REVISION OF PAPER 23

Dear Meta Reviewer and Referees:

We have substantially revised the paper following the suggestions of the Reviewers. For the convenience of Reviewers, changes to the paper are color-coded in blue.

We would like to thank Reviewers for valuable comments!

Below please find a summary of changes and our feedback to the comments.

Response to the comments of the Meta Reviewer.

[Meta Required Changes]

- MR.R1: Strengthen the motivation (Rev2.O1).

- MR.R2: Improve the write-up (Rev2.O4, Rev3.D1)

- MR.R3: Enhance the contribution (Rev4.O2)

- MR.R4. Extend the experimental evaluation (Rev1.O1, Rev2.O2/O3, Rev3.D2/D3, Rev4.O1/O3)

[A] Many thanks! We have revised the paper substantially and addressed all the issues raised.

(i) We strengthened the motivation by clarifying real-life applications of low-support but high-confidence FDs and by identifying limitations of prior work in discovering such rules (see the response to R2O1, pp. 1, 4);

(ii) We added the previously omitted attribute information in Figure 1 and in the case study (pp. 3, 12). We also included additional figures, tables, and examples to improve the readability of the technical derivations (see responses to R2O4, R3O2 and R3D1, pp. 3-4, 6, 12);

(iii) We discussed human-in-the-loop approaches to enhance the contribution, which includes involving domain experts to interpret BN structures and incorporating user-provided labels to annotate discovered FDs (see the response to R4O2 and R4O4, pp. 12);

(iv) We expanded the experimental evaluation with new datasets, baselines, and experiments (pp. 9-12). We (1) added the 70-attribute dataset Hepar2 (see the response to R1O1), (2) introduced two learning-based baselines FDX and FDM$_\epsilon$ (see responses to R2O2, R3D2 and R4O3), (3) evaluated the robustness of BSFD by injecting noise into two real-life datasets and clarified the skewness, imbalance and sparsity of adopted datasets (see responses to R2O2 and R4O1), (4) conducted ablation studies by omitting row sampling and by changing the correlated attributes extraction strategy (see responses to R2O3 and R3D3), (5) reported accuracy results of each method with support thresholds ranging from 0.05 to 0.90 (see responses to R3O3), and (6) provided additional details on the experimental setup and analysis (see response to R2O2 and R2O3).

Availability of artifacts

[R3O1 & R4 Availability] *There is no mention (I could not find the link in the paper) or link to publicly available source code, which limits reproducibility and transparency.*

[A] Thanks! We would like to clarify that the link (https://anonymous.4open.science/r/BSFD-45D7) to our source code has been provided in the detailed information page of the original submission, as pointed out by Reviewers #1 and #2. Moreover, we have added the code and datasets for the updated experiments to this repository as well.

Response to the comments of Referee #1.

[R1O1& MetaO4] *In their rationale, the authors state: "Most dependency mining algorithms suffer from prohibitive computational costs. As demonstrated by Papenbrock et al., almost all existing methods exceed acceptable runtime thresholds or consume over 100 GB of memory when applied to datasets with more than 60 attributes." However, the maximum number of attributes in the datasets involved in the experimental evaluation is 37. Conducting an ad hoc discussion or experiment on synthetic data with more than 60 attributes would improve the robustness of the provided results*

[A] Thanks! As suggested, we have added experiments on a larger synthetic dataset Hepar2 with 70 attributes, with detailed analysis on the results of efficiency and scalability (pp. 9). Specifically, BSFD is 6.71 times faster than FDX on Hepar2, while FDM$_\epsilon$ and DFD cannot execute due to the memory limit and TANE reaches the timeout threshold of three hours with no results returned. Moreover, our column-scalability study on Hepar2, with attribute number increasing from 58 to 70, verifies that BSFD continues to scale effectively beyond 60 attributes. These results show that BSFD remains robust and efficient when processing large datasets, whereas the baselines either become prohibitively slow or cannot execute (pp. 10).

Response to the comments of Referee #2.

[R2O1 & MetaO1] *A primary concern with this paper lies in its motivation. While the authors emphasize the limitations of existing methods in discovering relatively low-support and high-confidence FDs, it remains unclear why such FDs are of particular interest or importance. The paper would benefit from a clearer articulation of the practical or theoretical significance of identifying FDs in this regime. In addition, it is not sufficiently discussed whether the proposed framework is able to discover high-support FDs, as most existing methods do, and could further identify low-support, high-confidence FDs as a byproduct. Clarification on this point, along with supporting discussion, is necessary.*

It also appears that the BSFD framework is limited to discovering FDs with a single attribute on the right-hand side (RHS), as suggested by Definition 3.1 and several subsequent analyses and theorems. The scope of applicability should be clearly stated early in the paper if the proposal is indeed restricted to unary RHS FDs. Alternatively, if extension to more general cases is feasible, the authors should discuss how the framework could be adapted to handle multi-attribute RHS FDs.

Lastly, while the BSFD framework is technically well-structured, it largely builds upon and integrates existing techniques. This raises questions regarding its novelty, particularly in relation to recent and more advanced approaches. The paper would be strengthened by a deeper discussion of the technical challenges encountered in integrating these components and a more explicit comparative analysis highlighting differences and improvements over prior work.

[A] Thanks! First, we have strengthened the motivation of our work in Section I (pp. 1). As pointed out by previous studies, it is important to discover rare but meaningful patterns from data [1], known as rare association rule mining (RARM) [2], and BSFD agrees with this perspective and focuses on rare (low-support) but meaningful (high-confidence) FDs. Moreover, FDs with low support and high confidence are crucial for various real-life applications. For example, in outlier detection, rare but reliable rules help predict data anomalies [3]. In environmental science, air pollution can be inferred from rare associations between pollutants and environmental observations [4]. In public clinical service, the medical requirements of the minority groups, such as aging veterans and sensitive children, can be found in uncommon cases with low support, as demonstrated in the case study [5] (pp. 12). In order to discover only these low-support FDs via previous level-wise search methods, the support pruning threshold has to be set low [6]–[8], which significantly expands the search space and invalidates the pruning strategy [9], [10]. Thus, previous level-wise search-based methods can hardly discover low-support high-confidence FDs due to the memory limit and excessive execution time. The experiments also verify that BSFD is able to discover both low-support and high-support FDs by varying the support threshold (pp. 11). As shown in Figure 7(e), Figure 7(f), BSFD remains effective across thresholds from 0.05 to 0.20, which covers high-support rules.

Secondly, following the Armstrong's axioms [11], without loss of generality, our framework focuses on discovering FDs with a single attribute on the RHS as clarified in Section III [12] (pp. 3). Specifically, the Armstrong's axioms state that FDs with the same LHS can be joined together to form a valid FD with multiple attributes on the RHS, formally presented as: any FD $\varphi_1 : X \to A_1 A_2$ is equivalent to the pair $\varphi_2 : X \to A_1$ and $\varphi_3 : X \to A_2$ [13], [14]. Thus, researchers in the database community focus on mining FDs whose RHS has only one attribute [6], [12], [15]–[22], and FDs with multiple attributes on the RHS can be constructed by the discovered simple FDs [6], [13], [14]. As for supporting the discovery of multi-attribute RHS FDs, a trivial extension of BSFD is to split the RHS into single attributes, mine the FDs whose RHS contains the split single attribute, and join the discovered FDs whose LHS are the same, which follows the Armstrong's axioms.

Thirdly, we have clarified the novelty, challenges and main contributions of our work (pp. 2,4). To the best of our knowledge, we are the first to apply BN structure learning (probabilistic) to FD discovery (deterministic) with a theoretical performance guarantee provided, and based on which we propose an algorithm that efficiently discovers low-support-high-confidence FDs without incurring excessive memory or runtime costs. Although extensive theoretical and empirical studies have been conducted on BN structure learning, introducing the Bayesian approach to FD discovery remains non-trivial as follows. (1) BNs characterize uncertainty between attributes while FDs specify deterministic constraints, which can hardly be directly bridged together. (2) The aims of FDs discovery and BN structure learning are different, where FD discovery searches rules above support and confidence thresholds while BN structure learning finds a BN that best fits the probabilistic distribution of the data. Thus, BN structure learning can never be directly applied to FDs discovery without investigating the theoretical connection between the two. In view of these challenges, our main contribution is to prove the numerical equivalence between the aims of BN structure learning optimization and FD discovery through conditional entropy. This equivalence not only bridges probabilistic and deterministic approaches to rule discovery, but also provides the guarantee for discovering high-confidence FDs. Moreover, we reduce the search space by vertically partitioning the relation r into smaller independent sub-tables based on attribute correlations identified from the BN structure, which preserves high-confidence FDs and allows a lower support threshold without excessively enlarging the search space. Finally, given the partitioned independent sub-tables as inputs, we propose a data-parallelism-based FD discovery method that achieves significant efficiency improvements over traditional level-wise boosting strategies, with speedups of 882.57× on average (pp. 9).

[R2O2& MetaO4] *The experimental evaluation raises several concerns that merit further experiments or clarification:*

(i) The selected baselines, TANE and DFD, are relatively outdated and may not adequately represent the current state of the art. Incorporating more recent methods, particularly those cited in the related work, would provide a fairer and more informative comparison, better highlighting the contributions of BSFD.

(ii) The framework is evaluated with both TANE-based and DFD-based variants, but it is not clear which variant performs better overall. The authors should identify the superior variant and provide guidance for practitioners on which one to adopt in real-world scenarios.

(iii) The scalability experiments (Figure 6) involve inconsistent dataset usage across different analyses. A more coherent dataset selection strategy, or a clear rationale for the varied usage, is necessary to improve the credibility of the empirical results.

(iv) The robustness of BSFD to noise in real-world datasets is not systematically evaluated. Beyond the provided case study, empirical validation of noise tolerance should be added to strengthen the evaluation.

[A] Thanks!

(i) We have added FDM$_\epsilon$ [7] and FDX [16] as new state-of-the-art baselines, which incorporate covariance analysis and matrix decomposition into FDs discovery, respectively (pp. 9). Specifically, experimental results on various datasets show that BSFD on average beats FDM$_\epsilon$ and FDX by 998.22 and 21.20 times faster, respectively (pp. 9). Moreover, the rule discovery accuracy of BSFD is, on average, 1.95× and 9.00× higher than that of FDM$_\epsilon$ and FDX, respectively (pp. 10), which demonstrates the competitiveness of BSFD.

(ii) BSFD can easily integrate any existing FD discovery algorithm, not limited to TANE and DFD that are exemplary implementations in our work, and accelerates rule mining via data parallelism. Based on our experiments on nine datasets, we empirically find that the variant BSFD$_{\text{DFD}}$ generally performs better in terms of efficiency and accuracy (pp. 10). As for the detailed empirical comparison of various FD discovery algorithms, one can refer to the thorough experimental study of Papenbrock et. al [12].

(iii) We have added clarification and explanation to the scalability experiments as suggested (pp. 10). In fact, we have conducted scalability experiments on all datasets under multiple parameter settings (pp. 10-11). Due to the page limit, we have reported representative results for fair comparison where (1) a sufficient number of baseline methods are able to complete execution, and (2) the datasets have been widely adopted in previous works [7], [12], [16], [19].

(iv) We have evaluated the discovery performance of BSFD on noisy data to verify its robustness. Specifically, following [16], [23], we inject noise into two real-life datasets (Statlog and Amazon) to evaluate the noise tolerance of BSFD, with the noise rate varying from 0.01 to 0.3. We select Amazon and Statlog as test datasets since (1) All mining methods can execute and return discovered FDs on these datasets without failures of memory drain or timeout, which is suitable for fair comparison, and (2) there exist rich FDs in these datasets, e.g., 120 FDs discovered in Amazon dataset, which can be easily influenced by the injected noises. Discovery accuracy (F_1 score) is used as the evaluation metric where groundtruth FDs used for F_1 score computation are obtained using the two exact discovery methods (TANE and DFD), and results are compared with five other methods (TANE, DFD, FDM$_\epsilon$, FDX and BSFD$^{\text{G}}$) as reported in Section VIII (pp. 11-12). As the noise rate increases from 0.01 to 0.30, the F_1 scores of BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ drop at most by 0.034 and 0.097 on Amazon and Statlog, respectively. At the highest noise rate (0.30), the average F_1 scores of BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ remain 0.938 and 0.938, respectively. Across all noise levels, BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ on average rank the first with F_1 of 0.980, while BSFD$^{\text{G}}_{\text{DFD}}$ ranks the third with F_1 of 0.937, which is 0.043 below the top two.

[R2O3 & MetaO4] *The experimental evaluation would benefit from addressing the following questions:*

(i) The row sampling ratios used for each dataset had better be explicitly reported.

(ii) In the complexity analysis presented in Section VI.C, it would be helpful to include a comparative analysis of the theoretical complexity of BSFD versus that of the baseline methods. This would clarify the extent of efficiency improvement the proposal offers in theory.

(iii) While some components of the BSFD framework are evaluated, other key components are not. Given the framework's relatively complex design, a more comprehensive ablation study would better support the effectiveness of each component.

(iv) The rationale for setting the specific tuning ranges for the support and confidence thresholds in the scalability experiment should be described.

(v) In Table III, the authors highlight that BSFD outperforms its variant based on the global BN. However, it is noted that the TANE-based BSFD fails on the Studentfull dataset. Further analysis of this failure case would be useful to understand potential limitations of the proposed framework.

[A] Thanks!

(i) We have added specifications for the row sampling ratios in Table III (pp. 9) as suggested.

(ii) We have added the complexity analysis of BSFD in Section VI.C for better comparison (pp. 8). Specifically, there are four stages in BSFD, including row sampling, BN learning, correlated attribute identification, and parallel rule mining. It takes only $O(|r|)$ for hierarchical sampling as representative attribute selection requires one scan over all rows [24]. The complexity of the BN learning is $O(|R|^3|r|)$ [25], where the learning quickly converges to a local optimum within limited iterations in practice. Correlated attribute identification for each attribute costs $O(|R|)$ since in the worst case, the procedure examines at most $|R|-1$ other attributes to construct the correlated attribute set. After partitioning the original relation into multiple independent smaller relations based on the correlated attribute analysis in constant time, the size of the input to downstream discovery methods decreases (smaller $|R|$ for each partition) and the discovery of each partitioned relation can be executed in parallel on multiple processors. In contrast, the classical discovery algorithm TANE requires $O(|r|2^{|R|})$ for partition construction and $O(|R|^{2.5}2^{|R|})$ for lattice traversal [26], while DFD, though incorporating pruning and depth-first strategies, remains an exponential search space in the worst case [27].

(iii) We have added the ablation study to verify the effectiveness of each component (pp. 12). Specifically, we have (1) omitted the row sampling part and (2) changed the extraction strategy for correlated attributes to investigate the impact of these components on the BSFD framework on three challenging real-life datasets Flight, Statlog, and Amazon that, on average, contain over 68 FDs to discover. As row sampling is removed, the discovery time sharply increases by 105.04× on average (205.96× and 4.12× slower for BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$, respectively), while the discovery accuracy improves marginally, e.g., 0.005 on Amazon (pp. 12). This indicates that row sampling is essential for efficiency, with a reasonable trade-off for accuracy. Then we replace the per-node correlated attribute extraction strategy, which constructs a star-shaped BN for each attribute, with a global BN that covers all attributes (denoted as $\mathsf{BSFD}^{\mathsf{G}}$). After this substitution, the discovery accuracy decreases by 59.73% relative to BSFD and the runtime increases by 2.21×, which confirms the effectiveness of the correlated attributes extraction strategy that operates at each node (pp. 12). Moreover, we evaluate $\mathsf{BSFD}^{\mathsf{G}}$ and compare it extensively with TANE, DFD, FDM_{ϵ}, FDX, where the conclusions are consistent with those from the ablation study.

(iv) We have added explanations for the range selection of the support and confidence thresholds (pp. 10). We choose high confidence ranges since this configuration returns reliable dependencies rather than coincidental patterns [28], which ensures the evaluation focuses on meaningful and trustworthy rules in specific or less frequent cases. As for the support ranges, we choose 0.05 to 0.20 with a default of 0.10 which aligns with the motivation of BSFD: discovering low-support yet high-confidence rules within practical time and memory limits (see responses to R2O1). Evaluating scalability under low-support configuration is critical, as low support thresholds tend to invalidate classical pruning strategies [9], [10] and incur a large search space in practice. Moreover, this range suits real-life applications such as anomaly detection and clinic data analysis, where the rules hold at a relatively low frequency [2], [3], [29].

(v) We have updated the results for the TANE-based BSFD on Studentfull, where $\mathsf{BSFD}_{\mathsf{TANE}}$ achieves an F_1 of 1.000 that exceeds the score of $\mathsf{BSFD}^{\mathsf{G}}_{\mathsf{TANE}}$ (0.316), after three independent retests (pp. 10).

[R2O4 & MetaO2] *The writing and presentation of the paper could be improved in the following aspects:*

(i) In Figure 1, the attributes labeled as B, C, D, and E are not mapped to their corresponding attributes in the example dataset. Providing further information would enhance clarity and aid readers' comprehension.

(ii) The methodology section introduces a substantial number of mathematical notations, which may hinder readability. A summary table of key notations would be helpful for readers to follow the technical content more easily.

(iii) The case study would benefit from additional details. In particular, the meanings of "HOwner" and "EService" should be clarified to make the application context more transparent.

[A] Thanks!

(i) We have updated Figure 1 with full names of the attributes (pp. 3), which can easily be mapped to the exemplary dataset.

(ii) As suggested, we have added Table II in Section III (pp. 4), which summarizes the key notations for better readability.

(iii) We have expanded the abbreviations "HOwner" and "EService" to "HospitalOwner" and "EmergencyService" and added additional clarifications (pp. 12). Specifically, the two discovered FDs are rewritten as φ_1 : State $\xrightarrow{\sigma=0.1,\delta=0.9}$ HospitalType, and φ_2 : HospitalOwner, EmergencyService $\xrightarrow{\sigma=0.1,\delta=0.9}$ HospitalType (pp. 12). Their supports are 0.12 and 0.24, respectively, and both achieve confidence 0.91. Although the support is low, these dependencies are meaningful for minority groups. For example, from φ_1 and φ_2 we observe that only the states `AZ`, `AL`, or `AR` tend to have `Childrens` hospitals serving sensitive children, and that hospitals owned by `Government Federal` without emergency services are typically `Acute Care-VA Medical Centers`, which primarily provide services for aging veterans. These rules, therefore, can help aging veterans and sensitive children identify specific facilities.

Response to the comments of Referee #3.

[R3O2 & R3D1 & MetaO2] *The paper is densely written, and some sections, particularly those involving technical details, are difficult to follow without additional clarification or illustrative examples.*

[A] Thanks! As suggested, we have added an illustrative example in Section VI-A (pp. 6) to clarify the notations used in the proof of the sampling bounds, which improves the readability of the technical derivations. In addition, we have added Figure 4 to further illustrate the theoretical connection between FD discovery and BN structure learning, as established by Lemma 1, Lemma 2, Theorem 1 and Theorem 2. This figure explains the intuition behind the proofs (pp. 6).

[R3O3 & R3D3 & MetaO4] *The evaluation focuses almost exclusively on execution time, with little discussion of accuracy relative to baseline methods. A more comprehensive evaluation in terms of accuracy would improve the paper.*

[A] Thanks! We have added the accuracy evaluation of each method and reported the F_1 scores with support thresholds ranging from 0.05 to 0.9, where discovery results returned by level-wise search and depth-first search (TANE and DFD) are viewed as the groundtruth (pp. 11). Results on four real-life datasets show that BSFD consistently outperforms the other two learning-based methods, FDX and FDM_{ϵ}, achieving an average accuracy of 0.958. In addition, the high discovery accuracy of BSFD across all support thresholds confirms its effectiveness in discovering both low- and high-support rules.

[R3D2 & MetaO4] *While the related work section is thorough and lists many existing approaches, the experimental section evaluates their proposed approach against a relatively small subset of these methods. It could be interesting, if possible, to consider more baselines.*

[A] Thanks! We have added new baseline methods, including FDM_{ϵ} [7], FDX [16], as suggested. Experiments on multiple six datasets show that, on average, BSFD is 998.22× and 21.20 × faster than FDM_{ϵ} and FDX, respectively (pp. 9). Moreover, the rule discovery accuracy of BSFD is, on average, 1.95× and 9.00× higher than that of FDM_{ϵ} and FDX, respectively (pp. 10). These results verify both the efficiency and effectiveness of BSFD compared with state-of-the-art methods.

Response to the comments of Referee #4.

[R4O1 & R2O2 & MetaO4] *The paper could discuss how BSFD performs under skewed, imbalanced, or sparse distributions, which are common in real-world data lakes.*

[A] Thanks!

We have added experiments to evaluate the robustness of BSFD under noisy and imbalanced conditions using real-life datasets (see responses to R2O2). For real-life datasets, we randomly flipped varying ratios of cell values to perturb the data distribution (pp. 12). In addition, the datasets in the experiments naturally exhibit skewness, imbalance, or sparsity, as pointed by [30]–[32] (pp. 9). For example, in Hospital the attribute "State" exhibits the smallest imbalance ratio, with its majority class being $10.30\times$ the size of its minority class; in Amazon the majority class of "Status" constitutes 86% of all records. We also observe in Statlog that three attributes ("Duration", "Credit amount", and "Age") have distributions skewed toward larger values, and in Hospital that 12% of cells are missing.

[R4O2 & MetaO3] *Integrating mechanisms for interpreting or labeling discovered FDs semantically could enhance practical adoption, particularly in human-in-the-loop scenarios.*

[A] Thanks! In order to enhance practical adoption, promising future work includes (1) involving domain experts to interpret BN structures, which provides a better choice of correlated attributes, and (2) incorporating user-provided labels to annotate discovered FDs in human-in-the-loop scenarios, which produces FDs of user interest (pp. 12).

[R4O3 & MetaO4] *A brief discussion comparing BSFD with modern alternatives, such as deep learning-based dependency discovery or causal structure learning, could better position the contribution in current research trends.*

[A] Thanks! We have added new learning-based baselines for comparison, including FDM$_\epsilon$ [7] and FDX [16], which verify the efficiency and effectiveness of BSFD (pp. 9-10). Moreover, we have added causal structure learning in the related work to better position our contributions (pp. 2).

[R4O4] *A dedicated "Limitations and Future Work" section would help frame the method's applicability and clarify its assumptions and address the other opportunities for improvement written above.*

[A] Thanks! We have added discussions on limitations and future work in the Section VI and conclusion due to the page limit (pp. 8,12).

Specifically, there are two main limitations as follows: (1) attributes that are semantically correlated may be partitioned into different sub-relations by the correlated attribute sets selection, so that any FDs composed of these attributes will never be discovered and (2) the absence of a human-in-the-loop mechanism fails to improve sub-relation partitioning and interpret discovered FDs. As for future work, we plan to involve domain experts in interpreting learned BNs to guide attribute selection and to incorporate user interests in labeling discovered FDs for practical adoption.

Many thanks again for the helpful suggestions and supporting our work!

Fast Discovery of Functional Dependencies via Bayesian Network Learning

Siyi Yang[*], Shenglin Chen[*], Xi Wang, Yuhua Tang, Ruochun Jin[†]
College of Computer Science and Technology, National University of Defense Technology
Changsha, China
yangsi_@nudt.edu.cn, {13272068106, 18342211026,yhtang62}@163.com, jinrc@nudt.edu.cn

***Abstract*—Functional dependencies (FDs) are fundamental to data quality and query optimization. However, discovering high-confidence FDs from large-scale, noisy real-life datasets remains challenging, especially for those with low-support which can be early pruned. In view of this challenge, we propose BSFD, a scalable and parallel framework that leverages Bayesian network (BN) structure learning to guide the discovery of meaningful FDs with low support and high confidence (FD$_{\sigma,\delta}$s). We establish a numerical equivalence between FDs and parent-child relationships in BNs, which lays the statistical foundation of our approach. We have also proposed a stratified sampling strategy with a theoretical bound on structure correctness relative to the sampling ratio, which enables efficient BN learning with structural accuracy preserved. As for large datasets, BSFD vertically partitions the input relation into multiple smaller sub-tables using BN-derived correlated attribute sets, which significantly reduces the search space. Experiments on real-life and synthetic datasets demonstrate that BSFD achieves on average 490× and up to 7008× speedup over baseline methods while maintaining high discovery accuracy, with an average F_1 score of 0.98.**

***Index Terms*—Rule discovery, Bayesian approach, Search space reduction**

I. Introduction

Functional dependencies (FDs) [33] describe the relevance of attributes in relational databases, which are crucial for data quality management [34]–[38], query optimization [35], [36], data integration [37] and data cleaning [38]. A functional dependency $X \rightarrow A$ implies that the value of attribute A is uniquely determined by the value of the attribute set X in a relational schema R, where X and A are also referred as the left-hand side (LHS) and the right-hand side (RHS), respectively [39]. Typically, FD discovery aims to find all minimal and non-trivial functional dependencies from a given dataset [12].

During the discovery of FDs, the number of potential candidates increases exponentially with the number of attributes, reaching a total of $\sum_{k=1}^{|R|} \binom{|R|}{k} \cdot (|R| - k)$, where $|R|$ denotes the number of attributes [13], [19]. This exponential complexity makes exhaustive enumeration impractical for large datasets [19]. Although numerous efficient FD discovery algorithms have been proposed to mitigate this issue, most of them still face the following critical challenges when scaling to large datasets:

TABLE I: Example relation of Adult information

tid	Age	Discode	Education	Edu-num	Workclass	Relationship	Occupation
t_1	39	29874	HS-grad	*10*	Local-gov	Not-in-family	Craft-repair
t_2	74	29887	Masters	14	Self-emp-not-inc	Not-in-family	Exec-managerial
t_3	51	30008	HS-grad	9	Private	Husband	Transport-moving
t_4	26	33417	Masters	14	Self-emp-not-inc	Other-relative	Sales
t_5	67	212759	Bachelors	13	Local-gov	Husband	Other-service
t_6	28	338409	Bachelors	13	Private	Husband	Prof-specialty

(1) Most dependency mining algorithms suffer from their prohibitive computational costs. As demonstrated by Papenbrock et al. [12], nearly all existing methods either exceed acceptable runtime thresholds or consume over 100GB of memory when applied to datasets with more than 60 attributes. Although various pruning techniques have been proposed to reduce the search space by limiting attribute combinations, these methods often fail to achieve sufficient efficiency in practice [12], [13], which highlights the need for more scalable and resource-efficient solutions.

(2) While the support-based pruning method is widely used to reduce the search space, it often overlooks confidence (not anti-monotonic [10]), a critical metric for assessing FD quality [6], [7], [40], [41]. Support-based pruning methods usually limit discovery to high support FDs and risk missing valuable dependencies with relatively low support but high confidence. For example, as shown in Table I, all tuples with `Education` = `Masters` (i.e., t_1 and t_4) share the same `Workclass` = `Self-emp-not-inc`, making the FD {`Education`} → {`Workclass`} hold with confidence close to 1. However, since this pattern appears in only two tuples (relatively low support), such FD may possibly be neglected by support-based pruning approaches. These FDs are critical in domains where rare cases carry greater importance than common patterns [2], such as clinical service for minority groups [5], [29], pollution prediction [4], and outlier detection [3]. Although conditional functional dependencies (CFDs) [14] are able to capture such patterns, discovering CFDs is substantially more complex and incurs a much higher computational cost [42], [43].

(3) Real-life datasets often contain considerable noise, which further complicates the discovery [16], [40], [44]. Existing statistical methods typically address this issue by analyzing binary matrices that encode tuple pair differences [7], [16]. However, such binary representations exhibit high noise sensitivity, as minor data errors distort bit patterns and restrict models to linear statistical relationships, which cannot capture complex dependencies [16], [44]. Moreover, these methods

[*] These authors contributed equally to this work.
[†] Corresponding author.

incur substantial memory overhead due to tuple-wise comparisons. These limitations underscore the need for more robust and efficient techniques for accurate FD discovery in noisy data [45], [46].

Previous efforts on FD discovery raise the following key research questions: First, can we design a scalable method for discovering FDs on large datasets? Second, if we prune the search space using support thresholds, how can we ensure that high-confidence rules are retained? Finally, can we discover FDs efficiently and robustly from noisy data?

Contributions & organization. To address these challenges, we present a parallel framework BSFD, a **B**ayesian network **S**tructure learning guided **F**unctional dependency **D**iscovery method, which leverages Bayesian Network (BN) structure learning to efficiently discover FDs in large noisy datasets, while ensuring both high reliability and scalability. We firstly redefine support and confidence through a probabilistic interpretation of FDs, which establishes the foundation for linking support and confidence bounded FD discovery with BN structure learning. We then prove the numerical equivalence between the aim of BN structure learning optimization and FDs discovery, which demonstrates that BN structure learning inherently captures high-confidence dependencies between attributes and their predecessors (corresponding to RHS and LHS). Based on this numerical equivalence, we extract candidate LHS attribute sets for each RHS from BN structures learned over datasets to form a discovery sub-space, which significantly reduces the search space and enables the discovery of relatively low-support and high-confidence dependencies. Finally, we design a parallel strategy to accelerate the discovery process while preserving high discovery accuracy. To the best of our knowledge, we are the first to apply BN structure learning to FD discovery and provide a rigorous theoretical analysis of its effectiveness. Our main contributions are summarized as follows:

(1) Bayesian Perspective Analysis of FD *discovery (Section V).* We formally establish the numerical equivalence between the aim of high-confidence FD discovery and BN structure learning through conditional entropy. Furthermore, by defining a more fine-grained, per-value confidence measure, we show that conditional entropy $H(A|X) \to 0$ implies the confidence of the FD $\varphi : X \to A$ approaches 1. This result provides a theoretical foundation for leveraging BN structure optimization to guide the discovery of high-confidence FDs.

(2) Search Space Reduction (Section VI). To enhance computational efficiency, we propose an accuracy-bounded sampling method for efficient BN structure learning and vertically partition the relation r into smaller, independent sub-tables based on attribute correlations identified through statistical analysis. This BN-based partition of relation leads to a significant reduction in computational overhead.

(3) Data-parallelism-based FD *Discovery (Section VII).* Given the guarantees provided by our BN-based approach, we employ a relatively low support threshold for pruning the search space and discovering rules from sub-tables in parallel. This approach effectively reduces the computational complexity while preserving high-confidence FDs, thus facilitating the discovery of valuable rules with relatively low support but exceptionally high confidence. Experimental results show that BSFD achieves an average speedup of 672 × and an average F_1 score of 0.96.

II. Related Work

We categorize the related work as follows.

FD **discovery.** Such methods can be classified as follows. (1) Level-wise search. TANE [26], FUN [47], FD_Mine [48], and DepMiner [49] traverse the power set lattice using a level-wise approach and prune the search space based on the Apriori-Gen [50]. (2) Depth-first search. DFD [27] and FastFDs [51] extend a LHS in a depth-first manner until an FD holds. In our approach, we also consider all possible LHS column sets for each RHS in a single pass when generating candidate FDs. (3) Hybrid approaches. HyFD [18] couples column-efficient and row-efficient phases. (4) Approximate FDs discovery. This kind of methods allow few violations [19], [52], while CORDS [36] considers only single-column LHS and top-k variants [20], [45], [53] miss many rules. (5) Learning-based FD discovery. FDX [16] adopts graph lasso, and FDM$_\epsilon$ [7] executes covariance analysis, where both encode difference matrices and incur heavy cost. Causal structure learning [54] finds deterministic dependencies implied by FDs under noise-free data, but fails with noise.

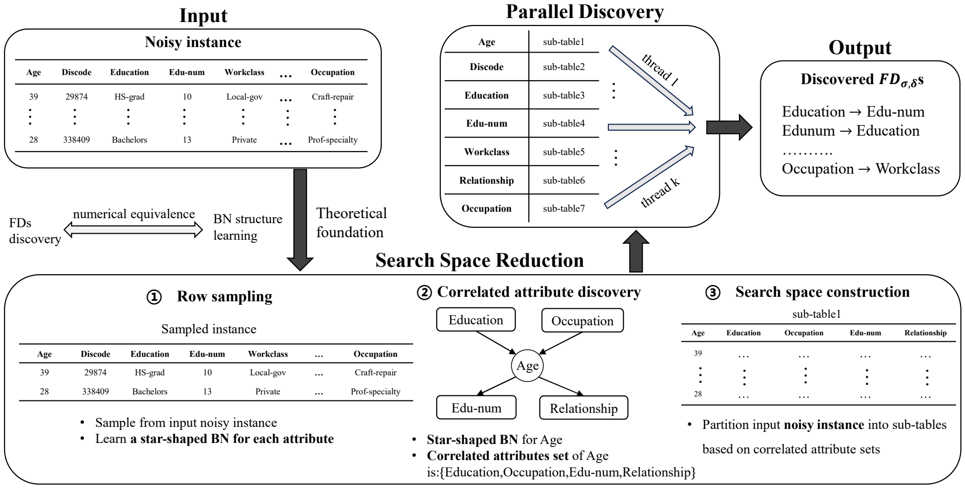
This work differs from prior work. (a) We reduce the search space while preserving high-confidence dependencies through probabilistic analysis. (b) We aim to discover FDs with low support and high confidence by leveraging conditional entropy as a principled measure of approximation. (c) We adopt BN structure learning to guide rule discovery.

Bayesian-based rule discovery. Prior work adopt methods in a Bayesian framework for more general rule mining [55]. Wang et al. [56] study rules in disjunctive normal form, place priors on length and complexity, and rank candidates by posteriors; they search the powerset yet prune by support and information gain. In healthcare, Bayes-based criteria drop uninformative associations and improve relevance [57].

Different from the above works, (a) we employ BN structure learning for rule discovery, (b) we focus on FD, a standardized and widely adopted form of rules in real-life applications.

Parallel rule discovery. Saxena et al. [58] decompose sequential miners into six primitives, which raises transformation cost and decreases scalability. A single-node, shared-everything design concentrates load on one machine [59]. Horizontal partitioning finds approximate FDs that hold only locally [60]. Vertical partitioning reduces search by restricting the LHS to one attribute, which limits expressiveness [61].

This work differs from the above studies as follows. (1) We adopt vertical partitioning based on BN-derived correlated attribute sets. (2) We apply classical sequential FD discovery algorithms in parallel across independent sub-tables.



Age	Discode	Education	Edu-num	Workclass	…	Occupation
39	29874	HS-grad	10	Local-gov	…	Craft-repair
⋮	⋮	⋮	⋮	⋮	⋮	⋮
28	338409	Bachelors	13	Private	…	Prof-specialty

Age	sub-table1
Discode	sub-table2
Education	sub-table3
Edu-num	sub-table4
Workclass	sub-table5
Relationship	sub-table6
Occupation	sub-table7

Age	Discode	Education	Edu-num	Workclass	…	Occupation
39	29874	HS-grad	10	Local-gov	…	Craft-repair
28	338409	Bachelors	13	Private	…	Prof-specialty

- Sample from input noisy instance
- Learn **a star-shaped BN for each attribute**

- **Star-shaped BN** for Age
- **Correlated attributes set** of Age is:{Education,Occupation,Edu-num,Relationship}

Age	Education	Occupation	Edu-num	Relationship
39	…	…	…	…
⋮	⋮	⋮	⋮	⋮
28	…	…	…	…

- Partition input **noisy instance** into sub-tables based on correlated attribute sets

Fig. 1: An overview of the dependency discovery algorithm BSFD.

III. PRELIMINARIES

In this section, we begin by reviewing the basic concepts of FDs and BN structure learning, and then formally define relatively low-support and high-confidence dependencies (FD$_{\sigma,\delta}$s). The notations of the paper are summarized in Table II.

A. Functional Dependencies

Definition 3.1: *Given a relational schema R and an instance r over R, a functional dependency φ is a statement $X \to A$, where $X \subseteq R$ and $A \in R$. φ holds on r if for all tuples $t_i, t_j \in r$, $t_i[X] = t_j[X]$ implies $t_i[A] = t_j[A]$. Here, X is the left-hand side (*LHS*) and A is the right-hand side (*RHS*).*

Without loss of generality [11], we focus on FDs with a single attribute on the RHS [6], [12], [15]–[22].

Approximate Functional Dependency. An approximate FD holds for most of the tuples, allowing for a few violations. A functional projection function $f : \mathrm{dom}(X) \to \mathrm{dom}(A)$ maps each value $x \in \mathrm{dom}(X)$ to a specific value $f(x)$ for attribute A [16]. The approximate FD $\varphi : X \to A$ holds if $p(A = f(x) \mid X = x) \geq 1-\epsilon$ where ϵ is a small constant representing the error tolerance [45], [54]. Note that φ becomes an exact FD as defined in Defintion 3.1 when $\epsilon = 0$.

To discover rules from noisy relations, we define the support and confidence measures for FDs following [6], [7].

Definition 3.2: *Given a relation r and an* FD *$\varphi : X \to A$, the support of φ is defined as: supp$(X \to A, r) = \left(\sum_{x \in \mathrm{dom}(X)} |\{t \in r \mid t[X] = x \wedge t[A] = f(x)\}|\right) / (|r|)$*

Definition 3.3: *Given a relation r and an* FD *$\varphi : X \to A$, the confidence of φ is defined as: conf$(X \to A, r) = \left(\sum_{x \in \mathrm{dom}(X)} |\{t \in r \mid t[X] = x \wedge t[A] = f(x)\}|\right) / \left(\sum_{x \in \mathrm{dom}(X)} |\{t \in r \mid t[X] = x\}|\right)$*

Example 1: Consider the Adult relation in Table I, where X is Education and A is Edu-num. Define the correct mappings of the projection f as $f(\texttt{HS-grad}) = 9$, $f(\texttt{Masters}) = 14$, $f(\texttt{Bachelors}) = 13$. We compute the support of the approximate FD $\varphi : \texttt{Education} \to \texttt{Edu-num}$ as $\mathrm{supp}(\varphi) = \frac{5}{6} \approx 0.83$, with the only violation occurring at t_1. The confidence of φ is computed as: $\mathrm{conf}(\varphi) = \frac{1+2+2}{2+2+2} = \frac{5}{6} \approx 0.83$, with t_1 being the only tuple where $t[\texttt{Edu-num}] \neq f(t[\texttt{Education}])$.

Definition 3.4: *Given a schema R and an instance r, an* FD$_{\sigma,\delta}$ *φ on R is defined as: $X \xrightarrow{\sigma,\delta} A$, where σ and δ are support and confidence thresholds, respectively. An* FD$_{\sigma,\delta}$ *holds on R if and only if supp$(\varphi, r) \geq \sigma$ and conf$(\varphi, r) \geq \delta$.*

Order of FD$_{\sigma,\delta}$s. We use $\preceq$ to define the order between two FD$_{\sigma,\delta}$s $\varphi : X \xrightarrow{\sigma,\delta} A$ and $\varphi' : X' \xrightarrow{\sigma,\delta} A$: if $X \subset X'$, then φ has lower order than φ' (denoted as $\varphi \preceq \varphi'$). Intuitively, φ' is more restrictive than φ, that is, φ is a generalization of φ', and φ' is a specialization of φ.

Minimality of FD$_{\sigma,\delta}$s. An FD$_{\sigma,\delta}$s φ: $X \xrightarrow{\sigma,\delta} A$ on R is *trivial* if $A \in X$. In this paper, we only consider non-trivial FD$_{\sigma,\delta}$s. An FD$_{\sigma,\delta}$s φ: $X \xrightarrow{\sigma,\delta} A$ is *left-reduced* on r iff $\forall Y \subset X$, $r \nvDash \varphi(Y \xrightarrow{\sigma,\delta} A)$. Intuitively, this means no attribute in X can be removed, i.e., the minimality of FD$_{\sigma,\delta}$s. A minimal FD$_{\sigma,\delta}$ φ on r is a non-trivial and left-reduced FD$_{\sigma,\delta}$s such that $r \vDash \varphi$. We say that a set Σ of dependencies φ is minimal if $\forall \varphi \in \Sigma$, Σ is not equivalent to $\Sigma \setminus \{\varphi\}$, i.e., there are no redundant dependencies in Σ. To discover all FD$_{\sigma,\delta}$s in relation r, it suffices to discover a set Σ of minimal, non-trivial FD$_{\sigma,\delta}$s.

B. Bayesian Network

A Bayesian network (BN) encodes the conditional independence structure of a probability distribution P using a directed acyclic graph (DAG) G, where each node represents a random variable X_i, and edges represent dependencies among variables [25], [62]. Formally, for each node X_i in G, let Pa_{X_i} denote its set of parent nodes, and let NonDes_{X_i} denote the set of non-descendants of X_i. The conditional independence semantics of a BN are defined as: $X_i \perp \mathrm{NonDes}X_i \mid \mathrm{Pa}_{X_i}, \forall X_i \in G$. Consequently, the joint distribution P factorizes as: $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid \mathrm{Pa}_{X_i})$.

TABLE II: Summary of main notations

symbols	notations
R, r, r_{sub}	relational schema, relation instance,a set of sub-tables
f	the projection from LHS to RHS
σ/δ	support and confidence thresholds, respectively
t,$t[A]$	tuples on r, projection of t onto the attribute A
$H(A)$,$H(A \mid X)$	entropy of A, conditional entropy of A given X
Pa_A	the set of parent nodes of A
Σ,Σ_s	the set of FD$_{\sigma,\delta}$s discovered from r and r_{sub},respectively

As a key component of our approach, we next provide a brief overview of BN structure learning:

Structure learning algorithms. Structure learning algorithms for BNs are generally classified into two categories: constraint-based methods and score-based methods [25], [63]. Given a relation r and its empirical distribution $\hat{P}$, these two approaches operate as follows:

(1) Constraint-based learning. Constraint-based methods start with a fully connected undirected graph and iteratively remove edges using conditional independence (CI) tests. A CI test evaluates if two variables are conditionally independent given a set of other variables.

(2) Score-based Learning. Score-based methods formulate structure learning as an optimization task that searches for the optimal DAG maximizing a scoring function. Common strategies include search strategy Hill-Climbing (HC), with popular score Bayesian Information Criterion (BIC) [25], which is defined as: $\text{score}_{BIC}(G) = |r| \sum_{i=1}^{n} I_{\hat{P}}(X_i; \text{Pa}_{X_i}) - |r| \sum_{i=1}^{n} H_{\hat{P}}(X_i) - \frac{\log |r|}{2} \cdot \text{Dim}[G]$, where $I_{\hat{P}}(X_i; \text{Pa}_{X_i})$ is the empirical mutual information between variable X_i and its parent set, $H_{\hat{P}}(X_i)$ is the empirical entropy, and $\text{Dim}[G]$ is the number of free parameters in the DAG.

C. Problem Statement

Given a noisy relation r of schema R with an underlying probability distribution P, support threshold σ, confidence threshold δ, the FD$_{\sigma,\delta}$ discovery aims to discover a set of minimal FD$_{\sigma,\delta}$s Σ that holds on r, where each minimal FD$_{\sigma,\delta}$ has a support of at least σ and a confidence at least δ. Specifically, an FD$_{\sigma,\delta}$ $\varphi : X \xrightarrow{\sigma,\delta} A$ is said to hold on r if: $\text{H}(A \mid X) = \epsilon \wedge \text{supp}(\varphi, r) \geq \sigma \wedge \text{conf}(\varphi, r) \geq \delta$, where ϵ is a small constant representing noise tolerance.

IV. SOLUTION OVERVIEW

We introduce algorithm BSFD for discovering FD$_{\sigma,\delta}$s.

Challenges. Although BN structure learning has been studied extensively, adopting a Bayesian approach to FD discovery is non-trivial.(1) BNs characterize uncertainty among attributes while FDs specify deterministic constraints, which can hardly be directly bridged together. (2) The aims of FDs discovery and BN structure learning are different, where FDs discovery searches rules above thresholds while BN structure learning fits the data distribution. Thus, BN structure learning can never be directly applied to FDs discovery.

In view of these, we present FD$_{\sigma,\delta}$s discovery algorithm BSFD. As shown in Fig. 1, BSFD takes a noisy relation r as input and outputs a set of discovered FD$_{\sigma,\delta}$s with support and confidence bound. The workflow of BSFD is as follows:

Bayesian Perspective Analysis of FDs. We establish a statistical foundation for utilizing BN in FD$_{\sigma,\delta}$s discovery through Bayesian perspective analysis. Specifically, we prove that BN structure learning and FDs discovery share the same optimization objective when considering a single attribute as the child node in BN edges and as the RHS in FDs. Given an attribute set X and a target attribute A, both tasks aim to form an edge $X \to A$ or declare an FD $\varphi : X \to A$ valid if the conditional entropy $H(A \mid X) \to 0$. We show that φ holds with high confidence when $H(A \mid X) \to 0$. This establishes the numerical equivalence between conditional entropy minimization and high-confidence FDs discovery.

Search Space Reduction. Based on the above analysis, we use BNs to guide FD$_{\sigma,\delta}$s discovery. BSFD first samples r to obtain a representative subset with theoretical correctness bounds. For each attribute, it constructs a star-shaped BN to identify strongly correlated attributes with low conditional entropy. Finally, the original relation r is vertically partitioned into independent sub-tables according to these sets, significantly reducing the search space for subsequent discovery.

Parallel Dependencies Discovery. Finally, to accelerate discovery, BSFD integrates existing sequential FDs discovery algorithms to process each sub-table in parallel. The discovered FD$_{\sigma,\delta}$s from all sub-tables are then merged and deduplicated. The output is a collection of minimal FD$_{\sigma,\delta}$s that satisfy the user-defined support threshold σ and confidence threshold δ.

V. BAYESIAN NETWORK PERSPECTIVE ON FD DISCOVERY

This section establishes the theoretical connection between BN structure learning and the discovery of FD$_{\sigma,\delta}$s with a single RHS attribute. We further show that minimizing conditional entropy in BN learning naturally leads to high-confidence FDs, providing a unified optimization objective.

A. From BN Structure Learning to FD Discovery

Before adopting BNs for FD$_{\sigma,\delta}$s discovery, we first explore the theoretical connection between BN structure learning and the identification of FDs.

Lemma 1: *Let G be the DAG learned via BN structure learning over a set of attributes R. Then, for a given attribute $A \in R$, an edge $X \to A$ is preferred if X minimizes the conditional entropy of A, i.e., $X = \arg\min_{Pa_A \subset R} H(A \mid Pa_A)$.*

Proof sketch: The proof examines two classical BN structure learning paradigms: constraint-based and score-based. The complete proof is in full version [64] due to the page limit.

(1) Constraint-based methods. These approaches determine the presence of an edge $X \to A$ through CI tests, typically using statistical measures such as χ^2 or mutual information (MI) to assess dependency strength [25]. A larger χ^2 value indicates stronger dependence, i.e., greater evidence against $X \perp A$. The MI is defined as $I(X; A) = H(A) - H(A \mid X)$, where $H(A)$ is the entropy of A. Since $H(A)$ is fixed for a

given distribution, maximizing $I(X;A)$ is equivalent to minimizing $H(A \mid X)$. Empirically, the two measures are related approximately by $\chi^2 \approx 2 \cdot I(X;A)$ [25], both increasing as $H(A \mid X)$ decreases. Thus, constraint-based methods tend to favor edges $X \rightarrow A$ that exhibit low conditional entropy, thereby indicating strong statistical dependence.

(2) Score-based methods. These score-based methods evaluate candidate graph structures utilizing scoring functions such as the BIC, which can be reformulated as [25]: $\text{score}_{BIC}(G) = -|r| \sum_{i=1}^{n} H_{\hat{P}}(X_i \mid \text{Pa}_{X_i}) - \frac{\log |r|}{2} \text{Dim}[G] = -|r| \sum_{i=1}^{n} H_{\hat{P}}(X_i \mid \text{Pa}_{X_i}) + C$, where the constant C consolidates terms independent of the parent sets. Thus, maximizing the BIC score minimizes the total conditional entropy across all nodes, and score-based methods favor edges that reduce conditional entropy. □

Building on the probabilistic interpretation of FDs in Section III-A, we have the following lemma:

Lemma 2: *Given a relation r and an approximate* FD *$X \rightarrow A$, the conditional entropy $H(A \mid X)$ under distribution P over r quantifies the strength of the dependency. In particular, $H(A \mid X) = 0$ iff $X \rightarrow A$ holds exactly under P [65].*

The above Lemma implies that the lower the value of $H(A \mid X)$, the closer the FD $X \rightarrow A$ is to holding on r under distribution P. Therefore, the numerical equivalence is presented as follows:

Theorem 1: *The aim of* FD *discovery with a single attribute on the* RHS *is numerically equivalent to BN structure learning.*

Proof: By Lemma 1, both constraint-based and score-based BN structure learning methods seek to build directed edges from X to A such that $H(A \mid X)$ is minimized, indicating that X provides maximal information about A. By Lemma 2, the degree to which an approximate FD $X \rightarrow A$ holds in a relation r under distribution P is measured by the conditional entropy $H(A \mid X)$. A smaller $H(A \mid X)$ implies a stronger dependency. Therefore, when the RHS of each FD contains a single attribute, both FD discovery and BN structure learning share the objective of minimizing the conditional entropy $H(A \mid X)$. This establishes a numerical equivalence between the two tasks under the given condition. □

Theorem 1 establishes the numerical equivalence between FD discovery and BN structure learning based on conditional entropy. We subsequently demonstrate that conditional entropy serves as a valid measure for representing approximate FDs, particularly when the LHS involves multiple attributes.

Conditional entropy for multi-attributes in LHS. It has been shown in [65] that Armstrong's Axioms remain sound under a conditional entropy-based definition. For instance, consider a candidate LHS $X = \{X_1, X_2, X_3\}$ and RHS A. If $H(A \mid X_1) < \epsilon$ for a small $\epsilon > 0$, then $X_1 \rightarrow A$ holds approximately. Since conditional entropy satisfies $H(A \mid X_1) \geq H(A \mid X_1, X_2)$ [66], it follows that $H(A \mid X_1, X_2) < \epsilon$, implying $X_1, X_2 \rightarrow A$ is also valid. This confirms that conditional entropy is a consistent measure of FDs even with multi-attribute LHS.

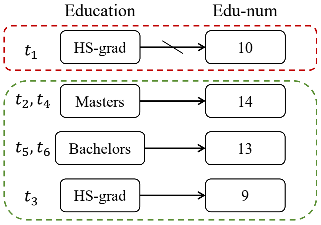


Fig. 2: Partition of the Adult relation

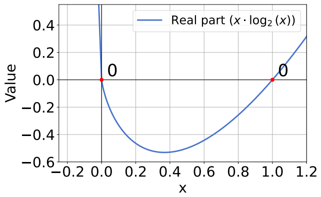


Fig. 3: Plot of $x \log_2 x$

B. High-Confidence FDs *Analysis*

Though based on the analysis in the prior section, we can treat child-parent combinations in a BN topology as FDs if the BN is ideal. However, it has been proven that finding the maximum-score BN structure is NP-hard [25], [67], making it impractical to deduce FDs directly from BN edges. Therefore, instead of aiming to directly extract FDs from a BN, we seek to prove the condition $H(A \mid X) \rightarrow 0$ guarantees high confidence in the FD $X \rightarrow A$.

Inspired by the per-value likelihood measure in [17], we define a per-value confidence as follows:

Definition 5.1: *For a* FD *$\varphi : X \rightarrow A$, with an arbitrary value $x \in \text{dom}(X)$, the per-value confidence of φ is defined as* pconf$(X \rightarrow A, x) = \frac{|x, f(x)|}{|x|}$.

Example 2: Consider the relation r in Table I, let X be `Education` and A be `Edu-num`. Based on the projection of values in Example 1, the pconf values for each $x \in \text{dom}(\texttt{Edu-num})$ are as follows:

1)pconf$(\texttt{Education} \rightarrow \texttt{Edu-num}, \texttt{HS-grad}) = \frac{|\texttt{HS-grad,9}|}{|\texttt{HS-grad}|} = \frac{1}{2}$,
2)pconf$(\texttt{Education} \rightarrow \texttt{Edu-num}, \texttt{Bachelors}) = \frac{|\texttt{Bachelors,13}|}{|\texttt{Bachelors}|} = \frac{2}{2}$,
3)pconf$(\texttt{Education} \rightarrow \texttt{Edu-num}, \texttt{Masters}) = \frac{|\texttt{Masters,14}|}{|\texttt{Masters}|} = \frac{2}{2}$.

Lemma 3: *For a* FD *$\varphi : X \rightarrow A$, for $\forall x \in \text{dom}(X)$, if* pconf$(X \rightarrow A, x) \geq C$, *then conf$(X \rightarrow A) \geq C$.*

Proof: For each x, let $\alpha_x = |x, f(x)|$, and $\beta_x = |x|$. Then, By the given condition, we have pconf$(X \rightarrow A, x) = \frac{\alpha_x}{\beta_x} \geq C$. Multiplying both sides of it by β_x (which is positive), we obtain: $\alpha_x \geq C \cdot \beta_x$. Summing over all $x \in \text{dom}(X)$ yields: $\sum_x \alpha_x \geq \sum_x C \cdot \beta_x = C \sum_x \beta_x$. Dividing both sides by $\sum_x \beta_x$ (which is positive because it is the total number of records where X is defined), we get: $\frac{\sum_x \alpha_x}{\sum_x \beta_x} \geq C$. By the definition given earlier, the left-hand side is exactly conf$(X \rightarrow A) = \frac{\sum_x \alpha_x}{\sum_x \beta_x}$. Therefore, we conclude: conf$(X \rightarrow A) \geq C$. □

Theorem 2: *Given a relation schema R and an instance r, $H(A \mid X) \rightarrow 0$ implies a high-confidence* FD *$\varphi : X \rightarrow A$ for attributes X and a single attribute A in R.*

Proof sketch: It is evident that pconf$(X \rightarrow A, x) = p(A = f(x) \mid X = x)$. For tuples in r that satisfy φ, $p(A = f(x) \mid X = x) \rightarrow 1$, and $p(A \neq f(x) \mid X = x) \rightarrow 0$. Since $x \log_2 x \rightarrow 0$ as $x \rightarrow 1$ or $x \rightarrow 0$, the proof has two parts.

(1) If $X \rightarrow A$ is an FD with high confidence, then $H(A \mid X) \rightarrow 0$. Partition tuples into those satisfying

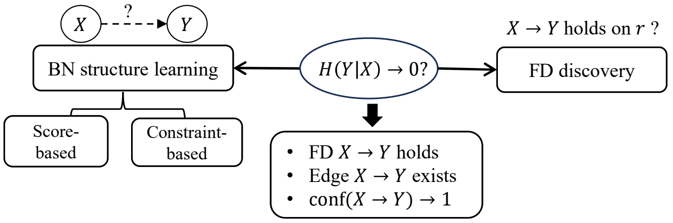


Fig. 4: Bayesian perspective analysis.

$t[A] = f(t[X])$ and those violating $t[A] \neq f(t[X])$. The conditional entropy $H(A \mid X)$ is then computed as in [65]: $H(A \mid X) = -\sum_{x\in X} p(x)\Big(\sum_{a=f(x)} p(a \mid x)\log p(a \mid x) + \sum_{a\neq f(x)} p(a \mid x)\log p(a \mid x)\Big)$. As shown in Fig. 3, $p(a \mid x)\log_2 p(a \mid x) \to 0$ as $p(a \mid x) \to 0$ or $p(a \mid x) \to 1$ hold. Therefore, With each $p(x) > 0$, $H(A \mid X) \to 0$ when φ has high pconf. By Lemma 3, we conclude that $H(A \mid X) \to 0$ for a high-confidence FD.

(2) $H(A \mid X) \to 0$ implies that the FD $X \to A$ has high confidence. Both $\sum_{a=f(x)} p(a \mid x)\log p(a \mid x) \leq 0$ and $\sum_{a\neq f(x)} p(a \mid x)\log p(a \mid x) \leq 0$ as $p(a \mid x) \in [0,1]$. When $H(A \mid X) \to 0$, we have each of these terms must also approach zero, which implies that $p(A = f(x) \mid X = x) \to 1$ and $p(A \neq f(x) \mid X = x) \to 0$. According to Lemma 3, if pconf$(X \to A, x) \to 1$ for every $x \in \text{dom}(X)$, then conf$(X \to A)$ also approaches 1. Therefore, $H(A \mid X) \to 0$ implies that conf$(X \to A) \to 1$. □

Based on Theorem 2, we can use $H(A \mid X) \to 0$ to validate whether an FD $X \to A$ holds on R and guide the attribute set discovery in a smaller search space. Fig. 4 illustrates the intuition that underlies our theoretical analysis.

Example 3: Continuing with values in Example 2, as shown in Fig. 2, tuples are split into those where $t[\texttt{Edu-num}] = f(\texttt{Education})$ (green dashed box) and those where $t[\texttt{Edu-num}] \neq f(\texttt{Education})$ (red dashed box). Then, we have $H(\texttt{Edu-num} \mid \texttt{Education}) = -\Big(\frac{1}{3}\big(\frac{1}{2}\log\frac{1}{2} + \frac{1}{2}\log\frac{1}{2}\big) + \frac{1}{3}\big(1\cdot\log 1\big) + \frac{1}{3}\big(1\cdot\log 1\big)\Big) = \frac{1}{3} \approx 0.333$. The last two terms vanish because the tuples t_2, t_4, t_5, t_6 satisfy the dependency, which significantly reduces the uncertainty of $\texttt{Edu-num}$ given $\texttt{Education}$ and thus leads to high confidence. Meanwhile, conf$(\texttt{Education} \to \texttt{Edu-num}) = 5/6 \approx 0.83$, which is relatively high.

VI. Search Space Reduction

Based on the above analysis, we employ BN structure learning to guide the discovery of high-confidence FDs by extracting correlated attribute sets that reduce the search space and capture high-confidence attribute dependency. In this section, we introduce a sampling strategy, a correlated attributes extraction method, and finally explain how these sets are used to construct a compact and practical search space.

A. Row Sampling

To ensure scalable BN structure learning, we introduce a sampling strategy to obtain a representative subset from the original data while preserving essential attribute relationships.

Definition 6.1: *Given a relational schema R, the representative attribute r_p is the one that induces the most distinct partitions, i.e., $r_p = \arg\max_{A\in R} |\pi_A|$. where π_A is the partitioning by attribute A.*

We adopt stratified sampling guided by the representative attribute r_p and formally establish a bound on the distributional deviation between the sampled subset and the original relation r, by extending the finite population analysis in [68]. Since similar distributions yield similar BN structures [25], the structure learned from the sample remains reliable.

Assumption. Let $P(r) = \frac{1}{|r|}\sum_{t\in r}\mathbb{I}[t[R] = r^*]$ be the true estimate over relation r, and $P(S) = \frac{1}{|S|}\sum_{t\in S}\mathbb{I}[t[R] = r^*]$ its empirical estimate on sample $S \subset r$ with $|S| = k|r|$, where r^* is the correct value and k is the sample ratio. Assume attribute A partitions r into $m = |\text{dom}(A)|$ strata. For each $a \in \text{dom}(A)$, let $r_a = \{t \in r \mid t[A] = a\}$ and $S_a \subset r_a$ with $|S_a| = k|r_a|$. Then the within-stratum estimates are $P(r_a) = \frac{1}{|r_a|}\sum_{t\in r_a}\mathbb{I}[t[R] = r^*]$ and $\widehat{P}(S_a) = \frac{1}{|S_a|}\sum_{t\in S_a}\mathbb{I}[t[R] = r^*]$. Finally, the overall estimates are given by $P(r) = \sum_{a=1}^{m}\frac{|r_a|}{|r|}P(r_a)$ and $\widehat{P}(S) = \sum_{a=1}^{m}\frac{|S_a|}{|S|}P(S_a)$ [69].

Example 4: Continue with Table I, $\texttt{Occupation}$ is chosen as r_p as has the most distinct values. Then the relation is partitioned into $m = |dom(\texttt{Occupation})| = 6$ strata by $\texttt{Occupation}$. Although each stratum has only one tuple in this toy example, in practical datasets, a stratum typically contains a large collection of tuples. For each value $a \in dom(\texttt{Occupation})$, let r_a be the set of tuples with $\texttt{Occupation} = a$ (e.g. $r_{\texttt{Craft-repair}}$ contains all tuples whose $\texttt{Occupation}$ is $\texttt{Craft-repair}$). Given a sampling ratio k, we sample $k\,|r_a|$ tuples from each stratum r_a (rounding as needed) to obtain its sample (e.g., $S_{\texttt{Craft-repair}}$). The combination of all per-stratum samples is the final sample S used for BN structure learning.

Theorem 3: *Given a sampling ratio $k \in (0,1]$, an error constant $\epsilon \in (0,1)$ and a confidence level $\alpha \in (0,1)$. Suppose k satisfy:*

$$k \geq \begin{cases} \frac{C(1+1/r_{min})}{r_{min}+C}, & C \leq \frac{r_{min}^2}{r_{min}+2}, \\ \frac{C\left(r_{min}-1\right) + \sqrt{C\left(Cr_{min}^2+2Cr_{min}+C+4r_{min}^2\right)}}{2r_{min}\left(C+r_{min}\right)}, & C > \frac{r_{min}^2}{r_{min}+2}, \end{cases}$$

where the constant $C = \frac{ln(1/\alpha)}{2\epsilon^2}$, $r_{min} = \min_a |r_a|$. Then, the deviation of probability distribution over r and S : $|P(\widehat{S}) - P(r)| < \epsilon$ holds with probability at least $1-\alpha$.

Proof sketch: Given $\widehat{P}_S = \sum_{a=1}^{m}\frac{|S_a|}{|S|}\widehat{P}(S_a)$ and $P(r) = \sum_{a=1}^{m}\frac{|r_a|}{|r|}P(r_a)$ and triangle inequality, we get $\left|\widehat{P}(S) - P(r)\right| \leq \sum_{a=1}^{m}\frac{|r_a|}{|r|}\left|\widehat{P}(S_a) - P(r_a)\right|$. Let $r_{\min} = \min_a r_a$ and $N_{min} = kr_{\min}$, $\beta = m\alpha$. Applying the union bound over all m strata [69], the tight bound is given by [68]: $\Pr\left[\left|\widehat{P}(S) - P(r)\right| \leq \sqrt{\frac{\rho_{r_{min}} \ln(m/\beta)}{2N_{\min}}}\right] \geq 1-\beta$, with $\rho_{r_{\min}} = 1 - \frac{N_{\min}-1}{r_{\min}}$ if $N_{\min} \leq \frac{r_{\min}}{2}$, and $\rho_{r_{\min}} = \left(1 - \frac{N_{\min}}{r_{\min}}\right)\left(1 + \frac{1}{N_{\min}}\right)$ if $N_{\min} > \frac{r_{\min}}{2}$. To satisfy ϵ, i.e., $\Pr\left[\left|\widehat{P}(S) - P(r)\right| \leq \epsilon\right] \geq 1 - \beta$. Define a constant $C = \frac{ln(1/\alpha)}{2\epsilon^2}$, recall that $N_{min} = kr_{min}$, following [68], we have $\epsilon^2 \geq \begin{cases} \frac{\ln(1/\alpha)}{2kr_{\min}}(1-k+1/r_{min}), & k \leq 1/2, \\ \frac{\ln(1/\alpha)}{2kr_{\min}}(1-k)(1+\frac{1}{kr_{min}}), & k > 1/2, \end{cases}$ then, for each case, we obtain the lower bound of k: when $C \leq \frac{r_{\min}^2}{r_{\min}+2}$, $k \geq \frac{C(1+1/r_{min})}{r_{min}+C}$, otherwise, $k \geq \frac{C\left(r_{min}-1\right) + \sqrt{C\left(Cr_{min}^2+2Cr_{min}+C+4r_{min}^2\right)}}{2r_{min}\left(C+r_{min}\right)}$. □

Example 5: Consider sampling from a relation r with $N = 10\,000$ tuples, partitioned into $m = 6$ strata of (nearly) equal size. Let $\epsilon = 0.01$ and $\alpha = 0.1$, resulting in $r_{\min} = \lfloor N/m \rfloor = 1\,667$. To guarantee $\Pr\left[|P(\widehat{S}) - P(r)| \leq \varepsilon\right] \geq 0.9$, we compute the constant $C = \frac{\ln 10}{2\times 0.01^2} \approx 1.1513 \times 10^4$. Since $C > \frac{1\,667^2}{1\,669} \approx 1.666 \times 10^3$, we apply the *second* bound in Theorem 3, giving $k \approx 0.874$. Thus, the required sample size is $n = kN \approx 0.874 \times 10\,000 = 8\,736$. Therefore, drawing 8736 tuples (i.e., 87.4% of the relation) suffices to bound the distribution error within ±0.01 at 90% confidence.

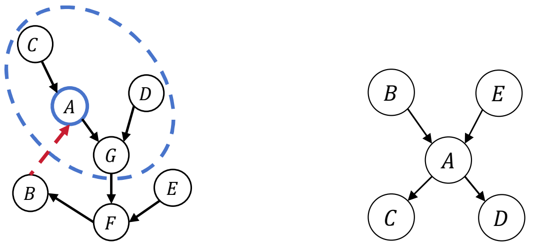
B. Correlated Attributes Discovery

Our method directly obtains correlated attribute sets for each RHS from the topology of a BN learned from the given instance r, unlike previous work that derive such sets via complex statistical correlation analyses. Similar to the search space construction strategy in DFD [27], we build a correlated attribute set for each RHS as a sub-space, where the attributes in the set serve as candidates for constructing the LHS.

Correlated attributes extraction for each RHS is guided by Markov blanket [70], as the Markov blanket of RHS is the minimal set of variables that contains all information needed to infer its values. In a BN, this set consists of the parents of RHS, its children, and the co-parents of its children.

Specifically, for each attribute A, we learn a star-shaped BN on r with A as the center, and every other attribute can either point to A or be pointed by A as illustrated in Fig. 5b. During the BN structure learning, an edge between A and an attribute X_i is established if the conditional entropy $H(A \mid X_i)$ is sufficiently small. The extraction process is as follows:

- **Constructing a star-shaped BN:** For each attribute A, we define: $blacklist(A) = \{(X_i, X_j) \in R \times R \mid X_i \neq A \wedge X_j \neq A\}$, which excludes all edges between non-A attributes during learning [71]. This constraint ensures that only edges directly connected to A are preserved, resulting in a star-shaped BN centered on A, denoted as $\mathcal{B}_{star}(A)$.
- **Extracting the correlated attribute set:** We extract the Markov blanket of attribute A from its corresponding star-shaped BN $\mathcal{B}_{star}(A)$, denoted as corr$(A) = MB(A)$, which includes all attributes adjacent to A regardless of edge direction (parents or children). We do not fix LHS/RHS roles at this stage: if either $H(A \mid X)$ or $H(X \mid A)$ is small, then X is informative for A, and the direction is determined during FD discovery. Therefore, the set $corr(A)$ forms the complete candidate attribute set for constructing the LHS of potential FDs with A as the RHS.



(a) Global BN (b) Star-shaped BN

Fig. 5: Comparison of (a) global BN of attributes and (b) star-shaped BN of attributes

We adopt star-shaped BNs rather than a single global BN, as the global DAG's acyclicity can prune edges between truly correlated attributes and thus yield incomplete correlated sets. As shown in Fig. 5a(a), when $H(A \mid B)$ is small enough (Lemma 1), the edge $B \rightarrow A$ (red dashed) is disallowed to avoid a cycle, which excludes B from A's Markov blanket (blue dashed). Our experiments confirm that star-shaped BNs yield more complete coverage of correlated attributes.

C. Search Space Construction

The correlated attribute sets discovery method enables the design of the SubLearner algorithm, which vertically partitions the relation r into multiple sub-tables. This decomposition significantly reduces the search space.

Algorithm. As shown in Algorithm 1, given schema R and relation r, and attribute-level constraints $blacklist(\cdot)$ of each attribute, SubLearner constructs sub-spaces for each RHS attribute using the correlated attributes extraction method introduced in Section VI-B. It begins by defining a BN structure learning strategy HC and a set of scoring functions S, initializing the BN learner $B(HC, S)$ (Lines 1–2). We adopt three commonly used scoring functions, AIC, BIC, and K2, since they yield similar optimization directions [25]. Next, two empty sets R_{sub} and r_{sub} are prepared for the resulting attribute subsets and their corresponding sub-tables (Lines 3–4). To construct a star-shaped BN for each attribute, SubLearner first computes constraints $blacklist(X_i)$ to focus BN learning exclusively on edges involving X_i. Using this constraint, it re-initializes the learner $B(HC, S, blacklist(X_i))$ and learn a star-shaped BN $\mathcal{B}_{star}(X_i)$ from r (Lines 5–8). Then, it extracts the Markov blanket of X_i from $\mathcal{B}_{star}(X_i)$ as correlated attribute set $corr(X_i)$ (Line 9). To enable the FD discovery on the sub-table for X_i, we append the X_i itself to $corr(X_i)$ so that X_i could be identified as LHS to form a valid dependency (Line 10). To make the discovery practical and efficient, it retains only the attribute subsets $corr(X_i)$ that contain at least two attributes, which is the minimum number required to form a valid dependency (Lines 11–12). Finally, for each

```
Algorithm 1: Algorithm SubLearner
Input  : Schema $R$, relation $r$, and constraint function
         $blacklist(\cdot)$ defined in Section VI-B.
Output: A set of attribute subsets $R_{sub}$ and their
         corresponding sub-relations $r_{sub}$
1  Scoring function set $S \leftarrow \{AIC, BIC, K2\}$;
2  BN structure learner $B \leftarrow B(HC, S)$;
3  $R_{sub} \leftarrow \emptyset$;
4  $r_{sub} \leftarrow \emptyset$;
5  foreach attribute $X_i \in R$ do
6      Construct constraint $blacklist(X_i)$;
7      Re-initialize BN structure learner
         $B(HC, S, blacklist(X_i))$;
8      Learn local BN $\mathcal{B}_{star}(X_i)$ centered on $X_i$ using
         $B(HC, S, blacklist(X_i))$;
9      Extract correlated attributes $corr(X_i)$ from $\mathcal{B}_{star}(X_i)$;
10     Add $X_i$ to $corr(X_i)$;
11     if $|corr(X_i)| \geq 2$ then
12         Add $corr(X_i)$ to $R_{sub}$;
13 foreach attribute subset $R_i \in R_{sub}$ do
14     Project relation $r$ onto schema $R_i$ to form sub-table $r_i$;
15     Add $r_i$ to $r_{sub}$;
16 return $R_{sub}, r_{sub}$
```

resulting attribute subset R_i, SubLearner projects the original relation r onto R_i to form sub-relation r_i and adds it to the set r_{sub} (Lines 13–15). The algorithm returns R_{sub} and r_{sub}. A limitation of this partitioning strategy is that attributes that are semantically correlated may be partitioned into different sub-relations by our correlated attribute sets selection, so that any FDs composed of these attributes will never be discovered.

Example 6: Consider to obtain *corr*(Education) from the Adult relation in Table I, we learn a star-shaped BN centered on Education by applying *blacklist*(Education) to initialize the structure learner. The resulting BN contains edges Education→ Edu-num, Education→ Workclass, and Relationship→ Education, yielding the correlated attributes as {Edu-num, Workclass, Relationship, Education}. As it contains more than two attributes, the set is added to R_{sub}, and Adult is partitioned accordingly.

Complexity. The worst-case complexity of Algorithm 1 is dominated by the BN structure learning step and subsequent relation projection operations. Assuming the complexity of the BN learning for each attribute is T_{BN}, the total complexity can be expressed as $O(|R| \cdot T_{BN}) + O(|R|^2|r|)$. In practice, the typical complexity of HC-based BN learning algorithms is polynomial, e.g., $O(|R|^3|r|)$ [25], [63]. Thus, under standard polynomial-time assumptions, Algorithm 1 has an overall complexity of $O(|R|^4|r|)$. In contrast, TANE costs $O(|r|2^{|R|})$ for partition construction and $O(|R|^{2.5}2^{|R|})$ for lattice traversal [26], while DFD, though incorporating pruning and depth-first strategies, remains exponential search space [27].

VII. Parallel Discovery

To fully leverage the reduced search spaces derived from correlated attribute sets, we propose a scalable parallel rule discovery algorithm PFMiner, designed to efficiently mine

```
Algorithm 2: Algorithm PFMiner
Input  : Thresholds $\sigma$, $\delta$, a sequential FD discovery
         algorithm $Alg_{seq}$, a set of sub-tables $r_{sub}$
Output: A minimal FD$_{\sigma,\delta}$s cover $\Sigma_s$
1  The number of threads $l \leftarrow |r_{sub}|$;
2  Create $l$ threads across available processors;
3  Set the thresholds of $Alg_{seq}$ as $\sigma$ and $\delta$;
4  $\Sigma_s \leftarrow \emptyset$;
5  foreach thread $t_i$ for $i \in [1, l]$ in parallel do
6      Assign sub-table $r_i$ to thread $t_i$;
7      Use $Alg_{seq}$ to discover a minimal FD$_{\sigma,\delta}$s cover $\Sigma_{s_i}$
         from $r_i$;
8  Merge results from all threads: $\Sigma_s \leftarrow \bigcup_{i=1}^{l} \Sigma_{s_i}$; return $\Sigma_s$
```

FD$_{\sigma,\delta}$s on large-scale datasets. The algorithm distributes the FD$_{\sigma,\delta}$s discovery process across the disjoint sub-tables produced by SubLearner, enabling each sub-table to be processed independently and concurrently.

Parallel scalability. We first revisit the widely adopted notion of parallel scalability [72]. Given an input instance r, a minimal cover Σ_s of discovered FD$_{\sigma,\delta}$s, and thresholds σ and δ for support and confidence, respectively. We denote the worst running time of the sequential FD discovery algorithm $\mathcal{A}$ as $t(|r|, |\Sigma_s|, \sigma, \delta)$. A parallel algorithm $\mathcal{A}_p$ is parallelly scalable relative to $\mathcal{A}$ if its running time by using n CPU processors can be expressed as:

$$T(|r|, |\Sigma_s|, \sigma, \delta) = \widetilde{O}(\frac{t(|r|, |\Sigma_s|, \sigma, \delta)}{n}),$$

where the notation $\widetilde{O}()$ hides $log(n)$ factors.

Algorithm. As detailed in Algorithm 2, PFMiner integrates sequential FD discovery algorithms and executes them in parallel. The inputs include: (i) a relatively low support threshold σ and a high confidence threshold δ, leveraging the fact that the sub-tables tend to preserve high-confidence dependencies, which permits a lower support threshold without compromising discovery quality; (ii) a sequential FD discovery algorithm (e.g., TANE or DFD), and (iii) the set of sub-tables r_{sub} generated by Algorithm 1. PFMiner initializes $l = |r_{\text{sub}}|$ threads across available processors, with each thread responsible for processing one sub-table (Lines 1–2). The sequential algorithm Alg_{seq} is configured with thresholds σ and δ for FD$_{\sigma,\delta}$s discovery, and an empty set Σ_s is prepared for results(Line 3-4). Each thread processes a sub-table $r_i \in r_{sub}$ (Lines 5-6), applying Alg_{seq} to compute a minimal cover of FD$_{\sigma,\delta}$s, denoted Σ_{s_i} (Line 7). After all threads complete, a post-processing step merges $\{\Sigma_{s_i}\}$ into the final result Σ_s to eliminate duplicates from overlapping sub-tables (Line 8).

Theorem 4: PFMiner *is parallelly scalable relative to the sequential algorithms* TANE *and* DFD*.*

Proof: For PFMiner, the worst-case time complexity of TANE and DFD is $t(|r|, |\Sigma_s|, \sigma, \delta)$. In PFMiner, the time complexity of workload assignment is $O(|r_{sub}|)$. The cost at each processor is dominated by the following: (a) fetch its corresponding data in time at most $O(|r|)$; (b) transmit the mined rules to the coordinator in at most $O(|r|)$; (c) balance its

TABLE III: Dataset statistic

Name	Type	#tuples	#attributes	#FDs	%sample
Adult	real-life	32561	15	13	15
Hospital	real-life	640000	17	2	10
Studentfull	real-life	1043	26	13	100
Flight	real-life	500000	20	24	10
Statlog	real-life	1000	21	17	50
Amazon	real-life	19379	24	120	25
Child	synthetic	100000	20	unknown	10
Alarm	synthetic	10000	37	unknown	10
Hepar2	synthetic	1000	70	unknown	100

workload, at most $O(|r|)$ data is sent to idle workers; and (d) locally conduct discovery in $\frac{t(|r|,|\Sigma_s|,\sigma,\delta)}{n}$, since the workload is evenly distributed by (c). Taken together, PFMiner takes at most $\frac{t(|r|,|\Sigma_s|,\sigma,\delta)}{n}$ time, and is thus parallelly scalable relative to sequential algorithms. □

Remark. We analyze the PFMiner algorithm from three perspectives: (1) *Minimality.* Each thread applies a standard sequential FD discovery algorithm (e.g., TANE), which guarantees minimality on its sub-table. After merging and deduplication, the final FD cover Σ_s remains minimal; (2) *Accuracy.* Each sub-table r_i is constructed based on the star-shaped BN of attribute A_i. The correlated attributes in r_i are selected to minimize $H(A_i \mid X)$, which ensure high-confidence LHS candidates; (3) *Efficiency.* By reducing the attribute set size in each sub-table relative to the original schema R, the search space is substantially reduced. Combined with parallel processing, this leads to significant runtime improvements.

VIII. EXPERIMENTAL STUDY

We experimentally evaluated (1) the efficiency of BSFD, (2) the scalability of BSFD, (3) the effectiveness and robustness of BSFD, (4) the effectiveness of $H(Y \mid X)$, (5) the effectiveness of the BN-based search space reduction, (6) ablation study for key components, and (7) case study on a real-life noisy dataset.

A. Experimental setting

Datasets. We employed six real-life and three synthetic datasets, as summarized in Table III. Most of these datasets have been used in prior work on FD discovery. The real-world datasets include: (1) Adult[1], which predicts individual income levels based on census features; (2) Hospital [5], [30], from the U.S. Department of Health & Human Services, containing patient records across multiple hospitals; (3) Studentfull [32], [73], which describes student performance in secondary education from two Portuguese schools; (4) Flight [30], containing flight departure and arrival information; (5) Statlog [31], [74], used for credit risk classification based on personal attributes; and (6) Amazon[2], providing profitability insights from online retail data. All rows with null values were removed before experimentation. To evaluate scalability, we generated additional synthetic datasets with a larger number of attributes using the method from [16], with the following configurations: (i) 20 columns and 100,000 rows, (ii) 37 columns and 10,000 rows, and (iii) 70 columns and 1000 rows.

Algorithms. We implement the following in C++: (1) TANE [26], a classical level-wise algorithm for FD discovery; (2) DFD [27], a depth-first search–based FDs discovery method; (3) FDX [16], a statistical approach that transforms FD discovery into structural learning via linear structural equation models, with code released[3]. (4) FDM$_\epsilon$ [7], a search-based method boosted by clustering-based correlated attributes extraction and covariance-guided pruning; (5) BSFD, our proposed FD discovery framework consisting of SubLearner and PFMiner. We adopt either DFD or TANE as the sequential discovery engine, yielding two variants: BSFD$_{\mathsf{DFD}}$ and BSFD$_{\mathsf{TANE}}$; (6) BSFD$^{\mathsf{G}}$, a variant of BSFD that extracts correlated attributes from a global BN instead of a star-shaped BN. Its variants are denoted as BSFD$^{\mathsf{G}}_{\mathsf{DFD}}$ and BSFD$^{\mathsf{G}}_{\mathsf{TANE}}$.

Accuracy Metrics. Let Σ and Σ_{s} denote the sets of minimal FD$_{\sigma,\delta}$s discovered by the same discovery algorithm on r and r_{sub}, respectively. We adopt the following metrics to evaluate the effectiveness: (1) Precision measures the fraction of discovered FD$_{\sigma,\delta}$s in Σ_{s} over the total number of FD$_{\sigma,\delta}$s in Σ; (2) Recall measures the fraction of FD$_{\sigma,\delta}$s in Σ that can be discovered from r_{sub} (i.e., appear in Σ_{s}); and (3) $F1$ score is defined as $2 \times \frac{Precision \times Recall}{Precision + Recall}$.

Default setting. All experiments were conducted on a single-node cluster with 192 CPU cores and 386GB of memory. Each experiment was repeated three times, and the average result is reported. We excluded the time spent on data loading and partition writing, but included the time for learning BNs for each RHS attribute. Unless otherwise specified, the support and confidence thresholds are set to 0.1 and 0.9, respectively, which focus on the discovery of low-support, high-confidence FDs. Groundtruth rules in Table III are obtained by applying TANE/DFD with support and confidence above the given thresholds, and domain experts further verify these rules before using them as groundtruth. Each experiment was subject to a maximum runtime of three hours, and crosses in figures indicate runs that exceeded this limit.

B. Experimental results

Exp-1: Time Efficiency. Fig. 6 shows the running time of all algorithms on real-world and synthetic datasets. Overall, BSFD is the fastest, where the largest gains appear for BSFD$_{\mathsf{TANE}}$ over TANE: on Hospital, BSFD$_{\mathsf{TANE}}$ achieves 2741.52× speedup, while TANE fails to complete on Studentfull and Flight due to memory and time limitations, respectively. BSFD$_{\mathsf{TANE}}$ shows substantial acceleration over TANE, achieving an average 882.57× speedup. On the high-dimensional Hepar2 dataset, BSFD runs 6.71× faster than FDX, while FDM$_\epsilon$ and DFD fail to complete due to memory limits and TANE times out after three hours without returning results. On average, BSFD outperforms FDX and FDM$_\epsilon$ by 21.20× and 998.22×, respectively, with a maximum of 7007.61×. Finally, BSFD also outperforms the BSFD$^{\mathsf{G}}$ variants in most cases, except on Adult where BSFD$_{\mathsf{TANE}}$ is slower than BSFD$^{\mathsf{G}}_{\mathsf{TANE}}$ by one second.

[1] https://archive.ics.uci.edu/datasets

[2] https://www.kaggle.com/

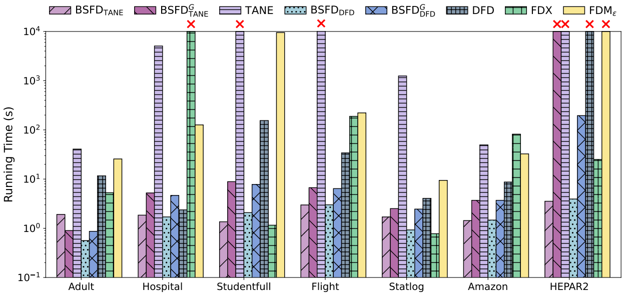


Fig. 6: Efficiency evaluation on real-life and synthetic datasets.

Analysis. BSFD and BSFD$^{\text{G}}$ benefit from the reduced search space determined by correlated attribute sets, achieving notable speedups over the baselines. By contrast, TANE and DFD face power-set explosion, and FDX and FDM$_{\epsilon}$ incur high cost due to binary difference matrices. However, BSFD$^{\text{G}}$ shows lower efficiency likely due to overfitting in the global BN that produces larger but less targeted attribute sets, while our star-shaped BN yields more compact sets which support faster discovery with lower computational overhead.

Exp-2: Scalability. We next studied the scalability of BSFD by varying (1) the number of rows $|r|$, (2) the number of columns $|R|$, (3) the support threshold σ, (4) the confidence threshold δ, and (5) the number n of CPU cores. The ranges of σ and δ are set to target low-support high-confidence FDs. We evaluate all datasets under multiple settings and report representative results to ensure that (i) a sufficient number of baselines complete for fair comparison, and (ii) the datasets are widely used in previous works [7], [12], [16], [19].

Varying $|r|$. We evaluate row scalability by varying the sampling ratio from 0.25 to 1.00 on Hospital (Fig. 7(a)) and synthetic Child datasets (Fig. 7(b)). As expected, the runtime of all methods generally increases with the sampling ratio, where BSFD$_{\text{TANE}}$ is up to 135.52×, 5603.73×, and 1088.44× faster than FDM$_{\epsilon}$, TANE, and FDX on Hospital, respectively, and BSFD$_{\text{DFD}}$ is up to 305.61× and 25.83× faster than FDM$_{\epsilon}$ and DFD on Child, and 1517.09× faster than FDX on Hospital. Although the runtime of BSFD is close to that of BSFD$^{\text{G}}$, BSFD shows more consistent efficiency across sampling ratios.

Varying $|R|$. As for column scalability, we tested all methods on Hepar2 and Flight, as shown in Fig. 7(c) and 7(d). BSFD consistently outperforms the baselines, and its runtime grows more slowly as $|R|$ increases, indicating strong scalability. Even on Hepar2 with large $|R|$ (58 to 70), both BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ take at most 3.94s, while TANE, DFD, FDX, and BSFD$^{\text{G}}_{\text{TANE}}$ fail due to memory limits. On Flight, BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ achieve 205.8× and 5.40× speedups over TANE and DFD, respectively. They are also faster than FDX and FDM$_{\epsilon}$ at smaller $|R|$, showing consistent scalability.

Varying σ. We evaluate the effect of the support threshold σ by varying it from 0.05 to 0.20 on Studentfull (Fig. 7(e)) and Alarm (Fig. 7(f)). Runtime generally decreases as σ increases, as higher thresholds prune more candidate dependencies. On Alarm, all methods except BSFD fail due to memory limits or, for FDX, a non-convergent solution caused by a violated sparsity assumption. Among executable methods, BSFD$_{\text{DFD}}$ achieves the largest speed-up, exceeding FDM$_{\epsilon}$ by 4066.10×, DFD by 75.32× and BSFD$^{\text{G}}_{\text{DFD}}$ by 3.26×.

TABLE IV: Evaluation on Real-life Datasets

Dataset	**Metric**	BSFD$_{\text{TANE}}$	BSFD$_{\text{DFD}}$	BSFD$^{\text{G}}_{\text{TANE}}$	BSFD$^{\text{G}}_{\text{DFD}}$	FDX	FDM$_{\epsilon}$
Adult	Precision	1.000	1.000	1.000	1.000	0.000	1.000
	Recall	0.846	0.846	0.615	0.615	0.000	0.154
	F_1	**0.917**	**0.917**	0.762	0.762	0.000	0.267
Hospital	Precision	1.000	1.000	1.000	1.000	–	0.000
	Recall	1.000	1.000	1.000	1.000	–	0.000
	F_1	**1.000**	**1.000**	**1.000**	**1.000**	–	0.000
Studentfull	Precision	1.000	1.000	0.500	0.600	0.000	0.000
	Recall	1.000	1.000	0.231	0.231	0.000	0.000
	F_1	**1.000**	**1.000**	0.316	0.333	0.000	0.000
Flight	Precision	1.000	1.000	1.000	1.000	0.000	1.000
	Recall	1.000	1.000	0.417	0.417	0.000	0.667
	F_1	**1.000**	**1.000**	0.588	0.588	0.000	0.800
Statlog	Precision	1.000	1.000	1.000	1.000	1.000	0.000
	Recall	1.000	1.000	0.294	0.294	0.059	0.000
	F_1	**1.000**	**1.000**	0.455	0.455	0.111	0.000
Amazon	Precision	1.000	1.000	1.000	1.000	0.000	1.000
	Recall	0.883	0.883	0.075	0.075	0.000	0.708
	F_1	**0.938**	**0.938**	0.140	0.140	0.000	0.829

'–' indicates that the method exceeded runtime or memory limit.

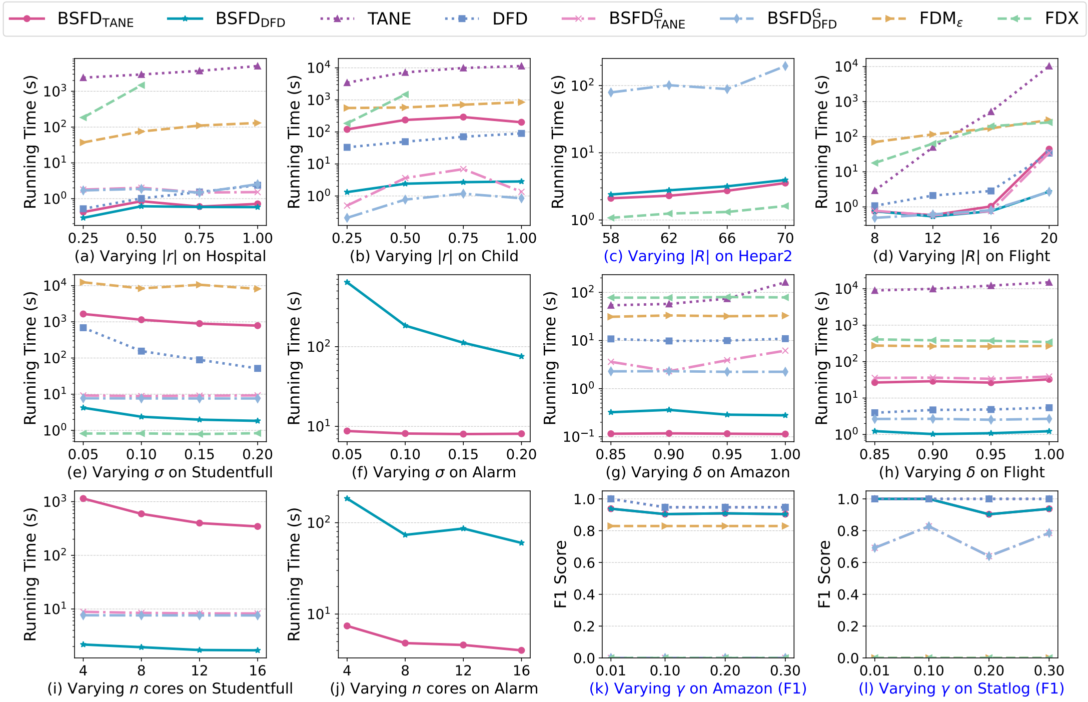
Varying δ. We evaluate the impact of the confidence threshold δ by varying it from 0.85 to 1.00 on Amazon and Flight, as shown in Fig. 7(g) and 7(h). Overall, methods exhibit stable runtime trends across different confidence, indicating low sensitivity to δ. For example, BSFD$_{\text{DFD}}$ becomes only 1.16× faster as δ increases from 0.85 to 1.00. Across all settings, BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ are consistently faster than TANE, DFD, BSFD$^{\text{G}}_{\text{TANE}}$ and BSFD$^{\text{G}}_{\text{DFD}}$.

Varying n. As shown in Fig. 7, BSFD demonstrates strong scalability with increasing number of machines. Specifically, BSFD$_{\text{TANE}}$ (resp. BSFD$_{\text{DFD}}$) achieve 3.31× (resp. 3.06×) speedups on Studentfull (resp. Alarm) as n increases from 4 to 16. In contrast, both variants of BSFD$^{\text{G}}$ exhibit minimal performance gains and fail to scale on larger datasets such as Alarm, even with additional CPU resources. These results confirm the scalability of our parallel strategy.

Analysis. Both BSFD and BSFD$^{\text{G}}$ exhibit superior scalability compared to baseline methods, with the following observations. (1) The runtime of BSFD and BSFD$^{\text{G}}$ does not vary monotonically with $|r|$ and $|R|$, as the number and size of sub-tables depend on the learned BN structure rather than the parameters. (2) Typically, as parameters vary, the runtime trend of BSFD aligns with that of TANE and DFD. In contrast, BSFD$^{\text{G}}$ occasionally exhibits the opposite trend as $|r|$ and δ change. (3) FDX on Hepar2 and BSFD$^{\text{G}}$ on Studentfull appear faster than BSFD. However, the apparent speedup results from early termination, since both algorithms find only a negligible number of FDs that meet the thresholds, causing the search to stop early. (4) BSFD$_{\text{DFD}}$ runs faster than BSFD$_{\text{TANE}}$ while attaining the same discovery accuracy.

Exp-3: Effectiveness of BSFD. We evaluate BSFD in terms of (1) the discovery accuracy on real-life data and (2) robustness to noise. Specifically, we compare BSFD, BSFD$^{\text{G}}$, FDM$_{\epsilon}$ and

[3] https://github.com/sis-ethz/Profiler-Public

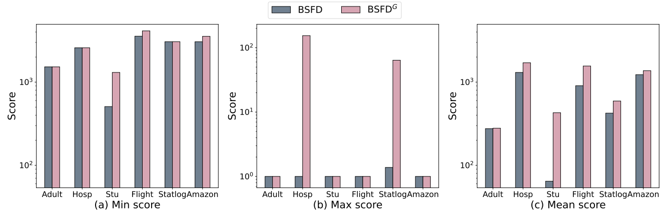


(a) Varying |r| on Hospital (b) Varying |r| on Child (c) Varying |R| on Hepar2 (d) Varying |R| on Flight

(e) Varying σ on Studentfull (f) Varying σ on Alarm (g) Varying δ on Amazon (h) Varying δ on Flight

(i) Varying n cores on Studentfull (j) Varying n cores on Alarm (k) Varying γ on Amazon (F1) (l) Varying γ on Statlog (F1)

Fig. 7: Scalability evaluation



(a) Min score (b) Max score (c) Mean score

Fig. 8: Contiditonal entropy evaluation

FDX on Precision, Recall, and F_1 using datasets with moderate attribute counts, since TANE and DFD fail when $|R| > 30$.

Accuracy. As shown in Table IV, BSFD achieves average F_1 of 0.976 on six real-life datasets, which exceeds all baselines. Specifically, BSFD matches TANE and DFD on Hospital, Flight, and Statlog with F_1=1.000. On average, the F_1 of BSFD is 208.26% and 4273.87% higher than that of FDM$_\epsilon$ and FDX, respectively, which demonstrates the competitiveness of BSFD compared with other probability-based methods. Among the variants, BSFD$_{\text{TANE}}$ exceeds BSFD$^{\text{G}}_{\text{TANE}}$ by 79.55%, and BSFD$_{\text{DFD}}$ exceeds BSFD$^{\text{G}}_{\text{DFD}}$ by 78.62%.

TABLE V: Effectiveness of BN-based space reduction

Dataset	#n	Reduc.	F_1
Adult	10	98.50%	0.917
Hospital	5	98.32%	1.000
Studentfull	19	93.01%	1.000
Flight	7	97.99%	1.000
Statlog	13	98.66%	1.000
Amazon	14	99.95%	0.938

TABLE VI: Discovery accuracy with varying support thresholds.

Dataset	Method	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	Avg
Adult	FDX	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	–	–	0.000
	FDM$_\epsilon$	0.250	0.000	0.000	0.000	0.000	0.000	0.000	0.000	–	–	0.031
	BSFD$_{\text{TANE}}$	0.833	0.917	0.875	0.923	0.909	0.889	0.800	0.667	–	–	**0.852**
	BSFD$_{\text{DFD}}$	0.833	0.917	0.875	0.923	0.909	0.889	0.800	0.667	–	–	**0.852**
Flight	FDX	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	FDM$_\epsilon$	0.769	0.800	0.800	0.800	0.800	0.800	0.800	0.800	0.800	0.800	0.797
	BSFD$_{\text{TANE}}$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	**1.000**
	BSFD$_{\text{DFD}}$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	**1.000**
Statlog	FDX	0.105	0.111	0.118	0.154	0.222	0.333	0.500	1.000	–	–	0.318
	FDM$_\epsilon$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	–	–	0.000
	BSFD$_{\text{TANE}}$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	–	–	**1.000**
	BSFD$_{\text{DFD}}$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	–	–	**1.000**
Amazon	FDX	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	FDM$_\epsilon$	0.829	0.829	0.830	0.830	0.830	0.830	0.830	0.830	0.824	0.824	0.829
	BSFD$_{\text{TANE}}$	0.964	0.938	0.977	0.985	0.985	0.985	0.985	0.985	0.989	0.989	**0.978**
	BSFD$_{\text{DFD}}$	0.964	0.938	0.977	0.985	0.985	0.985	0.985	0.985	0.989	0.989	**0.978**

'–' indicates no groundtruth.

Analysis. The comparison yields two key insights: (1) The high F_1 scores of BSFD demonstrate that SubLearner effectively selects correlated candidate LHS attributes, preserving key relationships for accurate FD$_{\sigma,\delta}$ discovery. In addition, the perfect precision across all datasets confirms that our partitioning strategy maintains the correctness of the sequential algorithms. (2) BSFD outperforms BSFD$^{\text{G}}$ due to its star-shaped BNs, which focus on attributes directly related to each RHS, resulting in higher-quality candidates. While BSFD$^{\text{G}}$ relies on a global BN that introduces irrelevant attributes, leading to lower candidate quality and reduced accuracy. (3) FDX and FDM$_\epsilon$ fail on most datasets as they provide no guarantee on support or confidence.

Impact of σ. Table VI illustrates the discovery accuracy (F_1) with support σ varying from 0.05 to 0.90. Groundtruth rules are those discovered by TANE/DFD at each σ threshold, since they enumerate all rules with support above σ [40]. BSFD consistently outperforms FDX and FDM$_\epsilon$, which shows that BSFD efficiently discovers low-support rules at the cost of little accuracy loss. The accuracy of BSFD remains high as σ varies, which shows that BSFD is able to recover both low- and high-support rules for practical use.

Robustness. Following [16], [23], we inject noise into Amazon and Statlog, that hold rich FDs sensitive to noise and support fair comparison as all baselines are executable on them, and the noise rate γ is the fraction of flipped cells. As shown in Fig. 7(k) and 7(l), (1) as γ increases from 0.01 to 0.30, the F_1 of BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ drop by at most 0.034 and 0.097 on Amazon and Statlog, respectively. (2) At the highest noise rate ($\gamma = 0.3$), the average F_1 of BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ remain 0.938 and 0.938, respectively. (3) Across all noise levels, BSFD$_{\text{TANE}}$ and BSFD$_{\text{DFD}}$ on average rank the first with F_1 of 0.980, while BSFD$^{\text{G}}_{\text{DFD}}$ ranks the third. These results verify the robustness of BSFD to noise.

Exp-4: Effectiveness of $H(A|X)$. We evaluate $H(A|X)$ as a criterion for identifying high-confidence FD$_{\sigma,\delta}$s by comparing standard BN structure scores [25] for BSFD and BSFD$^{\text{G}}$ across all datasets (Fig. 8), where lower values indicate a better fit. BSFD is scored using the star-shaped BN centered at that RHS, while BSFD$^{\text{G}}$ is scored using the RHS 's Markov blanket from the global BN. Across metrics, BSFD consistently attains lower or equal scores than BSFD$^{\text{G}}$, especially in the mean, which indicates more informative correlated attribute sets. Averaged over datasets, BSFD reduces the mean score by 289.32 and the minimum score by 308.91 relative to BSFD$^{\text{G}}$, which provides clear evidence that BSFD uncovers high-confidence FD$_{\sigma,\delta}$s that align better with the data distribution.

Analysis. The results confirm that lower conditional entropy $H(A \mid X)$ effectively guides the discovery of high-confidence FD$_{\sigma,\delta}$s, as evidenced by BSFD achieving both lower scores and higher F_1 values. In contrast, the lower F_1 of BSFD$^{\text{G}}$ is explained by its higher conditional entropy due to using globally constructed BNs.

Exp-5: Search Space Reduction. We evaluated the effectiveness of our search space reduction strategy. Given a vertical partitioning of relation r into n sub-relations, the total number of candidate LHS sets is reduced from $2^{|R|}$ to $\sum_{i=1}^{n} 2^{|R_i|}$, where $|R|$ is the number of attributes in r, and $|R_i|$ denotes the number in each sub-relation r_i. The overall reduction ratio is computed as $\frac{2^{|R|}-\sum_{i=1}^{n} 2^{|R_i|}}{2^{|R|}}$. As shown in Table V, all real-life datasets achieve over 90% reduction. This is mainly because the star-shaped BN limits the RHS to its most relevant attributes, reducing the search space even more. Interestingly, the reduction ratio does not monotonically decrease with larger $|R|$, as the number of candidate LHS attributes varies significantly across different RHS attributes, despite the growth in $|R|$ and the number of targets. Nonetheless, our approach maintains a high average F_1 score of 0.976, confirming that the reduced space preserves nearly all valid FD$_{\sigma,\delta}$s.

TABLE VII: Ablation evaluation of row sampling and correlated attributes discovery.

Method	Variant	Flight		Statlog		Amazon	
		F_1	Time	F_1	Time	F_1	Time
w/o Sample	TANE	1.000	1113.20	1.000	5.77	0.943	15.93
	DFD	1.000	11.87	1.000	0.58	0.943	15.88
	Average	**1.000**	562.54	**1.000**	3.18	**0.943**	15.91
BSFD$^{\text{G}}$	TANE	0.588	6.69	0.455	2.52	0.140	3.70
	DFD	0.588	6.41	0.455	2.46	0.140	3.70
	Average	0.588	6.55	0.455	2.49	0.140	3.70
BSFD	TANE	1.000	2.98	1.000	1.70	0.938	1.44
	DFD	1.000	3.00	1.000	0.93	0.938	1.47
	Average	**1.000**	**2.99**	**1.000**	**1.32**	0.938	**1.46**

Exp-6: Ablation Study. By omitting row sampling and replacing star-shaped BN with global-BN in correlated attributes discovery stages, we perform an ablation study on key components using Flight, Statlog and Amazon. Since the global BN variant BSFD$^{\text{G}}$ has been evaluated in Exp-1 to Exp-4, further analysis is omitted here. As shown in Table VII, removing sampling significantly increases runtime by 105.04× on average, while accuracy changes negligibly, which shows that sampling is crucial for efficiency.

Exp-7: Case Study. Finally, we conduct a case study on the real-world dataset Hospital. The discovered FD$_{\sigma,\delta}$s are (1) φ_1 : State $\xrightarrow{\sigma=0.1,\delta=0.9}$ HospitalType, with $\sigma = 0.24, \delta = 0.91$, and (2) φ_2 : HospitalOwner,EmergencyService $\xrightarrow{\sigma=0.1,\delta=0.9}$ HospitalType, with $\sigma = 0.12, \delta = 0.91$. Interestingly, the results differ from those obtained using classical FD discovery algorithms. Dependencies such as φ_1 and φ_2 exhibit high confidence but occur with relatively low support. These patterns would typically be pruned by traditional support-driven methods, but are preserved by our framework, demonstrating the value of using conditional entropy to guide discovery. Despite their low frequency, such dependencies are meaningful for minority groups. For example, from φ_1 and φ_2 we observe that only the states `AZ`, `AL`, or `AR` tend to have `Childrens` hospitals serving sensitive children, and that hospitals owned by `Government Federal` without emergency services are typically `Acute Care-VA Medical Centers`, which primarily provide services for aging veterans. These rules, therefore, can help aging veterans and sensitive children identify specific facilities.

IX. Conclusion

We propose an efficient framework for discovering low-support, high-confidence FDs from noisy datasets via BN. The key contributions are threefold: (1) a numerical equivalence between the aims of BN structure learning optimization and FD discovery through conditional entropy; (2) a BN-guided relation partitioning strategy that reduces validation overhead by filtering out irrelevant attributes; (3) a data-parallelism-based FD discovery method. The main limitation is the absence of a human-in-the-loop mechanism, which fails to improve sub-relation partitioning and interpret discovered FDs. As future work, we plan to involve domain experts in interpreting learned BNs to guide attribute selection and to incorporate user interests in labeling discovered FDs for practical adoption.

References

[1] F. Antonello, P. Baraldi, A. Shokry, E. Zio, U. Gentile, and L. Serio, "A novel association rule mining method for the identification of rare functional dependencies in complex technical infrastructures from alarm data," *Expert Systems with Applications*, vol. 170, p. 114560, 2021.

[2] K. Hu, L. Qiu, S. Zhang, Z. Wang, and N. Fang, "An incremental rare association rule mining approach with a life cycle tree structure considering time-sensitive data," *Applied Intelligence*, vol. 53, no. 9, p. 10800–10824, Aug. 2022. [Online]. Available: https://doi.org/10.1007/s10489-022-03978-3

[3] C. S. Hemalatha, V. Vaidehi, and R. Lakshmi, "Minimal infrequent pattern based approach for mining outliers in data streams," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1998–2012, 2015.

[4] A. Borah and B. Nath, "An efficient method for mining rare association rules: A case study on air pollution," *International Journal on Artificial Intelligence Tools*, vol. 30, no. 04, p. 2150018, 2021.

[5] X. Chu, I. F. Ilyas, and P. Papotti, "Holistic data cleaning: Putting violations into context," in *2013 IEEE 29th International Conference on Data Engineering (ICDE)*. IEEE, 2013, pp. 458–469.

[6] G. I. Webb, "Discovering significant rules," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 434–443.

[7] X. Wang, R. Jin, W. Huang, and Y. Tang, "Boosting meaningful dependency mining with clustering and covariance analysis," in *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 639–652.

[8] C. Giannella and E. Robertson, "On approximation measures for functional dependencies," *Information Systems*, vol. 29, no. 6, pp. 483–507, 2004, aDBIS 2002: Advances in Databases and Information Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306437903001054

[9] N. Asghar and A. Ghenai, "Automatic discovery of functional dependencies and conditional functional dependencies: a comparative study," *university of Waterloo*, 2015.

[10] J. Li, "On optimal rule discovery," *IEEE Transactions on knowledge and data engineering*, vol. 18, no. 4, pp. 460–471, 2006.

[11] W. W. Armstrong, "Dependency structures of data base relationships," in *Information Processing 74*. North-Holland, 1974, pp. 580–583.

[12] T. Papenbrock, J. Ehrlich, J. Marten, T. Neubert, J.-P. Rudolph, M. Schönberg, J. Zwiener, and F. Naumann, "Functional dependency discovery: An experimental evaluation of seven algorithms," *Proceedings of the VLDB Endowment*, vol. 8, no. 10, pp. 1082–1093, 2015.

[13] J. Liu, J. Li, C. Liu, and Y. Chen, "Discover dependencies from data—a review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 2, pp. 251–264, 2010.

[14] W. Fan, F. Geerts, J. Li, and M. Xiong, "Discovering conditional functional dependencies," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 5, pp. 683–698, 2010.

[15] J. D. Ullman, *Principles of Database and Knowledge-Base Systems: Volume II: The New Technologies*. USA: W. H. Freeman & Co., 1990.

[16] Y. Zhang, Z. Guo, and T. Rekatsinas, "A statistical perspective on discovering functional dependencies in noisy data," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 861–876.

[17] D. Z. Wang, X. L. Dong, A. D. Sarma, M. J. Franklin, and A. Y. Halevy, "Functional dependency generation and applications in pay-as-you-go data integration systems," in *International Workshop on the Web and Databases*, 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:15960115

[18] T. Papenbrock and F. Naumann, "A hybrid approach to functional dependency discovery," in *Proceedings of the 2016 International Conference on Management of Data*, 2016, pp. 821–833.

[19] T. Bleifuß, S. Bülow, J. Frohnhofen, J. Risch, G. Wiese, S. Kruse, T. Papenbrock, and F. Naumann, "Approximate discovery of functional dependencies for large datasets," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 1803–1812.

[20] F. Pennerath, P. Mandros, and J. Vreeken, "Discovering approximate functional dependencies using smoothed mutual information," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1254–1264.

[21] P. Schirmer, T. Papenbrock, S. Kruse, F. Naumann, D. Hempfing, T. Mayer, and D. Neuschäfer-Rube, "Dynfd: Functional dependency discovery in dynamic datasets." in *EDBT*, 2019, pp. 253–264.

[22] L. Caruccio, S. Cirillo, V. Deufemia, and G. Polese, "Incremental discovery of functional dependencies with a bit-vector algorithm," in *Sistemi Evoluti per Basi di Dati*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:195877475

[23] W. Fan, C. Hu, X. Liu, and P. Lu, "Discovering graph functional dependencies," *ACM Trans. Database Syst.*, vol. 45, no. 3, Sep. 2020. [Online]. Available: https://doi.org/10.1145/3397198

[24] P. Efraimidis and P. Spirakis, "Weighted random sampling," in *Encyclopedia of algorithms*. Springer, 2008, pp. 1024–1027.

[25] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[26] Y. Huhtala, J. Kärkkäinen, P. Porkka, and H. Toivonen, "Tane: An efficient algorithm for discovering functional and approximate dependencies," *The computer journal*, vol. 42, no. 2, pp. 100–111, 1999.

[27] Z. Abedjan, P. Schulze, and F. Naumann, "Dfd: Efficient functional dependency discovery," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, 2014, pp. 949–958.

[28] T. Scheffer, "Finding association rules that trade support optimally against confidence," *Intelligent Data Analysis*, vol. 9, pp. 381 – 395, 2001. [Online]. Available: https://api.semanticscholar.org/CorpusID:8611876

[29] Q. Pan, L. Xiang, and Y. Jin, "Rare association rules mining of diabetic complications based on improved rarity algorithm," in *2019 IEEE 7th International Conference on Bioinformatics and Computational Biology (ICBCB)*. IEEE, 2019, pp. 115–119.

[30] T. Rekatsinas, X. Chu, I. F. Ilyas, and C. Ré, "Holoclean: holistic data repairs with probabilistic inference," *Proc. VLDB Endow.*, vol. 10, no. 11, p. 1190–1201, Aug. 2017. [Online]. Available: https://doi.org/10.14778/3137628.3137631

[31] J. Quan and X. Sun, "Credit risk assessment using the factorization machine model with feature interactions," *Humanities and Social Sciences Communications*, vol. 11, pp. 1–10, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:267548711

[32] S. Begum and S. S. Padmannavar, "Prediction of student performance using genetically optimized feature selection with multiclass classification," *International Journal of Engineering Trends and Technology*, vol. 70, no. 4, pp. 223–235, 2022.

[33] E. F. Codd *et al.*, *Relational completeness of data base sublanguages*. IBM Corporation, 1972.

[34] W. Fan and F. Geerts, *Foundations of data quality management*. Morgan & Claypool Publishers, 2012.

[35] D. Simmen, E. Shekita, and T. Malkemus, "Fundamental techniques for order optimization," in *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, 1996, pp. 57–67.

[36] I. F. Ilyas, V. Markl, P. Haas, P. Brown, and A. Aboulnaga, "Cords: Automatic discovery of correlations and soft functional dependencies," in *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 2004, pp. 647–658.

[37] R. J. Miller, M. A. Hernández, L. M. Haas, L. Yan, C. Howard Ho, R. Fagin, and L. Popa, "The clio project: managing heterogeneity," *ACM Sigmod Record*, vol. 30, no. 1, pp. 78–83, 2001.

[38] T. Rekatsinas, X. Chu, I. F. Ilyas, and C. Ré, "Holoclean: Holistic data repairs with probabilistic inference," *arXiv preprint arXiv:1702.00820*, 2017.

[39] E. F. Codd, "Further normalization of the data base relational model," *Research Report / RJ / IBM / San Jose, California*, vol. RJ909, 1971. [Online]. Available: https://api.semanticscholar.org/CorpusID:45071523

[40] S. Kruse and F. Naumann, "Efficient discovery of approximate dependencies," *Proceedings of the VLDB Endowment*, vol. 11, no. 7, pp. 759–772, 2018.

[41] W. Fan, Z. Han, Y. Wang, and M. Xie, "Parallel rule discovery from large datasets by sampling," in *Proceedings of the 2022 international conference on management of data*, 2022, pp. 384–398.

[42] L. Golab, H. Karloff, F. Korn, D. Srivastava, and B. Yu, "On generating near-optimal tableaux for conditional functional dependencies," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 376–390, 2008.

[43] J. Rammelaere and F. Geerts, "Revisiting conditional functional dependency discovery: Splitting the "c" from the "fd"," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part II 18*. Springer, 2019, pp. 552–568.

[44] X. Ding, Y. Lu, H. Wang, C. Wang, Y. Liu, and J. Wang, "Dafdiscover: Robust mining algorithm for dynamic approximate

functional dependencies on dirty data," *Proc. VLDB Endow.*, vol. 17, pp. 3484–3496, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:272274464

[45] P. Mandros, M. Boley, and J. Vreeken, "Discovering reliable approximate functional dependencies," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 355–363.

[46] D. Sánchez, J.-M. Serrano, I. J. Blanco, M. J. Martín-Bautista, and M. A. Vila, "Using association rules to mine for strong approximate dependencies," *Data Mining and Knowledge Discovery*, vol. 16, pp. 313–348, 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:17634182

[47] N. Novelli and R. Cicchetti, "Fun: An efficient algorithm for mining functional and embedded dependencies," in *International Conference on Database Theory*. Springer, 2001, pp. 189–203.

[48] H. Yao, H. J. Hamilton, and C. J. Butz, in *Proceedings of the 2002 IEEE International Conference on Data Mining*, ser. ICDM '02. USA: IEEE Computer Society, 2002, p. 729.

[49] S. Lopes, J.-M. Petit, and L. Lakhal, "Efficient discovery of functional dependencies and armstrong relations," in *International Conference on Extending Database Technology*. Springer, 2000, pp. 350–364.

[50] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. I. Verkamo *et al.*, "Fast discovery of association rules." *Advances in knowledge discovery and data mining*, vol. 12, no. 1, pp. 307–328, 1996.

[51] C. Wyss, C. Giannella, and E. Robertson, "Fastfds: A heuristic-driven, depth-first algorithm for mining functional dependencies from relation instances extended abstract," in *Data Warehousing and Knowledge Discovery: Third International Conference, DaWaK 2001 Munich, Germany, September 5–7, 2001 Proceedings 3*. Springer, 2001, pp. 101–110.

[52] J. Kivinen and H. Mannila, "Approximate inference of functional dependencies from relations," *Theoretical Computer Science*, vol. 149, no. 1, pp. 129–149, 1995.

[53] L. Berti-Équille, H. Harmouch, F. Naumann, N. Novelli, and S. Thirumuruganathan, "Discovery of genuine functional dependencies from relational data with missing values," *Proceedings of the VLDB Endowment*, vol. 11, no. 8, pp. 880–892, 2018.

[54] C. Glymour, K. Zhang, and P. Spirtes, "Review of causal discovery methods based on graphical models," *Frontiers in genetics*, vol. 10, p. 524, 2019.

[55] S. Greco, Z. Pawlak, and R. Słowiński, "Can bayesian confirmation measures be useful for rough set decision rules?" *Engineering Applications of Artificial Intelligence*, vol. 17, no. 4, pp. 345–361, 2004.

[56] T. Wang, C. Rudin, F. Doshi-Velez, Y. Liu, E. Klampfl, and P. MacNeille, "A bayesian framework for learning rule sets for interpretable classification," *Journal of Machine Learning Research*, vol. 18, no. 70, pp. 1–37, 2017.

[57] O. Haas, L. I. Lopera Gonzalez, S. Hofmann, C. Ostgathe, A. Maier, E. Rothgang, O. Amft, and T. Steigleder, "Predicting anxiety in routine palliative care using bayesian-inspired association rule mining," *Frontiers in Digital Health*, vol. 3, p. 724049, 2021.

[58] H. Saxena, L. Golab, and I. F. Ilyas, "Distributed implementations of dependency discovery algorithms," *Proc. VLDB Endow.*, vol. 12, no. 11, p. 1624–1636, Jul. 2019. [Online]. Available: https://doi.org/10.14778/3342263.3342638

[59] E. Garnaud, N. Hanusse, S. Maabout, and N. Novelli, "Parallel mining of dependencies," in *2014 International Conference on High Performance Computing and Simulation (HPCS)*, 2014, pp. 491–498.

[60] W. Li, Z. Li, Q. Chen, T. Jiang, and Z. Yin, "Discovering approximate functional dependencies from distributed big data," in *Asia-Pacific Web Conference*. Springer, 2016, pp. 289–301.

[61] W. Li, Z. Li, Q. Chen, T. Jiang, and H. Liu, "Discovering functional dependencies in vertically distributed big data," in *Web Information Systems Engineering–WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Proceedings, Part II 16*. Springer, 2015, pp. 199–207.

[62] D. Margaritis, "Learning bayesian network model structure from data," 2003. [Online]. Available: https://api.semanticscholar.org/CorpusID:122677662

[63] N. K. Kitson, A. C. Constantinou, Z. Guo, Y. Liu, and K. Chobtham, "A survey of bayesian network structure learning," *Artificial Intelligence Review*, vol. 56, no. 8, pp. 8721–8814, 2023.

[64] "Code, datasets and full version," https://anonymous.4open.science/r/BSFD-45D7, 2025.

[65] R. Cavallo and M. Pittarelli, "The theory of probabilistic databases." in *VLDB*, vol. 87, 1987, pp. 1–4.

[66] A. Y. Khinchin, *Mathematical foundations of information theory*. Courier Corporation, 2013.

[67] D. E. Heckerman, D. Geiger, and D. M. Chickering, "Learning bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, pp. 197–243, 1994. [Online]. Available: https://api.semanticscholar.org/CorpusID:12224032

[68] R. Bardenet and O.-A. Maillard, "Concentration inequalities for sampling without replacement," 2015.

[69] O. Zuk, S. Margel, and E. Domany, "On the number of samples needed to learn the correct structure of a bayesian network," *arXiv preprint arXiv:1206.6862*, 2012.

[70] T. J. Koski and J. Noble, "A review of bayesian networks and structure learning," *Mathematica Applicanda*, vol. 40, no. 1, 2012.

[71] A. Ankan and J. Textor, "pgmpy: A python toolkit for bayesian networks," *Journal of Machine Learning Research*, vol. 25, no. 265, pp. 1–8, 2024. [Online]. Available: http://jmlr.org/papers/v25/23-0487.html

[72] C. P. Kruskal, L. Rudolph, and M. Snir, "A complexity theory of efficient parallel algorithms," *Theoretical Computer Science*, vol. 71, no. 1, pp. 95–132, 1990.

[73] P. Cortez, "Student Performance," UCI Machine Learning Repository, 2008, DOI: https://doi.org/10.24432/C5TG7T.

[74] H. Hofmann, "Statlog (German Credit Data)," UCI Machine Learning Repository, 1994, DOI: https://doi.org/10.24432/C5NC77.

Appendix

A. *Proof of Lemma 1*

The proof examines two classical BN structure learning paradigms: constraint-based and score-based.

(1) Constraint-based methods. These approaches determine the presence of an edge $X \rightarrow A$ through CI tests, typically using statistical measures such as χ^2 or mutual information (MI) to assess dependency strength [25].

- ◦ χ^2: Evaluates the divergence between observed and expected joint frequencies of X and A under the assumption of independence. A higher χ^2 value suggests stronger dependence, i.e., greater evidence against $X \perp A$.
- ◦ MI: From an information-theoretic perspective, MI is defined as $I(X;A) = H(A) - H(A \mid X)$, where $H(A)$ is the entropy of A. Since $H(A)$ is fixed for a given distribution, maximizing $I(X;A)$ is equivalent to minimizing $H(A \mid X)$.

Empirically, the two measures are related approximately by $\chi^2 \approx 2 \cdot I(X;A)$ [25], both increasing as $H(A \mid X)$ decreases. Thus, constraint-based methods tend to favor edges $X \rightarrow A$ that exhibit low conditional entropy, thereby indicating strong statistical dependence.

(2) Score-based methods. These methods evaluate candidate graph structures using scoring functions such as the BIC, which can be reformulated as [25]:

$$\begin{aligned}\text{score}_{BIC}(G) &= -|r| \sum_{i=1}^{n} H_{\hat{P}}(X_i \mid \text{Pa}_{X_i}) - \frac{\log |r|}{2}\text{Dim}[G] \\ &= -|r| \sum_{i=1}^{n} H_{\hat{P}}(X_i \mid \text{Pa}_{X_i}) + C. \end{aligned} \tag{1}$$

The constant C consolidates terms not dependent on the parent sets. Therefore, maximizing the BIC score corresponds to minimizing the total conditional entropy across all nodes. Consequently, score-based methods also favor edges that lead to lower conditional entropy. □

B. *Proof of Theorem 2*

It is evident that $\mathsf{pconf}(X \rightarrow A, x) = p(A = f(x) \mid X = x)$. Intuitively, for tuples in r that satisfy φ, $p(A = f(x) \mid X = x) \rightarrow 1$, and $p(A \neq f(x) \mid X = x) \rightarrow 0$. The proof consists of two parts:

(1) If $X \rightarrow A$ is an FD with high confidence, then $H(A \mid X) \rightarrow 0$. We divide the dataset into two parts: tuples that satisfy the FD, i.e., $t[A] = f(t[X])$, and those that violate it, i.e., $t[A] \neq f(t[X])$. Consider the FD `Education` → `Edu-num`, as illustrated in Fig. 2, tuples in the Adult relation are split into those where $t[\texttt{Edu-num}] = f(\texttt{Education})$ (green dashed box) and those where $t[\texttt{Edu-num}] \neq f(\texttt{Education})$ (red dashed box). The conditional entropy $H(A \mid X)$ is then computed as in [65]:

$$\begin{aligned} H(A \mid X) &= -\sum_{x \in X} p(x) \sum_{a \in A} p(a \mid x) \log p(a \mid x) \\ &= -\sum_{x \in X} p(x) \bigg(\sum_{a = f(x)} p(a \mid x) \log p(a \mid x) \\ &\quad + \sum_{a \neq f(x)} p(a \mid x) \log p(a \mid x) \bigg). \end{aligned} \tag{2}$$

As shown in Fig. 3, the limits $\lim_{p(a|x) \rightarrow 0} p(a \mid x) \log_2 p(a \mid x) = 0$ and $\lim_{p(a|x) \rightarrow 1} p(a \mid x) \log_2 p(a \mid x) = 0$ hold. Assuming $p(x) \neq 0$, this implies that both correct and incorrect mappings contribute negligibly to $H(A \mid X)$ when the FD $X \rightarrow A$ has high pconf. By Lemma 3, high pconf indicates high overall confidence, and thus $H(A \mid X) \rightarrow 0$ for high-confidence FDs.

(2) $H(A \mid X) \rightarrow 0$ implies that the FD $X \rightarrow A$ has high confidence. Since $x \log x \leq 0$ for $x \in [0,1]$, both terms $\sum_{a=f(x)} p(a \mid x) \log p(a \mid x)$ and $\sum_{a \neq f(x)} p(a \mid x) \log p(a \mid x)$ are non-positive and bounded above by zero. When $H(A \mid X) \rightarrow 0$, it follows that each of these terms must also approach zero. The expression $x \log_2 x \rightarrow 0$ holds only as $x \rightarrow 1$ or $x \rightarrow 0$, which implies that $p(A = f(x) \mid X = x) \rightarrow 1$ and $p(A \neq f(x) \mid X = x) \rightarrow 0$, respectively. According to Lemma 3, if $\mathsf{pconf}(X \rightarrow A, x) \rightarrow 1$ for every $x \in \text{dom}(X)$, then the overall confidence $\text{conf}(X \rightarrow A)$ also approaches 1. Therefore, $H(A \mid X) \rightarrow 0$ implies that the FD $X \rightarrow A$ has high confidence. □

C. *Proof of Theorem 3*

Since samples within each stratum are i.i.d. and without replacement, applying Hoeffding-serfling inequality in [68] yields: $\Pr\left[\left|\widehat{P}(S_a) - P(r_a)\right| \leq \sqrt{\frac{\rho_{S_a} \ln(1/\alpha)}{2|S_a|}}\right] \geq 1 - \alpha$, with: $\rho_{S_a} = \begin{cases} 1 - \frac{|S_a|-1}{|r_a|}, & |S_a| \leq |r_a|/2, \\ (1 - \frac{|S_a|}{|r_a|})(1 + \frac{1}{|S_a|}), & |S_a| > |r_a|/2. \end{cases}$ Based on $\widehat{P}_S = \sum_{a=1}^{m} \frac{|S_a|}{|S|} \widehat{P}(S_a)$ and $P(r) = \sum_{a=1}^{m} \frac{|r_a|}{|r|} P(r_a)$ and triangle inequality, we get $\left|\widehat{P}(S) - P(r)\right| \leq \sum_{a=1}^{m} \frac{|r_a|}{|r|} \left|\widehat{P}(S_a) - P(r_a)\right|$. Let $r_{\min} = \min_a r_a$ and $N_{min} = kr_{\min}$, $\beta = m\alpha$. Applying the union bound over all m strata [69], the tight bound is given by: $\Pr\left[\left|\widehat{P}(S) - P(r)\right| \leq \sqrt{\frac{\rho_{r_{min}} \ln(m/\beta)}{2N_{\min}}}\right] \geq 1 - \beta$, with $\rho_{r_{min}} = \begin{cases} 1 - \frac{N_{min}-1}{r_{min}}, & N_{min} \leq r_{min}/2, \\ (1 - \frac{N_{min}}{r_{min}})(1 + \frac{1}{N_{min}}), & N_{min} > r_{min}/2. \end{cases}$ To satisfy given error tolerance ϵ, i.e., $\Pr\left[\left|\widehat{P}(S) - P(r)\right| \leq \epsilon\right] \geq 1 - \beta$. Define a constant $C = \frac{ln(1/\alpha)}{2\epsilon^2}$, recall that $N_{min} = kr_{min}$, following [68], we have

$$\epsilon^2 \geq \begin{cases} \frac{\ln(1/\alpha)}{2kr_{\min}}(1 - k + 1/r_{min}), & k \leq 1/2, \\ \frac{\ln(1/\alpha)}{2kr_{\min}}(1 - k)(1 + \frac{1}{kr_{min}}), & k > 1/2, \end{cases}$$

then, for each case, we obtain the lower bound of k:

$$k \geq \begin{cases} \frac{C(1+1/r_{min})}{r_{min}+C}, & C \leq \frac{r_{\mathrm{min}}^2}{r_{\mathrm{min}}+2}, \\ \frac{C\left(r_{min}-1\right) + \sqrt{C\left(Cr_{min}^2+2Cr_{min}+C+4r_{min}^2\right)}}{2r_{min}\left(C+r_{min}\right)}, & C > \frac{r_{\mathrm{min}}^2}{r_{\mathrm{min}}+2}. \end{cases}$$

□