Improve learning combining crowdsourced labels by weighting Areas Under the Margin

Joseph Salmon
IMAG, Univ Montpellier, CNRS
Institut Universitaire de France (IUF)

Ongoing joint work with...

- Benjamin Charlier (IMAG, Univ Montpellier, CNRS)
- Alexis Joly (Inria, LIRMM, Univ Montpellier CNRS)
- **Tanguy Lefort** (IMAG, Inria, LIRMM, Univ Montpellier, CNRS)

Improve learning combining crowdsourced labels
by
weighting Areas Under the Margin

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

Mainly joint work with:

Camille Garcin (Univ. Montpellier, IMAG)
Maximilien Servajean (Univ. Paul-Valéry-Montpellier, LIRMM, Univ. Montpellier)
Alexis Joly (Inria, LIRMM, Univ. Montpellier)

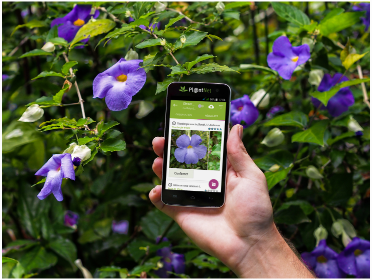
and: Pl@ntNet

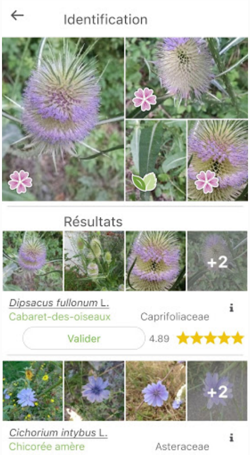
Pierre Bonnet (CIRAD, AMAP)
Antoine Affouard, **J-C. Lombardo**, **Titouan Lorieul**, **Mathias Chouet** (Inria, LIRMM, Univ. Montpellier)

- C. Garcin, A. Joly, 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*
- C. Garcin, M. Servajean, et al. (2022). "Stochastic smoothing of the top-K calibrated hinge loss for deep imbalanced classification". In: *ICML*

Plant classification with Pl@ntNet

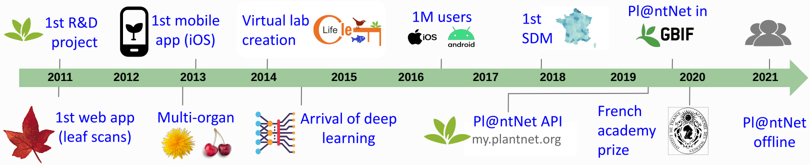
https://plantnet.org/





- ML assisted **citizen science**
- **> 40,000 species**
- **> 10,000,000 annotated images**
- **> 1Tb** of data $\Longrightarrow$ Reduction to share with community

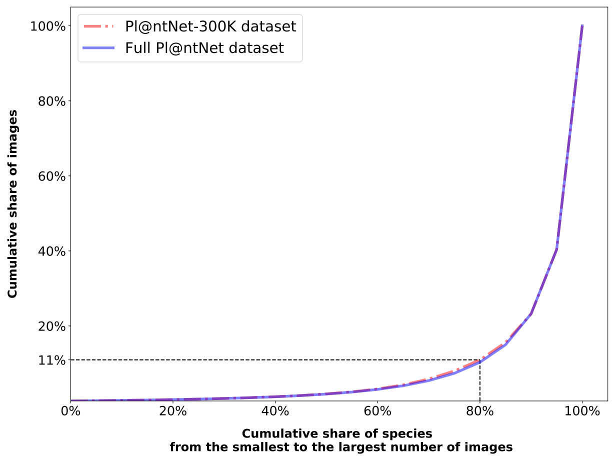
Pl@ntNet Key milestones



inria informatics mathematics · cirad · IRD · INRAE · agropolis fondation

Table of Contents

Long tailed distribution
Preserved with sampling of genera



80% of species account for only 11% of images

Intra-class variability
same label/species but very diverse images

Guizotia abyssinica — *Diascia rigescens* — *Lapageria rosea* — *Casuarina cunninghamiana* — *Freesia alba*

Plant species are challenging to model based on pictures only!

Inter-class ambiguity

different labels/species but similar images

Cirsium *rivulare*

Chaerophyllum *aromaticum*

Conostomium *kenyense*

Adenostyles *leucophylla*

Sedum *montanum*









Cirsium *tuberosum*

Chaerophyllum *temulum*

Conostomium *quadrangulare*

Adenostyles *alliariae*

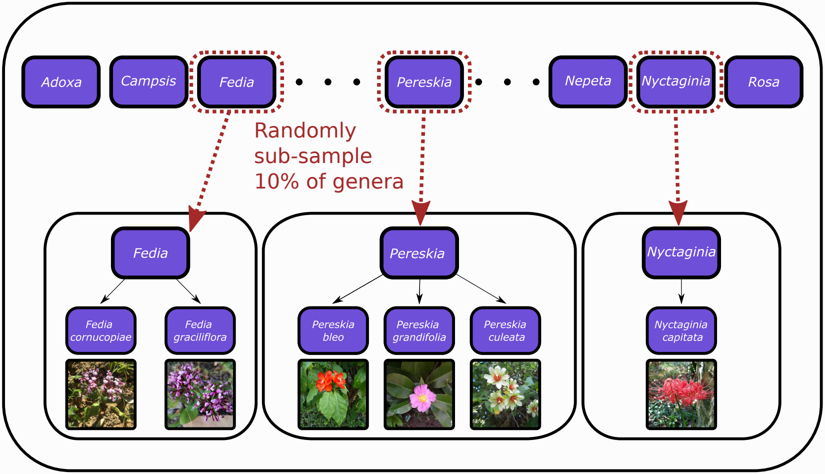
Sedum *rupestre*

Some species are visually similar (especially within genus)

Table of Contents

Construction of Pl@ntNet-300K

Subsampling of genera



Sample at genus level to preserve intra-genus ambiguity

Info / Links for Pl@ntNet-300K

- **306, 146** color images images
- Labels: **1, 081** species
- **2, 079, 003** workers (volunteers), with $\approx$ 2 labels per worker (on average)

Zenodo, 1 click download

`https://zenodo.org/record/5645731`

Code to train models:

`https://github.com/plantnet/PlantNet-300K`

Problem: Can we trust our data?

(1) A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.
(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?

…but labelling errors are common

`CIFAR10`[1]

$y^\star =$ **cat**

`Quickdraw`[2]

$y^\star =$ **T-shirt**

`MNIST`[3]

$y^\star =$ **6**

[1] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.

[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: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Popular datasets used for supervised learning (classification):
  `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

- Classical dataset: $(x_1, y_1), \dots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \dots, K\}$
- Popular datasets used for supervised learning (classification):
  `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

Questions:

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Popular datasets used for supervised learning (classification):
  `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

Questions:

- Where do the tasks come from?

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Popular datasets used for supervised learning (classification):
  `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

Questions:

- Where do the tasks come from? **Web scrapping**

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Popular datasets used for supervised learning (classification):
  `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

Questions:

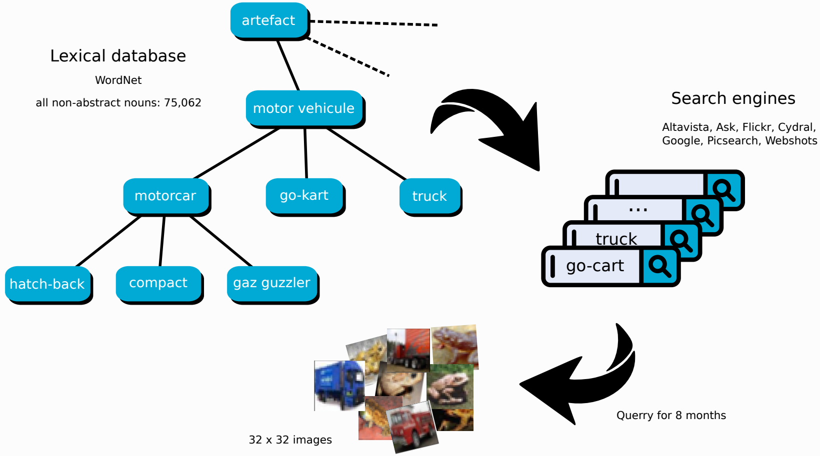
- Where do the tasks come from? **Web scrapping**
- Where do the labels come from?

- Classical dataset: $(x_1, y_1), \ldots, (x_{n_{\text{task}}}, y_{n_{\text{task}}})$
  Features/tasks $\times$ labels pairs: $(x_i, y_i) \in \mathcal{X} \times [K] = \{1, \ldots, K\}$
- Popular datasets used for supervised learning (classification): `CIFAR10`, `CIFAR100`, `ImageNet`, `MNIST`, `Quickdraw`, `LabelMe`, etc.

Questions:

- Where do the tasks come from? **Web scrapping**
- Where do the labels come from? **Crowdsourcing**

CIFAR10, an archetypal example
Step 1: data collection (80 Million Tiny Images)

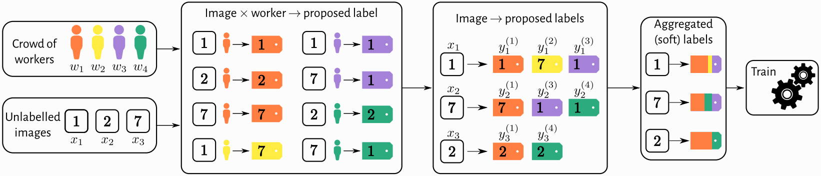


80 Million Tiny Images

Note: some issues on this process[4]

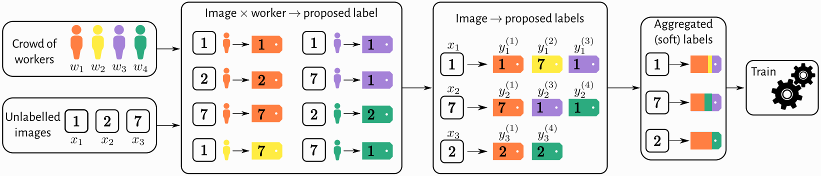
[4] V. Uday Prabhu and A. Birhane (June 2020). "Large image datasets: A pyrrhic win for computer vision?" In: *arXiv e-prints*, arXiv:2006.16923, arXiv:2006.16923.

CIFAR10, an archetypal example
Step 2: label collection and crowdsourcing



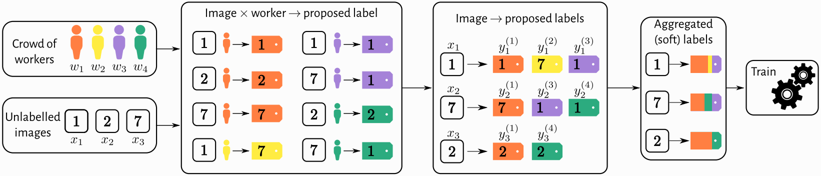
(5) A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.

CIFAR10, an archetypal example
Step 2: label collection and crowdsourcing



Quotes[5] :

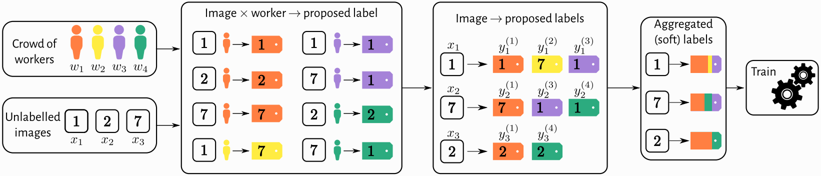
[5] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.



Quotes[5] :

- "We paid students to label a subset of the tiny images dataset[...]. The labelers were paid a fixed sum per hour spent labeling."

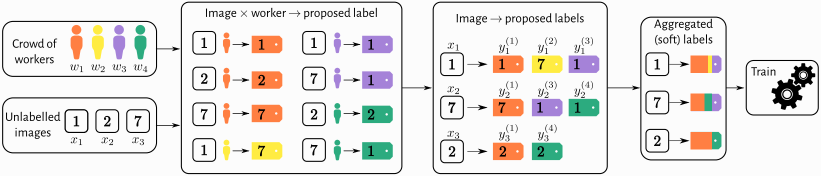
[5] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.



Quotes [5] :

- "We paid students to label a subset of the tiny images dataset[...]. The labelers were paid a fixed sum per hour spent labeling."
- "Since each image in the dataset already comes with a noisy label (the search term used to find the image), all we needed the labelers to do was to filter out the mislabeled images."

[5] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.



Quotes[5] :

- "We paid students to label a subset of the tiny images dataset[...]. The labelers were paid a fixed sum per hour spent labeling."
- "Since each image in the dataset already comes with a noisy label (the search term used to find the image), all we needed the labelers to do was to filter out the mislabeled images."
- "Furthermore, we personally verified every label submitted by the labelers" : *errare humanum est*

[5] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.

CIFAR10H[6]

Peterson *et al.* (2019) "Our final CIFAR10H behavioral dataset consists of **511,400** human categorization decisions over the n_{taks}=10,000-image testing subset of CIFAR10 (approx. 50 judgments per image)."

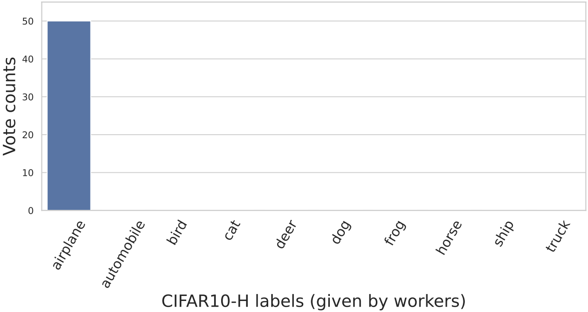
- Total number of workera: $n_{\text{worker}} = \mathbf{2{,}571}$ (via Amazon Mechanical Turk)
- Processing: every 20 trials, an obvious image is presented as an attention check, and participants who scored below 75% on these were removed from the final analysis (14 total).

Note: workers were paid $1.50 total.

[6] J. C. Peterson et al. (2019). "Human Uncertainty Makes Classification More Robust". In: *ICCV*, pp. 9617–9626.

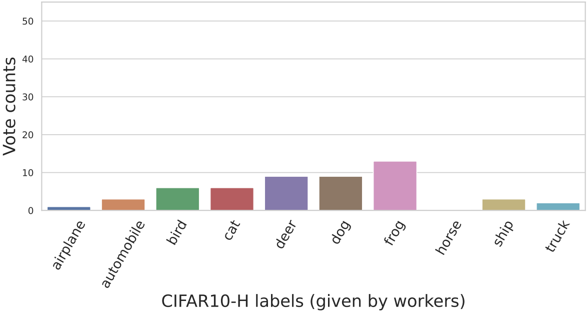
CIFAR10HPeterson et al., 2019
More human in the loop

Image #7681
CIFAR10 label: airplane



CIFAR10HPeterson et al., 2019
More human in the loop

Image #6750
CIFAR10 label: deer



CIFAR10HPeterson et al., 2019

More human in the loop

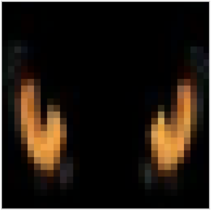
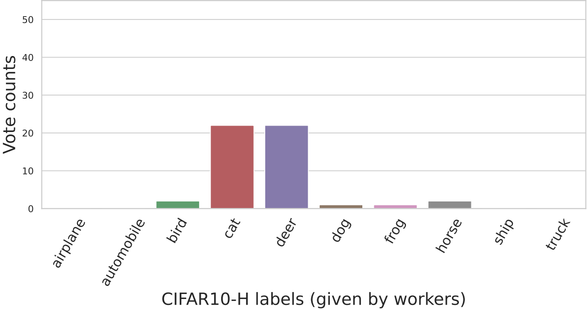


Image #9246
CIFAR10 label: cat



CIFAR10H Peterson et al., 2019

More human in the loop

Image #3724
CIFAR10 label: frog

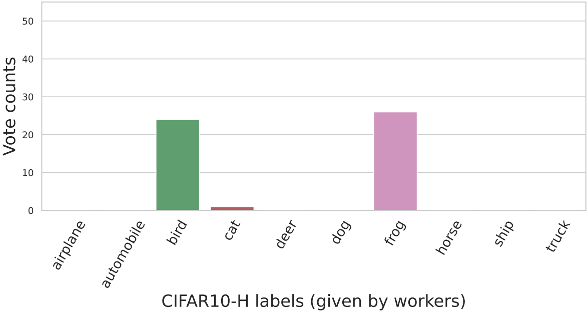
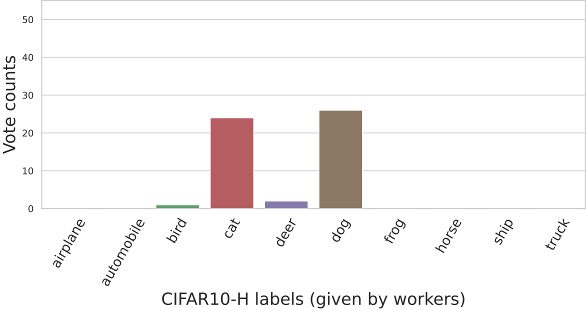
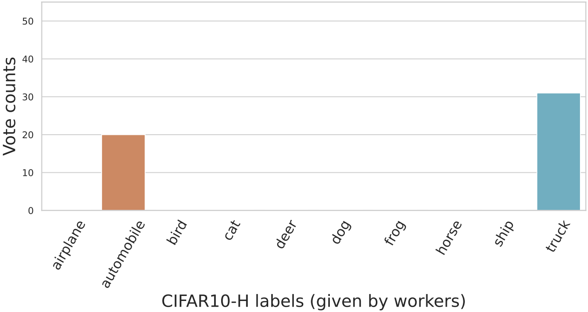


Image #1353
CIFAR10 label: cat



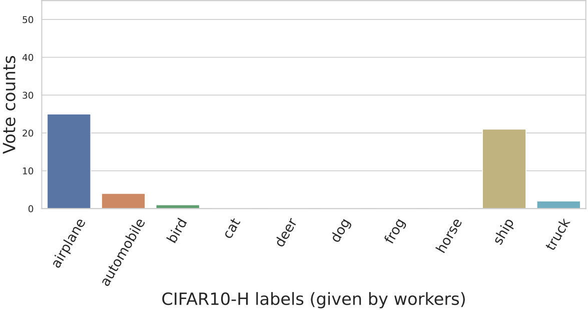
CIFAR10HPeterson et al., 2019
More human in the loop

Image #7455
CIFAR10 label: automobile



CIFAR10HPeterson et al., 2019
More human in the loop

Image #8872
CIFAR10 label: ship



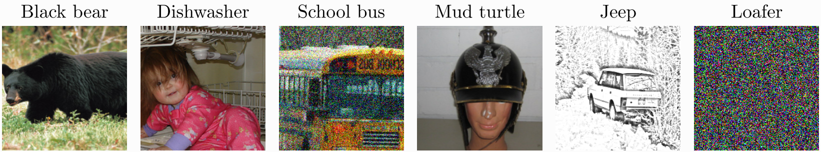
Simple strategies:

- **Majority voting (MV)**:
  naive but ineffective for borderline cases
- First label reaching a **consensus** of p workers (often $p = 5$)[7]
  $\rightarrow$ arbitrary choice of p
- Leverage label distribution, say with **entropy**:
  not always reliable (*e.g.*, with few labels), biases, psychology mechanisms spammers

Intermission : see app for entropy visualization

[7] 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.

- curated set of probes[8] in the training data (OOD=Out Of Distribution)
  e.g.,: `ImageNet`[9] $+14$ millions tasks, $K = 1000$ classes
  $$(\texttt{task}_i, \texttt{label}_i, \texttt{metadata}_i) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{M}$$



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

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

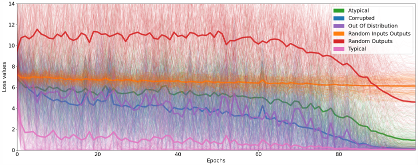
[9] O. Russakovsky et al. (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *Int. J. Comput. Vision* 115.3, pp. 211–252.

A FIRST SOLUTION: CLASSIFY THE QUALITY
IMAGENET ODDITIES

- curated set of probes[8] in the training data (OOD=Out Of Distribution)
  e.g.,: `ImageNet`[9] +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

[8] S. A. Siddiqui et al. (2022). *Metadata Archaeology: Unearthing Data Subsets by Leveraging Training Dynamics.*
[9] O. Russakovsky et al. (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *Int. J. Comput. Vision* 115.3, pp. 211–252.

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

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

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

(12) 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.

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

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

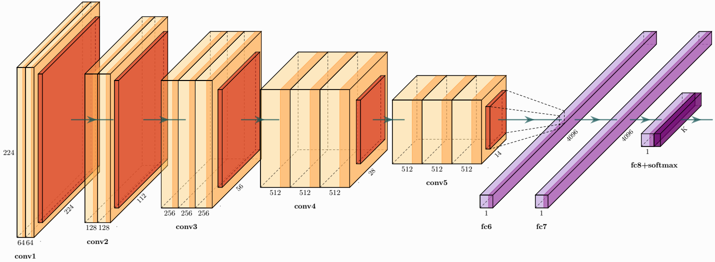
Assuming we have a hard label($\in [K]$):

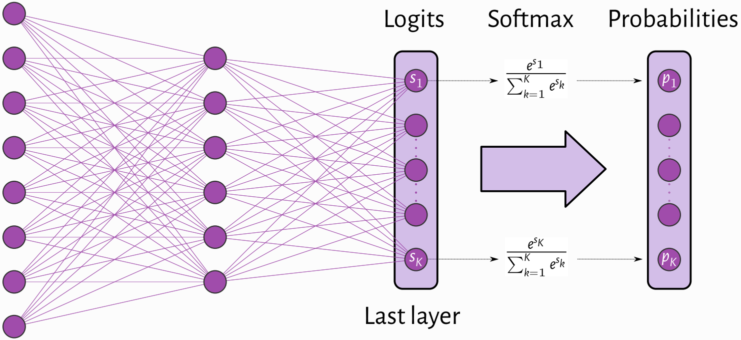
- Confident learning[(10)]: estimate joint distribution between noisy (given) and true labels (unknown)
- Self learning[(11)]: train a model + extract features and similarity metric on a subset + retrain with modified weighted loss
- Representative Sampling (`CleanNet`[(12)]): trapping set + encoders + task similarity with constraints on loss
- Our focus here: study the learning dynamic,
  - AUM[(13)] (Area Under the Margin): study margin during training

---

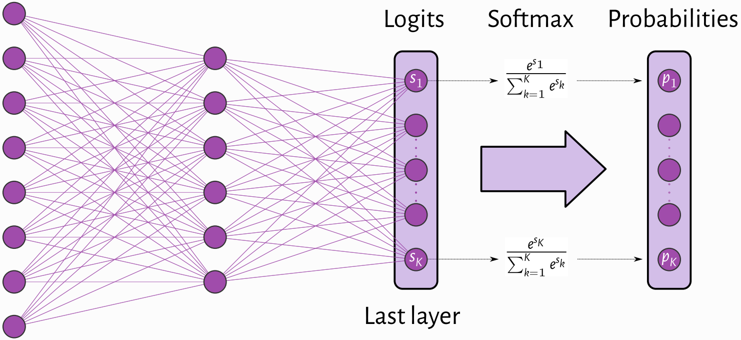
[(10)] C. Northcutt, L. Jiang, and I. Chuang (2021). “Confident learning: Estimating uncertainty in dataset labels”. In: *J. Artif. Intell. Res.* 70, pp. 1373–1411.
[(11)] J. Han, P. Luo, and X. Wang (2019). “Deep self-learning from noisy labels”. In: *ICCV*, pp. 5138–5147.
[(12)] K.-H. Lee et al. (2018). “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.
[(13)] G. Pleiss et al. (2020). “Identifying mislabeled data using the area under the margin ranking”. In: *NeurIPS*.

Deep learning
Notation mostly





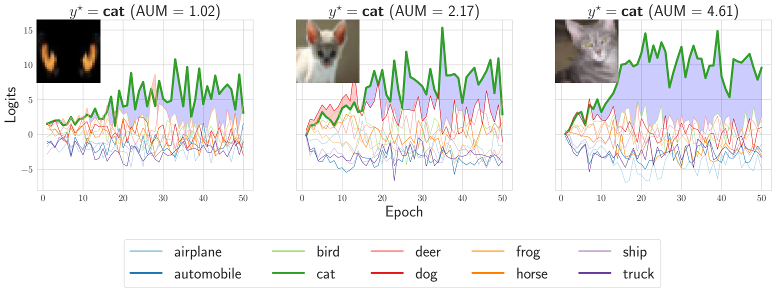
Logits
Softmax
Probabilities
s_1
s_K
$\frac{e^{s_1}}{\sum_{k=1}^{K} e^{s_k}}$
$\frac{e^{s_K}}{\sum_{k=1}^{K} e^{s_k}}$
p_1
p_K
Last layer



- From an image, get a score vector $s = (s_1, \dots, s_L)^\top \in \mathbb{R}^L$ (aka logits)
- s_k : score for class k
- Train for T epochs (say with SGD)

Area under the margins[14]

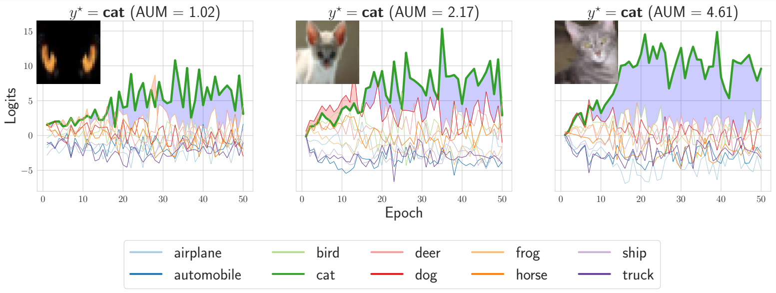
A step back with one label per task



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

Area under the margins[14]

A step back with one label per task



Motivation: the logit scores (average) value along learning epochs give insights on the task difficulty

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

Settings:

- $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times [K]$ (images, labels) pairs
- Classifier: at epoch $t \in [T]$, $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores (logits)

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

Dissecting the AUM
Building to the crowdsourced extension

Settings:

- $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times [K]$ (images, labels) pairs
- Classifier: at epoch $t \in [T]$, $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores (logits)

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

Challanging for crowdsourcing:

- No single y_i, multiple $y_i^{(j)}$: one for each worker w_j answering task x_i

Dissecting the AUM

Building to the crowdsourced extension

Settings:

- $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times [K]$ (images, labels) pairs
- Classifier: at epoch $t \in [T]$, $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores (logits)

Average = Stability

Margin between scores: content of Hinge loss

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

Score of assigned label

Other maximum score

Challanging for crowdsourcing:

- No single y_i, 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

Settings:

- $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times [K]$ (images, labels) pairs
- Classifier: at epoch $t \in [T]$, $\mathcal{C}^{(t)}(x_i) \in \mathbb{R}^K$ a vector of scores (logits)

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

Challanging for crowdsourcing:

- No single y_i, 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 $\ell \neq y_i$.

Settings:

- $(x_i, y_i^{(j)})_{i\in[n_{\mathtt{task}}], j\in[n_{\mathtt{worker}}]}$: (task,labels) crowdsourced pairs

Averaging workers AUM

$$\widetilde{\mathrm{AUM}}(x_i) = \frac{1}{|\mathcal{A}(x_i)|}\sum_{j\in\mathcal{A}(x_i)} \frac{1}{T}\sum_{t=1}^{T} \overbrace{\left[\mathcal{C}^{(t)}(x_i)_{y_i^{(j)}} - \max_{\ell\neq y_i^{(j)}} \mathcal{C}^{(t)}(x_i)_\ell \right]}^{\text{Margin between scores: content of Hinge loss}}$$

Score of assigned label by worker w_j

Other maximum score

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

Dissecting the AUM
On the way to a crowdsourced extension

Settings:

- $(x_i, y_i^{(j)})_{i\in[n_{\texttt{task}}], j\in[n_{\texttt{worker}}]}$: (task,labels) crowdsourced pairs

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

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

Reliability issue:

- Expert = random workers $\implies$ **weight** AUM per worker

- Introduce weights $s^{(j)}(x_i)$ as the trust score in worker j for task x_i

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

with $S = \sum_{j\in\mathcal{A}(x_i)} s^{(j)}(x_i)$

Modifying the margin:

- Scale effects in the scores discarded, need normalization(15)
- Better margin (in theory, for top-k classification(16))

(15) 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.

(16) 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.

Modifying the margin:

- Scale effects in the scores discarded, need normalization[15]
- Better margin (in theory, for top-k classification[16])

Notation:

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

Weighted average of AUM

Trust score of w_j for x_i

Average = Stability

Margin between scores: content of Hinge loss

$$\mathrm{WAUM}(x_i) := \frac{1}{S}\sum_{j\in\mathcal{A}(x_i)} s^{(j)}(x_i)\, \frac{1}{T}\sum_{t=1}^{T} \left[\mathrm{softmax}^{(t)}_{y_i^{(j)}}(x_i) - \mathrm{softmax}^{(t)}_{[2]}(x_i)\right]$$

Probability of assigned label by worker w_j

Second maximum probability

---

[15] 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.

[16] 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.

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 !

(17) 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 !

DS: Dawid and Skene[17]

Assumption: each worker answers independently
j-th worker **confusion matrix**: $\pi^{(j)} \in \mathbb{R}^{K\times K}$: $\quad \pi^{(j)}_{\ell,k} = \mathbb{P}(y_i^{(j)} = \ell | y_i^\star = k)$

$$y_i^{(j)} \,|\, y_i^\star = \ell \sim \mathcal{M}\mathrm{ultinomial}\left(\pi^{(j)}_{\ell\bullet}\right)$$

Note : diagonal elements of $\pi^{(j)}$ represents worker ability to be correct

[17] 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.

DS likelihood

Likelihood:

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

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

Likelihood:

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

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

Likelihood:

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

- 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:

$$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell\prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\pi^{(j)}_{\ell,k}\right]^{T_{i\ell}}$$

- 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:

$$\prod_{i \in [n_{\mathtt{task}}]} \prod_{\ell \in [K]} \left[\rho_\ell \prod_{j \in [n_{\mathtt{worker}}]} \prod_{k \in [K]} \left(\pi^{(j)}_{\ell,k} \right) \right]^{T_{i\ell}}$$

Indicator of class ℓ for task i

Prevalence of class ℓ

Likelihood:
$$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell \prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)\right]^{T_{i\ell}}$$

Probability for worker j to answer k with truth ℓ

Indicator of class ℓ for task i

Prevalence of class ℓ

Likelihood:

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

Probability for worker j to answer k with truth ℓ

---

1 **Soft labels initialization:**

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

DS vanilla algorithm

Indicator of class ℓ for task i

Prevalence of class ℓ

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

Probability for worker j to answer k with truth ℓ

Soft labels initialization:
$\forall i \in [n_{\texttt{task}}], \forall \ell \in [K],\ \hat{T}_{i\ell} = \frac{1}{|\mathcal{A}(x_i)|}\sum_{j\in\mathcal{A}(x_i)} \mathbb{1}_{\{y_i^{(j)}=\ell\}}$

while *not converged* **do**

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

DS VANILLA ALGORITHM

Likelihood:

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

Soft labels initialization:

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

while *not converged* **do**

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

$$\forall (\ell, k) \in [K]^2,\ \hat{\pi}^{(j)}_{\ell k} \leftarrow \frac{\sum_{i \in [n_{\mathtt{task}}]} \hat{T}_{i\ell} \mathbb{1}_{\{y_i^{(j)} = k\}}}{\sum_{k' \in [K]} \sum_{i' \in [n_{\mathtt{task}}]} \hat{T}_{i'\ell} \mathbb{1}_{\{y_{i'}^{(j)} = k'\}}}$$

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

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

DS VANILLA ALGORITHM

Likelihood:

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

Soft labels initialization:

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

while *not converged* **do**

```
// M-step: Get π̂ and ρ̂ assuming T̂s are known
```

$\forall(\ell,k)\in[K]^2,\ \hat{\pi}^{(j)}_{\ell k} \leftarrow \frac{\sum_{i\in[n_{\mathtt{task}}]}\hat{T}_{i\ell}\mathbb{1}_{\{y_i^{(j)}=k\}}}{\sum_{k'\in[K]}\sum_{i'\in[n_{\mathtt{task}}]}\hat{T}_{i'\ell}\mathbb{1}_{\{y_{i'}^{(j)}=k'\}}}$

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

```
// E-step: Estimate T̂s with current π̂ and ρ̂
```

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

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

GLAD[18]

Generative model of Labels, Abilities, and Difficulties

- DS assumption: errors only come from workers (no task modelling)

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

GLAD[18]

Generative model of Labels, Abilities, and Difficulties

- DS assumption: errors only come from workers (no task modelling)

GLAD: incorporating task difficulty

Model labelling errors as a function of worker ability and task difficulty:

- worker j has an ability $\alpha_j \in \mathbb{R}$
- task i has a difficulty $\beta_i \in \mathbb{R}_+^\star$

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

Note: assume uniform errors on other labels

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

Proposed scores:

- Keep the product of a worker term and a task term
- Use multidimensionality of DS confusion matrices
- Use a neural network as control agent[19]

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

Proposed scores:

- Keep the product of a worker term and a task term
- Use multidimensionality of DS confusion matrices
- Use a neural network as control agent[19]

$$s^{(j)}(x_i) = \Big\langle \underbrace{\mathrm{diag}(\hat{\pi}^{(j)})}_{\text{Worker } j \text{ overall ability } \ell} \;\Big|\; \underbrace{\mathrm{softmax}^{(T)}(x_i)}_{\text{Difficulty of task } i} \Big\rangle \in [0, 1]$$

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

- Estimate confusion matrices $\pi^{(j)} \in \mathbb{R}^{K\times K}$

- Estimate confusion matrices $\pi^{(j)} \in \mathbb{R}^{K \times K}$
- For each worker j
  - Train a network on $\left\{ (x_i, y_i^{(j)});\ x_i \text{ is answered by worker } j \right\}$
  - Compute $\mathrm{AUM}(x_i, y_i^{(j)})$ for the answered tasks x_i
  - Compute trust scores $s^{(j)}(x_i)$
  - For each task i compute $\mathrm{WAUM}(x_i)$

- Estimate confusion matrices $\pi^{(j)} \in \mathbb{R}^{K\times K}$
- For each worker j
  - Train a network on $\left\{ \left(x_i, y_i^{(j)}\right);\ x_i \text{ is answered by worker } j \right\}$
  - Compute $\mathrm{AUM}(x_i, y_i^{(j)})$ for the answered tasks x_i
  - Compute trust scores $s^{(j)}(x_i)$
  - For each task i compute $\mathrm{WAUM}(x_i)$

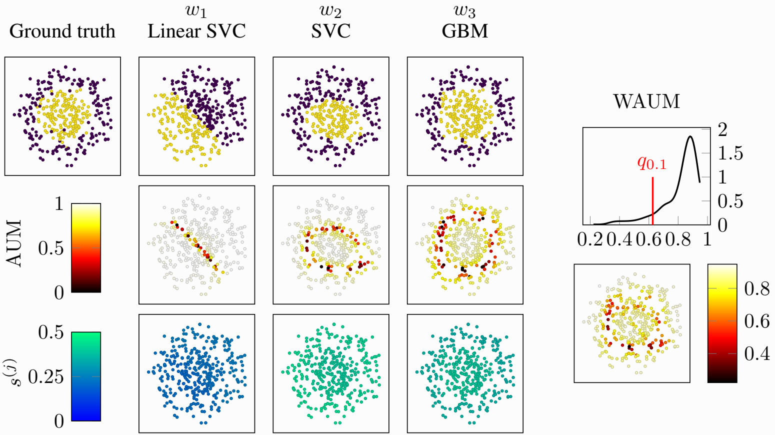
- Estimate confusion matrices $\pi^{(j)} \in \mathbb{R}^{K\times K}$
- For each worker j
  - Train a network on $\left\{\left(x_i, y_i^{(j)}\right);\ x_i \text{ is answered by worker } j\right\}$
  - Compute $\mathrm{AUM}(x_i, y_i^{(j)})$ for the answered tasks x_i
  - Compute trust scores $s^{(j)}(x_i)$
  - For each task i compute $\mathrm{WAUM}(x_i)$

Usage (for learning):

- **Prune** x_i's with $\mathrm{WAUM}(x_i)$ below quantile q_α
- **Estimate confusion matrices** $\hat{\pi}^{(j)}$ on pruned training dataset
- Get **soft labels**: normalize $\hat{y}_i = \left(\sum_{j\in\mathcal{A}(x_i)} \pi^{(j)}_{k,k} \mathbb{1}_{\{y_i^{(j)}=k\}}\right)_{k\in[K]} \in \mathbb{R}^K$
- **Train** a classifier on the pruned dataset (with soft label as above)

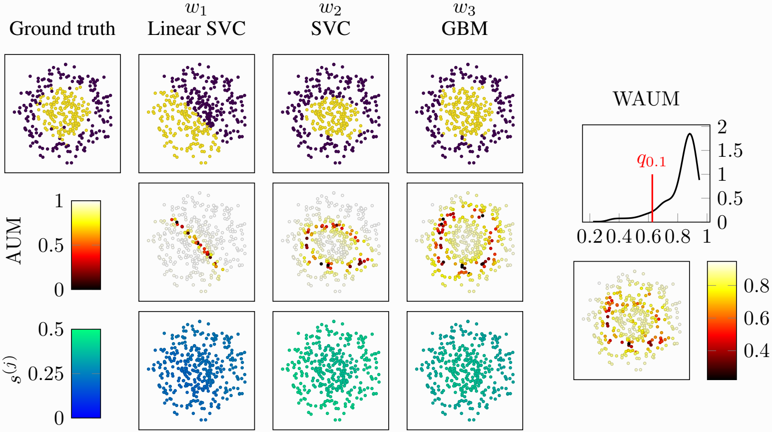
Simulation with circles

Binary setting



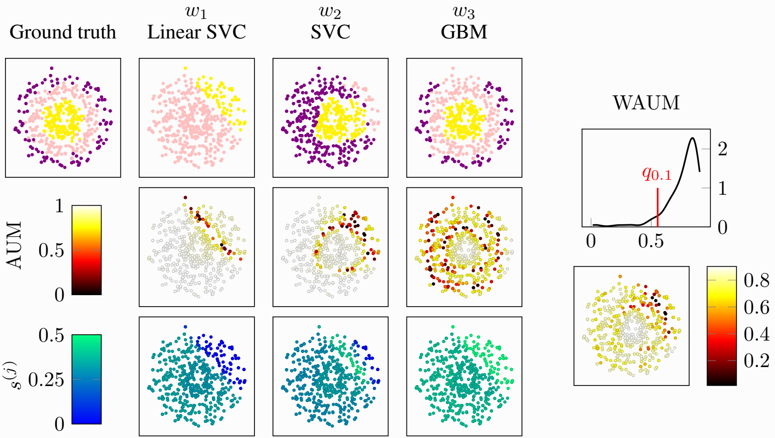
Simulation with circles

Binary setting



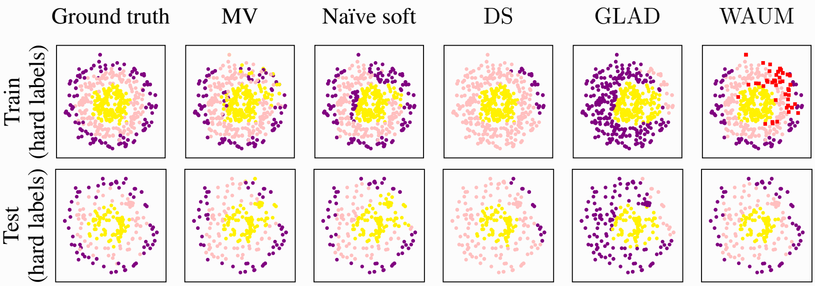
- Workers = simulated classifiers (answering 500 tasks)
- Normalized trust scores

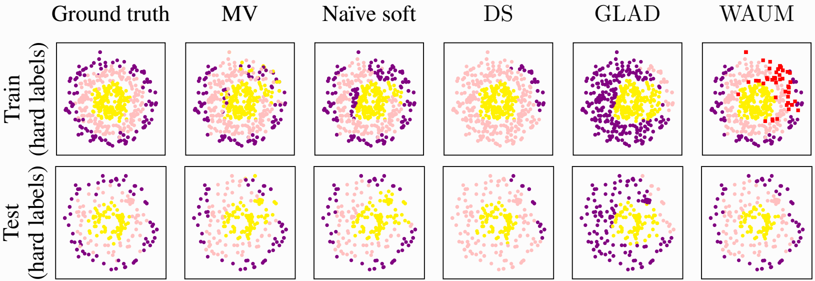
Simulation with circles
Three classes



- 3 classes with 250 tasks per class
- Normalized trust scores
- Neural Network: 3-dense layers' artificial neural network (30, 20, 20)

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





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

"3 answers per task is not enough!"

(20) C. Garcin, A. Joly, 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*.

(21) 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[20] (future work), LabelMe[21]
- LabelMe 1000 images (subset of LabelMe image segmentation project)
- Each image was labelled by 1, 2 or 3 workers

[20] C. Garcin, A. Joly, 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*.

[21] 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[20] (future work), LabelMe[21]
- LabelMe 1000 images (subset of LabelMe image segmentation project)
- Each image was labelled by 1, 2 or 3 workers

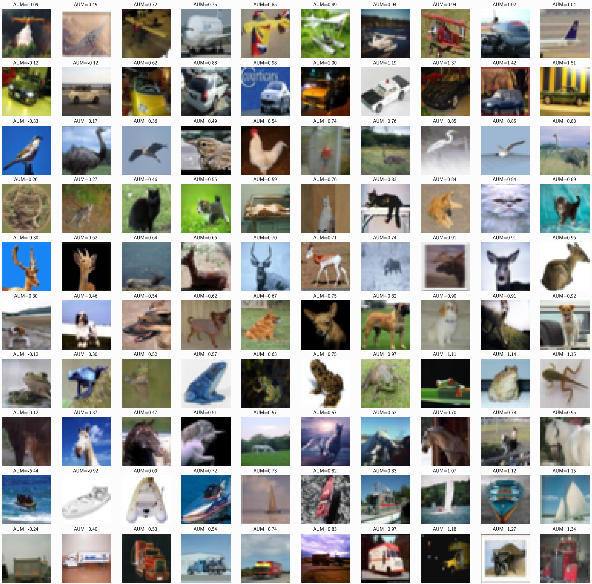
LabelMe and task difficulty

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

(20) C. Garcin, A. Joly, 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*.

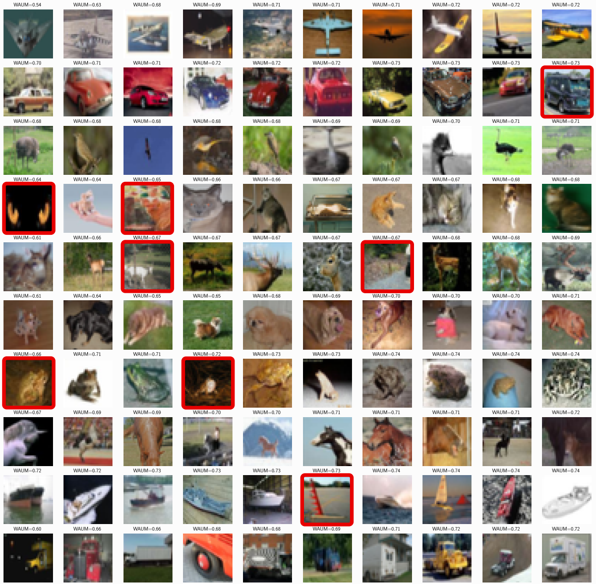
(21) F. Rodrigues and F. Pereira (2018). "Deep learning from crowds". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

Results on CIFAR10H
Improved mislabeled detections: worst AUM/WAUM



Results on CIFAR10H

Improved mislabeled detections: worst AUM/WAUM



Intermission

Back to the application of the AUM/WAUM to the CIFAR10H dataset.

Prediction performance

Table: Label recovery, generalization performance and calibration error on the CIFAR-10H dataset by a Resnet-18

Aggregation method	Test accuracy (on CIFAR10-train)	ECE (expected calibration error)
MV	69.533 ± 0.84	0.175 ± 0.00
Naive soft	**72.149** ± 2.74	**0.132** ± 0.03
DS (vanilla)	70.268 ± 0.93	0.173 ± 0.00
DS (spam identification)	70.053 ± 0.81	0.174 ± 0.0
GLAD	66.569 ± 8.48	0.173 ± 0.01
WAUM	**72.747** ± 1.93	**0.124** ± 0.00

"Can I use the WAUM in my framework?"

Most probably yes

- Most frameworks are built on DS model
  - the WAUM only needs a neural 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

Take home message(s)

- Crowdsourcing / Label uncertainty : helpful for **data curating**

Conclusion

Take home message(s)

- Crowdsourcing / Label uncertainty : helpful for **data curating**
- Improved **data quality** $\Rightarrow$ **improved learning** performance

Take home message(s)

- Crowdsourcing / Label uncertainty : helpful for **data curating**
- Improved **data quality** $\Rightarrow$ **improved learning** performance
- (Fast) "stacked" WAUM : the presented version requires **one neural network per worker** (stacked version : **one neural network per dataset**)

Conclusion

Take home message(s)

- Crowdsourcing / Label uncertainty : helpful for **data curating**
- Improved **data quality** $\Rightarrow$ **improved learning** performance
- (Fast) "stacked" WAUM : the presented version requires **one neural network per worker** (stacked version : **one neural network per dataset**)

Future work & wishful thinking

- Soon a crowdsourced module in `benchopt` `https://benchopt.github.io/`
- Pl@ntnet crowdsourced dataset: coming, but it's messy (**2M workers**, 2 labels per task on average,...)

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

Contact:

joseph.salmon@umontpellier.fr

http://josephsalmon.eu

Github: @josephsalmon

Twitter: @salmonjsph

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., A. Joly, 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*.
- Garcin, C., M. Servajean, et al. (2022). "Stochastic smoothing of the top-K calibrated hinge loss for deep imbalanced classification". In: *ICML*.
- 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*. Tech. rep. University of Toronto.
- 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.
- Peterson, J. C. et al. (2019). "Human Uncertainty Makes Classification More Robust". In: *ICCV*, pp. 9617–9626.
- 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: *Int. J. Comput. Vision* 115.3, pp. 211–252.
- Servajean, M. et al. (2017). “Crowdsourcing thousands of specialized labels: A Bayesian active training approach”. In: *IEEE Transactions on Multimedia* 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.
- Uday Prabhu, V. and A. Birhane (June 2020). “Large image datasets: A pyrrhic win for computer vision?” In: *arXiv e-prints*, arXiv:2006.16923, arXiv:2006.16923.

References IV

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.