Supervised learning by crowdsourcing

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

Ongoing work with several colleagues...

- **Tanguy Lefort** (IMAG, Inria, LIRMM, Univ Montpellier, CNRS) Ph.D. student, looking for a post-doc next year!
- Benjamin Charlier (IMAG, Univ Montpellier, CNRS)
- Alexis Joly (Inria, LIRMM, Univ Montpellier CNRS)
- Maximilien Servajean (Paul Valery University, LIRMM, Univ Montpellier CNRS)
- Axel Dubar (IMAG, Univ Montpellier, CNRS)

Problem: Can we trust our data?

(Deep) Learning pipeline with huge labeled dataset (of images):

Problem: Can we trust our data?

(Deep) Learning pipeline with huge labeled dataset (of images):

...but labeling errors are common

Problem: Can we trust our data?

(Deep) Learning pipeline with huge labeled dataset (of images):

…but labeling errors are common

CIFAR10 [(1)]

Given label: cat

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

Problem: Can we trust our data?

(Deep) Learning pipeline with huge labeled dataset (of images):

...but labeling errors are common

CIFAR10[(1)]

Given label: `cat`

Quickdraw[(2)]

Given label: `T-shirt`

[(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`)

Problem: Can we trust our data?

(Deep) Learning pipeline with huge labeled dataset (of images):

. . . but labeling errors are common

CIFAR10[(1)]

Given label: cat

Quickdraw[(2)]

Given label: T-shirt

MNIST[(3)]

Given label: 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". *Proceedings of the IEEE* 86.11, pp. 2278–2324)

- **Notation and setting**
  - Dataset : $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n_{\mathtt{train}}+n_{\mathtt{val}}+n_{\mathtt{test}}} = \mathcal{D}_{\mathtt{train}} \cup \mathcal{D}_{\mathrm{val}} \cup \mathcal{D}_{\mathrm{test}}$
  - Splitting : $|\mathcal{D}| = n_{\mathtt{train}} + n_{\mathtt{val}} + n_{\mathtt{test}}$
  - Tasks : $x_i \subset \mathcal{X}$ (images here)
  - Labels : $y_i \in [K] = \{1, \ldots, K\}$

- **Notation and setting**
  - Dataset : $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n_{\mathtt{train}}+n_{\mathtt{val}}+n_{\mathtt{test}}} = \mathcal{D}_{\mathtt{train}} \cup \mathcal{D}_{\mathrm{val}} \cup \mathcal{D}_{\mathrm{test}}$
  - Splitting : $|\mathcal{D}| = n_{\mathtt{train}} + n_{\mathtt{val}} + n_{\mathtt{test}}$
  - Tasks : $x_i \subset \mathcal{X}$ (images here)
  - Labels : $y_i \in [K] = \{1, \dots, K\}$

- **Popular datasets for classification**
  - `CIFAR10`[4]
  - `CIFAR100`[4]
  - `ImageNet`[5]
  - `MNIST`[6]
  - `Quickdraw`[7]
  - `LabelMe`[8]

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

(5) (J. Deng et al. [2009]. “ImageNet: A Large-Scale Hierarchical Image Database”. *CVPR*)

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

(7) ([n.d.]. `https://github.com/googlecreativelab/quickdraw-dataset`)

(8) (F. Rodrigues and F. Pereira [2018]. “Deep learning from crowds”. *AAAI*. vol. 32)

▶ Notation and setting

- Dataset : $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n_{\texttt{train}}+n_{\texttt{val}}+n_{\texttt{test}}} = \mathcal{D}_{\texttt{train}} \cup \mathcal{D}_{\text{val}} \cup \mathcal{D}_{\text{test}}$
- Splitting : $|\mathcal{D}| = n_{\texttt{train}} + n_{\texttt{val}} + n_{\texttt{test}}$
- Tasks : $x_i \subset \mathcal{X}$ (images here)
- Labels : $y_i \in [K] = \{1, \dots, K\}$

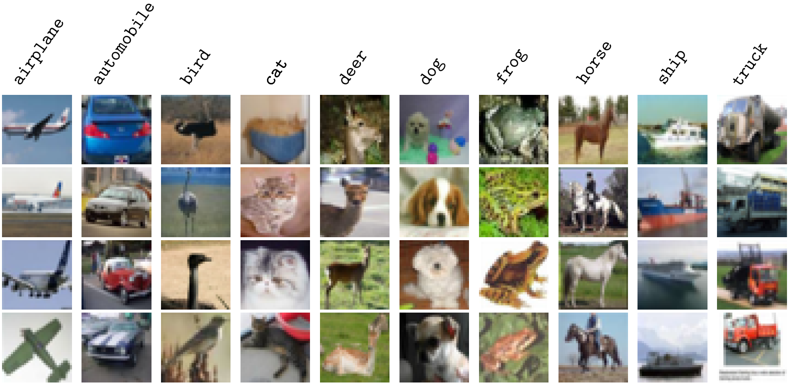
▶ Popular datasets for classification

- ▶ `CIFAR10`[4]
- ▶ `CIFAR100`[4]
- ▶ `ImageNet`[5]
- ▶ `MNIST`[6]
- ▶ `Quickdraw`[7]
- ▶ `LabelMe`[8]

(4) (A. Krizhevsky and G. Hinton [2009]. *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto)
(5) (J. Deng et al. [2009]. “ImageNet: A Large-Scale Hierarchical Image Database”. *CVPR*)
(6) (Y. LeCun et al. [1998]. “Gradient-based learning applied to document recognition”. *Proceedings of the IEEE* 86.11, pp. 2278–2324)
(7) ([n.d.]. `https://github.com/googlecreativelab/quickdraw-dataset`)
(8) (F. Rodrigues and F. Pereira [2018]. “Deep learning from crowds”. *AAAI*. vol. 32)

CIFAR10

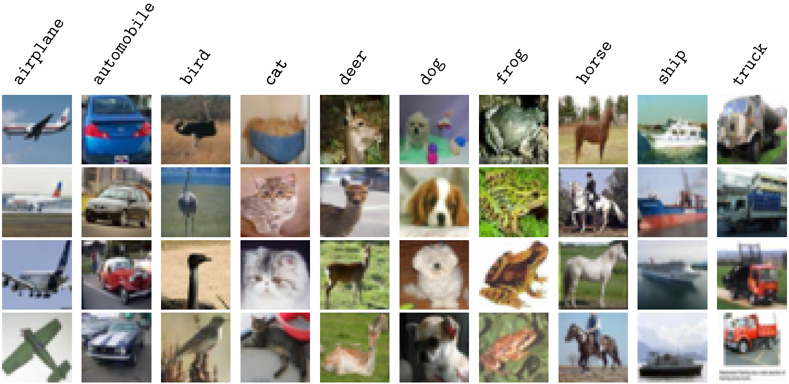
A simple dataset example for modern deep learning[9]



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

CIFAR10

A SIMPLE DATASET EXAMPLE FOR MODERN DEEP LEARNING[9]



- $K = 10$ classes
- x_i: 32×32 RGB images
- $n_{\texttt{train}} + n_{\texttt{val}} = \mathbf{50\,000}$
- $n_{\texttt{test}} = \mathbf{10\,000}$

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

Questions:

- Where do the tasks come from?

Questions:

- Where do the tasks come from? $\hookrightarrow$ **Web scrapping**

Questions:

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

Questions:

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

Dataset construction
How do we create such a dataset?

Questions:

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

Notation:

- Tasks: $\mathcal{X}_{\texttt{train}} = \{x_1, \dots, x_{n_{\texttt{task}}}\}$
- True labels: $(y_i^\star)_{i\in[n_{\texttt{task}}]}$ unobserved
- Workers: $(w_j)_{j\in[n_{\texttt{worker}}]}$, label some images
- Label answered by worker w_j for a task x_i: $y_i^{(j)} \in [K]$
- **A**nnotators set: $\mathcal{A}(x_i) = \{j \in [n_{\texttt{worker}}] : \text{worker } w_j \text{ labeled task } x_i\}$

$$\mathcal{D}_{\texttt{train}} = \bigcup_{i=1}^{n_{\texttt{task}}} \left\{ \left(x_i, \left(y_i^{(j)}\right)\right) \text{ for } j \in \mathcal{A}(x_i) \right\}$$

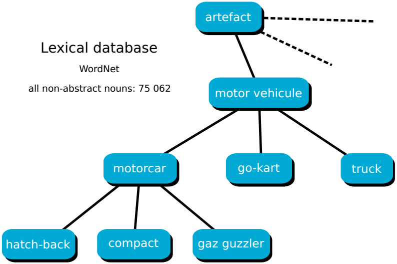
CIFAR10, an archetypal example

Step 1: data collection (80 Million Tiny Images)

Lexical database

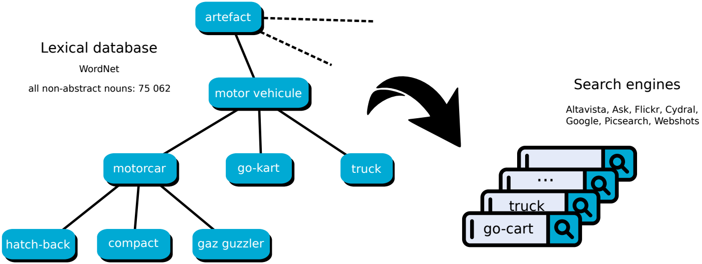
WordNet

all non-abstract nouns: 75 062

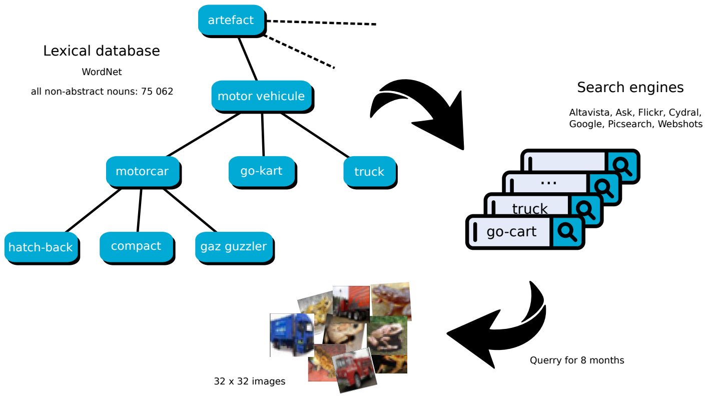


CIFAR10, an archetypal example

Step 1: data collection (80 Million Tiny Images)



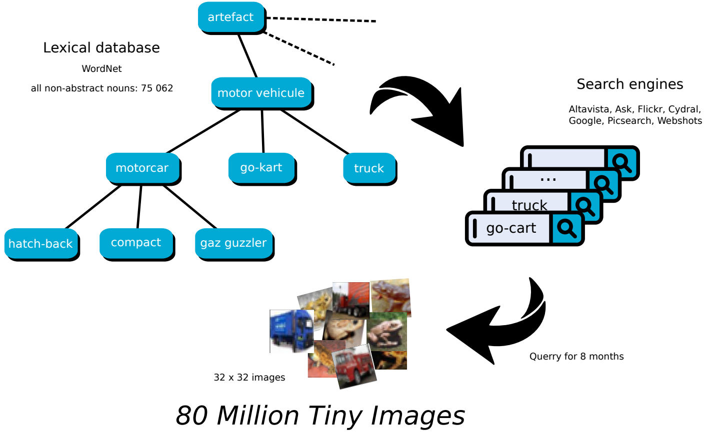
CIFAR10, an archetypal example
Step 1: data collection (80 Million Tiny Images)



80 Million Tiny Images

CIFAR10, an archetypal example

Step 1: data collection (80 Million Tiny Images)

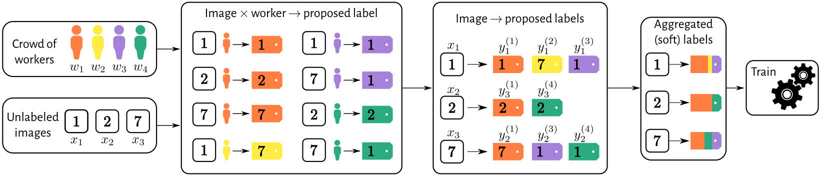


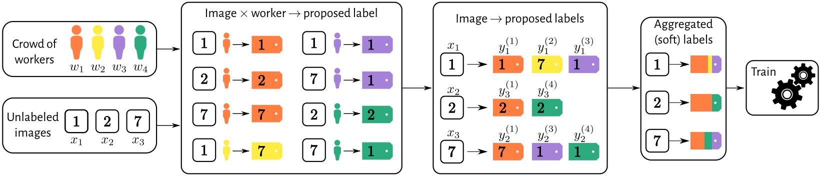
Many issues raised[10]: opacity, anonymity (face search/reverse image search), perpetuate stereotypes, etc.

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

CIFAR10, an archetypal example

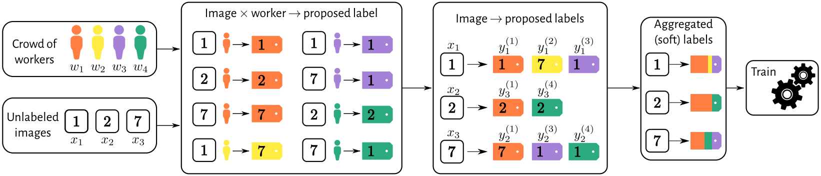
Step 2: label collection and crowdsourcing





Krizhevsky and Hinton[11] :

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

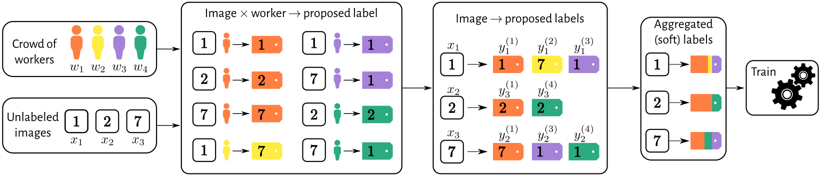


Krizhevsky and Hinton[11] :

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

---

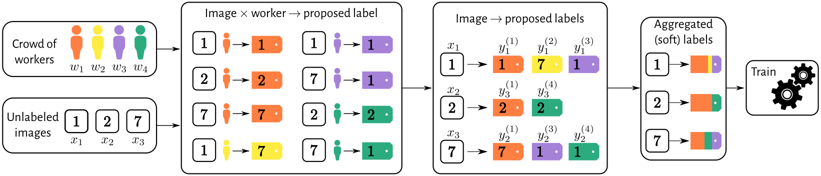
[11] A. Krizhevsky and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.



Krizhevsky and Hinton[11] :

- "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."

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



Krizhevsky and Hinton[11] :

- "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*

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

CIFAR-10H[13]

Peterson *et al.* (2019): "Our final CIFAR-10H behavioral dataset consists of **511 400** human categorization decisions over the **10 000**-image testing subset of CIFAR10 (approx. 50 judgments per image)."

- Total number of workers: $n_{\text{worker}} = \mathbf{2571}$ (via Amazon Mechanical Turk)
- **Processing**: (After an initial training phase) 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, according to the authors...we could not reproduce that).

Note: workers were paid $1.50 (average completion time ≈ 5 mn); poor worker conditions[12]

For learning: we will consider $n_{\texttt{train}} = 9500$ and $n_{\texttt{val}} = 500$

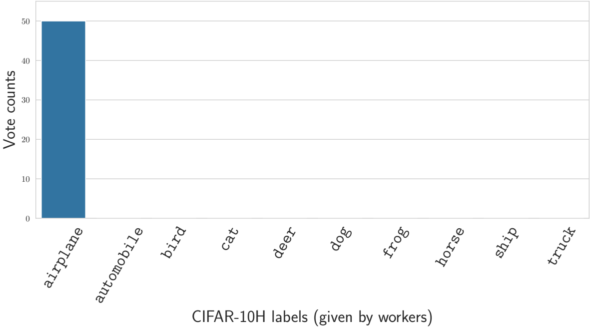
[12] https://time.com/6247678/openai-chatgpt-kenya-workers/

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

CIFAR-10H

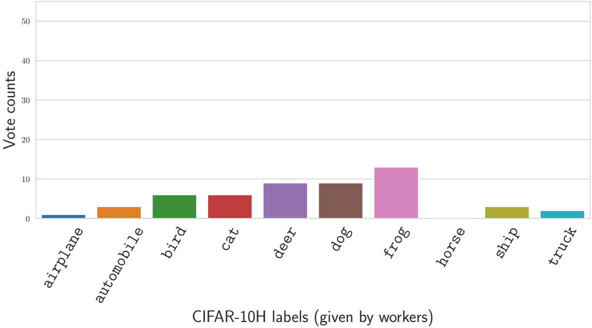
More human in the loop

Image # 7681
CIFAR-10 label: `airplane`



CIFAR-10H
More human in the loop

Image # 6750
CIFAR-10 label: deer



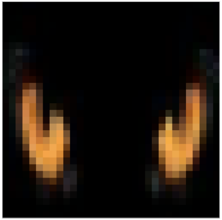


Image # 9246
CIFAR-10 label: cat

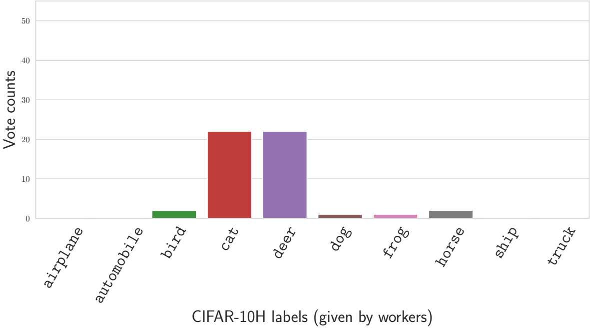


Image # 3724
CIFAR-10 label: `frog`

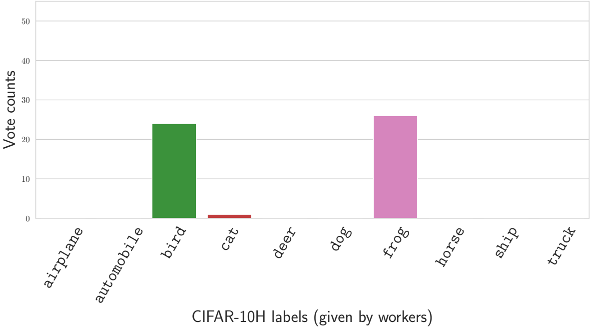
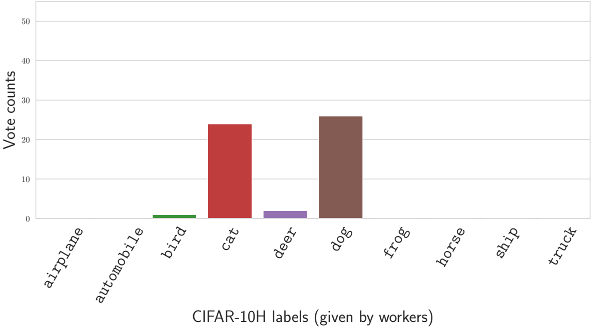


Image # 1353
CIFAR-10 label: cat



CIFAR-10H

More human in the loop

Image # 7455
CIFAR-10 label: `automobile`

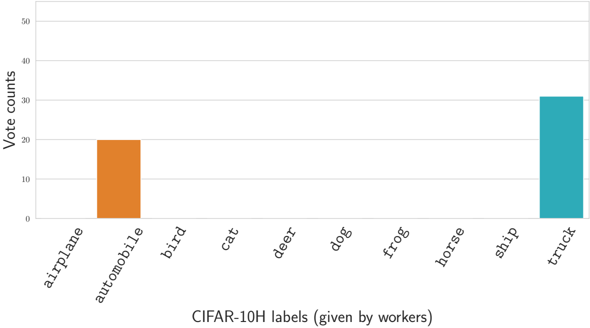
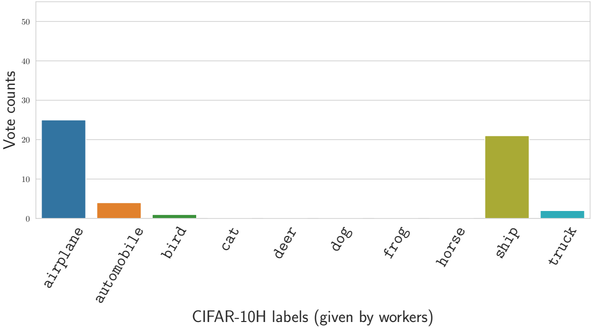
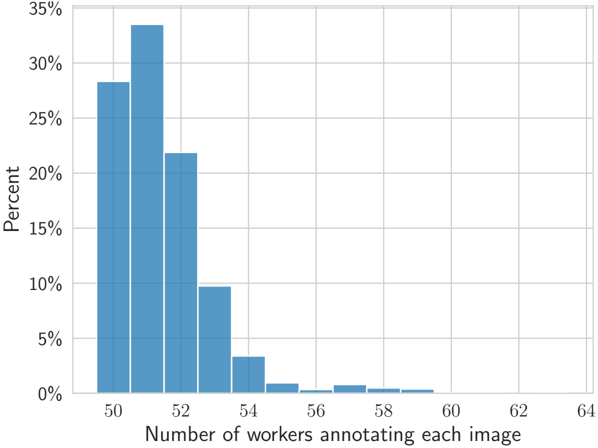


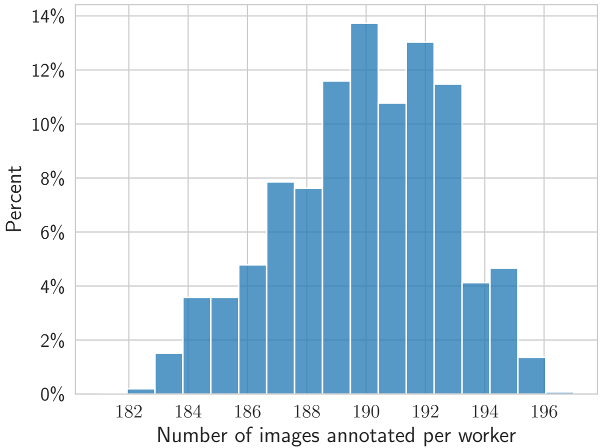
Image # 8872
CIFAR-10 label: `ship`





Feedback effort per task distribution

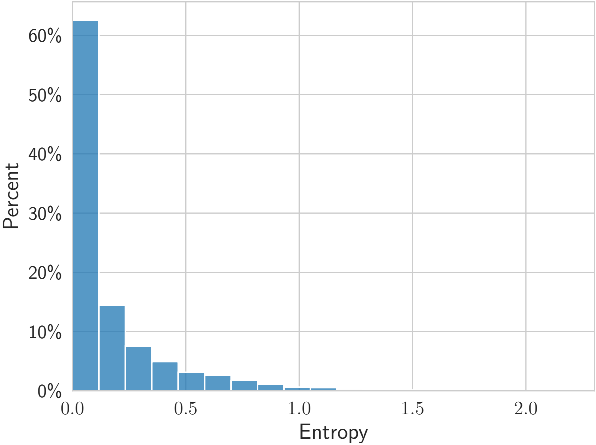
CIFAR-10H: DATASET VISUALIZATION
Statistics on our training set ($n_{\texttt{train}} = 9\,500$)



Load per worker distribution

CIFAR-10H: dataset visualization

Statistics on our training set ($n_{\mathtt{train}} = 9\,500$)



Naive soft labels, entropy distribution

Definition: Majority Voting (MV)

Majority Voting outputs the most answered label:

$$\forall x_i \in \mathcal{X}_{\texttt{train}}, \quad \hat{y}_i^{\mathrm{MV}} = \operatorname*{arg\,max}_{k\in[K]} \Big(\sum_{j\in\mathcal{A}(x_i)} \mathbb{1}_{\{y_i^{(j)}=k\}} \Big)$$

Properties:

- ✓ simple
- ✓ adapted for any number of workers
- ✓ usually efficient, often few labelers sufficient (say[14] <5)
- ✗ ineffective for borderline cases
- ✗ suffer from spammers / adversarial workers

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

Definition: Weighted Majority Voting (WMV)

Majority voting but weighted by a confidence score per worker w_j:

$$\forall x_i \in \mathcal{X}_{\texttt{train}}, \quad \hat{y}_i^{\mathrm{WMV}} = \underset{k \in [K]}{\arg\max} \left(\sum_{j \in \mathcal{A}(x_i)} \alpha_j \mathbb{1}_{\{y_i^{(j)} = k\}} \right)$$

$\alpha_j > 0$: reflects the confidence in worker w_j

- ✓ simple
- ✓ adapted for any number of workers
- ✓ usually efficient
- ✓ can leverage expert workers
- ✗ ineffective for borderline cases
- ✗ suffer from spammers / adversarial workers
- ✗ requires prior knowledge of the workers

Notation : for $z \in \mathbb{R}_+^d, \forall i \in [d]$, $\quad \mathrm{Norm}(z)_i = \dfrac{z_i}{\sum_{i'=1}^{d} z_{i'}}$

Definition: Naive Soft (NS) labels

Naive soft outputs the empirical distribution of the answered votes:

$$\forall x_i \in \mathcal{X}_{\texttt{train}}, \quad \hat{y}_i^{\mathrm{NS}} = \mathrm{Norm}(\tilde{y}_i), \quad \text{where } \tilde{y}_i = \Big(\sum_{j \in \mathcal{A}(x_i)} \mathbb{1}_{\{y_i^{(j)} = k\}} \Big)_{k \in [K]}$$

- ✓ simple
- ✓ adapted for any number of workers
- ✓ can reflect workers variability & task ambiguity
- ✗ suffer from spammers/adversarial workers

Dawid and Skene[15] (DS)

Assumption: each worker answers independently
The j-th worker has is own **confusion matrix** : $\pi^{(j)} \in \mathbb{R}^{K\times K}$

$$\pi^{(j)}_{\ell,k} = \mathbb{P}(y_i^{(j)} = k | y_i^\star = \ell)$$

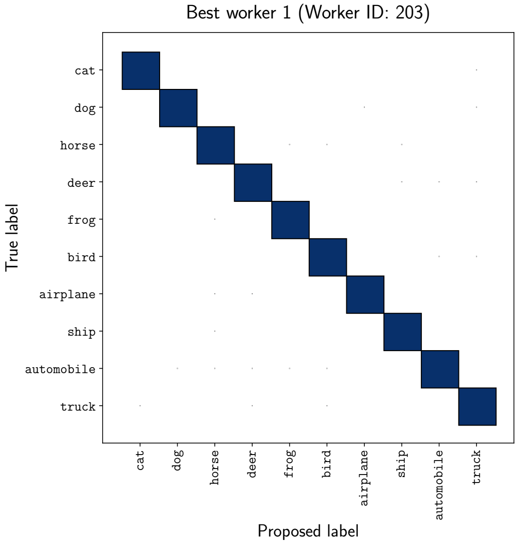
Conditionally on the true label, the j-th worker answers as follows:

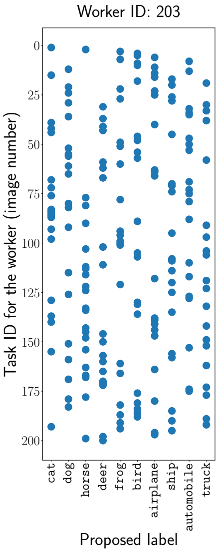
$$y_i^{(j)} \,|\, y_i^\star = \ell \sim \mathcal{M}\mathrm{ultinomial}\left(\pi^{(j)}_{\ell,:}\right)$$

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

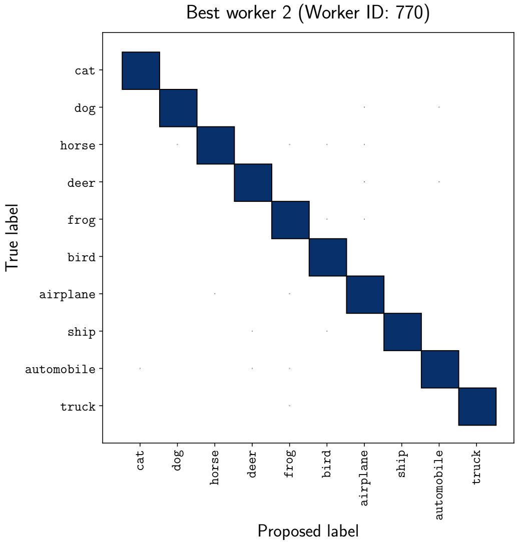
(Estimated) confusion matrices

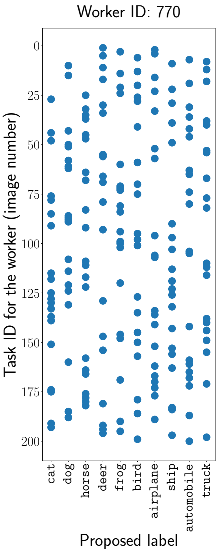
Illustration and interpretation





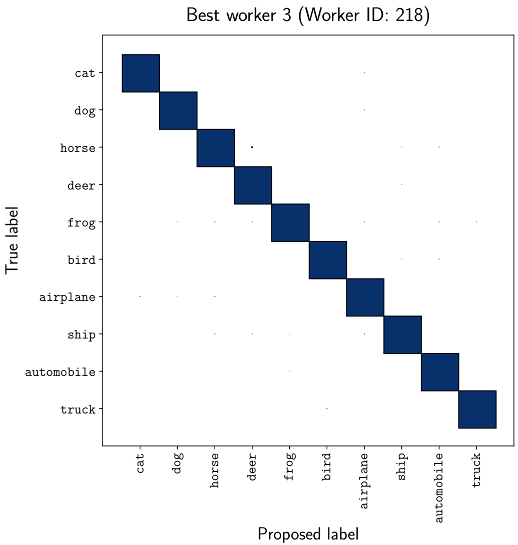
(Estimated) confusion matrices
Illustration and interpretation

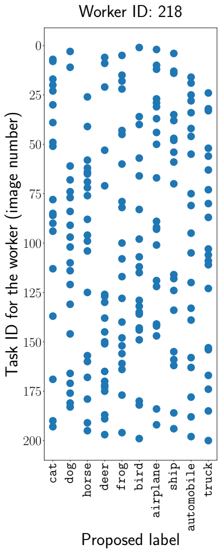




(Estimated) confusion matrices

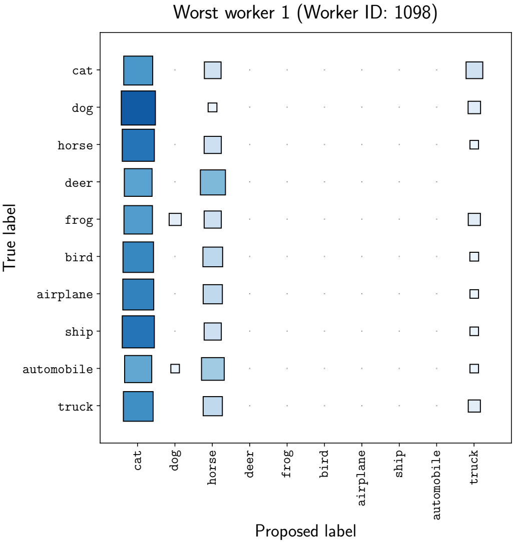
Illustration and interpretation

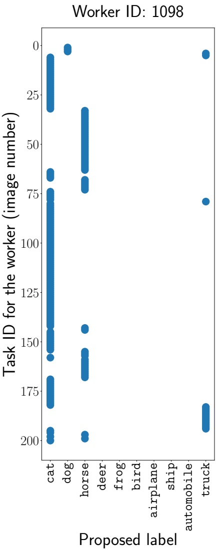


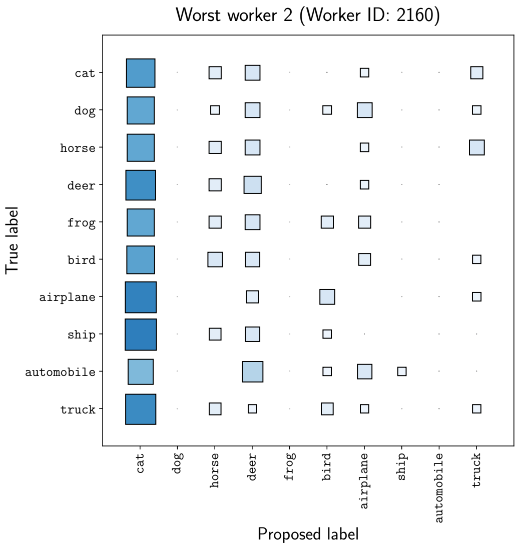


(Estimated) confusion matrices

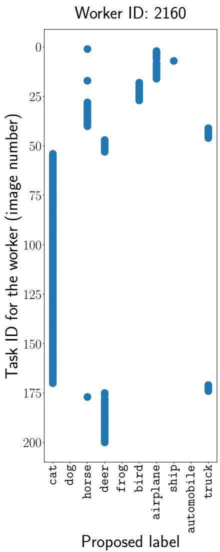
Illustration and interpretation



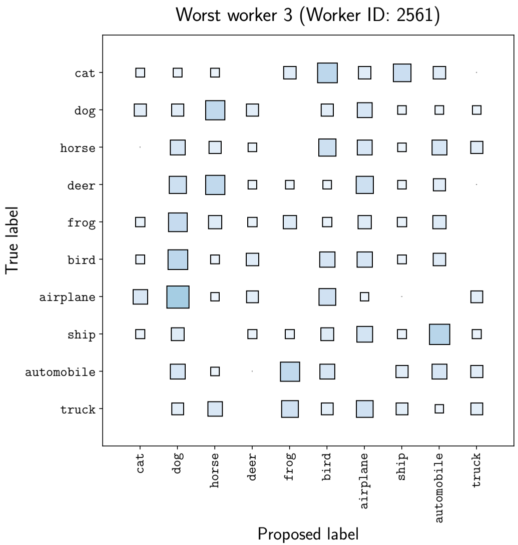


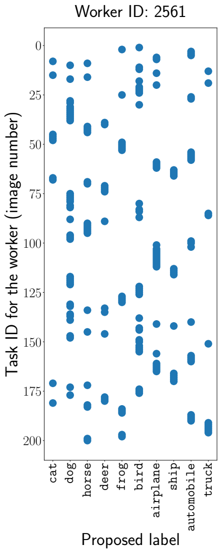


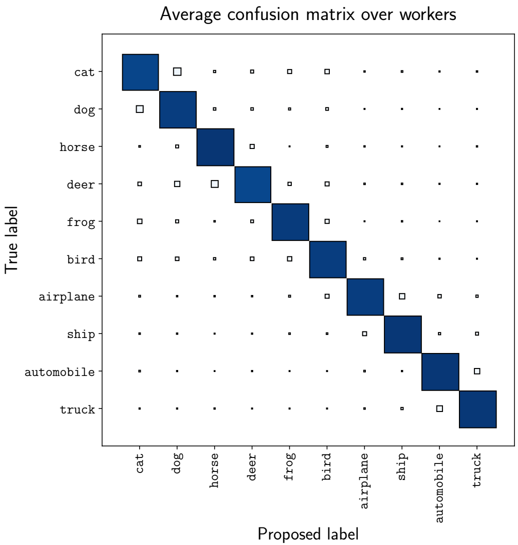
Worst worker 2 (Worker ID: 2160)
True label
cat
dog
horse
deer
frog
bird
airplane
ship
automobile
truck
Proposed label



Worker ID: 2160
Task ID for the worker (image number)
0
25
50
75
100
125
150
175
200
cat
dog
horse
deer
frog
bird
airplane
ship
automobile
truck
Proposed label


Worst worker 3 (Worker ID: 2561)
True label
cat
dog
horse
deer
frog
bird
airplane
ship
automobile
truck
Proposed label


Worker ID: 2561
Task ID for the worker (image number)
0
25
50
75
100
125
150
175
200
cat
dog
horse
deer
frog
bird
airplane
ship
automobile
truck
Proposed label



Average confusion matrix over workers
True label
Proposed label
cat
dog
horse
deer
frog
bird
airplane
ship
automobile
truck

Likelihood:

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

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

Likelihood:

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

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

Likelihood:

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

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

Likelihood:

$$\prod_{i\in[n_{\texttt{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{T_{i,\ell}}$$

- Mutlinomial with 1 task, 1 worker and 1 answer conditioned on $y_i^\star=\ell$
- Multiple workers answer independently
- Remove conditioning assumption on $y_i^\star$: $\mathbb{P}(y_i^\star=\ell)=\rho_\ell$ (**prevalence**)
- Tasks independence and $T_{i,\ell}=\mathbb{1}_{\{y_i^\star=\ell\}}$ (1 if task i has true label ℓ, 0 otherwise)

DS algorithm

Likelihood: $$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell \prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{T_{i,\ell}}$$

Likelihood:

$$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\Bigg[\underbrace{\rho_\ell}_{\text{Prevalence of class }\ell}\prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\Big(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Big)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\Bigg]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class }\ell\text{ for task } i}}$$

Indicator of class ℓ for task i

Prevalence of class ℓ

Likelihood: $$\prod_{i\in[n_{\mathrm{task}}]}\prod_{\ell\in[K]}\left[\boxed{\rho_\ell}\prod_{j\in[n_{\mathrm{worker}}]}\prod_{k\in[K]}\left(\boxed{\pi^{(j)}_{\ell,k}}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{\boxed{T_{i,\ell}}}$$

Probability for worker j to answer k with truth ℓ

a) Estimate $\rho\in\Delta_{K-1}:=\{p\in\mathbb{R}^K,\sum_{k=1}^K p_k=1, p_k\geq 0\}$ assuming **known** $T_{i,\ell}$s and the constraints $\sum_{\ell\in[K]}T_{i,\ell}=1$ for all i

$$\hat\rho\in\underset{\rho\in\Delta_{K-1}}{\arg\max}\left(\log\prod_{i\in[n_{\mathrm{task}}]}\prod_{\ell\in[K]}\left[\rho_\ell\prod_{j\in[n_{\mathrm{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{T_{i,\ell}}\right)$$

$$\iff\hat\rho\in\underset{\rho\in\Delta_{K-1}}{\arg\max}\sum_{i\in[n_{\mathrm{task}}]}\sum_{\ell\in[K]}T_{i,\ell}\log\left[\rho_\ell\prod_{j\in[n_{\mathrm{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]$$

$$\iff\hat\rho\in\underset{\rho\in\Delta_{K-1}}{\arg\max}\sum_{i\in[n_{\mathrm{task}}]}\sum_{\ell\in[K]}T_{i,\ell}\log(\rho_\ell)$$

$$\iff\hat\rho=\frac{1}{n_{\mathrm{task}}}\sum_{i\in[n_{\mathrm{task}}]}\hat T_{i,:}$$ (use Lagrange multipliers to get the solution)

Likelihood:

$$\prod_{i\in[n_{\text{task}}]}\prod_{\ell\in[K]}\left[\overbrace{\rho_\ell}^{\text{Prevalence of class }\ell}\prod_{j\in[n_{\text{worker}}]}\prod_{k\in[K]}\left(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class }\ell\text{ for task } i}}$$

b) Estimate $\pi^{(j)}_{\ell,:}\in\Delta_{K-1}$ assuming **known** $T_{i,\ell}$s and the constraints $\sum_{\ell\in[K]}T_{i,\ell}=1$ for all i

$$\hat{\pi}^{(j)}_{\ell,:}\in\operatorname*{arg\,max}_{\pi^{(j)}\in\Delta_{K-1}}\left(\log\prod_{i\in[n_{\text{task}}]}\prod_{\ell'\in[K]}\left[\rho_{\ell'}\prod_{j'\in[n_{\text{worker}}]}\prod_{k\in[K]}\left(\pi^{(j')}_{\ell',k}\right)^{\mathbb{1}_{\{y_i^{(j')}=k\}}}\right]^{T_{i,\ell'}}\right)$$

$$\iff\hat{\pi}^{(j)}_{\ell,:}\in\operatorname*{arg\,max}_{\pi^{(j)}\in\Delta_{K-1}}\sum_{i\in[n_{\text{task}}]}T_{i,\ell}\log\left[\rho_\ell\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]$$

$$\iff\hat{\pi}^{(j)}_{\ell,:}\in\operatorname*{arg\,max}_{\pi^{(j)}\in\Delta_{K-1}}\sum_{i\in[n_{\text{task}}]}\sum_{k\in[K]}T_{i,\ell}\cdot\mathbb{1}_{\{y_i^{(j)}=k\}}\log(\pi^{(j)}_{\ell,k})$$

$$\iff\hat{\pi}^{(j)}_{\ell,:}=\sum_{i\in[n_{\text{task}}]}T_{i,\ell}\cdot\mathbb{1}_{\{y_i^{(j)}=:\}}\Big/\sum_{i\in[n_{\text{task}}]}\sum_{k'\in[K]}T_{i,\ell}\cdot\mathbb{1}_{\{y_i^{(j)}=k'\}}$$

DS algorithm

Likelihood:

$$\prod_{i\in[n_{\texttt{task}}]}\prod_{\ell\in[K]}\left[\overbrace{\rho_\ell}^{\text{Prevalence of class }\ell}\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\left(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class } \ell \text{ for task } i}}$$

c) Estimate $T_{i,\ell}$s as probabilities with ρ and $\pi^{(j)}$s known, with the constraints $\sum_{\ell\in[K]} T_{i,\ell} = 1$ for all i

$$\begin{aligned}
\hat{T}_{i,\ell} &= \mathbb{P}(T_{i,\ell}=1|\mathcal{D}_{\texttt{train}}) \\
&\propto \mathbb{P}(\mathcal{D}_{\texttt{train}}|T_{i,\ell}=1)\mathbb{P}(T_{i,\ell}=1) \\
&\propto \prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\cdot\rho_\ell \\
&\propto \prod_{j\in\mathcal{A}(x_i)}\prod_{k\in[K]}\left(\pi^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\cdot\rho_\ell
\end{aligned}$$

Likelihood:

$$\prod_{i\in[n_{\mathtt{task}}]}\prod_{\ell\in[K]}\Bigg[\underbrace{\rho_\ell}_{\text{Prevalence of class }\ell}\prod_{j\in[n_{\mathtt{worker}}]}\prod_{k\in[K]}\Big(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Big)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\Bigg]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class }\ell\text{ for task } i}}$$

1 **Soft labels initialization:**

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

Likelihood:

$$\prod_{i\in[n_{\texttt{task}}]}\prod_{\ell\in[K]}\Bigg[\overbrace{\rho_\ell}^{\text{Prevalence of class } \ell}\prod_{j\in[n_{\texttt{worker}}]}\prod_{k\in[K]}\Big(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Big)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\Bigg]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class } \ell \text{ for task } i}}$$

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

Likelihood:

$$\prod_{i\in[n_{\text{task}}]}\prod_{\ell\in[K]}\left[\overbrace{\boxed{\rho_\ell}}^{\text{Prevalence of class } \ell}\prod_{j\in[n_{\text{worker}}]}\prod_{k\in[K]}\Bigg(\underbrace{\boxed{\pi^{(j)}_{\ell,k}}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Bigg)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\right]^{\overbrace{\boxed{T_{i,\ell}}}^{\text{Indicator of class } \ell \text{ for task } i}}$$

Soft labels initialization:

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

while *not converged* **do**

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

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

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

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

Likelihood:

$$\prod_{i\in[n_{\text{task}}]}\prod_{\ell\in[K]}\Bigg[\overbrace{\rho_\ell}^{\text{Prevalence of class }\ell}\prod_{j\in[n_{\text{worker}}]}\prod_{k\in[K]}\Big(\underbrace{\pi^{(j)}_{\ell,k}}_{\text{Probability for worker } j \text{ to answer } k \text{ with truth } \ell}\Big)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}\Bigg]^{\overbrace{T_{i,\ell}}^{\text{Indicator of class }\ell\text{ for task } i}}$$

Soft labels initialization:

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

while *not converged* **do**

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

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

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

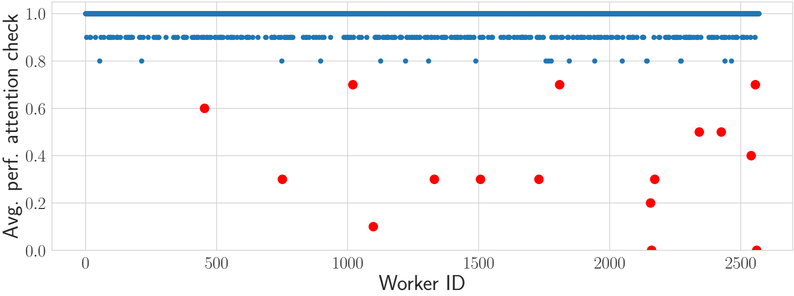
  // **E-step:** Estimate $\hat{T}$s knowing $\hat{\pi}$ and $\hat{\rho}$

  $\forall(i,\ell)\in[n_{\text{task}}]\times[K],\ \hat{T}_{i\ell}\leftarrow\frac{\prod_{j\in\mathcal{A}(x_i)}\prod_{k\in[K]}\hat{\rho}_\ell\cdot\left(\hat{\pi}^{(j)}_{\ell,k}\right)^{\mathbb{1}_{\{y_i^{(j)}=k\}}}}{\sum_{\ell'\in[K]}\prod_{j'\in\mathcal{A}(x_i)}\prod_{k'\in[K]}\hat{\rho}_{\ell'}\cdot\left(\hat{\pi}^{(j')}_{\ell',k'}\right)^{\mathbb{1}_{\{y_i^{(j')}=k'\}}}}$

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

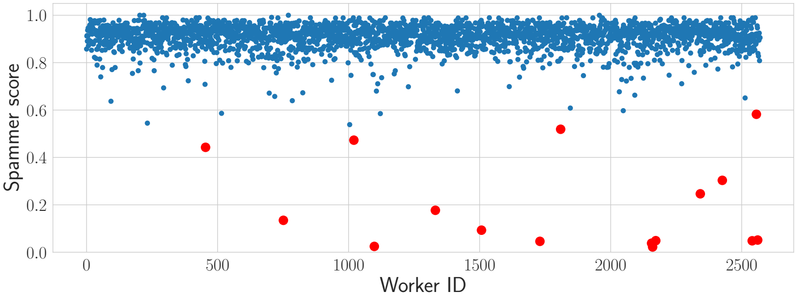
Sorting workers by quality
Use case on CIFAR10H

- Use attention check / Trapping sets: 10 images per worker (out of 200) whose true label is known $\implies$ get an average score for each worker (red: 16 workers < 0.8)



- Use spammer score[16]: measure the distance between $\hat{\pi}^j$ and rank 1 matrices (since a spammer has a distribution of answers independent of the true label)

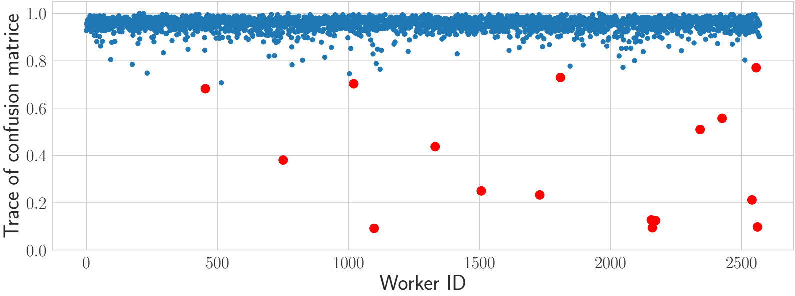
$$\min_{v_j \in \mathbb{R}^K} \left\| \hat{\pi}^{(j)} - \mathbf{1}_K v_j \right\|_F^2$$



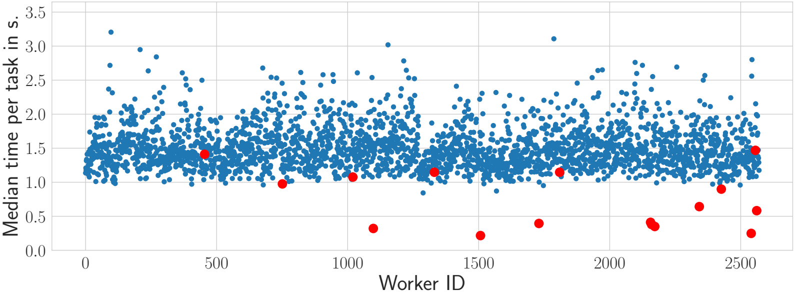
[16] V. C. Raykar and S. Yu (2011). "Ranking annotators for crowdsourced labeling tasks". *NeurIPS*, pp. 1809–1817.

- Use DS: diagonal elements of $\hat{\pi}^{(j)}$ represents worker ability to be correct, get the average success across all labels with

$$\frac{1}{K}\,\mathrm{trace}(\hat{\pi}^{(j)})$$



- Use time spent: get the median time spent per task



Conclusion

More to come after a short break

Contact:

Joseph Salmon

joseph.salmon@umontpellier.fr

https://josephsalmon.eu

Github: @josephsalmon

Mastodon: @josephsalmon@sigmoid.social

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". *J. R. Stat. Soc. Ser. C. Appl. Stat.* 28.1, pp. 20–28.
- Deng, J. et al. (2009). "ImageNet: A Large-Scale Hierarchical Image Database". *CVPR*.
- Krizhevsky, A. and G. Hinton (2009). *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto.
- LeCun, Y. et al. (1998). "Gradient-based learning applied to document recognition". *Proceedings of the IEEE* 86.11, pp. 2278–2324.
- Peterson, J. C. et al. (2019). "Human Uncertainty Makes Classification More Robust". *ICCV*, pp. 9617–9626.
- Raykar, V. C. and S. Yu (2011). "Ranking annotators for crowdsourced labeling tasks". *NeurIPS*, pp. 1809–1817.
- Rodrigues, F. and F. Pereira (2018). "Deep learning from crowds". *AAAI*. Vol. 32.

Snow, R. et al. (2008). “Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks”. *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?” *arXiv e-prints*, arXiv:2006.16923.