Algorithmic Crowdsourcing

Denny Zhou
Microsoft Research Redmond
Dec 9, NIPS13, Lake Tahoe
Main collaborators

John Platt (Microsoft)
Xi Chen (UC Berkeley)
Chao Gao (Yale)
Nihar Shah (UC Berkeley)
Qiang Liu (UC Irvine)
Machine learning + crowdsourcing

• Almost all machine learning applications need training labels
• By crowdsourcing we can obtain many labels in a short time at very low cost
<table>
<thead>
<tr>
<th></th>
<th>Orange (O)</th>
<th></th>
<th>Orange (O)</th>
<th></th>
<th>Orange (O)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Kid 1</td>
<td>M</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>Kid 2</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>Kid 3</td>
<td>O</td>
<td>M</td>
<td>O</td>
<td>O</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>Kid 4</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td></td>
</tr>
</tbody>
</table>

Repeated labeling: Orange (O) vs. Mandarin (M)
Repeated labeling: Orange (O) vs. Mandarin (M)
Repeated labeling: Orange (O) vs. Mandarin (M)

<table>
<thead>
<tr>
<th></th>
<th>Orange</th>
<th>Mandarin</th>
<th>Orange</th>
<th>Mandarin</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>O</td>
<td>O</td>
<td></td>
<td>O</td>
</tr>
<tr>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
<td>M</td>
</tr>
<tr>
<td>O</td>
<td>M</td>
<td>O</td>
<td></td>
<td>M</td>
</tr>
<tr>
<td>M</td>
<td>M</td>
<td>M</td>
<td></td>
<td>M</td>
</tr>
</tbody>
</table>
How to make assumptions?

• Intuitively, label quality depends on worker ability and item difficulty. But,
  – How to measure worker ability?
  – How to measure item difficulty?
  – How to combine worker ability and item difficulty?
  – How to infer worker ability and item difficulty?
  – How to infer ground truth?
A single assumption for all questions

Our assumption: measurement objectivity

Invariance: No matter which scale, A is twice larger than B
A single assumption for all questions

Our assumption: measurement objectivity

How to formulate invariance for mental measuring?
A single assumption for all questions

Our assumption: measurement objectivity

Assume a set $\aleph$ of equally difficult questions:

$R_A$: number of right answers

$W_A$: number of wrong answers

\[
\frac{R_A}{W_A} \quad \frac{R_B}{W_B}
\]
A single assumption for all questions

Our assumption: measurement objectivity

\[ R'_A : \text{number of right answers} \quad W'_A : \text{number of wrong answers} \]

Assume another set \( \aleph' \) of equally difficult questions:

\[ \frac{R'_A}{W'_A} = 2 \times \frac{R'_B}{W'_B} \]
A single assumption for all questions

Our assumption: measurement objectivity

\[
\frac{R_A/W_A}{R_B/W_B} = \frac{R'_A/W'_A}{R'_B/W'_B}
\]
A single assumption for all questions

Our assumption: measurement objectivity

For multiclass labeling, we count the number of misclassifications from one class to another
Measurement objectivity assumption leads to a unique model!

\[
P(X_{ij} = k | Y_j = c) = \frac{1}{Z} \exp[\sigma_i(c, k) + \tau_j(c, k)]
\]

worker \(i\), item \(j\) labels \(c, k\)  
data matrix \(X_{ij}\)  
true label \(Y_j\)

worker confusion matrix  
item confusion matrix
Estimation procedure

• First, estimate the worker and item confusion matrices by maximizing marginal likelihood
• Then, estimate the labels by using Bayes’ rule with the estimated confusion matrices

Two steps can be seamlessly unified in EM!
Expectation-Maximization (EM)

• Initialize label estimates via majority vote
• Iterate till converge:
  – Given the estimates of labels, estimate worker and item confusion matrices
  – Given the estimates of worker and item confusion matrices, estimate labels
Prevent overfitting

• Equivalently formulate our solution into minimax conditional entropy
• Prevent overfitting by a natural regularization
Minimax conditional entropy

• True label distribution: $Q(Y)$

• Define two 4-dim tensors
  – Empirical confusion tensor
    $$\hat{\phi}_{ij}(c, k) = Q(Y_j = c)\mathbb{I}(x_{ij} = k)$$
  – Expected confusion tensor
    $$\phi_{ij}(c, k) = Q(Y_j = c)P(X_{ij} = k|Y_j = c)$$
Minimax conditional entropy

• Jointly estimate $P$ and $Q$ by

$$\min_{Q} \max_{P} H(X|Y)$$

subject to

$$\sum_{j} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = 0, \ \forall i, k, c,$$

$$\sum_{i} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = 0, \ \forall j, k, c.$$
Minimax conditional entropy

• Exactly recover the previous model via the dual of maximum entropy

\[
P(X_{ij} = k | Y_j = c) = \frac{1}{Z} \exp[\sigma_i(c, k) + \tau_j(c, k)]
\]

• Estimate true labels by minimizing maximum entropy (= maximizing likelihood)

Nothing but Lagrangian multipliers!
Regularization

• Move from exact matching to approximate matching:

$$\sum_{j} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] \approx 0, \forall i, k, c$$

$$\sum_{i} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] \approx 0, \forall j, k, c$$
Regularization

• Move from exact matching to approximate matching:

\[
\sum_{j} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = \xi_i(c, k), \quad \forall i, k, c.
\]

\[
\sum_{i} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = \zeta_j(c, k), \quad \forall j, k, c.
\]

• Penalize large fluctuations:

\[
\min_{Q} \max_{P} H(X|Y) - \alpha \|\xi\|^2 - \beta \|\zeta\|^2
\]
Experimental results

• Bluebirds data (error rates)

<table>
<thead>
<tr>
<th># Workers</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimax Entropy</td>
<td>0.150 ± 0.061</td>
<td>0.122 ± 0.026</td>
<td>0.097 ± 0.017</td>
<td>0.090 ± 0.016</td>
</tr>
<tr>
<td>Dawid &amp; Skene</td>
<td>0.142 ± 0.039</td>
<td>0.132 ± 0.025</td>
<td>0.114 ± 0.017</td>
<td>0.117 ± 0.017</td>
</tr>
<tr>
<td>Belief Propagation</td>
<td>0.143 ± 0.040</td>
<td>0.133 ± 0.026</td>
<td>0.117 ± 0.018</td>
<td>0.121 ± 0.019</td>
</tr>
<tr>
<td>Majority Vote</td>
<td>0.244 ± 0.065</td>
<td>0.234 ± 0.052</td>
<td>0.240 ± 0.034</td>
<td>0.242 ± 0.030</td>
</tr>
</tbody>
</table>

- Belief propagation: Variational Inference for Crowdsourcing (Liu et al. NIPS 2013)
- Other methods in the literature cannot outperform Dawid & Skene (1979)
- Some are even worse than majority voting
- Data: The multidimensional wisdom of crowds (Welinder et al, NIPS 2010)
Experimental results

• Web search data (error rates)

<table>
<thead>
<tr>
<th></th>
<th>Majority Vote</th>
<th>Dawid &amp; Skene</th>
<th>Latent Trait</th>
<th>Minimax Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0 Error</td>
<td>0.269</td>
<td>0.170</td>
<td>0.201</td>
<td>0.111</td>
</tr>
<tr>
<td>L1 Error</td>
<td>0.428</td>
<td>0.205</td>
<td>0.211</td>
<td>0.131</td>
</tr>
<tr>
<td>L2 Error</td>
<td>0.930</td>
<td>0.539</td>
<td>0.481</td>
<td>0.419</td>
</tr>
</tbody>
</table>

- Latent trait analysis code from: http://www.machinedlearnings.com
- Data: Learning from the wisdom of crowds by minimax entropy (Zhou et al., NIPS 2012)
Crowdsourced ordinal labeling

• Ordinal labels: web search, product rating
• Our assumption: adjacency confusability

1

likely to confuse

2

3

4

5

unlikely to confuse
Ordinal minimax conditional entropy

- Minimax conditional entropy with the ordinal-based worker and item constraints:

\[
\sum_{c\Delta s} \sum_{k\nabla s} \sum_{j} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = \xi_{is}^{\Delta, \nabla}, \forall i, s,
\]

\[
\sum_{c\Delta s} \sum_{k\nabla s} \sum_{i} \left[ \phi_{ij}(c, k) - \hat{\phi}_{ij}(c, k) \right] = \zeta_{is}^{\Delta, \nabla}, \forall j, s
\]

for all \( \Delta \) and \( \nabla \) taking values from \( \{\geq, <\} \)
Constraints: indirect label comparison
Ordinal labeling model

- Obtain the same model except confusion matrices are subtly structured by

\[
\sigma_i(c, k) = \sum_{s \geq 1} \sum_{\Delta, \nabla} \sigma_{i,s}^{\Delta, \nabla} I(c\Delta s, k\nabla s)
\]

\[
\tau_j(c, k) = \sum_{s \geq 1} \sum_{\Delta, \nabla} \tau_{j,s}^{\Delta, \nabla} I(c\Delta s, k\nabla s)
\]

- Fewer parameter, less model complexity
Experimental results

Web search data

<table>
<thead>
<tr>
<th></th>
<th>Majority Vote</th>
<th>Dawid &amp; Skene</th>
<th>Latent Trait</th>
<th>Entropy (M)</th>
<th>Entropy (O)</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0 Error</td>
<td>0.269</td>
<td>0.170</td>
<td>0.201</td>
<td>0.111</td>
<td>0.104</td>
</tr>
<tr>
<td>L1 Error</td>
<td>0.428</td>
<td>0.205</td>
<td>0.211</td>
<td>0.131</td>
<td>0.118</td>
</tr>
<tr>
<td>L2 Error</td>
<td>0.930</td>
<td>0.539</td>
<td>0.481</td>
<td>0.419</td>
<td>0.384</td>
</tr>
</tbody>
</table>

- Latent trait analysis code from: [http://www.machinedlearnings.com](http://www.machinedlearnings.com)
- Entropy(M): regularized minimax conditional entropy for multiclass labels
- Entropy(O): regularized minimax conditional entropy for ordinal labels
- Data: Learning from the wisdom of crowds by minimax entropy (Zhou et al., NIPS 2012)
Experimental results

Price estimation data

<table>
<thead>
<tr>
<th></th>
<th>Majority Vote</th>
<th>Dawid &amp; Skene</th>
<th>Latent Trait</th>
<th>Entropy (M)</th>
<th>Entropy (O)</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0 Error</td>
<td>0.675</td>
<td>0.650</td>
<td>0.688</td>
<td>0.675</td>
<td>0.613</td>
</tr>
<tr>
<td>L1 Error</td>
<td>1.125</td>
<td>1.050</td>
<td>1.063</td>
<td>1.150</td>
<td>0.975</td>
</tr>
<tr>
<td>L2 Error</td>
<td>1.605</td>
<td>1.517</td>
<td>1.504</td>
<td>1.643</td>
<td>1.492</td>
</tr>
</tbody>
</table>

- Latent trait analysis code from: [http://www.machinedlearnings.com](http://www.machinedlearnings.com)
- Entropy(M): regularized minimax conditional entropy for multiclass labels
- Entropy(O): regularized minimax conditional entropy for ordinal labels
- Data: 7 price ranges from least expensive to most expensive (Liu et al. NIPS 13)
Why latent trait model doesn’t work?

When solving a given problem try to avoid solving a more general problem as an intermediate step.

- Vladimir Vapnik

ordinal label = score range
Error bounds: problem setup

• Observed Data: \( X = (X_{ij})_{I \times J} \)
• Unknown true labels: \( Y = (Y_1, \ldots, Y_J) \)
• Unknown workers’ accuracies: \( p = (p_1, \ldots, p_I) \)
• Simplified model of Dawid and Skene (1979) and minimax conditional entropy
Dawid-Skene estimator (1979)

• Complete likelihood: $\mathbb{P}(X, Y | \rho)$

• Marginal likelihood: $\mathbb{P}(X | \rho)$

• Estimating workers’ accuracy by

$$\hat{\rho} = \arg \max \mathbb{P}(X | \rho)$$

• Estimating true labels by (plug-in)

$$\hat{z}_j = \mathbb{P}(Y_j = 1 | X, \hat{\rho})$$
Theorem (lower bound)

For any estimator, there exists a least favorable parameter space $p \in \mathcal{P}_{q, \bar{\mu}}$

\[
\frac{1}{J} \sum_{j} |\hat{z}_j - Y_j| \gtrsim \exp \left( -8I \max \left\{ 2q, \frac{1}{2}D(\bar{\mu} || 1 - \bar{\mu}) \right\} \right)
\]

\[
\mathcal{P}_{q, \bar{\mu}} = \left\{ p \in \mathbb{R}^I : \frac{1}{I} \sum_{i} (2p_i - 1)^2 = q, \frac{1}{I} \sum_{i} p_i = \bar{\mu} \right\}
\]
Theorem (upper bound)

Under mild assumptions, Dawid-Skene estimator is optimal in Wald’s sense

\[ \frac{1}{J} \sum_{j} |\hat{z}_j - Y_j| \lesssim \exp \left( - \frac{1}{4} I \max \left\{ 2q, \frac{1}{2} D(\bar{\mu}||1 - \bar{\mu}) \right\} \right) \]
Budget-optimal crowdsourcing

We propose the following formulation:

• Given $n$ biased coins, we want to know which are biased to heads and which are biased to tails
• We have a budget of tossing $m$ times in total
• Our goal is to maximize the accuracy of prediction based on observed tossing outcomes

Optimistic Knowledge Gradient Policy for Optimal Budget Allocation in Crowdsourcing (Chen et al, ICML 2013)
Summary

Measurement Objectivity → Maximum Conditional Entropy

Maximum Likelihood → Minimum Conditional Entropy

Minimax Conditional Entropy

Regularized Minimax Conditional Entropy

Minimax optimal rates → Budget-optimal crowdsourcing

12/9/2013  NIPS 2013 Workshop on Crowdsourcing


• Xi Chen, Qihang Lin, and Dengyong Zhou. Optimistic Knowledge Gradient Policy for Optimal Budget Allocation in Crowdsourcing, in Proceedings of the 30th International Conference on Machine Learning (ICML), 2013
