Streaming Bayesian Inference for Crowdsourced Classification

UNIVERSITY OF Southampton

Edoardo Manino, Long Tran-Thanh, Nicholas R. Jennings

Imperial College London

E.Manino@soton.ac.uk

Crowdsourcing in a nutshell

We pay a group of people to annotate our data:

- they are experts, non-experts or even bots;
- we get noisy and unreliable data.

The need for speed

The **online** approach saves us money [4]:

- we collect new data only where there is no consensus;
- thus, we need fewer data points to achieve a target accuracy.

However, it requires **real-time** decision making to work.

Theoretical guarantees?

Data comes at a cost. We want to predict in advance:

- how many data points we need to achieve a given target accuracy;
- which algorithm will work the best given the budget constraints.

Contributions

Existing crowdsourcing algorithms [2,3] are either slow, offer no accuracy guarantees or are built on unrealistic assumptions.

We address this gap in the literature as follows:

- we propose the Streaming Bayesian Inference for Crowdsourcing algorithm (**SBIC**);
- we introduce a first variant, **Fast SBIC**, which is as fast as the simple majority rule;
- we introduce a slower variant, **Sorted SBIC**, which delivers state-of-the-art accuracy;
- we derive **asymptotic bounds** on both the offline and online accuracy of SBIC.

Acknowledgements

This research is funded by the UK Research Council project ORCHID, grant EP/I011587/1.

The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton.

Bibliography

[1] Broderick, Boyd, Wibisono, Wilson and Jordan. *Streaming Variational Bayes*. 26th International Conference on Neural Information Processing Systems (NIPS). 2013.

[2] Bonald and Combes. *A Minimax Optimal Algorithm for Crowdsourcing*. 30th International Conference on Neural Information Processing Systems (NIPS). 2017

[3] Liu, Peng and Ihler. *Variational Inference for Crowdsourcing*. 25th International Conference on Neural Information Processing Systems (NIPS). 2012.

[4] Manino, Tran-Thanh and Jennings. *On the Efficiency of Data Collection for Crowdsourced Classification*. 27th International Joint Conference on Artificial Intelligence (IJCAI). 2018.

The SBIC algorithm

The SBIC algorithm falls under the umbrella of the streaming variational Bayes framework [1]. First, we define a mean-field approximation on the posterior:

$$\mathbb{P}(\boldsymbol{y}^t, \boldsymbol{p}^t | X^t, \theta) \approx \prod_{i \in M} \mu_i^t(y_i) \prod_{j \in N} \nu_j^t(p_j)$$

Legend:
$\boldsymbol{y}$ **- task classes**
$\boldsymbol{p}$ **- workers' accuracy**
X **- dataset**
θ **- prior**
$\boldsymbol{\mu}$ **- task factors**
$\boldsymbol{\nu}$ **- worker factors**

Second, we initialise all task factors μ_i^0 to an uninformative prior.

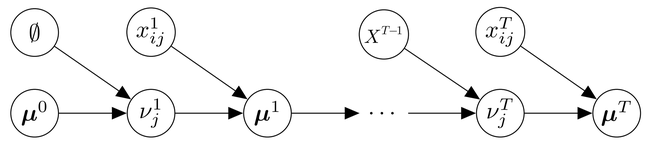
Third, we assume that the worker factors ν_j^t are Beta-distributed, and we compute their mean accuracy $\bar{p}_j^t$ on the subset of tasks M_j^{t-1} they have already worked on:

$$\bar{p}_j^t = \frac{\sum_{i \in M_j^{t-1}} \mu_i^{t-1}(x_{ij}) + \alpha}{|M_j^{t-1}| + \alpha + \beta} \tag{2}$$

Fourth, we process one extra data point x_{ij}^t, and update the task factors as follows:

$$\mu_i^t(y_i) \propto \begin{cases} \mu_i^{t-1}(y_i)\bar{p}_j^t & \text{if } x_{ij}^t = y_i \\ \mu_i^{t-1}(y_i)\left(1 - \bar{p}_j^t\right) & \text{if } x_{ij}^t \neq y_i \end{cases} \tag{3}$$

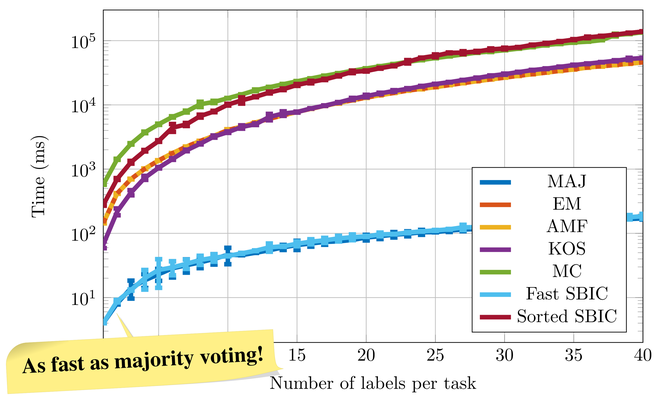
Fifth, we repeat for all $t \in [1, T]$ until the whole dataset is processed.



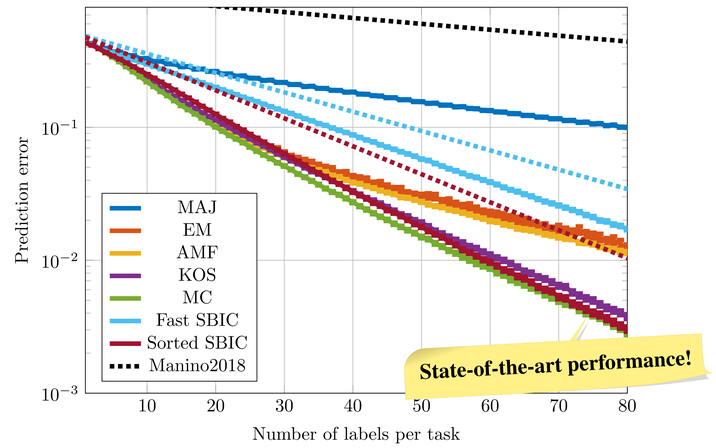
Intuition: at each time step t, we trust our current estimates to be correct, and build on top of them (see sketch above). If everything goes well, the task factors μ_i converge to the ground truth over time. In turn, our estimates of the worker accuracy $\bar{p}_j$ become more informative as we observe more data points.

Fast or Sorted? The SBIC algorithm is sequential. A vanilla implementation (Fast SBIC) computes all the estimates in one single pass over the dataset. However, the last data points that we process have a larger impact on our predictions. By reordering the dataset (Sorted SBIC), we can extract more predictive power at the price of some extra computation.

Computational Speed



Prediction Error (offline)



Asymptotic accuracy bounds

Assume that the crowd of workers has accuracy $p_j \sim \text{Beta}(\alpha, \beta)$, each worker provides L data points, and each task receives an average of R data points.

The probability of an error in the **offline** case is bounded by:

$$\mathbb{P}(error) \leq \exp\big(-R \log f(L, \alpha, \beta) + o(1)\big) \tag{4}$$

The probability of an error in the **online** case is bounded by:

$$\mathbb{P}(error) \leq \exp\big(-Rg(L, \alpha, \beta) + o(1)\big) \tag{5}$$

where $f(L, \alpha, \beta)$ and $g(L, \alpha, \beta)$ depend on the variant of SBIC we use.

Prediction Error (online)