# Learning from the Crowd

Yuemei Zhang

November 27, 2017

# Outline

# An Example of Crowdsourced Labeling: the ESP Game

Players try to "agree" on as many images as they can in 2.5 minutes.

# An Example of Crowdsourced Labeling: the ESP Game

Players try to "agree" on as many images as they can in 2.5 minutes.


## Characteristics of Crowdsourcing

- A large pool of labelers
- High level of noise

# The Setting

- Realizable PAC learning
  - The instance space $\mathcal{X}$
  - Labels $\mathcal{Y} = \{+1, -1\}$
  - A distribution $D$ over $\mathcal{X} \times \mathcal{Y}$
  - The hypothesis class $\mathcal{F}$
  - A true classifier $f^* \in \mathcal{F}$: $err_D(f^*) = 0$
  - $\mathrm{err}_D(f) = \Pr_{(x,f^*(x))\sim D}[f(x) \neq f^*(x)]$

# The Setting

- Realizable PAC learning
  - The instance space $\mathcal{X}$
  - Labels $\mathcal{Y} = \{+1, -1\}$
  - A distribution $D$ over $\mathcal{X} \times \mathcal{Y}$
  - The hypothesis class $\mathcal{F}$
  - A true classifier $f^* \in \mathcal{F}$: $err_D(f^*) = 0$
  - $\mathrm{err}_D(f) = \mathrm{Pr}_{(x, f^*(x)) \sim D}[f(x) \neq f^*(x)]$
- A set of labelers $L$: each labeler $i$ is a classification function $g_i : \mathcal{X} \to \mathcal{Y}$

# The Setting

- Realizable PAC learning
  - The instance space $\mathcal{X}$
  - Labels $\mathcal{Y} = \{+1, -1\}$
  - A distribution $D$ over $\mathcal{X} \times \mathcal{Y}$
  - The hypothesis class $\mathcal{F}$
  - A true classifier $f^* \in \mathcal{F}$: $err_D(f^*) = 0$
  - $\mathrm{err}_D(f) = \Pr_{(x, f^*(x)) \sim D}[f(x) \neq f^*(x)]$
- A set of labelers $L$: each labeler $i$ is a classification function $g_i : \mathcal{X} \to \mathcal{Y}$
- *Perfect* labelers: $err_D(g_i) = 0$

# The Setting

- Realizable PAC learning
  - The instance space $\mathcal{X}$
  - Labels $\mathcal{Y} = \{+1, -1\}$
  - A distribution $D$ over $\mathcal{X} \times \mathcal{Y}$
  - The hypothesis class $\mathcal{F}$
  - A true classifier $f^* \in \mathcal{F}$: $err_D(f^*) = 0$
  - $\mathrm{err}_D(f) = \mathrm{Pr}_{(x, f^*(x)) \sim D}[f(x) \neq f^*(x)]$
- A set of labelers $L$: each labeler $i$ is a classification function $g_i : \mathcal{X} \to \mathcal{Y}$
- *Perfect* labelers: $err_D(g_i) = 0$
- Uniform distribution $P$ over all labelers

# The Setting

- Realizable PAC learning
  - The instance space $\mathcal{X}$
  - Labels $\mathcal{Y} = \{+1, -1\}$
  - A distribution $D$ over $\mathcal{X} \times \mathcal{Y}$
  - The hypothesis class $\mathcal{F}$
  - A true classifier $f^* \in \mathcal{F}$: $err_D(f^*) = 0$
  - $\text{err}_D(f) = \Pr_{(x, f^*(x)) \sim D}[f(x) \neq f^*(x)]$
- A set of labelers $L$: each labeler $i$ is a classification function $g_i : \mathcal{X} \to \mathcal{Y}$
- *Perfect* labelers: $err_D(g_i) = 0$
- Uniform distribution $P$ over all labelers
- Fraction of perfect labelers $\alpha = \Pr_{i \sim P}[err_D(g_i) = 0]$

# The Setting

The Learning Algorithm

- Draw unlabeled instances according to $D$.
- Query labelers on these instances.
- Use the oracle $\mathcal{O}_{\mathcal{F}}$ that for a set of labeled samples $S$, returns a function $f \in \mathcal{F}$ consistent with $S$.

Goal

- Low error rate
- A small number of label queries

# The Setting

The Learning Algorithm

- Draw unlabeled instances according to $D$.
- Query labelers on these instances.
- Use the oracle $\mathcal{O}_{\mathcal{F}}$ that for a set of labeled samples $S$, returns a function $f \in \mathcal{F}$ consistent with $S$.

Goal

- Low error rate
- A small number of label queries

Recall the label complexity of traditional PAC learning (VC theory):

$$m_{\epsilon,\delta} = O\left(\frac{d}{\epsilon}\left(\log\frac{1}{\epsilon} + \log\frac{1}{\delta}\right)\right).$$

# The Setting

The Learning Algorithm

- Draw unlabeled instances according to $D$.
- Query labelers on these instances.
- Use the oracle $\mathcal{O}_{\mathcal{F}}$ that for a set of labeled samples $S$, returns a function $f \in \mathcal{F}$ consistent with $S$.

Goal

- Low error rate
- A small number of label queries

Recall the label complexity of traditional PAC learning (VC theory):

$$m_{\epsilon,\delta} = O\left(\frac{d}{\epsilon}\left(\log\frac{1}{\epsilon} + \log\frac{1}{\delta}\right)\right).$$

*Cost per labeled example* : $\#$ label queries$/m_{\epsilon,\delta}$

# A Baseline Algorithm

Consider the case of a strong majority of perfect labelers ($\alpha = 1/2 + \Theta(1)$).

# A Baseline Algorithm

Consider the case of a strong majority of perfect labelers ($\alpha = 1/2 + \Theta(1)$).

## BASELINE

- Draw $m = m_{\epsilon,\delta}$ samples.
- Label each sample using the majority vote of $k$ labelers, where

$$k = O\left(\frac{\log(m/\delta)}{(\alpha - 1/2)^2}\right).$$

- Use the supervised learning oracle and return $\mathcal{O}_{\mathcal{F}}(S)$.

# A Baseline Algorithm

Consider the case of a strong majority of perfect labelers ($\alpha = 1/2 + \Theta(1)$).

## BASELINE

- Draw $m = m_{\epsilon,\delta}$ samples.
- Label each sample using the majority vote of $k$ labelers, where

$$k = O\left(\frac{\log(m/\delta)}{(\alpha - 1/2)^2}\right).$$

- Use the supervised learning oracle and return $\mathcal{O}_{\mathcal{F}}(S)$.

## Improvement over BASELINE

- Improve the $\log(m/\delta)$ cost per labeled example.
- Generalize to the case where $\alpha < 1/2$.

# Outline

# Overview of Techniques: Boosting

Combines three classifiers of error $p < 1/2$ to get a classifier of error $O(p^2)$.

### Theorem

*Boosting (Schapire 1990): For any* $p < 1/2$ *and distribution* $D$*, consider three classifiers:*

# Overview of Techniques: Boosting

Combines three classifiers of error $p < 1/2$ to get a classifier of error $O(p^2)$.

**Theorem**

*Boosting (Schapire 1990): For any $p < 1/2$ and distribution $D$, consider three classifiers:*

1. $h_1$: $\mathrm{err}_D(h_1) \leq p$;

# Overview of Techniques: Boosting

Combines three classifiers of error $p < 1/2$ to get a classifier of error $O(p^2)$.

**Theorem**

*Boosting (Schapire 1990): For any $p < 1/2$ and distribution $D$, consider three classifiers:*

1. $h_1$: $\mathrm{err}_D(h_1) \leq p$;
2. $h_2$: $\mathrm{err}_{D_2}(h_2) \leq p$, *where* $D_2 = \frac{1}{2}D_C + \frac{1}{2}D_I$, $D_C$ *is* $D$ *conditioned on* $\{x | h_1(x) = f^*(x)\}$, *and* $D_I$ *is* $D$ *conditioned on* $\{x | h_1(x) \neq f^*(x)\}$;

# Overview of Techniques: Boosting

Combines three classifiers of error $p < 1/2$ to get a classifier of error $O(p^2)$.

**Theorem**

*Boosting (Schapire 1990): For any $p < 1/2$ and distribution $D$, consider three classifiers:*

1. $h_1$: $\mathrm{err}_D(h_1) \leq p$;
2. $h_2$: $\mathrm{err}_{D_2}(h_2) \leq p$, *where* $D_2 = \frac{1}{2}D_C + \frac{1}{2}D_I$, $D_C$ *is* $D$ *conditioned on* $\{x|h_1(x) = f^*(x)\}$, *and* $D_I$ *is* $D$ *conditioned on* $\{x|h_1(x) \neq f^*(x)\}$;
3. $h_3$: $\mathrm{err}_{D_3}(h_3) \leq p$. $D_3$ *is* $D$ *conditioned on* $\{x|h_1(x) \neq h_2(x)\}$.

# Overview of Techniques: Boosting

Combines three classifiers of error $p < 1/2$ to get a classifier of error $O(p^2)$.

### Theorem

*Boosting (Schapire 1990): For any $p < 1/2$ and distribution $D$, consider three classifiers:*

1. $h_1$: $\text{err}_D(h_1) \leq p$;
2. $h_2$: $\text{err}_{D_2}(h_2) \leq p$, *where* $D_2 = \frac{1}{2}D_C + \frac{1}{2}D_I$, $D_C$ *is* $D$ *conditioned on* $\{x|h_1(x) = f^*(x)\}$, *and* $D_I$ *is* $D$ *conditioned on* $\{x|h_1(x) \neq f^*(x)\}$;
3. $h_3$: $\text{err}_{D_3}(h_3) \leq p$. $D_3$ *is* $D$ *conditioned on* $\{x|h_1(x) \neq h_2(x)\}$.

*Then, the majority vote of* $h_1$, $h_2$ *and* $h_3$ *has error* $\leq 3p^2 - 2p^3$ *under distribution* $D$.

# The Algorithm (Overview)

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers
- Phase 1
  - Draw a set of samples $S_1$ from $D$
  - $\overline{S_1}$ = CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers
- Phase 1
  - Draw a set of samples $S_1$ from $D$
  - $\overline{S_1}$ = CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw a set of samples $\overline{W}$ to simulate distribution $D_2$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers
- Phase 1
  - Draw a set of samples $S_1$ from $D$
  - $\overline{S_1} = $ CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw a set of samples $\overline{W}$ to simulate distribution $D_2$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$
- Phase 3
  - Draw a set of samples $S_3$ from $D_3$
  - $\overline{S_3} = $ CORRECT-LABEL($S_3$, $\delta/6$)
  - $h_3 = \mathcal{O}_{\mathcal{F}}(\overline{S_3})$

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers
- Phase 1
  - Draw a set of samples $S_1$ from $D$
  - $\overline{S_1} = \text{CORRECT-LABEL}(S_1, \delta/6)$
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw a set of samples $\overline{W}$ to simulate distribution $D_2$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$
- Phase 3
  - Draw a set of samples $S_3$ from $D_3$
  - $\overline{S_3} = \text{CORRECT-LABEL}(S_3, \delta/6)$
  - $h_3 = \mathcal{O}_{\mathcal{F}}(\overline{S_3})$
- Return the majority vote of $h_1$, $h_2$ and $h_3$

# The Algorithm (Overview)

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of a set of labelers
- Phase 1
  - Draw a set of samples $S_1$ from $D$
  - $\overline{S_1} = \text{CORRECT-LABEL}(S_1, \delta/6)$
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw a set of samples $\overline{W}$ to simulate distribution $D_2$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$
- Phase 3
  - Draw a set of samples $S_3$ from $D_3$
  - $\overline{S_3} = \text{CORRECT-LABEL}(S_3, \delta/6)$
  - $h_3 = \mathcal{O}_{\mathcal{F}}(\overline{S_3})$
- Return the majority vote of $h_1$, $h_2$ and $h_3$

# Overview of Techniques: Filtering

**Algorithm 1** FILTER($S$, $h_1$)

Returns a set of instances mislabeled by $h_1$ to simulate $D_2$.

- Let $S_I = \emptyset$ and $N = \log(1/\epsilon)$
- For each $x \in S$
  - For $t = 1, \ldots, N$
    - Draw a labeler $i \sim P$ and let $y_t = g_i(x)$.
    - If $t$ is odd and the majority vote of $y_{1:t}$ agrees with $h_1$ on $x$, then goto the next $x$.
  - If the majority vote of $y_{1:t}$ never agrees with $h_1$ on $x$, then add $x$ to $S_I$.
- Return $S_I$

# Outline

# Algorithm 2

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of $k$ labelers, where $\mathrm{k} = O(\log \frac{|S|}{\delta})$

# Algorithm 2

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of $k$ labelers, where k $= O(\log \frac{|S|}{\delta})$
- Phase 1
  - Draw $S_1$ of size $2m_{\sqrt{\epsilon},\delta/6}$ from $D$
  - $\overline{S_1} =$ CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$

# Algorithm 2

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of $k$ labelers, where k $= O(\log \frac{|S|}{\delta})$
- Phase 1
  - Draw $S_1$ of size $2m_{\sqrt{\epsilon},\delta/6}$ from $D$
  - $\overline{S_1} =$ CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw $S_2$ of size $\Theta(m_{\epsilon,\delta})$, $S_C$ of size $\Theta(m_{\sqrt{\epsilon},\delta})$ from $D$
  - $S_I =$ FILTER($S_2$, $h_1$)
  - CORRECT-LABEL($S_I \cup S_C$, $\delta/6$)
  - Divide the labeled set into $\overline{W_I}$ and $\overline{W_C}$ according to whether the label agrees with $h_1$
  - Draw $\overline{W}$ of size $\Theta(m_{\sqrt{\epsilon},\delta})$ from a distribution that equally weights $\overline{W_I}$ and $\overline{W_C}$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$

# Algorithm 2

- CORRECT-LABEL($S$, $\delta$): label each instance in $S$ with the majority vote of $k$ labelers, where k $= O(\log \frac{|S|}{\delta})$
- Phase 1
  - Draw $S_1$ of size $2m_{\sqrt{\epsilon},\delta/6}$ from $D$
  - $\overline{S_1} =$ CORRECT-LABEL($S_1$, $\delta/6$)
  - $h_1 = \mathcal{O}_{\mathcal{F}}(\overline{S_1})$
- Phase 2
  - Draw $S_2$ of size $\Theta(m_{\epsilon,\delta})$, $S_C$ of size $\Theta(m_{\sqrt{\epsilon},\delta})$ from $D$
  - $S_I =$ FILTER($S_2$, $h_1$)
  - CORRECT-LABEL($S_I \cup S_C$, $\delta/6$)
  - Divide the labeled set into $\overline{W_I}$ and $\overline{W_C}$ according to whether the label agrees with $h_1$
  - Draw $\overline{W}$ of size $\Theta(m_{\sqrt{\epsilon},\delta})$ from a distribution that equally weights $\overline{W_I}$ and $\overline{W_C}$
  - $h_2 = \mathcal{O}_{\mathcal{F}}(\overline{W})$
- Phase 3
  - Draw $S_3$ of size $2m_{\sqrt{\epsilon},\delta/6}$ from $D_3$
  - $\overline{S_3} =$ CORRECT-LABEL($S_3$, $\delta/6$)
  - $h_3 = \mathcal{O}_{\mathcal{F}}(\overline{S_3})$

# Main Result

**Theorem**

*Algorithm 2 returns* $f \in \mathcal{F}$ *with* $\mathrm{err}_D(f) \leq \epsilon$ *with probability* $1 - \delta$*, using* $O\left(m_{\sqrt{\epsilon},\delta} \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right) + m_{\epsilon,\delta}\right)$ *labels.*

# Main Result

**Theorem**

*Algorithm 2 returns $f \in \mathcal{F}$ with $\mathrm{err}_D(f) \leq \epsilon$ with probability $1-\delta$, using $O\left(m_{\sqrt{\epsilon},\delta} \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right) + m_{\epsilon,\delta}\right)$ labels.*

Note that when

$$\frac{1}{\sqrt{\epsilon}} \geq \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right),$$

the cost per labeled example is $O(1)$.

# Correctness of FILTER

### Lemma

*If* $h_1(x) = f^*(x)$, *then* $x \in$ *FILTER(*$S$, $h_1$*) with probability* $< \sqrt{\epsilon}$.
*If* $h_1(x) \neq f^*(x)$, *then* $x \in$ *FILTER(*$S$, $h_1$*) with probability* $\geq 1/2$.

# Correctness of FILTER

**Lemma**

*If $h_1(x) = f^*(x)$, then $x \in FILTER(S, h_1)$ with probability $< \sqrt{\epsilon}$.*
*If $h_1(x) \neq f^*(x)$, then $x \in FILTER(S, h_1)$ with probability $\geq 1/2$.*

**Proof.**

- Part 1 ($h_1(x) = f^*(x)$):
  $E_1 = \mathbb{1}\{\text{the majority vote of } y_{1:N} \text{ is incorrect}\}$, where
  $N = O(\log \frac{1}{\sqrt{\epsilon}})$. By Hoeffding inequality, we have $\Pr[E_1] < \sqrt{\epsilon}$.

# Correctness of FILTER

**Lemma**

*If $h_1(x) = f^*(x)$, then $x \in$ FILTER($S$, $h_1$) with probability $< \sqrt{\epsilon}$.*
*If $h_1(x) \neq f^*(x)$, then $x \in$ FILTER($S$, $h_1$) with probability $\geq 1/2$.*

**Proof.**

- Part 1 ($h_1(x) = f^*(x)$):
  $E_1 = \mathbb{1}\{\text{the majority vote of } y_{1:N} \text{ is incorrect}\}$, where $N = O(\log \frac{1}{\sqrt{\epsilon}})$. By Hoeffding inequality, we have $\Pr[E_1] < \sqrt{\epsilon}$.
- Part 2 ($h_1(x) \neq f^*(x)$):
  $E_2 = \mathbb{1}\{\exists t : \text{the majority vote of } y_{1:t} \text{ is incorrect}\}$. Using the probability of return in biased random walks,

$$\Pr[E_2] = \left(1 - \left(\frac{\alpha}{1-\alpha}\right)^N\right) / \left(1 - \left(\frac{\alpha}{1-\alpha}\right)^{N+1}\right) < \frac{1-\alpha}{\alpha} < \frac{1}{2}.$$

$\square$

# Label Complexity of FILTER

**Lemma**

*With probability at least* $1 - \exp(-\Omega(|S|\sqrt{\epsilon}))$, *FILTER(S,* $h_1$*) makes* $O(|S|)$ *label queries.*

# Label Complexity of FILTER

## Lemma

*With probability at least* $1 - \exp(-\Omega(|S|\sqrt{\epsilon}))$, *FILTER(*$S$*,* $h_1$*) makes* $O(|S|)$ *label queries.*

## Proof.

Using Chernoff bound, with probability $1 - \exp(-|S|\sqrt{\epsilon})$ the total number of points in $S$ where $h_1$ disagrees with $f^*$ is $O(|S|\sqrt{\epsilon})$. The number of queries spent on these points is at most $O(|S|\sqrt{\epsilon}\log(1/\epsilon)) \leq O(|S|)$.

# Label Complexity of FILTER

### Lemma

*With probability at least* $1 - \exp(-\Omega(|S|\sqrt{\epsilon}))$*, FILTER(*$S$*,* $h_1$*) makes* $O(|S|)$ *label queries.*

### Proof.

Using Chernoff bound, with probability $1 - \exp(-|S|\sqrt{\epsilon})$ the total number of points in $S$ where $h_1$ disagrees with $f^*$ is $O(|S|\sqrt{\epsilon})$. The number of queries spent on these points is at most $O(|S|\sqrt{\epsilon}\log(1/\epsilon)) \leq O(|S|)$.

For each $x$ such that $h_1(x) = f^*(x)$, let $N_i$ be the expected number of queries until we have $i$ more correct labels than incorrect ones. Then $N_1 \leq \alpha + (1-\alpha)(N_2 + 1)$. $N_2 = 2N_1$.
$\Rightarrow N_1 \leq 1/(2\alpha - 1)$.

# Proof (continued)

Let $L_x$ be the total number of queries on $x$ before we have one more correct label than incorrect labels. Then $\mathbb{E}[L_x] \leq 1/(2\alpha - 1)$. We can show that for some positive real number $L$ and any $k > 1$,

$$\mathbb{E}[(L_x - \mathbb{E}[L_x])^k] \leq \frac{1}{2}\mathbb{E}[(L_x - \mathbb{E}[L_x])^2]L^{k-2}k!.$$

# Proof (continued)

Let $L_x$ be the total number of queries on $x$ before we have one more correct label than incorrect labels. Then $\mathbb{E}[L_x] \leq 1/(2\alpha - 1)$. We can show that for some positive real number $L$ and any $k > 1$,

$$\mathbb{E}[(L_x - \mathbb{E}[L_x])^k] \leq \frac{1}{2}\mathbb{E}[(L_x - \mathbb{E}[L_x])^2]L^{k-2}k!.$$

Using the Bernstein inequality,

$$\Pr\left[\sum_{h_1(x)=f^*(x)} L_x - |S|\mathbb{E}[L_x] \geq O(|S|)\right] \leq \exp(-|S|).$$

Therefore, the total number of queries over all points $x \in S$ is $O(|S|)$ with probability at least $1 - \exp(-|S|\sqrt{\epsilon})$.

# Correctness of Phase 2

**Lemma**

*With probability* $1 - 2\delta/3$, $\mathrm{err}_{D_2}(h_2) \leq \sqrt{\epsilon}/2$.

# Correctness of Phase 2

**Lemma**

*With probability* $1 - 2\delta/3$, $\mathrm{err}_{D_2}(h_2) \leq \sqrt{\epsilon}/2$.

**Proof.**

- Part 1. With very high probability, $\mathrm{err}_D(h_1) \leq \frac{1}{2}\sqrt{\epsilon}$.

# Correctness of Phase 2

**Lemma**

*With probability* $1 - 2\delta/3$, $\mathrm{err}_{D_2}(h_2) \leq \sqrt{\epsilon}/2$.

**Proof.**

- Part 1. With very high probability, $\mathrm{err}_D(h_1) \leq \frac{1}{2}\sqrt{\epsilon}$.
- Part 2. With probability $1 - \exp(-\Omega(m_{\sqrt{\epsilon},\delta}))$, $\overline{W_I}$, $\overline{W_C}$ and $S_I$ all have size $\Theta(m_{\sqrt{\epsilon},\delta})$.

# Correctness of Phase 2

**Lemma**

*With probability* $1 - 2\delta/3$, $\mathrm{err}_{D_2}(h_2) \leq \sqrt{\epsilon}/2$.

**Proof.**

- Part 1. With very high probability, $\mathrm{err}_D(h_1) \leq \frac{1}{2}\sqrt{\epsilon}$.
- Part 2. With probability $1 - \exp(-\Omega(m_{\sqrt{\epsilon},\delta}))$, $\overline{W_I}$, $\overline{W_C}$ and $S_I$ all have size $\Theta(m_{\sqrt{\epsilon},\delta})$.
- Part 3. Let $D'$ be the distribution that equally weights $\overline{W_I}$ and $\overline{W_C}$, $\rho'(x)$ be the density of $x$ in $D'$, and $\rho_2(x)$ be the density of $x$ in $D_2$. Then for all $x$, $\rho'(x) \geq c \cdot \rho_2(x)$ for a constant $c > 0$.

# Correctness of Phase 2

### Lemma

*With probability* $1 - 2\delta/3$, $\text{err}_{D_2}(h_2) \leq \sqrt{\epsilon}/2$.

### Proof.

- Part 1. With very high probability, $\text{err}_D(h_1) \leq \frac{1}{2}\sqrt{\epsilon}$.
- Part 2. With probability $1 - \exp(-\Omega(m_{\sqrt{\epsilon},\delta}))$, $\overline{W_I}$, $\overline{W_C}$ and $S_I$ all have size $\Theta(m_{\sqrt{\epsilon},\delta})$.
- Part 3. Let $D'$ be the distribution that equally weights $\overline{W_I}$ and $\overline{W_C}$, $\rho'(x)$ be the density of $x$ in $D'$, and $\rho_2(x)$ be the density of $x$ in $D_2$. Then for all $x$, $\rho'(x) \geq c \cdot \rho_2(x)$ for a constant $c > 0$.
- Part 4. There exists a constant $c' > 1$ such that with a labeled sample set $S$ of size $c' m_{\sqrt{\epsilon},\delta}$ drawn from $D'$, $\mathcal{O}_{\mathcal{F}}(S)$ has error of at most $\frac{1}{2}\sqrt{\epsilon}$ under distribution $D_2$.

# Proof of Part 3

If $h_1(x) = f^*(x)$, then

$$\begin{aligned}
\rho'(x) &= \frac{1}{2}\mathbb{E}\left[\frac{\#\text{ occurrences of } x \text{ in } \overline{W_C}}{|\overline{W_C}|}\right] \\
&\geq \frac{\mathbb{E}[\#\text{ occurrences of } x \text{ in } \overline{W_C}]}{c_1 m_{\sqrt{\epsilon},\delta}} \\
&\geq \frac{\mathbb{E}[\#\text{ occurrences of } x \text{ in } S_C]}{c_1 m_{\sqrt{\epsilon},\delta}} \\
&= \frac{|S_C| \cdot \rho(x)}{c_1 m_{\sqrt{\epsilon},\delta}} \\
&= \frac{|S_C| \cdot \rho_C(x) \cdot (1 - \sqrt{\epsilon}/2)}{c_1 m_{\sqrt{\epsilon},\delta}} \\
&\geq c_2 \rho_C(x) \\
&= \frac{1}{2} c_2 \rho_2(x).
\end{aligned}$$

# Proof of Part 3 (continued)

If $h_1(x) \neq f^*(x)$, then

$$
\begin{aligned}
\rho'(x) &= \frac{1}{2}\mathbb{E}\left[\frac{\#\text{ occurrences of } x \text{ in } \overline{W_I}}{|\overline{W_I}|}\right] \\
&\geq \frac{\mathbb{E}[\#\text{ occurrences of } x \text{ in } \overline{W_I}]}{c_1' m_{\sqrt{\epsilon},\delta}} \\
&\geq \frac{\mathbb{E}[\#\text{ occurrences of } x \text{ in } S_I]}{c_1' m_{\sqrt{\epsilon},\delta}} \\
&\geq \frac{\frac{1}{2}|S_2| \cdot \rho(x)}{c_1' m_{\sqrt{\epsilon},\delta}} \\
&= \frac{\frac{1}{2}|S_2| \cdot \rho_I(x) \cdot \sqrt{\epsilon}/2}{c_1' m_{\sqrt{\epsilon},\delta}} \\
&\geq c_2' \rho_C(x) \\
&= \frac{1}{2} c_2' \rho_2(x).
\end{aligned}
$$

# Main Result

**Theorem**

*Algorithm 2 returns $f \in \mathcal{F}$ with $\mathrm{err}_D(f) \le \epsilon$ with probability $1-\delta$, using $O\left(m_{\sqrt{\epsilon},\delta} \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right) + m_{\epsilon,\delta}\right)$ labels.*

**Proof.**

- Phase 1 and Phase 3 use $O\left(m_{\sqrt{\epsilon},\delta} \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right)\right)$ labels
- Phase 2:
  - FILTER uses $O(m_{\epsilon,\delta})$ labels
  - CORRECT-LABEL uses $O\left(m_{\sqrt{\epsilon},\delta} \log\left(\frac{m_{\sqrt{\epsilon},\delta}}{\delta}\right)\right)$ labels

$\square$

# Outline

# The General Case of Any $\alpha$

The fraction of perfect labelers $\alpha < \frac{1}{2} + o(1)$.

## Key Challenges

- CORRECT-LABEL($S, \delta$) may return a highly noisy labeled sample set.
- FILTER($S, h_1$) may filter the instances incorrectly.

# The General Case of Any $\alpha$

The fraction of perfect labelers $\alpha < \frac{1}{2} + o(1)$.

## Key Challenges

- CORRECT-LABEL$(S, \delta)$ may return a highly noisy labeled sample set.
- FILTER$(S, h_1)$ may filter the instances incorrectly.

## "Golden Queries"

- We have access to an "expert" and get the correct label of an example.
- If we make a golden query when the size of the majority vote is less than a fraction $1 - \alpha/2$ of labelers, then at least an $\alpha/2$ fraction of labelers can be pruned.
- After making $O(1/\alpha)$ golden queries, the good labelers form a strong majority.

# No Perfect Labelers

In this setting, crowdsourced learning reduces to the difficult agnostic learning problem.

Goal: identify the set of all *good* labelers.

## The Setting

- a pool of $n$ labelers
- good labelers have error at most $\epsilon$
- bad labelers have error at least $4\epsilon$
- at least $\lfloor \frac{n}{2} \rfloor + 1$ labelers are good

We can identify all good labelers with probability $1 - \delta$, using $O(\frac{1}{\epsilon} \log \left(\frac{n}{\delta}\right))$ queries per labeler.