Strategic Classification with Crowdsourcing

Yang Liu
(joint work with Yiling Chen)

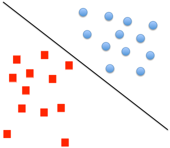
yangl@seas.harvard.edu
Harvard University

Nov. 2016

(Non-strategic) Classification

Non-strategic classification

$$y_i = f^*(\mathbf{x}_i), \quad f^* : \mathbb{R}^d \to \{-1, +1\}$$



- Observing a set of training data, to learn f

$$\tilde{f} = \text{argmin}_{f \in \mathcal{F}} \sum_{i=1}^{n} l(f(\mathbf{x}_i), y_i).$$

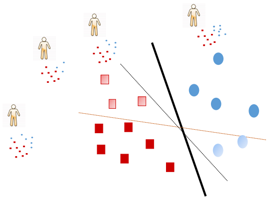
Strategic classification

When data comes from strategic data sources...

- Outsource $\mathbf{x}_i$ to get a label $\tilde{y}_i$.
- Crowdsourcing, survey, human reports etc.

Such training data carries noise

- *Intrinsic*: due to limited worker expertise.
- *Strategic*: lack of incentives.



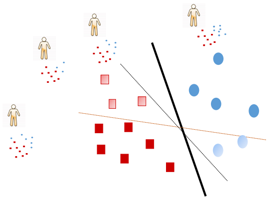
Strategic classification

When data comes from strategic data sources...

- Outsource $\mathbf{x}_i$ to get a label $\tilde{y}_i$.
- Crowdsourcing, survey, human reports etc.

Such training data carries noise

- *Intrinsic*: due to limited worker expertise.
- *Strategic*: lack of incentives.



Goal to achieve

The leaner wants to learn a good, unbiased classifier

- Workers' observations come from a flipping error model p_+, p_-.
- Workers are effort sensitive.
- Elicit high quality data from workers. (better performance)

Goal to achieve

~~The leaner wants to learn a good, unbiased classifier~~

- Workers' observations come from a flipping error model p_+, p_-.
- Workers are effort sensitive.
- Elicit high quality data from workers. (better performance)

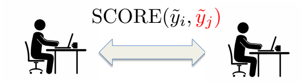
Goal to achieve

~~The leaner wants to learn a good, unbiased classifier~~

- Workers' observations come from a flipping error model p_+, p_-.
- Workers are effort sensitive.
- Elicit high quality data from workers. (better performance)

Information elicitation without verification

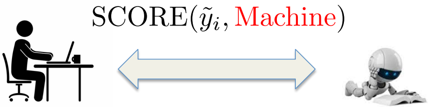
- Peer prediction: $\mathrm{SCORE}(\tilde{y}_i, \tilde{y}_j)$
- DG13, RF15, SAFP16, KS16...
- Exerting effort to have a high quality data usually a good equilibria.



Our method

Joint learning and information elicitation:

- $\text{SCORE}(\tilde{y}_i, \tilde{y}_j) \Rightarrow \text{SCORE}(\tilde{y}_i, \text{Machine})$
- "Machine Prediction"
- How to obtain a good machine answer?



Classification with flipping errors [Natarajan et al. 13]

- Suppose workers are truthfully reporting, how to de-bias?

$$\tilde{l}(t,y) := \frac{(1-p_{-y})l(t,y) - p_y l(t,-y)}{1-p_+-p_-},\ p_+ + p_- < 1.$$

- Why does it work? [un-biased in expectation]

$$\mathbb{E}_{\tilde{y}}[\tilde{l}(t,\tilde{y})] = l(t,y), \forall t.$$

- Find $\tilde{f}^*_{\tilde{l}}$ via minimizing the empirical risk w.r.t. $\tilde{l}(t,y)$:

$$\tilde{f}^*_{\tilde{l}} = \text{argmin}_f \hat{R}_{\tilde{l}}(f) := \frac{1}{N}\sum_{j=1}^{N} \tilde{l}(f(\mathbf{x}_j), \hat{y}_j).$$

Classification with flipping errors [Natarajan et al. 13]

- Suppose workers are truthfully reporting, how to de-bias?

$$\tilde{l}(t,y) := \frac{(1-p_{-y})l(t,y) - p_y l(t,-y)}{1-p_+-p_-},\ p_+ + p_- < 1.$$

- Why does it work? [un-biased in expectation]

$$\mathbb{E}_{\tilde{y}}[\tilde{l}(t,\tilde{y})] = l(t,y), \forall t.$$

- Find $\tilde{f}^*_{\tilde{l}}$ via minimizing the empirical risk w.r.t. $\tilde{l}(t,y)$:

$$\tilde{f}^*_{\tilde{l}} = \text{argmin}_f \hat{R}_{\tilde{l}}(f) := \frac{1}{N}\sum_{j=1}^{N} \tilde{l}(f(\mathbf{x}_j), \hat{y}_j).$$

Classification with flipping errors [Natarajan et al. 13]

- Suppose workers are truthfully reporting, how to de-bias?

$$\tilde{l}(t,y) := \frac{(1-p_{-y})l(t,y) - p_y l(t,-y)}{1-p_+-p_-}, \; p_+ + p_- < 1.$$

- Why does it work? [un-biased in expectation]

$$\mathbb{E}_{\tilde{y}}[\tilde{l}(t,\tilde{y})] = l(t,y), \forall t.$$

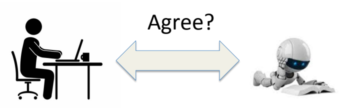
- Find $\tilde{f}^*_{\tilde{l}}$ via minimizing the empirical risk w.r.t. $\tilde{l}(t,y)$:

$$\tilde{f}^*_{\tilde{l}} = \text{argmin}_f \hat{R}_{\tilde{l}}(f) := \frac{1}{N}\sum_{j=1}^{N} \tilde{l}(f(\mathbf{x}_j), \hat{y}_j).$$

Our mechanism

For each worker i:

- Estimate flipping errors $\tilde{p}_{i,+}, \tilde{p}_{i,-}$ based on $\{\mathbf{x}_j, \tilde{y}_j\}_{j \neq i}$.
- Train $\tilde{f}^*_{\tilde{l},-i}$ using [Natarajan et al. 13] with data from $j \neq i$.



$\hat{y}_i$ $\tilde{f}^*_{\tilde{l},-i}(\mathbf{x}_i)$

How to estimate error rate

How do we estimate without ground-truth?

$\mathcal{P}_+[p_{i,+}^2 + (1-p_{i,+})^2] + \mathcal{P}_-[p_{i,-}^2 + (1-p_{i,-})^2] =$ Pr(mathcing)

$\mathcal{P}_+ p_{i,+} + \mathcal{P}_-(1-p_{i,-}) =$ Fraction of -1 labels observed

- Lemma: There is a unique pair of $\tilde{p}_{i,+}, \tilde{p}_{i,-}$ s.t. $\tilde{p}_{i,+} + \tilde{p}_{i,-} < 1$
  $\Rightarrow$*Bayesian informative:* $\Leftrightarrow \Pr(y_i = s | \tilde{y}_i = s) > \text{Prior}(s)$, $s \in \{+, -\}$

How to estimate error rate

How do we estimate without ground-truth?

$$\mathcal{P}_+[p_{i,+}^2 + (1 - p_{i,+})^2] + \mathcal{P}_-[p_{i,-}^2 + (1 - p_{i,-})^2] = \text{Pr(mathcing)}$$

$$\mathcal{P}_+ p_{i,+} + \mathcal{P}_-(1 - p_{i,-}) = \text{Fraction of -1 labels observed}$$

- Lemma: There is a unique pair of $\tilde{p}_{i,+}, \tilde{p}_{i,-}$ s.t. $\tilde{p}_{i,+} + \tilde{p}_{i,-} < 1$
  $\Rightarrow$ *Bayesian informative:* $\Leftrightarrow \Pr(y_i = s | \tilde{y}_i = s) > \text{Prior}(s)$, $s \in \{+, -\}$

How to estimate error rate

How do we estimate without ground-truth?

$$\mathcal{P}_+[p_{i,+}^2 + (1-p_{i,+})^2] + \mathcal{P}_-[p_{i,-}^2 + (1-p_{i,-})^2] = \text{Pr(mathcing)}$$
$$\mathcal{P}_+ p_{i,+} + \mathcal{P}_-(1-p_{i,-}) = \text{Fraction of -1 labels observed}$$

- Lemma: There is a unique pair of $\tilde{p}_{i,+}, \tilde{p}_{i,-}$ s.t. $\tilde{p}_{i,+} + \tilde{p}_{i,-} < 1$ $\Rightarrow$*Bayesian informative:* $\Leftrightarrow \Pr(y_i = s | \tilde{y}_i = s) > \text{Prior}(s)$, $s \in \{+, -\}$

Results

Effort exertion is a BNE.

Benefits of doing so?

- Less redundant assignment: not all tasks are re-assigned $\Rightarrow$ budget efficient.
- Better incentive: Reporting symmetric uninformative signal & permutation signal is not an equilibrium.
- More learning flavor: no requirement of knowing workers' data distribution.
- Better privacy preserving etc...

Results

Effort exertion is a BNE.

Benefits of doing so?

- Less redundant assignment: not all tasks are re-assigned $\Rightarrow$ budget efficient.
- Better incentive: Reporting symmetric uninformative signal & permutation signal is not an equilibrium.
- More learning flavor: no requirement of knowing workers' data distribution.
- Better privacy preserving etc...

Results

Effort exertion is a BNE.

Benefits of doing so?

- Less redundant assignment: not all tasks are re-assigned $\Rightarrow$ budget efficient.
- Better incentive: Reporting symmetric uninformative signal & permutation signal is not an equilibrium.
- More learning flavor: no requirement of knowing workers' data distribution.
- Better privacy preserving etc...

Results

Effort exertion is a BNE.

Benefits of doing so?

- Less redundant assignment: not all tasks are re-assigned $\Rightarrow$ budget efficient.
- Better incentive: Reporting symmetric uninformative signal & permutation signal is not an equilibrium.
- More learning flavor: no requirement of knowing workers' data distribution.
- Better privacy preserving etc...

Results

Effort exertion is a BNE.

Benefits of doing so?

- Less redundant assignment: not all tasks are re-assigned $\Rightarrow$ budget efficient.
- Better incentive: Reporting symmetric uninformative signal & permutation signal is not an equilibrium.
- More learning flavor: no requirement of knowing workers' data distribution.
- Better privacy preserving etc...

A case study: collusion is not an equlibria

Suppose $j \neq i$ collude by reporting -1

$$\mathcal{P}_+(p_{i,+}^2 + (1-p_{i,+})^2] + \mathcal{P}_-(p_{i,-}^2 + (1-p_{i,-})^2] = 1.$$
$$\mathcal{P}_+ p_{i,+} + \mathcal{P}_{-1}(1-p_{i,-}) = 1.$$

$\Rightarrow \tilde{p}_{i,+} = 1$

$\Rightarrow$ the solution interprets the missing of $+1$ as high error rate.

$$\tilde{l}(t, y=-1) := \frac{\cancel{(1-\tilde{p}_{i,+})l(t,-1)} - \tilde{p}_{i,-} l(t,+1)}{\cancel{1-\tilde{p}_{i,+}} - \tilde{p}_{i,-}} = l(t,+1)$$

$\Rightarrow$ the surrogate loss punishes this particular class

$\Rightarrow$ better to report $+1$ to match.

A case study: collusion is not an equlibria

Suppose $j \neq i$ collude by reporting -1

$$\mathcal{P}_+(p_{i,+}^2 + (1-p_{i,+})^2] + \mathcal{P}_-(p_{i,-}^2 + (1-p_{i,-})^2] = 1.$$
$$\mathcal{P}_+ p_{i,+} + \mathcal{P}_{-1}(1-p_{i,-}) = 1.$$
$$\Rightarrow \tilde{p}_{i,+} = 1$$

$\Rightarrow$ the solution interprets the missing of $+1$ as high error rate.

$$\tilde{l}(t, y=-1) := \frac{\cancel{(1-\tilde{p}_{i,+})l(t,-1)} - \tilde{p}_{i,-} l(t,+1)}{\cancel{1-\tilde{p}_{i,+}} - \tilde{p}_{i,-}} = l(t,+1)$$

$\Rightarrow$ the surrogate loss punishes this particular class

$\Rightarrow$ better to report $+1$ to match.

A case study: collusion is not an equlibria

Suppose $j \neq i$ collude by reporting -1

$$\begin{aligned}&\mathcal{P}_+(p_{i,+}^2 + (1-p_{i,+})^2] + \mathcal{P}_-(p_{i,-}^2 + (1-p_{i,-})^2] = 1.\\ &\mathcal{P}_+ p_{i,+} + \mathcal{P}_{-1}(1-p_{i,-}) = 1.\end{aligned} \quad \Rightarrow \tilde{p}_{i,+} = 1$$

$\Rightarrow$ the solution interprets the missing of $+1$ as high error rate.

$$\tilde{l}(t, y=-1) := \frac{\cancel{(1-\tilde{p}_{i,+})l(t,-1)} - \tilde{p}_{i,-} l(t,+1)}{\cancel{1-\tilde{p}_{i,+}} - \tilde{p}_{i,-}} = l(t,+1)$$

$\Rightarrow$ the surrogate loss punishes this particular class

$\Rightarrow$ better to report $+1$ to match.

Summary

What we achieve

- A classification problem with strategic data sources.
- A classification aided approach to elicit information.
- Enjoy several favorable properties.

Hope to see more on how machine learning can help information elicitation

Thank you!