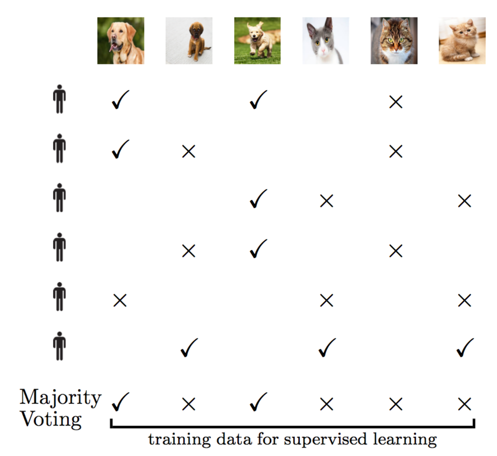
Learning From Noisy Singly-labeled Data

Ashish Khetan   Zachary C. Lipton   Animashree Anandkumar

amazon web services

Crowdsourcing: Noisy annotations



Problem Formulation

- n i.i.d. samples $(X,Y) \in (\mathcal{X} \times \mathcal{K}) \sim \mathcal{D}$
- r noisy labels $\{Z_{ij}\}_{j\in[r]}$ on each i-th example X_i
- given by workers $\{w_{ij}\}_{j\in[r]}$, $w_{ij} \in [m]$
- want to learn $\widehat{f} \in \mathcal{F}$ such that $\widehat{f}(X) = Y$ w.h.p.

Dawid Skene Model (DS)

- Each a-th worker is characterized by its confusion matrix $\pi^{(a)}$
- $\pi^{(a)} \in [0,1]^{K\times K}$: $\sum_{s\in\mathcal{K}} \pi_{ks} = 1$
- $\mathbb{P}\Big[Z_{ij} = s \Big| Y_i = k, w_{ij} = a\Big] = \pi^{(a)}_{ks}$

Learning with noisy labels

Posterior probability weighted loss:

$$\ell_{\widehat{\pi}}(f(X), Z^{(r)}, w^{(r)}) \equiv$$

$$\sum_{k\in\mathcal{K}} \mathbb{P}_{\widehat{\pi}}[Y = k | Z^{(r)}; w^{(r)}] \ell(f(X), Y = k)$$

Model Bootstrapped EM (MBEM)

Input: data $\{(X_i, Z_i^{(r)}, w_i^{(r)})\}_{i\in[n]}$
Output: deep learning model $\widehat{f}$

Initialize posterior distribution using weighted majority vote

$\mathbb{P}_{\widehat{\pi}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}] \leftarrow (1/r)\sum_{j=1}^{r} \mathbb{I}[Z_{ij} = k]$, for $k \in \mathcal{K}$

Repeat T times:

learn predictor function by minimizing probability weighted loss $\widehat{f} \leftarrow \arg\min_{f\in\mathcal{F}} \frac{1}{n}\ell_{\widehat{\pi}}(f(X_i), Z)i^{(r)}, w_i^{(r)})$

predict on the training examples
$t_i \leftarrow \arg\max_{k\in\mathcal{K}} \widehat{f}(X_i)_k$, for $i \in [n]$

estimate confusion matrices $\widehat{\pi}$ given model predictions $\{t_i\}_{i\in[n]}$
$\widehat{\pi}^{(a)} \leftarrow$ MLE under the DS model assuming $\{t_i\}$ are true labels, $a \in [m]$

estimate label posterior distribution given $\widehat{\pi}$
$\mathbb{P}_{\widehat{\pi}}[Y_i = k \mid Z_i^{(r)}; w_i^{(r)}] \leftarrow$ MLE under the DS model assuming $\widehat{\pi}$ are true confusion matrices, for $k \in \mathcal{K}, i \in [n]$

Return $\widehat{f}$

ℓ-risk under $\mathcal{D}$

Let $\ell(f(X), Y)$ denote a loss function.

$$R_{\ell,\mathcal{D}}(f) \triangleq \mathbb{E}_{(X,Y)\sim\mathcal{D}}\left[\ell(f(X), Y)\right] .$$

Main Theorem

- $N \triangleq nr$. For any hypothesis class $\mathcal{F}$ with a finite VC dimension V, and binary classification with 0-1 loss ℓ.
- There exists a universal constant C such that for any $\delta < 1$, if N is large enough (characterized in the paper) then $\widehat{f}$ returned by the MBEM algorithm after $T = 2$ iterations satisfies

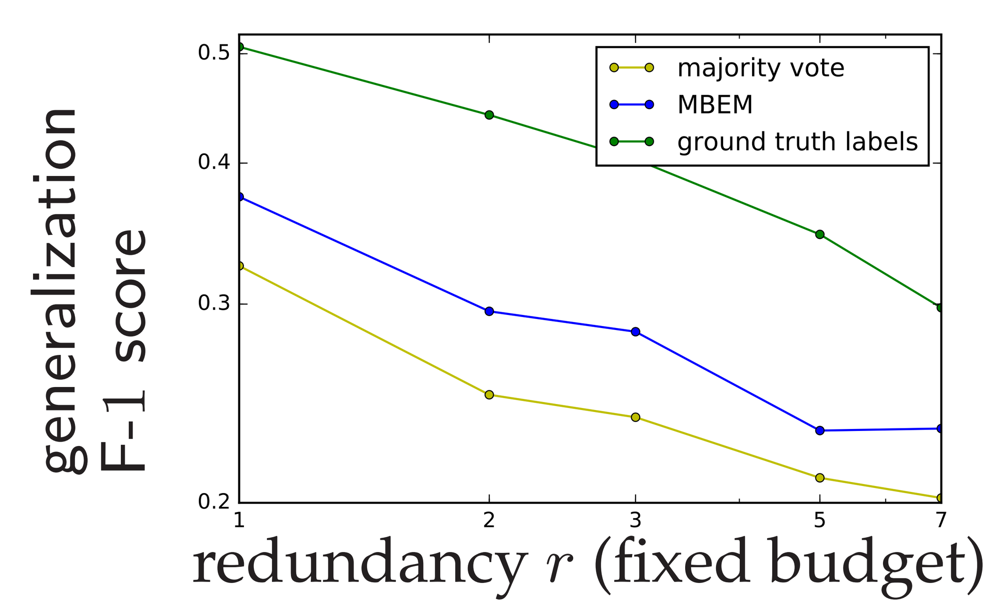
$$R_{\ell,\mathcal{D}}(\widehat{f}) - \min_{f\in\mathcal{F}} R_{\ell,\mathcal{D}}(f)$$
$$\leq \frac{C\sqrt{r}}{1 - 2g(\rho, r)}\left(\sqrt{\frac{V}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right).$$

- $g(\rho, r)$ is an analytical function of worker quality ρ and redundancy r.
- If ρ is above a threshold then $\arg\min_{r\in\mathbb{N}} \frac{\sqrt{r}}{1-2g(\rho,r)} = 1$.

Labeling once is optimal. It is also seen in all the experiments.

MS-COCO: Real annotations

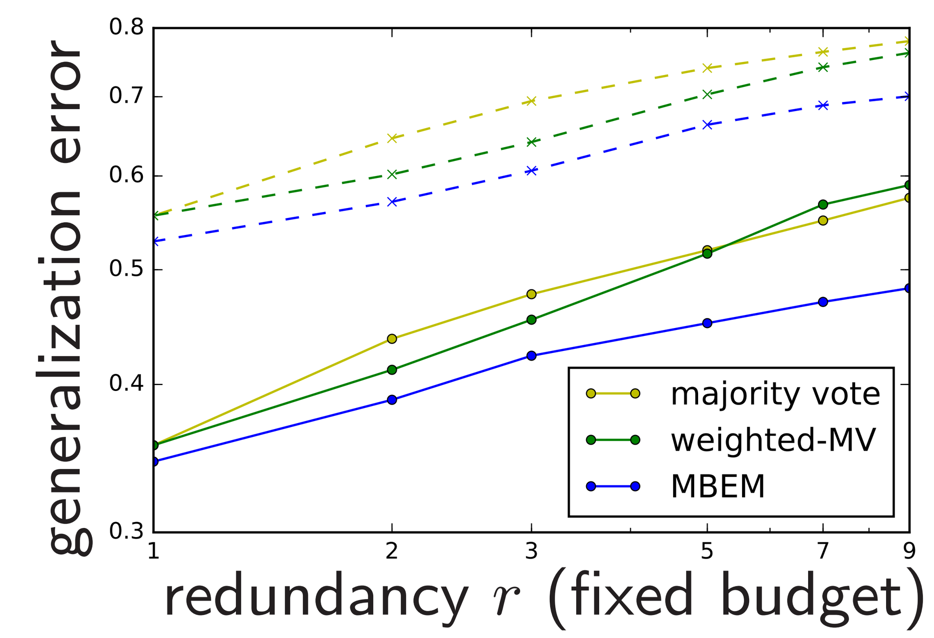
Labeling Once is Optimal.



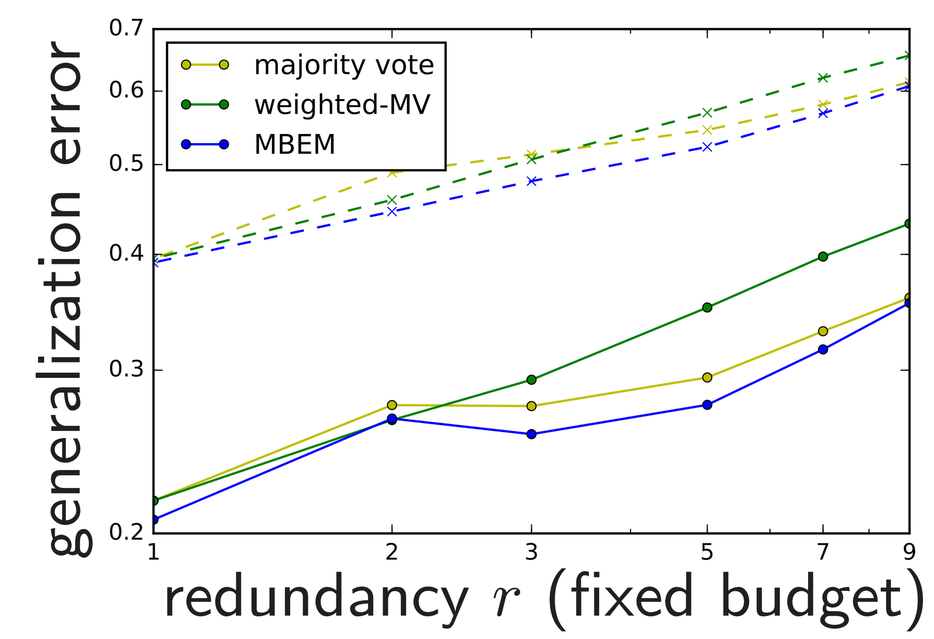
ImageNet 1K: Simulated workers

- class-wise hammer-spammer workers: Always correct with probability γ for each class independently.

Labeling Once is Optimal.



$\gamma = 0.2$

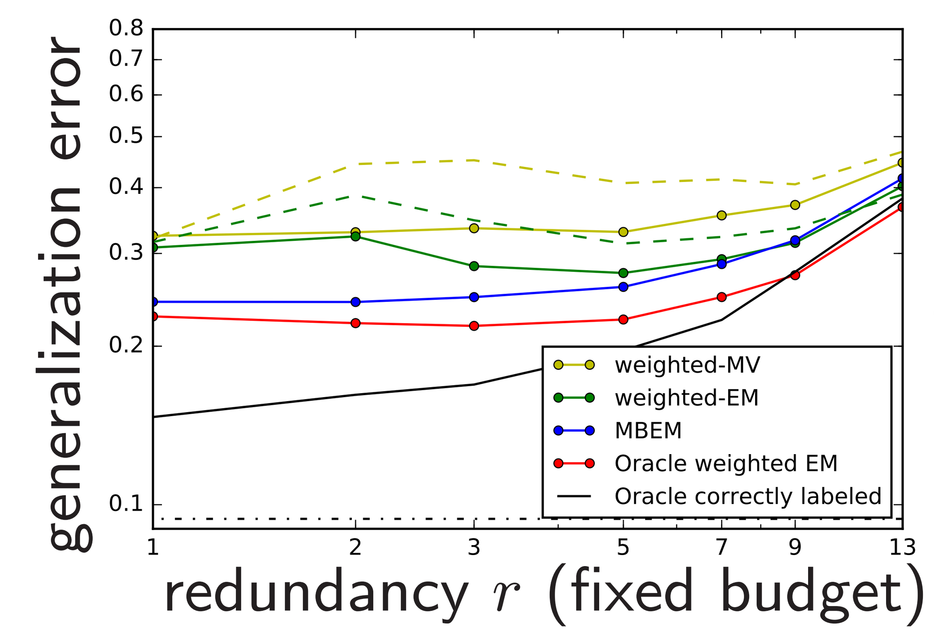


$\gamma = 0.6$

CIFAR10: Simulated workers

class-wise hammer spammer workers

Labeling Once is Optimal.



$\gamma = 0.2$