pereira (2018). "Deep learning from crowds". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.
- Russakovsky, O. et al. (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252.

References III

- Servajean, M. et al. (2017). “Crowdsourcing thousands of specialized labels: A Bayesian active training approach”. In: *IEEE Trans. Multimed.* 19.6, pp. 1376–1391.
- Siddiqui, S. A. et al. (2022). *Metadata Archaeology: Unearthing Data Subsets by Leveraging Training Dynamics*.
- Snow, R. et al. (2008). “Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks”. In: *Conference on Empirical Methods in Natural Language Processing*. EMNLP 2008. Association for Computational Linguistics, pp. 254–263.
- Whitehill, J. et al. (2009). “Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise”. In: *NeurIPS*. Vol. 22.
- Yang, F. and S. Koyejo (2020). “On the consistency of top-k surrogate losses”. In: *ICML*, pp. 10727–10735.