Towards Confidence in the Truth: A Bootstrapping based Truth Discovery Approach

Houping Xiao1, Jing Gao1, Qi Li1, Fenglong Ma1, Lu Su1
Yunlong Feng2, and Aidong Zhang1
1 SUNY Buffalo, Buffalo, NY USA 2 KU Leuven, Leuven, Belgium
{houpingx.jing,qili22,fenglong,lusu}@buffalo.edu
yunlong.feng@esat.kuleuven.be, azhang@buffalo.edu

ABSTRACT
The demand for automatic extraction of true information (i.e., truths) from conflicting multi-source data has soared recently. A variety of truth discovery methods have witnessed great successes via jointly estimating source reliability and truths. All existing truth discovery methods focus on providing a point estimator for each object’s truth, but in many real-world applications, confidence interval estimation of truths is more desirable, since confidence interval contains richer information. To address this challenge, in this paper, we propose a novel truth discovery method (ETCIBoot) to construct confidence interval estimates as well as identify truths, where the bootstrapping techniques are nicely integrated into the truth discovery procedure. Due to the properties of bootstrapping, the estimators obtained by ETCIBoot are more accurate and robust compared with the state-of-the-art truth discovery approaches. Theoretically, we prove the asymptotical consistency of the confidence interval obtained by ETCIBoot. Experimentally, we demonstrate that ETCIBoot is not only effective in constructing confidence intervals but also able to obtain better truth estimates.

Categories and Subject Descriptors
H.2.8 [Database Management]: Database Applications—Data mining

Keywords
Truth Discovery; Confidence Interval; Bootstrapping

1. INTRODUCTION
Today, we are living in a data-rich world, and the information on an object (e.g., population/weather/air quality of a particular city) is usually provided by multiple sources. Inevitably, there exist conflicts among the multi-source data due to a variety of reasons, such as background noise, hardware quality or malicious intent to manipulate data, etc. An important question is how to identify the true information (i.e., truths) among the multiple conflicting pieces of information. Because of the volume issue, we cannot expect people to detect truth for each object manually. Thus, the demand for automatic extraction of truths from conflicting multi-source data has soared recently.

A commonly used multi-source aggregation strategy is averaging or voting. The main drawback of these approaches is that they treat the reliability of each source as the same. In real-world applications, however, different sources may have different degrees of reliability and more importantly, their reliability degrees are usually unknown a priori. To address this problem, a variety of truth discovery methods [2–5, 7, 11, 12, 15–18, 20–25] have been proposed. Although these methods vary in many aspects, they share a common underlying principle: If a piece of information is provided by a reliable source, it is more likely to be trustworthy, and the source that more often provides trustworthy information is more reliable. Following this principle, existing methods are designed to jointly estimate source reliability and truths by assigning larger weights to the reliable sources which in return play more important roles in the data aggregation.

All existing truth discovery methods [2–5, 7, 11, 12, 15–19, 21, 22] focus on providing a point estimator for each object’s truth, i.e., the estimate is a single value. However, important confidence information is missing in this single-value estimate. For example, two objects A and B receive the same truth estimate, e.g., 25. Even though the estimates are the same, the confidence in these estimates could differ significantly—A may receive 1000 claims around 25 while B only receives one claim of 25, and clearly the confidence in A’s truth estimate is much higher. Therefore, instead of a point estimation, an estimated confidence interval of the truth is more desirable. An α-level confidence interval [8] is an interval (a, b) such that $\Pr(\theta \in (a, b)) = \alpha$, where $\theta$ denotes the truth in our scenario. The width of the interval reflects the confidence in the estimate—A smaller interval indicates the higher confidence in the estimate and a larger interval means that the estimate has more possible choices within the interval. In the example we just mentioned, suppose the 95% confidence interval of A and B’s estimates are (24.9, 25.1) and (0, 50) respectively. Although both truth estimates are 25, we are more certain that the truth of A is close to 25. With such confidence information, the decision makers can use the truth estimates more wisely. However, such important confidence information cannot be

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD '16, August 13-17, 2016, San Francisco, CA, USA
© 2016 ACM. ISBN 978-1-4503-4232-2/16/08...$15.00
DOI: http://dx.doi.org/10.1145/2939672.2939831
obtained by the traditional point estimation strategy adopted by existing truth discovery methods.

The estimation of confidence intervals for objects' truths can benefit any truth discovery scenario by providing additional information (i.e., confidence) in the output, but its advantage is more obvious on long-tail data. A multi-source data is said to be long-tail in the sense that most objects receive a few claims from a small number of sources and only a few objects receive many claims from a large number of sources. As discussed in the aforementioned example, the difference in the confidence of the truth estimates is usually caused by the difference in the number of claims received by the objects. When an object receives more claims, a smaller confidence interval is obtained, and thus the estimate of this truth is more certain. It is essential to provide confidence intervals rather than points for the truth estimates on such long-tail data, which are ubiquitous. The Flight Status and Game applications used in our experiments are examples of such long-tail phenomena (The details are deferred in Subsection 4.3). In Figure 1, we present the histograms in terms of the number of claims and fit them into an exponential distribution, a typical long-tail distribution, respectively.

![Histogram](image)

**(a) Flight Status Dataset** *(b) Game Dataset*

*Figure 1: The long-tail phenomenon*

To address the problem, in this paper, we propose a novel method, Estimating Truth and Confidence Interval via Bootstrapping (ETCIBoot) to construct confidence interval estimates for truth discovery tasks. We adopt the iterative two-step procedure used in traditional truth discovery methods: 1) Update truth estimates based on the current estimates of source weights (source reliability degrees), and 2) update source weights based on the current estimates of truths. At the truth computation step, instead of giving a point estimation, we now adopt the following procedure to obtain confidence interval estimates. ETCIBoot obtains multiple estimates of an object’s truth, using bootstrapping techniques. Each estimate is obtained by calculating the weighted averaging or voting on a new set of sources which are bootstrapped from available sources. A statistic $T$ that involves the truths is constructed. Its distribution $F$ is usually unknown a priori. Based on these multiple estimates obtained via bootstrapping, we derive an estimator $\hat{T}$ of $T$ and further approximate $F$ by $\hat{F}$ (i.e., the distribution of $\hat{T}$). The confidence intervals of the truths are naturally implied in the distribution of $\hat{T}$ (i.e., $\hat{F}$). Theoretically, we prove that $\hat{T}$ is asymptotically consistent to $T$ in distribution, and the end points of the confidence intervals converge to the true ones at $O_p(n^{-\frac{1}{2}})$, where $n$ is the number of claims.

Besides providing confidence intervals, ETCIBoot is also able to provide more accurate and robust truth estimates if we use the average of the multiple estimates as the point estimator. Existing truth discovery methods typically compute weighted mean in the truth computation step, and thus the truth estimates can be quite sensitive to some outlying claims. In contrast, ETCIBoot adopts bootstrapping procedure which improves the robustness of the estimation. The truth estimates are obtained by computing the mean of bootstrap samples. These samples capture the distribution of claims in which the outlying claims’ effect can be greatly reduced.

We conduct experiments on both simulated and real-world datasets. Experimental results show that the proposed ETCIBoot can effectively construct confidence intervals for each objects and achieve better truth estimates compared with the state-of-the-art truth discovery methods.

To sum up, the paper makes the following contributions:

- To the best of our knowledge, we are the first to illustrate the importance of confidence interval estimation in truth discovery, and propose an effective method (ETCIBoot) to address the problem.
- Theoretically, we prove that the confidence interval obtained by ETCIBoot is asymptotically consistent.
- The point estimates obtained by ETCIBoot are more accurate and robust compared with existing approaches due to the properties of bootstrap sampling, which is nicely integrated into the truth discovery procedure in ETCIBoot.
- Experimental results demonstrate the effectiveness of ETCIBoot in constructing confidence intervals as well as identifying truths.

## 2. PROBLEM SETTING

We first introduce terminologies and notations in this section. Then, the problem is formally defined.

**Definition 1.** An object is an item of interest. Its true information is defined as a truth.

**Definition 2.** The reliability of a source measures the quality of its information. A source weight is proportional to its reliability, i.e., the higher the quality of a source’s information, the higher its reliability, and the higher its weight. Assume that there are $S := \{s\}_i^N$ sources, providing claims on objects $N := \{n\}_{i=1}^N$, which are unknown a priori. For the $n$-th object, $S_n$ represents the set of sources which provide claims for it. The long-tail phenomenon denoted as $\{x_n^s\}_{n=1}^N$, which are unknown a priori. For the $n$-th object, $S_n$ represents the set of sources which provide claims for it. The long-tail phenomenon denoted as $\{x_n^s\}_{n=1}^N$, where $x_n^s$ represents the claim provided by the $s$-th source for the object $n$. The whole data collection on objects $N$ is further denoted as $\mathcal{X} := \bigcup_{n=1}^N X_n$. For the $s$-th source, we assume the difference $\epsilon_s$ between its claims and truths follows a normal distribution with mean 0 and variance $\sigma^2_s$, i.e., $\epsilon_s \sim \text{Normal}(0, \sigma^2_s)$. This assumption is commonly used in existing truth discovery works [11, 12, 22]. $\epsilon_s$ captures the error of source $s$. As a small $\epsilon_s$ means that the claims are close to the truths, $\sigma^2_s$ measures the quality of the claims provided by the $s$-th source. We further denote the weight of source $s$ as $w_s$. Definition 2 implies that the larger $\sigma^2_s$, the smaller $w_s$.

**Truth Discovery Task.** Truth discovery task is formally defined as follows: Given the multi-source data $\mathcal{X}$, the goal of a truth discovery approach is to obtain estimates $\hat{x}_n$ which are as close to $x_n^s$ as possible (\forall n \in \mathcal{N})$. Besides, for any $\alpha \in (0, 1)$, we can also provide an $\alpha$-level two-sided confidence interval for each object.

We summarize the notations in Table 1.
3. METHODOLOGY

In this section, we first review some preliminaries about truth discovery and confidence interval in Subsection 3.1. We then introduce two main components of ETCIBoot: a novel strategy for data aggregation (ETBoot) and a method for confidence interval construction (CIC) in Subsections 3.2 and 3.3, respectively. The proposed ETCIBoot is further summarized in Subsection 3.4. Finally, we present the theoretical analysis of the confidence interval estimates obtained by ETCIBoot in Subsection 3.5.

3.1 Preliminary

Truth Discovery

The goal of a truth discovery task is to identify objects’ truths (i.e., true information) from conflicting multi-source data. Many truth discovery methods have been proposed to estimate truths and weights iteratively. Details can be found in Section 5. We briefly introduce two iterative steps as follows.

Weight Update. Source weights play important roles in truth discovery. The underlying principle is that: If a source more often provides reliable information, it has a larger weight, and consequently this source contributes more in the truth estimation step discussed below. Based on this principle, various weight update strategies have been proposed. In this paper, we adopt the weight estimation introduced in [11]. Specifically, a source weight is inversely proportional to its total difference from the estimated truth, namely,

\[ \omega_s \propto \frac{\chi^2_{\alpha}(|N|)}{\sum_{n \in N} (\hat{x}_n - x_n)^2}, \]

(1)

where \( \chi^2_{\alpha}(|N|) \) is the \( \alpha \)-th percentile of a \( \chi^2 \)-distribution with \( |N| \) degree. It is used to capture the effect of the number of claims so that small sources get their weights reduced.

Truth Estimation. A commonly used strategy is weighted averaging for continuous data or weighted voting for categorical data, namely,

\[ \hat{x}_n = \frac{\sum_{s \in S_n} \omega_s x_n^s}{\sum_{s \in S_n} \omega_s}, \]  
\[ \text{or}, \quad \hat{x}_n = \arg\max_x \sum_{s \in S_n} \omega_s I(x_n^s, x) \sum_{s \in S_n} \omega_s, \]

(2)

where \( I(x_n^s, x) = 1 \) if \( x_n^s = x \); otherwise it is 0. The weights are obtained at the Weight Update step; the truth estimated at this step will be used to update weights based on (1).

Providing proper initializations, Weight Update and Truth Estimation are iteratively executed until the convergence condition is satisfied.

Confidence Interval

Assume that an experiment has a sample set \( X = \{x_1, \cdots, x_n\} \) from \( F_\mu(x) \), where \( F_\mu \) is an accumulative density function (c.d.f.) with a parameter \( \mu \). An \( \alpha \)-level confidence interval for the parameter \( \mu \) is defined as follows:

**Definition 3.** For any \( \alpha \in (0,1) \), \( (\mu_L, \mu_R) \) is called an \( \alpha \)-level two-sided confidence interval of a parameter \( \mu \) if it satisfies the following condition:

\[ P(\mu \in (\mu_L, \mu_R)) = \alpha. \]

(3)

The immediately preceding probability statement (3) can be read: Prior to the repeated independent trials of the random experiment, \( \alpha \) is the probability that the random interval \( (\mu_L, \mu_R) \) includes the unknown parameter \( \mu \).

Given the distribution of the experiment sample set \( X \), the exact end points of a confidence interval is defined as:

**Definition 4.** The exact end points of an \( \alpha \)-level two-sided confidence interval of \( \mu \) with a known c.d.f. \( F \) are:

\[ \mu_L, \mu_R \] = \begin{cases} \mu - \sqrt{\frac{\text{Var}(\mu)}{\alpha}} F^{-1}(1 - \alpha), \\ \mu + \sqrt{\frac{\text{Var}(\mu)}{\alpha}} F^{-1}(\alpha), \end{cases} \]

(4)

where \( F^{-1}(\cdot) \) is the inverse function of c.d.f. \( F \), \( \text{Var}(\mu) \) is the variance of \( \mu \), and \( n \) is the number of observed samples.

However, (4) is always unknown a prior because \( F \) is unknown. The major task in this paper is to construct a confidence interval estimate for each truth.

3.2 ETBoot Strategy

In this subsection, we introduce a novel bootstrapping-based strategy for the truth discovery task. We term this strategy as Estimating Truth via Bootstrapping (ETBoot). All existing truth discovery methods apply weighted averaging or voting using all sources’ information. In contrast, ETBoot first bootstraps multiple sets of sources and then on each set of the bootstrapped sources it obtains a truth estimate based on (2). The final truth estimator is defined as the mean of these estimates. Due to the properties of bootstrapping techniques which are nicely integrated into the truth discovery procedure, ETBoot is more robust to the outlying claims and then achieves a better estimate of the truth. Moreover, as shown in Subsection 3.3, ETBoot is able to construct an \( \alpha \)-level two-sided confidence interval of the estimated truth for any \( \alpha \in (0,1) \).

The detailed procedure of ETBoot is as follows: for the \( n \)-th object, it obtains \( B \) estimates of its truth, i.e., \( \{\hat{x}_b^n\}_{b=1}^B \), where \( \hat{x}_b^n \) is obtained by the following two-step procedure:

- **Step 1: Source Bootstrap.** In this step, we randomly sample \( |S_b| \) sources \( S_b \) from \( S_n \) with replacement. The sampled data is denoted as \( X^n_b = \{x^n_s\}_{s \in S_b} \).

- **Step 2: Truth Computation.** Based on the sampled data \( X^n_b = \{x^n_s\}_{s \in S_b} \), \( \hat{x}_b^n \) is calculated based on (2).

The final estimator \( \hat{x}_n^{\text{Boot}} \) \(^{1}\) for the \( n \)-th object’s truth is further defined as:

\[ \hat{x}_n^{\text{Boot}} = \frac{1}{B} \sum_{b=1}^B \hat{x}_b^n. \]

---

1We use \( ^{\text{Boot}} \) to represent the estimator obtained by Bootstrapping throughout the paper.
Compared with existing truth discovery methods which use (2), the proposed ETBoot combines results from multiple bootstrap samples instead of using all the sources at once. This enables ETBoot to obtain more robust estimates and confidence interval estimates as explained in Subsection 3.3.

The pseudo code of ETBoot for the n-th object is summarized in Algorithm 1.

**Algorithm 1 ETBoot**

**Input:** $S_n$, $x_n$, $\{o_s\}_{s \in S_n}$, and $B$

1: for the b-th iteration ($b = 1, \ldots, B$) do
2: Bootstrap $S_n^b$ from $S_n$;
3: Extract $X_n^b$ from $X_n$;
4: Calculate $x_n^b$ according to (2);
5: end for
6: Calculate $\hat{x}_{n}^{\text{Boot}}$ according to (5);

**Output:** $\hat{x}_{n}^{\text{Boot}}$.

### 3.3 Confidence Interval Construction

In this subsection, we introduce the procedure of constructing an $\alpha$-level two-sided confidence interval of an object’s truth. We illustrate it for the n-th object. The procedure is similar for other objects.

We denote the estimator we are interested in as $\hat{\theta}(X_n)$ corresponding to the dataset $X_n = \{x_n^s\}_{s \in S_n}$. In our scenario, $\hat{\theta}(X_n)$ denotes the truth estimate. For simplicity, we ignore the subscript $n$ for $X_n$. In a truth discovery task, the truth estimate is calculated as $\hat{\theta}(X) = \sum_{s \in S_n} x_n^s \omega_s^2$, yielding,

$$E(\hat{\theta}(X)) = x_n^*,$$

and,

$$\text{Var}(\hat{\theta}(X)) = \sum_{s \in S_n} \omega_s^2 \sigma_s^2 \left(\sum_{\sigma \in S_n} \omega_\sigma^2\right)^{-2}.$$ (6)

The corresponding estimate of $\text{Var}(\hat{\theta}(X))$ is defined as $\hat{\text{Var}}(\hat{\theta}(X)) = \sum_{s \in S_n} \omega_s^2 \sigma_s^2$ where $\hat{\sigma}_s^2 = \sum_{\sigma \in S_n} (x_n^s - \hat{x}_{n}^{\text{Boot}})^2/\sum_{\sigma \in S_n} \omega_\sigma^2$, and $\hat{x}_{n}^{\text{Boot}}$ is obtained by ETBoot. To obtain a confidence interval of the truth $x_n^*$, we first construct a statistic $T$ which is related to $x_n^*$, and then estimate the accumulated density function of $T \sim F(t)$. In our scenario, $T$ is defined as follows:

$$T = \frac{\hat{\theta}(X) - x_n^*}{\sqrt{\text{Var}(\hat{\theta}(X))}/\sqrt{|S_n|}},$$ (7)

which measures the error between $\hat{\theta}(X)$ and $x_n^*$. The confidence interval of $x_n^*$ is available once the distribution of $T$ is determined. More precisely, let $T^{(\alpha)}$ indicate the (100 $\alpha$)-th percentile of $T$, i.e., $\alpha = \int_{-\infty}^{T^{(\alpha)}} dF(t)$. Thus, we have that

$$\mathbb{P}\left(T^{(\alpha/2)} \leq \frac{\hat{\theta}(X) - x_n^*}{\sqrt{\text{Var}(\hat{\theta}(X))}/\sqrt{|S_n|}} \leq T^{(1 - \alpha/2)}\right) = \alpha.$$(8)

Moreover, an $\alpha$-level two-sided confidence interval of $x_n^*$ is naturally implied in (8), that is,

$$\frac{\hat{\theta}(X) - T^{(1 - \alpha/2)}[\sqrt{|S_n|}]^1}{\sqrt{\text{Var}(\hat{\theta}(X))/\sqrt{|S_n|}}},$$ (9)

$$\hat{\theta}(X) - T^{(\alpha/2)}[\sqrt{|S_n|}]^1 \leq \frac{\hat{\theta}(X) - T^{(\alpha/2)}[\sqrt{|S_n|}]^1}{\sqrt{\text{Var}(\hat{\theta}(X))/\sqrt{|S_n|}} \leq \hat{\theta}(X).$$ (10)

Thus, the width of the confidence interval is proportional to $\sqrt{|S_n|}$. It implies that if an object is claimed by more sources then the width of its truth’s confidence level is smaller, and vice versa. Especially, when the long-tail multi-source data is involved, this phenomenon is clearer.

However, as the $T$-percentile is usually unknown a priori, estimation of $T^{(\alpha)}$ is required. One commonly used strategy is bootstrap sampling [1, 6, 8, 10, 14]. Note that at the b-th iteration of ETBoot (Algorithm 1), we have bootstrapped $X_n^b$. Based on $X_n^b$, we are able to calculate both $\hat{\theta}(X_n^b)$ and $\sqrt{\text{Var}(\hat{\theta}(X_n^b))}$, yielding an estimator $\hat{T}_b$ for the statistic $T$, that is,

$$\hat{T}_b = \frac{\hat{\theta}(X_n^b) - \hat{\theta}(X)}{\sqrt{\text{Var}(\hat{\theta}(X_n^b))/\sqrt{|S_n|}}}.$$ (11)

Moreover, the estimate of $T^{(\alpha)}$ is defined as follows:

$$\hat{T}^{(\alpha)} = \sup \left\{ t \in \{\hat{T}_1, \ldots, \hat{T}_B\} : \frac{\#(\hat{T}_b \leq t)}{B} \leq \alpha \right\}.$$ (12)

(12) provides estimates of (9) and (10). Thus, the estimate of an $\alpha$-level two-sided confidence interval is defined as follows:

$$\hat{\theta}(X) - \frac{\hat{T}^{(1 - \alpha/2)}[\sqrt{|S_n|}]^1}{\sqrt{\text{Var}(\hat{\theta}(X))}/\sqrt{|S_n|}} \leq \hat{T}^{(\alpha/2)}[\sqrt{|S_n|}]^1 \leq \hat{\theta}(X).$$ (13)

We summarize the procedure of constructing confidence intervals as CIC, i.e., Confidence Interval Construction. Its pseudo is presented in Algorithm 2 for the n-th object.

**Algorithm 2 CIC**

**Input:** $\{X_n^b\}_{b=1}^B$, $\hat{x}_{n}^{\text{Boot}}$, and $\alpha$.

1: Calculate $\sigma_s^2$ for $s \in S_n$;
2: for the iteration $b$ ($b = 1, \ldots, B$) do
3: Calculate $\sqrt{\text{Var}(\hat{\theta}(X_n^b))}$ and $\hat{T}_b$ according to (11);
4: end for
5: Choose $\hat{T}^{(1 - \alpha/2)}$ and $\hat{T}^{(\alpha/2)}$ according to (12);

**Output:** Endpoints calculated based on (13) and (14).

### 3.4 ETCIBoot Algorithm

So far, we introduce the update for source weights (i.e., (1)), a new truth estimation strategy, ETBoot, and the construction of confidence intervals for truths via CIC. Combining them together, we propose a novel truth discovery approach, Estimating Truth and Confidence Interval via Bootstrapping (ETCIBoot), to automatically construct confidence intervals as well as identify objects’ truths. The main component of the proposed ETCIBoot consists of the following three steps:

(i) **Weight Update.** Given initialization of truth $\{x_n^0\}$, source weights are updated based on (1).

(ii) **Truth Estimation.** With source weights computed from previous step, for each object $n$, we obtain truth estimators via ETBoot at this step to obtain $\hat{x}_{n}^{\text{Boot}}$ associated with $\{X_n^b\}_{b=1}^B$.

(iii) **Confidence Interval Construction.** For all objects, the estimation of confidence intervals for their truths are obtained via CIC.
The above two steps are executed iteratively until no truth estimates change anymore. The pseudo code of the proposed ETCIBoot algorithm is shown in Algorithm 3.

Algorithm 3 ETCIBoot

**Input**: the whole data collection $X$, confidence level $\alpha$, and the number of bootstrapping samples $B$.

1: Initialize truths $x^0_1, \ldots, x^0_N$ as average;
2: while the convergence condition is not satisfied do
3:  Compute $\omega_s$ for each source $s$ according to (1);
4:  for each object $n (n = 1, \ldots, N)$ do
5:  Conduct ETBoot to obtain $\hat{x}^*_n$;
6:  Calculate the confidence interval $CI(n, \alpha)$ via $P$;
7:  end for
8: end while
**Output**: $\{\hat{x}^*_n\}_1^N$ and confidence interval $\{CI(n, \alpha)\}_1^N$.

### 3.5 Theoretical Analysis

In this subsection, we present the theoretical analysis on the confidence interval estimates, i.e., (13) and (14), obtained via ETCIBoot. We first prove that $\hat{T}$ converges to $T$ in distribution and present it in Proposition 1.

**Proposition 1.** Assume that $x^s_n \sim N(x^*_n, \sigma^2_n)$, for any $s \in S_n$. Let $T$ and $T^*$ be defined as (7) and (11), respectively. Then, we have that

$$\lim_{|S_n| \to \infty} \|P^*(\hat{T} \leq t) - P(T \leq t)\| = 0, \quad \text{a.s.}$$

where $P^*$ is the probability calculated based on the bootstrapping sample distribution, $|S_n|$ is the cardinality of $S_n$, $t$ is any real number, and a.s. means 'almost surely'.

**Proof.** See Appendix A for a detailed proof. □

Proposition 1 is a straightforward result from Theorem 1 in [14], where the author provides sufficient conditions to guarantee the convergence of the bootstrapping samples. Thus, the proof of Proposition 1 is to testify whether the ETCIBoot satisfies these sufficient conditions, as shown in Appendix A. Proposition 1 shows that the bootstrapping estimator $\hat{T}$ converges to $T$ in distribution. It enables us to use the bootstrapping distribution to approximate the unknown distribution $F$ for confidence interval construction.

Next, in Proposition 2, we show that the upper end point of an $\alpha$-level one-sided confidence interval obtained via ETCIBoot is close to that from the theoretical distribution.

**Proposition 2.** Given $T \sim F(x), \hat{T} \sim \hat{F}(x)$ and a dataset $X$, we have that

$$\hat{\theta}_{\hat{T}}(x) = \hat{\theta}_{T}(x) + O_p(n^{-3/2}),$$

where $P^*(\theta(x) \leq \hat{\theta}_{\hat{T}}(x)) = \alpha$, $P(\theta(x) \leq \hat{\theta}_{T}(x)) = \alpha$, $n = |X|$, and $O_p$ means the order holds in probability.

**Proof.** See Appendix B for a detailed proof. □

Proposition 2 shows that the endpoint of an $\alpha$-level one-sided confidence interval obtained by bootstrapping $\hat{T}$ is close to that obtained by $T$, provided that there are enough samples. As any $\alpha$-level two-sided confidence interval can be obtained by two one-sided confidence intervals, the results (16) also hold for (13) and (14). In truth discovery tasks, ETCIBoot is able to provide more accurate confidence intervals for the objects’ truths, if they receive more claims. This result is more obvious especially on long-tail data.

### 4. EXPERIMENTS

In this section, we evaluate the proposed ETCIBoot method on both simulated and real-world datasets. We first introduce the experimental setup in Subsection 4.1. Then, we test the ETCIBoot and baselines on simulated datasets generated in different scenarios and real-world datasets in Subsections 4.2 and 4.3, respectively. Experimental results show that: (1) ETCIBoot outperforms the state-of-the-art truth discovery methods in most cases, and (2) ETCIBoot can provide accurate confidence interval estimates.

#### 4.1 Experimental Setup

In this part, we introduce the baseline methods and discuss the measurements for evaluation.

**Baseline Methods.** For all truth discovery methods, we conduct them on the same input data in an unsupervised manner. Although ground truths are available, we only use them for evaluation. For different data types, different baselines are adopted, including both the naive conflict resolution methods and the state-of-the-art truth discovery methods. More precisely, for continuous data we use Median, Mean, CATD [11], CRH [12] and GTM [22]. Baselines used for categorical data include: Voting, Accusim [5], 3-estimate [7], CRH [12], Investment [18], CATD [11], ZenCrowd [3], Dawid-Kraskov [2], and TruthFinder [21]. Details of baselines are discussed in the related work (Section 5).

**Measurements.** As the experiments involve both continuous and categorical data, we introduce different measurements. For data of continuous type, we adopt both the mean of absolute error (MAE) and the root of mean square error (RMSE); Error Rate is used for data of categorical type. The details of the measurements are:

- **MAE: MAE** measures the $L^1$-norm between the methods’ output and the ground truths. It tends to penalize more on small errors.
- **RMSE: RMSE** measures the $L^2$-norm between the methods’ output and the ground truths. It tends to penalize more on the large distance and less on the small distance comparing with MAE.
- **Error Rate: Error Rate** is defined as the percentage of mismatched values between the output of each method and the ground truths.

*Note That: the smaller the measurement value, the closer to ground truths the methods’ output. Therefore, for all measurements, the smaller the value, the better the method.*
Table 2: Comparison on simulated data: all scenarios

<table>
<thead>
<tr>
<th>Method</th>
<th>Scenario 1 (Uniform(0,1))</th>
<th>Scenario 2 (Gamma(1,3))</th>
<th>Scenario 3 (FoldedNormal(1,2))</th>
<th>Scenario 4 (Beta(1, 1/2))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAE (10^{-3})</td>
<td>RMSE (10^{-1})</td>
<td>MAE (10^{-3})</td>
<td>RMSE (10^{-1})</td>
</tr>
<tr>
<td>ETCIBoot</td>
<td>1.200</td>
<td>2.724</td>
<td>0.700</td>
<td>1.599</td>
</tr>
<tr>
<td>CATD</td>
<td>1.300</td>
<td>2.903</td>
<td>0.800</td>
<td>1.708</td>
</tr>
<tr>
<td>Median</td>
<td>4.000</td>
<td>9.275</td>
<td>2.800</td>
<td>6.648</td>
</tr>
<tr>
<td>Mean</td>
<td>8.000</td>
<td>17.224</td>
<td>11.700</td>
<td>24.988</td>
</tr>
<tr>
<td>GTM</td>
<td>4.000</td>
<td>8.549</td>
<td>4.000</td>
<td>8.593</td>
</tr>
</tbody>
</table>

Data Generation. The procedure of generating simulated data is shown as follows:

(i) We first generate a vector of the number of claims $C$, e.g., $C = (5, 10, 15, \ldots, 50)$.

(ii) For each $e_i \in C$, there are $o_i = e_i^7 \cdot c_i^{-1.5}$ objects which will receive $c_i$ claims. This power law function is used to create the long-tail multi-source data. Thus, there are totally $O = \sum_i o_i$ objects and $S = \max\{o_i\}$ sources.

(iii) For each source, we randomly generate its reliability $\sigma_s^2 \sim F$, where $F$ is a pre-defined distribution. Thus, for each source, its claims are generated from $\text{Normal}(0, \sigma_s^2)$. Here, $\sigma_s^2$ captures reliability degree of the $s$-th source’s information. The larger value the $\sigma_s^2$, the lower reliability degree of the $s$-th source.

Experiments. In the following experiments, we simulate different scenarios via different source reliability distribution $F$. We set $C = 70 : 100$; thus, there are 31 objects and 100 sources. Note that the number of objects is not large. This is used to better display the experimental results on the confidence interval estimates. To reduce the randomness, we repeat the experiment 100 times and report the average results. As the simulated data is continuous, MAE and RMSE are used for evaluation. We simulate 4 scenarios and the detail of each scenario is discussed as follows. Note that $\sigma_s^2$ represents the source reliability degree. The larger value the $\sigma_s^2$, the lower reliability degree the source.

Scenario 1: $\sigma_s^2 \sim \text{Uniform}(0,1)$. In this scenario, all source reliability degrees are uniformly distributed in (0, 5).

Scenario 2: $\sigma_s^2 \sim \text{Gamma}(1,3)$. In this scenario, most of the sources are reliable with high reliability degrees. However, there are a few unreliable sources with very small reliability degrees.

Scenario 3: $\sigma_s^2 \sim \text{FoldedNormal}(1,2)$. As Folded Normal is a long-tail distribution, in this scenario, it generates a few unreliable sources. Compared with Scenarios 1 and 2, the reliable sources have higher reliability degrees.

Scenario 4: $\sigma_s^2 \sim \text{Beta}(1, 1/2)$. In this scenario, source reliability degrees are within $0 \sim 1$. Compared with other scenarios, there are much more reliable sources.

We show the histograms of the source variances in Figure 2, which implies that the simulated data covers various scenarios with varying source reliability distributions. We report the results in terms of MAE and RMSE in Table 2.

Comparison with Baselines. Table 2 shows that the proposed ETCIBoot outperforms all baselines in all scenarios in terms of both MAE and RMSE. When estimating the truth for each object $n$, ETCIBoot obtains multiple truth estimates which are calculated according to (2) based on the bootstrapped claims. Then, the final truth estimator is defined as the average of these estimates. Experimentally, we generate 10 * $|S|$ bootstrapping samples. Due to the properties of bootstrapping, ETCIBoot is robust to the outliers. We perform experiments on the confidence interval estimates. To reduce the randomness, we repeat the experiment 100 times and report the average results. As the simulated data is continuous, MAE and RMSE are used for evaluation. We simulate 4 scenarios and the detail of each scenario is discussed as follows. Note that $\sigma_s^2$ represents the source reliability degree. The larger value the $\sigma_s^2$, the lower reliability degree the source.

Confidence Interval Comparison. For confidence interval comparison, we compare the results of ETCIBoot with that obtained by theoretical distribution, i.e., normal distribution. Note that $\hat{x}_n \sim \text{Normal}(\bar{x}_n, \frac{\sum_s o_s \cdot \sigma_s^2}{\sum_s o_s \cdot \bar{o}_s})$ (based on (2)). As the true $\sigma_s^2$ is known for each source, we know the theoretical distribution for $\hat{x}_n$, based on which we can further obtain the 95%-level confidence interval. We term the confidence interval obtained in this way as CI-Normal. The confidence interval (i.e., (13) and (14)) for the truth estimators, which is obtained by the ETCIBoot using the bootstrapping technique, is referred to as CI-ETCIBoot.

We report the results in Scenarios 1 ~ 4 in Figures 3 ~ 6, respectively. From Figures 3 ~ 6, we can draw the following conclusions: (1) The CI-ETCIBoot is much smaller than CI-Normal in all simulated scenarios. Note that the smaller the confidence interval, the more confident the estimator. For example, in Scenario 1 the shaded area (i.e., the area between the lower and upper bound curves) of CI-Normal in Figure 3(a) is larger than that of CI-ETCIBoot in Figure 3(b). Similar conclusions can be drawn in other
scenarios. Thus, the experimental results show the power of the ETCIBoot on constructing effective confidence intervals. (2) As most sources are reliable in Scenarios 2 ~ 4, comparing with Scenario 1, the width of CI-ETCIBoot or CI-Normal in other scenarios is smaller, which indicates the higher overall confidence in these scenarios.

Next we conduct experiments to illustrate the relationship between the width of confidence interval and the number of claims on long-tail data. We follow the same procedure to generate the simulated data, except that we choose the number of claims as 2 to 30. If there is only one claim, it is impossible to construct the confidence interval. We present the width of CI-Normal and CI-ETCIBoot in all scenarios in Figures 7(a) and 7(b), respectively. Meanwhile, we also fit them into a polynomial function of $N \cdot (N^{-1/2})$, respectively. The red line with square marker represents the fitting line, averaging over all scenarios. From Figure 7, we can see that the width of the 95% confidence interval, obtained via either normal distribution or ETCIBoot, decreases with respect to the number of claims at an error rate $N^{-1/2}$, where $N$ is the number of claims. It confirms the theoretical analysis that if an object receives more claims then its estimator is more accurate. Moreover, the width of CI-ETCIBoot is much smaller than that of CI-Normal, which demonstrates that ETCIBoot is able to provide a more confident estimator. This advantage is achieved by incorporating bootstrapping techniques into truth discovery procedure in ETCIBoot.

4.3 Real-World Datasets

In this subsection, we present the experimental results on two continuous datasets and two categorical datasets. Experiments show that the proposed ETCIBoot is able to obtain more accurate estimates of truths comparing with baselines. We first introduce the description of the datasets and then report the results.

Continuous Data

Dataset Description. The following datasets of continuous data type are used in experiments:

- Flight Status Dataset: The flight data [13] is collected from 38 sources on every day in December 2011. There are totally 247 users and 129 objects (i.e., indoor hallways). The ground truth of the hallway length is obtained by manually measuring the indoor hallways. The goal is to estimate the distance of indoor hallways from the data provided by a crowd of users.

- Indoor Floorplan Dataset: We develop an Android App that can estimate the walking distances of smartphone users though multiplying their step sizes by step count inferred using the in-phone accelerometer. There are totally 247 users and 129 objects (i.e., indoor hallways). The ground truth of the hallway length is obtained by manually measuring the indoor hallways. The goal is to estimate the distance of indoor hallways from the data provided by a crowd of users.

- Simulated data in all scenarios: Confidence Interval width w.r.t. the number of claims ($N$)
Results Analysis. We present the results of ETCIBoot and baselines with respect to MAE and RMSE on the continuous datasets in Table 3. The experimental results show that the proposed ETCIBoot can achieve the best performance on both datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Indoor Floorplan MAE ($10^3$)</th>
<th>Indoor Floorplan RMSE ($10^3$)</th>
<th>Flight Status MAE ($10^3$)</th>
<th>Flight Status RMSE ($10^3$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETCIBoot</td>
<td>.9219</td>
<td>1.2992</td>
<td>.0310</td>
<td>.9933</td>
</tr>
<tr>
<td>CATD</td>
<td>0.9960</td>
<td>1.3845</td>
<td>1.077</td>
<td>8.120</td>
</tr>
<tr>
<td>CRH</td>
<td>1.1929</td>
<td>1.5955</td>
<td>1.074</td>
<td>8.094</td>
</tr>
<tr>
<td>Median</td>
<td>1.3797</td>
<td>1.7860</td>
<td>1.070</td>
<td>8.020</td>
</tr>
<tr>
<td>Mean</td>
<td>1.7851</td>
<td>2.2846</td>
<td>1.055</td>
<td>7.893</td>
</tr>
<tr>
<td>GTM</td>
<td>1.2845</td>
<td>1.6823</td>
<td>1.078</td>
<td>8.132</td>
</tr>
</tbody>
</table>

Table 3: Comparison on continuous data

On Indoor Floorplan dataset, as the number of objects is small, we also present the confidence intervals obtained by ETCIBoot for each object in Figure 8. The figure shows that in most cases the confidence intervals provided by ETCIBoot contain the corresponding objects’ truths. However, there are some confidence intervals which do not contain truths. A possible reason is: These objects are claimed by a few sources and the information provided by these sources is far away from the truth. Take the 9-th object for example. There are only 4 sources which provide claims, among which the smallest value is 14.3 that is still very larger than the ground truth 10.8. As a result, it is impossible to correctly identify these objects’ truths for any truth discovery method. Therefore, the confidence interval estimates obtained by ETCIBoot do not contain the truths for these objects.

On Flight Status dataset, the data on each day is treated as a single data collection. As there are many flights only claimed by a few sources, the performance of baselines is not satisfactory. We conduct a case study on Day 1 dataset. We count the statistics on how many claims of an object receives from multiple users based on a TV game show “Who Wants to Be a Millionaire” via an Android App. There are 37,029 Android users and 2,103 questions. Ground truths are available for evaluation. The goal is to identify each question’s answer from the users’ answers.

![Figure 8: Indoor Floorplan dataset: CI-ETCIBoot](image)

Categorical Data

Dataset Description. We introduce the details of two categorical datasets and their tasks as follows:

- SFV Dataset: SFV dataset is built upon the annual Slot Filling Validation (SFV) competition of the NIST Text Analysis Conference Knowledge Base Population track [9]. In this task, given a query (an object), e.g., the birthday of Obama, 18 slot filling systems (sources) extract useful claims independently from a large-scale corpus. The 2011 SFV dataset contains 2,538 claims from 18 sources for 328 objects. The goal is to extract the true answer for each query from the systems’ claims.

- Game Dataset: Game dataset [11] collects answers from multiple users based on a TV game show “Who Wants to Be a Millionaire” via an Android App. There are 37,029 Android users and 2,103 questions. Ground truths are available for evaluation. The goal is to identify each question’s answer from the users’ answers.

Results Analysis. For categorical data, we first encode the claims into probability vectors and then apply the methods proposed for continuous data, such as ETCIBoot, CATD, etc. The detailed procedure is: For a question with 4 possible choices, the first choice is encoded into a 4-element vector (1,0,0,0). In Tables 4 and 5, we present the experimental results of the proposed ETCIBoot as well as baselines on the SFV and Game datasets, respectively.

On SFV dataset, there are only 18 sources, so we have a limited number of sources to bootstrap at each iteration of ETCIBoot. Thus, the result of the proposed ETCIBoot (.0945) is not the best, but still comparable with the two best methods: AccuSim (.0701) and TruthFinder (.0793).

On Game dataset, the number of sources (37,029) is sufficient for bootstrapping. Although CATD performs best among all baselines, the proposed ETCIBoot achieves even better performance compared with CATD. Especially, on the Levels 8, 9, and 10, the proposed ETCIBoot improves the results by 33.28%, 50.00% and 33.30%, respectively, when...
Truth discovery has become an eye-catching term recently and many truth discovery methods have been proposed to identify true information (i.e., truths) from the conflicting multi-source data. The advantage of truth discovery over the naive aggregation methods such as averaging or voting is that it can capture the variance in sources’ reliability degrees. Therefore, truth discovery methods can estimate source reliability automatically from the data, which is integrated into truth computation as source weight. Consequently, the more reliable sources contribute more in the final aggregation step.

A large variety of truth discovery methods have been designed to jointly estimate truths and source reliability. In [12], the authors formulate the truth discovery task into an optimization framework (CRH). They propose to minimize the overall weighted distance between claims from sources and aggregated results. CATD [11] is a statistical method that has been proposed to deal with long-tail phenomenon in truth discovery tasks, where confidence interval is incorporated in source weight estimation. However, CATD does not consider the long-tail phenomenon on objects, which can be solved by ETCIBoot. In [22], the authors propose a probabilistic model based truth discovery framework (GTM). Both AccuSim [5] and TruthFinder [21] adopt Bayesian analysis to estimate source reliability and update truths iteratively. In [18], the authors take the prior knowledge on truth and background information into consideration and propose a truth discovery method Investment. In [7], 3-Estimate considers the difficulty of getting the truth for each object when calculating source weights as well as complement vote. Dawid&Skene [2] and ZenCrowd [3] propose to use Expectation-Maximization technique to update source weights and truths simultaneously, based on a confusion matrix.

However, most existing truth discovery methods have the following limitations: (1) As most of them apply weighted averaging, they are sensitive to outlying claims, and (2) they focus on point estimation of the truth, where important confidence information is missing. To the best of our knowledge, this is the first paper to illustrate the importance of confidence interval estimation in truth discovery, and proposes an effective method (ETCIBoot) to address it. By integrating bootstrapping into truth discovery, ETCIBoot is robust compared with the state-of-the-art truth discovery methods.

### 6. CONCLUSIONS

In this paper, we first illustrate the importance of confidence interval estimation in truth discovery, which has never been discussed in existing work. To address the problem, we propose a novel truth discovery method (ETCIBoot) to construct confidence interval estimates as well as identify truths. The bootstrapping techniques are nicely integrated into the truth discovery procedure in ETCIBoot. Due to the properties of bootstrapping, the estimators obtained by ETCIBoot are more accurate and robust compared with the state-of-the-art truth discovery approaches. Theoretically, we prove that the confidence interval obtained by ETCIBoot is asymptotically consistent. Experimentally, we demonstrate that ETCIBoot is not only effective in constructing confidence intervals but also able to obtain better truth estimates.

### 7. ACKNOWLEDGEMENTS

This work was sponsored in part by US National Science Foundation under grant IIS-1319973, IIS-1553411 and CNS-1566374. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

---

**Table 4: Comparison on SFV dataset**

<table>
<thead>
<tr>
<th>Method</th>
<th>Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETCIBoot</td>
<td>.0945</td>
</tr>
<tr>
<td>CATD</td>
<td>.1037</td>
</tr>
<tr>
<td>CRH</td>
<td>.0854</td>
</tr>
<tr>
<td>ZenCrowd</td>
<td>.1010</td>
</tr>
<tr>
<td>AccuSim</td>
<td>.0701</td>
</tr>
<tr>
<td>3-Estimates</td>
<td>.1128</td>
</tr>
<tr>
<td>Voting</td>
<td>.1128</td>
</tr>
<tr>
<td>Dawid&amp;Skene</td>
<td>.0985</td>
</tr>
<tr>
<td>Investment</td>
<td>.2896</td>
</tr>
<tr>
<td>TruthFinder</td>
<td>.0793</td>
</tr>
</tbody>
</table>

**Table 5: Comparison on Game dataset**

<table>
<thead>
<tr>
<th>Method</th>
<th>Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETCIBoot</td>
<td>.0165</td>
</tr>
<tr>
<td>CATD</td>
<td>.0132</td>
</tr>
<tr>
<td>CRH</td>
<td>.0264</td>
</tr>
<tr>
<td>ZenCrowd</td>
<td>.0330</td>
</tr>
<tr>
<td>AccuSim</td>
<td>.0264</td>
</tr>
<tr>
<td>3-Estimates</td>
<td>.0297</td>
</tr>
<tr>
<td>Dawid&amp;Skene</td>
<td>.0330</td>
</tr>
<tr>
<td>Voting</td>
<td>.0305</td>
</tr>
<tr>
<td>Investment</td>
<td>.0407</td>
</tr>
<tr>
<td>TruthFinder</td>
<td>.0693</td>
</tr>
<tr>
<td>SFV</td>
<td>.1010</td>
</tr>
<tr>
<td>Voting</td>
<td>.1128</td>
</tr>
<tr>
<td>TruthFinder</td>
<td>.1239</td>
</tr>
<tr>
<td>TruthFinder</td>
<td>.1581</td>
</tr>
</tbody>
</table>

**Error Rate**

<table>
<thead>
<tr>
<th>Method</th>
<th>Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETCIBoot</td>
<td>.0132</td>
</tr>
<tr>
<td>CATD</td>
<td>.0165</td>
</tr>
<tr>
<td>CRH</td>
<td>.0264</td>
</tr>
<tr>
<td>ZenCrowd</td>
<td>.0330</td>
</tr>
<tr>
<td>AccuSim</td>
<td>.0264</td>
</tr>
<tr>
<td>3-Estimates</td>
<td>.0297</td>
</tr>
<tr>
<td>Dawid&amp;Skene</td>
<td>.0330</td>
</tr>
<tr>
<td>Voting</td>
<td>.0305</td>
</tr>
<tr>
<td>Investment</td>
<td>.0407</td>
</tr>
<tr>
<td>TruthFinder</td>
<td>.0693</td>
</tr>
</tbody>
</table>
8. REFERENCES


APPENDIX

A. PROOF OF PROPOSITION 1

For the object $n$, we have $S_n$ with $|S_n| = n$. Denote the distribution of a sample $x_i^n$, as $G_i(\cdot)$. Based on the assumption, $x_i^n \sim \text{Normal}(\hat{x}_i^n, \sigma_i^n) = G_i(x_i^n)$. As shown in [14], to prove Proposition 1 we only need to prove the following conditions:

(a) There exists a non-lattice distribution $H$ with mean 0 and variance 1, and a sequence $k_n$ with $\frac{k_n}{\log n} \to \infty$, such that $k_n$ of the population $G_i$'s are of the form $G_i(x) = H(x - \frac{\hat{x}_i^n}{\sigma_i^n})$ with $\sigma_i^n$'s bounded away from 0;

(b) $E(|X_i|^{1+\delta}) \leq M_1 < \infty$ for some $\delta > 0$;

(c) lim $\inf_n v_n^2 > 0$ and $\frac{1}{2} \sum_{i=1}^n (\mu_i - \mu_n)^2 = o(n^{-\frac{1}{2}})$;

(d) $H$ is continuous; $(\exists \delta > 0) E(|X_i|^{1+\delta}) \leq M_2 < \infty$, where $\mu_i = \hat{x}_i^n$ and $\mu = \frac{1}{n} \sum_{i=1}^n \mu_i$. Namely, $\forall i, \mu_i = \hat{x}_i^n = \mu$. Next, we prove the sufficient conditions point by point.

Proof of (a). As introduced in Section 2, $x_i^n \sim \text{Normal}(\hat{x}_i^n, \sigma_i^n)$, where $\sigma_i^n > 0$. Let $H$ be the standard normal distribution, i.e., Normal(0, 1). As any continuous distribution is non-lattice, $H$ is a non-lattice distribution. Moreover, let $k_n = n$ and $G_i = H(x_{\sigma_i^n})$. We have $\frac{k_n}{\log n} \to \infty$.

Proof of (b). For the normal distribution, we have that

$$E(|X_i|^p) = \sigma_i^p \frac{\Gamma(\frac{p+1}{2})}{\sqrt{\pi}}.$$

(17)

where $\Gamma(\cdot)$ is the gamma function, i.e., $\Gamma(n+1) = n! \Gamma(n)$. Let $\delta_0 = 1$, we have that $E(|X_i|^1) = \sigma_i^2 \frac{\Gamma(\frac{3}{2})}{\sqrt{\pi}} = 3\sigma_i^2 \hat{M}_1 < \infty$.

Proof of (c). (i) $\forall i, \sigma_i^n > 0$ and $v_n^2 = \frac{1}{2} \sum_{i=1}^n \sigma_i^n > 0$. (ii) $\forall i, \mu_i = \mu$. Proof of (d). As shown in the proof of (a), $H$ is a normal distribution which is continuous. Let $\theta = 1$ and combine with (17), yielding that $E(|X_i|^p) = \sigma_i^p \frac{\Gamma(\frac{p+1}{2})}{\sqrt{\pi}} = 105\sigma_i^p < \infty$.

As shown above, all the conditions are satisfied. Thus,

$$P(T \leq t) = \Phi(t) + \frac{\mu_n}{\sqrt{n}} \frac{(2t^2 + 1)(\phi(t) + o(n^{-1/2}))}{\sqrt{\pi}}$$

and

$$P^*(T \leq t) = \Phi(t) + \frac{K_{3,n}}{6\sqrt{n}} \frac{(2t^2 + 1)(\phi(t) + o(n^{-1/2}))}{\sqrt{\pi}}.$$

(18)

Proofs of (b) and (c) also show that $K_{3,n} - \mu_{n,3} \to 0$, yielding that $P^*(T) \leq t \approx P(T \leq t) + O_p(n^{-1/2})$. Then, Proposition 1 has been proven.

B. PROOF OF PROPOSITION 2

Note that $P(\theta(\mathbf{X}) \leq \hat{\theta}(\mathbf{X}) - \frac{E^{-1(1-\alpha)|\bar{V}_n(\hat{\theta}(\mathbf{X}))|^2}{\sqrt{|S_n|}}) = \alpha$. as $P(T \leq t) = \alpha$. So, $\hat{\theta}_{T,\mathbf{X}}(\alpha) = \hat{\theta}(\mathbf{X}) - \frac{E^{-1(1-\alpha)|\bar{V}_n(\hat{\theta}(\mathbf{X}))|^2}{\sqrt{|S_n|}}$. For the bootstrapping, $\hat{\theta}_{\hat{F},\mathbf{X}}(\alpha) = \hat{\theta}(\mathbf{X}) - \frac{E^{-1(1-\alpha)|\bar{V}_n(\hat{\theta}(\mathbf{X}))|^2}{\sqrt{|S_n|}}$. Combing the facts that $\hat{F}^{-1} = F^{-1} + O_p(n^{-1})$ and $\bar{V}_n(\hat{\theta}(\mathbf{X})) = \bar{V}_n(\hat{\theta}(\mathbf{X})) + O_p(n^{-1})$ from [6], the proof of Proposition 2 is straightforward.