Aegis: A Correlation-Based Data Masking Advisor for Data-Sharing Ecosystems

OMAR ISLAM LASKAR, Indian Institute of Technology Delhi, India
FATEMEH RAMEZANI KHOZESTANI, New Jersey Institute of Technology, USA
ISHIKA NANKANI, Indian Institute of Technology Delhi, India
SOHRAB NAMAZI NIA, New Jersey Institute of Technology, USA
SENJUTI BASU ROY, New Jersey Institute of Technology, USA
KAUSTUBH BEEDKAR, Indian Institute of Technology Delhi, India

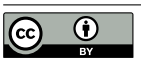
Data-sharing ecosystems connect providers, consumers, and intermediaries to facilitate the exchange and use of data for a wide range of downstream tasks. In sensitive domains such as healthcare, privacy is enforced as a hard constraint—any shared data must satisfy a minimum privacy threshold. However, among all masking configurations that meet this requirement, the utility of the masked data can vary significantly, posing a key challenge: how to efficiently select the *optimal* configuration that preserves maximum utility. This paper presents Aegis, a middleware framework that selects optimal masking configurations for machine learning datasets with features and class labels. Aegis incorporates a utility optimizer that minimizes *predictive utility deviation*—quantifying shifts in feature-label correlations due to masking. Our framework leverages limited data summaries (such as 1D histograms) or none to estimate the feature-label joint distribution, making it suitable for scenarios where raw data is inaccessible due to privacy restrictions. To achieve this, we propose a joint distribution estimator based on iterative proportional fitting, which allows supporting various feature-label correlation quantification methods such as mutual information, chi-square, or g3. Our experimental evaluation of real-world datasets shows that Aegis identifies optimal masking configurations over an order of magnitude faster, while the resulting masked datasets achieve predictive performance on downstream ML tasks on par with baseline approaches and complements privacy anonymization data masking techniques.

CCS Concepts: • **Information systems → Data management systems**; • **Security and privacy → Data anonymization and sanitization**.

Additional Key Words and Phrases: Data Markets, Data Sharing Ecosystems, Data Masking, Compliant Data Management

ACM Reference Format:
Omar Islam Laskar, Fatemeh Ramezani Khozestani, Ishika Nankani, Sohrab Namazi Nia, Senjuti Basu Roy, and Kaustubh Beedkar. 2025. Aegis: A Correlation-Based Data Masking Advisor for Data-Sharing Ecosystems. *Proc. ACM Manag. Data* 3, 6 (SIGMOD), Article 292 (December 2025), 26 pages. https://doi.org/10.1145/3769757

Authors' Contact Information: Omar Islam Laskar, Indian Institute of Technology Delhi, New Delhi, India, omar.laskar.cs522@cse.iitd.ac.in; Fatemeh Ramezani Khozestani, New Jersey Institute of Technology, New Jersey, USA, fr46@njit.edu; Ishika Nankani, Indian Institute of Technology Delhi, New Delhi, India, ishika.nankani.cs122@cse.iitd.ac.in; Sohrab Namazi Nia, New Jersey Institute of Technology, New Jersey, USA, sn773@njit.edu; Senjuti Basu Roy, New Jersey Institute of Technology, New Jersey, USA, senjuti.basuroy@njit.edu; Kaustubh Beedkar, Indian Institute of Technology Delhi, New Delhi, India, kbeedkar@cse.iitd.ac.in.



ACM 2836-6573/2025/12-ART292
https://doi.org/10.1145/3769757

1 Introduction

The increasing adoption of data-driven decision-making and machine learning (ML) in finance, healthcare, and public policy has made data-sharing ecosystems an integral component of modern information infrastructure. These ecosystems bring together data providers, consumers, and intermediaries—such as AWS Data Exchange [3], Snowflake Data Marketplace [8], OpenMined [40], DataSHIELD [19], GAIA-X [24]) and open data portals [7]—to facilitate the access, exchange, and use of data for research, innovation, and economic development across both public and private sectors.

As data-sharing practices become more widespread, concerns around data privacy and proprietorship have led to the need for strict compliance with data regulations like GDPR [22], CCPA [14], HIPAA [55] among others. As a result, data providers often anonymize their datasets by masking sensitive attributes using masking functions like generalization, which replaces specific values with broader categories; suppression, which redacts sensitive details; and perturbation, which introduces controlled noise. While anonymization is essential for regulatory compliance, it often transforms key properties of the data—posing challenges for downstream ML tasks, where preserving statistical structure and predictive performance is crucial. Moreover, for any given dataset, there are often multiple *masking configurations*, each prescribing and differing in how specific attribute values must be masked [43].

For example, consider the following scenario. In a real-world data marketplace for healthcare, hospitals and clinics share patient-level Electronic Health Records (EHRs) to third parties, such as insurance companies. In such settings (e.g., HIPAA in the U.S.) *privacy is binary*—the data is considered private and eligible for sharing if it satisfies a threshold, such as k-anonymity with a specific k-value [21, 44]; While a minimum k-value of 3 is sometimes suggested, a common recommendation in practice is to use $k = 5$. However, some scenarios might require higher values, like $k = 10$ or 15, to reduce the risk of re-identification to an acceptable level is used. When the specific privacy guarantee is not reached, the data is not private and cannot be shared. This binary treatment is crucial for compliance and auditability. To satisfy the privacy requirement, the marketplace applies various masking configurations to the data.

In the above scenario for instance, consider a dataset for predicting health based on features like age, weight, and zip code. A detailed running example based on this dataset is presented in Section 2. Masking configurations may involve generalizing age into ranges (e.g., $23 \rightarrow 20$–30), blurring weight by rounding to the nearest ten (e.g., $53.5 \rightarrow 50$, or generalizing zip codes into broader geographic regions (e.g., $12345 \rightarrow 123{*}{*}$). Alternatively, they might suppress zip codes entirely or add noise to sensitive numerical values. The system only allows those masking configurations that surpass the k-anonymity threshold, are considered valid, and only those versions of the data can be shared. In general, while each of the masking strategies that meet the binary privacy rule is acceptable, they affect the feature-label distributions differently and, consequently, may lead to substantial variation in the accuracy of the trained model.

In the context of a data-sharing ecosystem, this creates a fundamental tension. Data providers often do not know the specific downstream task consumers intend to perform. They thus cannot select the masking configuration that will preserve their dataset's utility for a particular model or use case. For example, Figure 1 shows the results of an experiment (see Section 5 for details) that shows how a dataset's utility (based on model accuracy) changes across different masking configurations for a dataset depending on the downstream ML task. On the other hand, for data consumers, the state-of-the-art approach for finding the optimal masking configuration involves exhaustively applying each configuration, building a model, and evaluating its predictive utility (see Figure 2a). After testing all configurations, the one with the highest utility (e.g. highest accuracy)

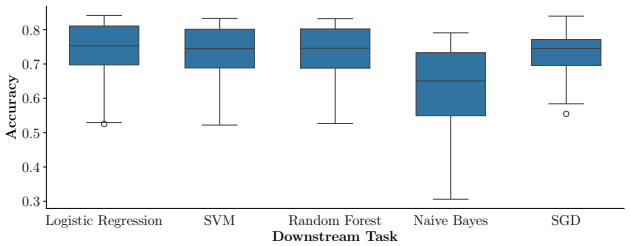


Fig. 1. Distribution of model accuracy across 50 masking configurations for Air Quality dataset [2].

is selected. However, this exhaustive process is computationally expensive and impractical for large-scale real-world datasets and complex models. This makes it difficult to efficiently find an effective masking configuration from a set of possible ones- one that preserves utility across a range of possible uses- both critical and nontrivial.

Contributions. In this paper, we present Aegis (Figure 2b), a framework designed to efficiently selecting optimal masking configurations for machine learning datasets—without requiring exhaustive evaluation of the entire pipeline. Our work is an effort to select masking configurations and not to create new ones that are privacy aware. Our work complements existing privacy preservation based masking techniques—Aegis can incorporate any k-anonymity, differential privacy, structural anonymization, or masking operation to evaluate and select the configuration that best preserves predictive utility in a model-agnostic way. In that sense, we are orthogonal to existing work on privacy preserving masking configuration generation related work.

At the core of Aegis is its *predictive utility estimator*, which estimates the predictive value of each feature with respect to the class label, even in scenarios where the feature, the label, or both are masked. This capability makes Aegis well-suited as a middleware solution for data-sharing ecosystems where access to raw data is restricted.

Aegis builds upon the understanding that the predictive utility of a dataset is often evaluated through model-agnostic techniques, commonly referred to as correlation measures. A variety of methods exists for quantifying such relationships. Among the most widely used are: Mutual Information (MI), which captures the statistical dependency between variables [10, 15, 23, 27, 31, 42, 56]; the Chi-Square Test of Independence [16], which determines whether the two categorical variables are related by comparing observed and expected frequencies. More recently, [32] has connected measures of approximate functional dependency, particularly the g_3 error [29], to theoretical upper bounds on classification accuracy. Our predictive utility quantifier can accommodate any of these correlation measures as inputs and quantifies the impact of masking configurations on the dataset's predictive utility. This allows masking configurations to be determined in a task-agnostic fashion. Aegis is orthogonal to k-anonymity [52] and other privacy-preserving techniques [33, 36], allowing it to integrate seamlessly with existing privacy-preserving methods to maintain both privacy and predictive utility.

We propose the notion of *predictive utility deviation* inside *predictive utility estimator* in Aegis. Given a set of predictors, a class label, a set of masking functions to be applied to the predictors, and a correlation measure, predictive utility deviation quantifies *the total change in correlation between the unmasked predictors and the class label compared to the masked predictors and the class label.* We then define the **Optimal Predictive Utility Aware Masking Configuration Selection Problem**, as follows: Given a set of possible masking configurations to be applied over a set of predictors (attributes) and a class label, where the joint frequency distributions between the

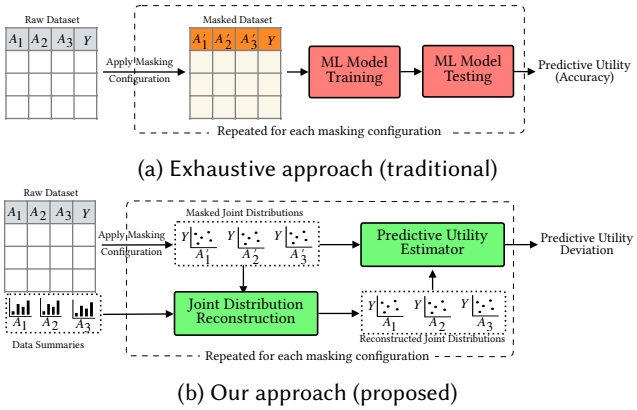


Fig. 2. Determining optimal masking configuration using (a) traditional approach and (b) our proposed approach.

unmasked predictors and the class label are unknown (due to the sensitive nature of the predictors), select the masking configuration to be applied on the predictors, such that, the *predictive utility deviation is minimized.*

To minimize predictive utility deviation, the grand challenge is to be able to estimate the unmasked predictor from the known information and reconstruct the joint distribution of the predictor and the class label from there. Examples of known information could be 1D histograms of the predictor, e.g., for *Age*, for example, there exists 4 individuals with age 10, 12 individuals 17, 4 with age 43, and so on. Given the masked *Age*, it may be known that there exist 20 young and 80 old individuals in the masked Age, and the masking function dictates that $Young \longrightarrow 10-45$, $Old \longrightarrow > 45$. From this, the reconstructed Age has to satisfy the following: there are 100 individuals of *Age* between $10-80$, among them 4 individuals are of age 10, 12 individuals of age 17, 4 of age 43, and so on. Clearly, a prohibitively large number of possible distributions of *Age* satisfies all these constraints, and any of these could be a likely solution. Without additional information, we assume that any value of *Age* between $10-80$ is equally likely. We formalize this as an optimization problem, i.e., estimate the joint distribution of each predictor and the class label (i.e., *Age* and *Health*) such that the distribution of the predictor (i.e., *Age*) is as uniform as possible within the given range (i.e., between $10-80$,). Still, the reconstructed joint distribution satisfies all constraints imposed on the marginals (e.g., available known information on *Age*). Once the joint distribution is estimated for each predictor-masking-function pair per configuration, the problem selects the one that minimizes the predictive utility deviation. We solve this problem by adapting Iterative Proportional Fitting (or IPF) [28, 30], an iterative algorithm designed to adjust statistics distribution.

We study two variants of the *Optimal Predictive Utility Aware Masking Configuration Selection Problem* : (a) When 1D histograms of unmasked predictors are available. Additionally, masking functions to be applied on the predictors and the masked predictors are also known; (b) When no distributional statistics of the unmasked predictors are available, but the masking functions to be applied on the predictors are known along with their masked values. We formalize both variants as optimization problems and demonstrate how IPF could be adapted to solve both. When more information is known about the predictors (case a), the joint estimation becomes more accurate.

We demonstrate the efficacy of the proposed framework using three real-world datasets considering three different correlation measures, namely, *Mutual Information, Chi-Squared Test, and g_3 error.* Our experiments demonstrate that Aegis can determine an optimal masking configuration order of

Age	Weight	Zip	Health
10	30	21162	**G**ood
10	31	21168	**G**ood
43	63	22170	**M**oderate
65	71	23175	**P**oor
75	80	23173	**V**ery **P**oor
80	78	25165	**V**ery **G**ood

(a) Excerpt of a dataset

Age	**10**	**17**	**43**	**55**	**60**	**65**	**75**	**80**
Count	4	12	4	30	20	10	10	10

Health	**VG**	**G**	**M**	**P**	**VP**
Count	14	16	30	25	15

(b) Marginal distributions

Fig. 3. (a) Example ML dataset with three attributes (Age, Weight, and Zip) and a label (Health); (b) marginal Age and Health distributions of 100 individuals.

magnitude faster while achieving comparable predictive performance for various downstream ML tasks compared to baseline approaches.

Section 2 presents data model and preliminaries, Section 3 presents Aegis. In Section 4, we present the algorithms inside Aegis. Section 5 presents our experimental evaluation, we discuss related works in Section 6, and we conclude in Section 7.

2 Preliminaries

We introduce the data model we are considering and the related concepts. Table 1 summarizes the notations we commonly use throughout the paper.

Dataset. We consider ML datasets, where a dataset is a collection of related observations comprising features/ attributes/predictors and a label organized in tabular format. More formally, denote by $D = \{ (x_i, y_i) \}_{i=1}^{N}$ a dataset with N rows (data points). The i-th data point $x_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \in \mathbb{X}^m$) is a feature vector over m attributes/predictors ($\mathcal{A} = \{A_1, A_2 \ldots A_m\}$). For the i-th data point, $y_i \in \mathbb{Y}$ is the label associated.

Example 1. *Figure 3a presents a sample dataset intended for machine learning tasks, where* `age`*,* `weight`*, and* `zip` *are attributes (predictors), and* `health` *serves as the label. Each feature contributes to predicting the health (class label).*

Domain of an attribute. For an attribute A, $\texttt{Dom}(A)$ denotes its domain, referring to all possible values that A can take. The j-th domain value for A is called a^j. Similarly, the domain of the label $\mathbb{Y}$ is $\texttt{Dom}(\mathbb{Y})$. For example, in Figure 3a, we have Dom(Health)= {Very Good, Good, Moderate, Poor, Very Poor}.

Masking Functions. In data-sharing ecosystems, datasets are often shared after anonymizing them by using masking functions. Moreover, an attribute (e.g., `Age`) can be masked differently using different masking functions (e.g., generalizing age to ranges into categories). For each attribute $A \in \mathcal{A}$, let $\mathcal{C}_A$ be the set of candidate masking functions applicable to A. When an attribute A is masked using a masking function $M \in \mathcal{C}_A$, the resulting masked attribute is denoted by $M(A)$. More formally, a *masking function*[1] M is a function that takes as input the i-th domain value $a^i \in \texttt{Dom}(A)$ (typically with other function parameters) and outputs a masked value $a^{i'}$, such that: $M(a^i) = a^{i'}, \quad \forall a^i \in \text{Dom}(A)$. The masked attribute $M(A)$ is then defined as the attribute whose domain consists of the values $\{M(a^i) \mid a^i \in \text{Dom}(A)\}$.

[1]In this paper, we consider scalar and value-based masking functions. In general, this is not a limitation. The techniques proposed here can be extended to include other masking functions.

Table 1. Table of frequently used notations.

Symbol	Description
M, $\boldsymbol{M}$	Masking function, Masking configuration
A, $M(A)$, $\mathbb{Y}$	Attribute, Masked attribute, Class label
D, $D_{\boldsymbol{M}}$	Dataset, Masked dataset over configuration $\boldsymbol{M}$
$f(a^i)$, $f(a^i, y^j)$	Frequency of attribute value a^i, attribute-label a^i, y^j
$\mathbb{F}(A, \mathbb{Y})$, $\rho(A; \mathbb{Y})$	Joint distribution and Predictive utility measure

Domain of a masked attribute. Let $\texttt{Dom}(M(A))$ denote the domain of attribute A masked by the masking function M. Using our running example, we have $\texttt{Dom}(M(\texttt{Age})) = \{\text{Young}, \text{Old}\}$.

Example 2. *Consider the* Age *values in Figure 3a and a masking function M that returns 'Young' for values $10-45$ and 'Old' for values ≥ 45. Then, for masked attribute M(Age) will have values $\langle$ Young, Young, Young, Old, Old, Old $\rangle$.*

Masking Configuration & Masked Dataset. A *masking configuration* is a tuple $\boldsymbol{M} = (M_1, M_2, ..., M_m)$, which applies the masking function M_j on attribute A_j, where $M_j \in \mathcal{C}_{A_j}$. Applying $\boldsymbol{M}$ to D produces a *masked dataset* $D_{\boldsymbol{M}} = \left\{ (x'_i, y_i) \right\}_{i=1}^{N}$, where $x'_i = (M_1(x_{i1}), M_2(x_{i2}), ..., M_m(x_{im}))$.

Example 3. *Consider two masking configurations that can be used for masking the example dataset of Figure 3a (for brevity, we omit other function parameters).*

$\boldsymbol{M}_1$ = (**Bu**cketize(Age), **Bl**ur(Weight), **S**uppress(Zipcode))

$\boldsymbol{M}_2$ = (**G**eneralize(Age),**Bu**cketize(Weight), Zipcode)

$D_{\boldsymbol{M}_1}$ (below left) and $D_{\boldsymbol{M}_2}$ (below right) show the excerpt of masked datasets after applying $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$ to D, respectively.

B(Age)	*Bl(Weight)*	*S(Zip)*	*Health*
1-10	*3**	*	***G***
1-10	*3**	*	***G***
31-40	*6**	*	***M***
61-70	*7**	*	***P***
71-80	*8**	*	***VP***
71-80	*7**	*	***VG***

G(Age)	*Bu(Weight)*	*Zip*	*Health*
Young	*30-35*	*21162*	***G***
Young	*30-35*	*21168*	***G***
Young	*60-65*	*22170*	***M***
Old	*70-75*	*23175*	***P***
Old	*80-85*	*23173*	***VP***
Old	*75-80*	*25165*	***VG***

Predictive Utility Measures of a Dataset. The *predictive utility* of a dataset refers to the extent to which it supports accurate predictions in a machine learning (ML) task. In simpler terms, it quantifies how effectively the features (inputs) in the dataset can explain or predict the target variable (label). Various model-specific metrics—such as precision, recall, accuracy, and AUC [26]—can be used to assess predictive utility. In this paper, however, we focus on model-agnostic, correlation-based measures [26], such as mutual information, Chi-Square, and related statistics (see Section 3.2).

High-Level Problem. In data-sharing ecosystems, the original data is anonymized and never directly accessible. We address the problem of selecting an optimal masking configuration in such settings. In some scenarios, besides the masked predictors, the original marginal distributions of the predictors may or may not be available.[2] Given a set of possible masking configurations and a dataset that can only be accessed in its masked form under one of these configurations, the goal is to select the configuration that maximizes the dataset's predictive utility. Continuing with

[2]In practice, frequency distributions are often accessible through histograms or summary statistics.

our running example, the objective is to determine which of two masking configurations—$\boldsymbol{M}_1$ or $\boldsymbol{M}_2$—yields the highest predictive utility according to the selected measure (e.g., accuracy).

3 Correlation-Based Data Masking Advisor

We propose Aegis, a middleware framework designed for data-sharing ecosystems that enables efficient determination of optimal masking configurations. Aegis acts as an "advisor" to data providers by recommending configurations that maximize predictive utility, independent of any specific downstream task. In the following, we give an overview of Aegis and formalize the problem of optimal masking configuration selection that we consider in this paper.

3.1 Aegis Overview

Figure 4 presents a schematic overview of our proposed framework, Aegis. We assume a collaborative setting where data providers work with privacy experts to determine appropriate masking configurations. Each configuration, when applied, produces a masked dataset that meets the minimum privacy guarantees and allows the dataset to be shared within a data-sharing ecosystem. In this context, Aegis facilitates the identification of an optimal masking configuration *independently* of the downstream task or model, making it particularly useful in scenarios where access to the raw dataset is restricted (e.g., due to privacy regulations). Our approach is grounded in the key principle that the predictive utility of a dataset can be assessed in a model-agnostic manner—i.e., without reliance on specific models—using techniques such as correlation-based measures. To achieve this, Aegis is comprises two main components:

Joint Distribution Reconstruction. As a first step, Aegis computes the joint distribution of (masked) attribute-label pairs for each masking configuration. When available, it further leverages marginal distributions (i.e., 1D histograms of the dataset's attributes) to reconstruct the joint distribution of *unmasked* attribute-label pairs.

Predictive Utility Estimator. Given the estimated joint distribution for each (attribute-label, masking configuration) pair[3], the predictive utility estimator computes a *predictive utility deviation* score (formally defined below) for each configuration. The estimator supports well-established, model-agnostic utility measures (see below) to evaluate utility deviation and recommends an optimal masking configuration accordingly.

In what follows, we formally define model-agnostic predictive utility measures, introduce the notion of predictive utility deviation, and describe the problem of selecting an optimal masking configuration that minimizes this deviation.

3.2 Model Agnostic Predictive Utility

Understanding the predictive utility of a dataset while being agnostic about the downstream model or task is crucial in data-sharing settings. In this section, we first explore model-agnostic approaches for estimating predictive utility, focusing on correlation-based measures for discrete and discretized continuous data. We then introduce a novel, model-agnostic metric called Predictive Utility Deviation to quantify utility loss under different masking configurations.

3.2.1 Predictive Utility Measures. The predictive utility of an attribute/feature refers to the amount of information it contributes to predicting the class label $\mathbb{Y}$. This utility is assessed without training a specific machine-learning model in a model-agnostic setting. Instead, statistical or information-theoretic measures such as correlation or association between predictors and the class label are used to estimate how informative the features are. Our framework supports a wide range of measures

[3]In practice, the exact masking functions need not be known—an identifier is sufficient.

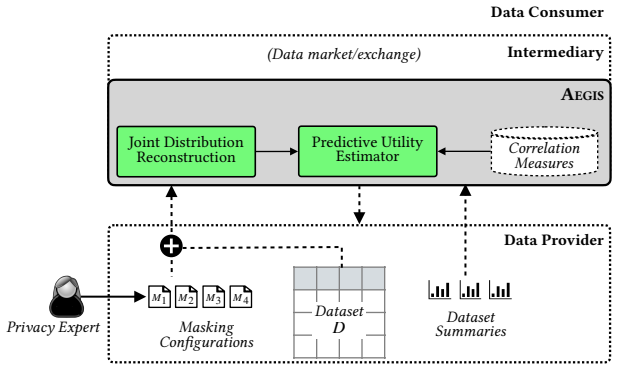


Fig. 4. Aegis Overview

designed for discrete or categorical data as long as the predictive utility can be captured through the joint distribution of the predictor and the class label. For continuous attributes, we assume that the data has been effectively discretized. Some of the correlation and association measures that we closely study are:

Mutual Information. The mutual information between predictor A_j and class label Y is defined as:

$$I(A;\mathbb{Y}) = \sum_{a^i \in Dom(A)} \sum_{y^j \in Dom(\mathbb{Y})} P(a^i y^j) \log\left(\frac{P(a^i y^j)}{P(a^i)P(y^j)}\right)$$

Where: $P(a^i y^j)$ is the joint distribution of A and $\mathbb{Y}$; and $P(a^i)$ and $P(y^j)$ are the marginal probability distributions of A and $\mathbb{Y}$, respectively.

g_3 **Error.** Given the attribute A and class label $\mathbb{Y}$, g_3, the error counts the number of records that must be deleted to satisfy the full functional dependency between $A \rightarrow \mathbb{Y}$.

$$g_3(A;\mathbb{Y}) = \frac{1}{N}(N - max\{|s| \mid s \in \mathcal{D},\ s \models A \rightarrow \mathbb{Y}\})$$

In other words, g_3 error captures how far a dataset is from satisfying a functional dependency $A \rightarrow \mathbb{Y}$ as a functional dependency.

Chi-Square Statistic. The Chi-Square statistic between attribute A and class label $\mathbb{Y}$ is given by:

$$\chi^2(A;\mathbb{Y}) = \sum_{\forall a^i \in Dom(A), y^j \in Dom(\mathbb{Y})} \frac{(O_{a^i,y^j} - E_{a^i,y^j})^2}{E_{a^i,y^j}}$$

Where: χ^2 = Chi-Square statistic; O_{a^i,y^j} = Observed frequency for the i, j-th domain values of A and $\mathbb{Y}$, respectively; and E_{a^i,y^j} = Expected frequency for the i, j-th domain values of A and $\mathbb{Y}$, respectively.

Example 4. *Continuing the running example for the dataset in Figure 3(a), we have* $I(\text{Age};\text{Health}) \approx 0.640$*;* $g_3(\text{Age};\text{Health}) = 0.53$*; and* $\chi^2(\text{Age};\text{Health}) \approx 85.96$.

3.2.2 Predictive Utility Deviation. For a given dataset D, a feature/predictor A, and a class label $\mathbb{Y}$ let ρ be the predictive utility measure (e.g., Mutual Information, Chi-Square, etc.), where $\rho(D, A, \mathbb{Y})$ returns a scalar value that reflects the predictive utility of A for $\mathbb{Y}$. If the m predictors in D are masked using a masking configuration $\boldsymbol{M} = (M_1, M_2, ..., M_m)$, we define the *predictive utility deviation* (PUD)

as the sum of the absolute differences between the predictive utility computed on the original dataset D and that of the masked dataset $D_{\boldsymbol{M}}$:

$$\Delta_{PU}(D, D_{\boldsymbol{M}}, \rho) = \frac{1}{m} \sum_{i=1}^{m} |\rho(A_i; \mathbb{Y}) - \rho(M(A_i); \mathbb{Y})| \tag{1}$$

3.3 Problem Definition

Problem Statement. **(Identify Optimal Masking Configuration to Minimize Predictive Utility Deviation.)** *Given a dataset D, a predictive utility measure ρ, and a set $\mathcal{M}_{set} = \{ \boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_K \}$ of K predefined masking configurations, find the masking configuration $\boldsymbol{M}^* \in \mathcal{M}_{set}$ that minimizes predictive utility deviation, i.e.,*

$$\boldsymbol{M}^* = \underset{\boldsymbol{M} \in \mathcal{M}_{set}}{\operatorname{argmin}} \, \Delta_{PU}(D, D_{\boldsymbol{M}}, \rho)$$

The primary challenge is that to compute any predictive utility measure ρ, the joint frequency distribution between each A and $\mathbb{Y}$ must be known. When the raw data D itself is not accessible, the joint distributions between the predictors and the class label are unknown, so Δ_{PU} (Equation 1) cannot be directly calculated from the data. We study two variants of this problem to that end.

Problem Statement. **(Variant I: Identify optimal masking configuration in the presence of marginal distributions.)** *Given a dataset D where the joint distributions of the attributes and the class label are not accessible, but the marginal distributions $Dist(A)$ of each masked predictor A is available. A set $\mathcal{M}_{set} = \{ \boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_K \}$ of K predefined masking configurations, for a given predictive utility measure ρ, find the masking configuration $\boldsymbol{M}^* \in \mathcal{M}_{set}$ that minimizes the predictive utility deviation.*

Problem Statement. **(Variant II: Identify optimal masking configuration in the absence of marginal distributions.)** *Given a dataset D where neither marginals nor joint distributions are accessible, for a given predictive utility measure ρ, and a given set $\mathcal{M}_{set} = \{ \boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_K \}$ of K predefined masking configurations, find the masking configuration $\boldsymbol{M}^* \in \mathcal{M}_{set}$ that minimizes the predictive utility deviation.*

4 Algorithms inside Aegis

We now turn attention to the algorithms inside Aegis. We first describe the algorithm for selecting the masking configuration with the minimum predictive utility deviation (Section 4.1). We will then discuss how Aegis reconstructs joint distributions to estimate the utility of masked datasets (Sections 4.2 and 4.3).

4.1 Overall Algorithm

Algorithm 1 outlines the end-to-end process for selecting the optimal masking configuration. For each configuration $\boldsymbol{M}_k$, the algorithm iterates over every masked predictor–label pair $(M(A), \mathbb{Y}) \in \boldsymbol{M}_k$ and first reconstructs the original joint distribution $\mathbb{F}(A, \mathbb{Y})$ by invoking Subroutine `Reconstruction` (Algorithm 2, line4). Then, using a chosen correlation or association measure ρ (see Section 3.2), it computes the predictive utility deviation between the masked distribution $\mathbb{F}(M(A), \mathbb{Y})$ and the reconstructed distribution $\mathbb{F}(A, \mathbb{Y})$ (lines 5–7). These individual deviations are averaged across all masked attributes in $\boldsymbol{M}_k$ to yield the total predictive utility deviation as defined in Equation 1 (line8). Finally, the configuration with the smallest total deviation is selected (line 9). Through this sequence of reconstruction, deviation computation, and aggregation, Algorithm 1 allows identifying the masking configuration that minimizes loss of predictive information. It is worth noting computing $\mathbb{F}(M(A), \mathbb{Y})$ does not require materializing the masked dataset for each configuration.

Algorithm 1 Finding the Best Masking Configuration

Require: Set of attributes $\mathcal{A}$, target $\mathbb{Y}$, set of masking configurations $\mathcal{M}_{\text{set}}$, available constraints, predictive utility measure ρ

Initialize $\boldsymbol{M}^* \leftarrow$ None
for each $\boldsymbol{M}_k \in \mathcal{M}_{\text{set}}$ **do**
  for each attribute A **do**
    $\mathbb{F}(A, \mathbb{Y}) \leftarrow$ Subroutine `Reconstruction`(A, constraints)
    Compute predictive utility $\mathbb{F}(A, \mathbb{Y}), \rho)$
    Compute predictive utility $\mathbb{F}(M(A), \mathbb{Y}), \rho)$
    Compute predictive utility deviation $(\mathbb{F}(A, \mathbb{Y}), \mathbb{F}(M(A), \mathbb{Y}), \rho)$
  Compute total predictive utility deviation of $\boldsymbol{M}_k$.
$\boldsymbol{M}^* \leftarrow$ configuration with the smallest predictive utility deviation.
return $\boldsymbol{M}^*$

Algorithm 2 Subroutine `Reconstruction`

Require: Uniform joint distribution $\mathbb{F}_{uni}(A, \mathbb{Y})$, constraints on rows of $\mathbb{F}(A, \mathbb{Y})$, constraints on columns of $\mathbb{F}(A, \mathbb{Y})$, convergence criteria
Ensure: Adjusted joint distribution $\mathbb{F}(A, \mathbb{Y})$

Initialize $\mathbb{F}(A, \mathbb{Y}) \leftarrow \mathbb{F}_{uni}(A, \mathbb{Y})$
repeat
  for each row i **do**
    Adjust $\mathbb{F}(A, \mathbb{Y})_{i,:}$ such that row constraints are satisfied
  for each column j **do**
    Adjust $\mathbb{F}(A, \mathbb{Y})_{:,j}$ such that column constraints are satisfied
until Convergence criteria is met
return $\mathbb{F}(A, \mathbb{Y})$

Lemma 1. *Running time of Algorithm 1 is* $\mathcal{O}(|\boldsymbol{M}_k| \times |\mathcal{M}_{set}| \times \mathbb{T})$*, where* $\mathbb{T}$ *is the time it takes to run Subroutine* `Reconstruction` *once.*

4.2 Reconstructing Joint Distributions

We now describe reconstructing the joint distribution, i.e., given a masking function M, a masked attribute $M(A)$, label $\mathbb{Y}$, and how to reconstruct $\mathbb{F}(A, \mathbb{Y})$.

The masked attribute $M(A)$ and the masking function M together reveal more information: imagine the following simple example: the attribute *Age* is masked, and the masked data contains 20 young and 80 old individuals. The masking function M imposes further constraints: *Young* $\rightarrow 10 - 45$, *Old* $\rightarrow >$ 45. From this, the reconstructed Age has to satisfy the following: 20 individuals of Age between $10 - 45$, 80 individuals are above 45. Additionally, when the marginal of *Age*, i.e. $Dist(A)$ is known, that imposes further constraints. The challenge, however, is that a prohibitively large number of possible distributions of *Age* satisfies all these constraints, and any of these could be a likely solution. Without any additional information, we assume that these values are equally likely. We formalize this as an optimization problem, i.e., estimate the joint distribution of each predictor and the class label (i.e., *Age* and *Health*) such that the distribution of the predictor (i.e., *Age*) is as uniform as possible within the given range. Still, the reconstructed joint distribution satisfies all constraints.

When the marginal distribution of the attribute is known, i.e., $Dist(A)$ is available, further information is known. For example, for predictor *Age*, there exists 4 individuals with age 10, 12 individuals 17, 4 with 43, and so on.

Constraints. Formally speaking, to be able to estimate the joint distribution of attribute A and the class label $\mathbb{Y}$, i.e., $\mathbb{F}(A, \mathbb{Y})$, the reconstruction process must satisfy several constrains "implied" by the input: the marginal distribution of A (denoted $Dist(A)$; the masking function M together with the resulting masked attribute $M(A)$; the joint distribution of masked A and $\mathbb{Y}$, i.e., $\mathbb{F}(M(A), \mathbb{Y})$; and the total number of records N.

- Type 1: $Dist(A)$ is known, it imposes the following set of constraints: the frequency of each unique domain value a^i of actual A, i.e., $a^i_{\in Dom(A)} f(a^i)$.
- The masking function M reveals, if the masked value of the predictor is $a^{i'}$ what is the likely set of a^i values, i.e., a mapping between $a^{i'} \rightarrow \{a^i\}, \forall a^{i'} \in Dom(M(A))$.
- Type 2: The masking function M and masked attribute $M(A)$ thus imposes an additional set of constraints: the frequency of each unique domain value of the masked attribute A: i.e., $a^{i'}_{\in Dom(M(A))} f(a^{i'})$.
- Type 3: The joint distribution of masked $M(A)$ and $\mathbb{Y}$, i.e., $\mathbb{F}(M(A), \mathbb{Y})$. This joint distribution $\mathbb{F}(M(A), \mathbb{Y})$ imposes another set of constraints, the frequency of each unique i', j, $a^{i'}_{\in Dom(M(A))}, y^j_{\in Dom(\mathbb{Y})} f(a^{i'} y^j)$.
- Finally, the total number of records N is known.

The task is to estimate the joint distribution $\mathbb{F}(A, \mathbb{Y})$ of a predictor A and class label $\mathbb{Y}$ from these above known constraints, such that all constraints are satisfied and $\mathbb{F}(A, \mathbb{Y})$ is as uniform as possible.

To be able to formalize the optimization problem, let $\mathbb{F}_{uni}(A, \mathbb{Y})$ be the joint distribution where $\mathbb{F}_{uni}(A, \mathbb{Y})$ is fully uniform. This distribution has $|Dom(A)|$ number of A values and $|Dom(\mathbb{Y})|$ number of $\mathbb{Y}$ values. Since this distribution is fully uniform, N is equally divided between $|Dom(A)| \times |Dom(\mathbb{Y})|$ number of possibilities.

Example 5. *Let N be 100, $|Dom(A)| = 4$,$|Dom(\mathbb{Y})| = 5$, $|Dom(A)| \times |Dom(\mathbb{Y})| = 20$, then $\mathbb{F}_{uni}(A, \mathbb{Y})$ should initially "look" like:*

	Very Poor	***Poor***	***Moderate***	***Good***	***Very Good***
Age10	*5*	*5*	*5*	*5*	*5*
Age12	*5*	*5*	*5*	*5*	*5*
Age17	*5*	*5*	*5*	*5*	*5*
Age43	*5*	*5*	*5*	*5*	*5*

When no constraints are imposed, the "maximally uninformative" choice for the joint distribution of $(A, \mathbb{Y})$ is the uniform law

$$\mathbb{F}_{uni}(A, \mathbb{Y}) \;=\; \frac{N}{|\mathtt{Dom}(A)|\;\;|\mathtt{Dom}(\mathbb{Y})|} \quad \forall\, a \in \mathtt{Dom}(A),\; y \in \mathtt{Dom}(\mathbb{Y})\,.$$

However, this $\mathbb{F}_{uni}$ will almost certainly violate the marginals (of A and $M(A)$), the joint $(M(A), \mathbb{Y})$ frequencies, and the total count N. To enforce all of those constraints, we must perturb $\mathbb{F}_{uni}$ into some new distribution $\mathbb{F}$ that exactly matches every input statistic. Yet among the infinitely many distributions satisfying those linear constraints, we still wish to remain as "close" as possible to the original uniform prior, i.e. to introduce no additional bias beyond what the constraints demand. The objective function could be formally written as follows:

$$\textbf{Minimize} \quad \text{DIST}\left(\mathbb{F}_{\text{uni}}(A, \mathbb{Y}),\ \mathbb{F}(A, \mathbb{Y})\right) \ \textbf{subject to}$$
$$\text{satisfy constraints } f(a^i), \quad a^i \in \text{Dom}(A)$$
$$\text{satisfy constraints } f(a^{i'}) \quad \forall a^{i'} \in \text{Dom}(M(A))$$
$$\text{satisfy constraints } f\left(a^{i'} y^j\right), \forall a^{i'} \in \text{Dom}(M(A)),\ \forall y^j \in \text{Dom}(\mathbb{Y}).$$

The concept of distance DIST between the two joint distributions can be formalized in multiple ways, depending on the choice of the norm used to measure the difference. It could be ℓ_2 norm $\sqrt{\Sigma_{\forall i,j}[\mathbb{F}_{\text{uni}}(A, \mathbb{Y})_{i,j} - \mathbb{F}(A, \mathbb{Y})_{i,j}]^2}$, which is the Euclidean distance between the matrices, treating them as flattened vectors. This is heavily used in practice due to its nice mathematical properties (differentiable, smooth). On the other hand, ℓ_1 norm, $\Sigma_{\forall i,j}|\mathbb{F}_{\text{uni}}(A, \mathbb{Y})_{i,j} - \mathbb{F}(A, \mathbb{Y})_{i,j}|$, is the sum of absolute differences between corresponding entries of the two distribution. This measure is more robust to outliers compared to ℓ_2. Finally, ℓ_∞ norm $Max_{\forall i,j}|\mathbb{F}_{\text{uni}}(A, \mathbb{Y})_{i,j} - \mathbb{F}(A, \mathbb{Y})_{i,j}|$ captures the worst case deviation between the two joint distributions.

If the problem is formalized in ℓ_2 this gives rise to an Integer Quadratic Programming problem, which is non-convex and NP-hard in general [57]. ℓ_2 error, defined as the sum of squared errors, is the most popular error function because of its key mathematical and statistical properties [13]. It leads to simpler and faster parameter estimation due to the availability of closed-form solutions. This makes it computationally efficient, particularly for large datasets. Statistically, in the presence of Gaussian noise, ℓ_2 corresponds to the maximum likelihood estimator. Another key advantage of ℓ_2 error is that its error function is differentiable, allowing for the application of calculus-based optimization techniques such as gradient descent. We have updated the revision accordingly.

Instead of solving using integer quadratic programming, we adapt iterative proportional fitting (IPF)[28, 30] to solve this problem efficiently. See subroutine `Reconstruction` in Algorithm 2. It starts with a uniform joint distribution $\mathbb{F}_{uni}(A, \mathbb{Y})$. The constraints derived from known masking functions masked 2-D distributions and unmasked 1-D distribution are used to impose a set of row (see Type 1 constraint above) and column (cf. Type 2 and 3 above) constraints on $\mathbb{F}(A, \mathbb{Y})$. The algorithm takes turns and readjusts rows to satisfy all row-related constraints. Then, it returns, takes turns, and readjusts columns to satisfy all constraints. Adjusting row constraints will likely perturb the column constraints (and vice versa). This process is repeated iteratively until $\mathbb{F}(A, \mathbb{Y})$ converges —i.e., the adjusted values closely align with the target marginals within a specified tolerance.

It is known that all the observed values in the 2-D data table are strictly positive (greater than zero) [47], then the existence and uniqueness of the maximum likelihood estimates are ensured, and thus convergence of IPF is guaranteed. For our problem, the underlying joint frequency distribution table satisfies this condition, hence IPF converges in our case.

In general, however, IPF converges to fit marginal totals within a selected level of closeness. The algorithm typically uses criteria such as the maximum number of iterations or a convergence criterion (e.g., largest proportional change in an expected cell count) to determine when to stop iterating. Lowering the tolerance for convergence will require more iterations.

A challenge still remains is that the Subroutine `Reconstruction` can give fractional numbers in the joint distribution, whereas the reconstructed joint distribution must only contain integral values. To address this issue and enforce integrality, we apply a randomized rounding procedure as a post-processing step at the end of the `Reconstruction` subroutine. In this step, each fractional value in the joint distribution is normalized and interpreted as the probability of rounding up to the next integer (or down to the integer before), and independent random trials are used to generate integer entries. This ensures that the final distribution consists entirely of integers while

preserving, in expectation, the original fractional values. Exploring the approximation factor of this randomized rounding [45] is left to future work.

Running Time Analysis. Running time of Subroutine `Reconstruction` is $\mathcal{O}(\#rows \times \#columns)$ of $\mathbb{F}(A, \mathbb{Y})$ per iterations. Let $\mathbb{I}$ denote that time. If the process takes t iterations for convergence, the total running time $\mathbb{T}$ is $\mathcal{O}(t \times \mathbb{I})$.

Case I: 1D Histogram Known When ID histograms are known, Subroutine `Reconstruction` is run, and the input constraints are of Type 1, Type 2, and Type 3.

Case II: 1D Histogram Unknown When the marginal of the predictors are unknown, Subroutine `Reconstruction` is run, but the input constraints are of only Type 2 and Type 3. Clearly, with fewer constraints, the subroutine converges faster.

4.3 Predictive Utility Estimation

Our algorithm to estimate the predictive utility deviation of a masking configuration $\boldsymbol{M}_k$ takes a predictive utility measure ρ as an input, as well as two joint distributions: $\mathbb{F}(M(A), \mathbb{Y})$ is the masked joint distribution, and $\mathbb{F}(A, \mathbb{Y})$ is the reconstructed joint distribution from Subroutine `Reconstruction` for each attribute A. It then computes the predictive utility deviation of $\boldsymbol{M}_k$. The masking configuration with the smallest deviation is then determined as the optimal masking configuration.

Example 6. *Consider the masked joint distribution* $\mathbb{F}(M(A), \mathbb{Y})$ *of* Age, Health *as shown below.*

M(Age)↓ / Health →	***VP***	***P***	***M***	***G***	***VG***	***Row Total***
Young (10–45)	*0*	*0*	*1*	*7*	*12*	*20*
Old (>45)	*15*	*25*	*29*	*9*	*2*	*80*
Column Total	*15*	*25*	*30*	*16*	*14*	*100*

Then for mutual information (MI) as the predictive utility measure, we have $p(\text{Young}) = 0.2$, $p(\text{Old}) = 0.8$, $p(\text{VeryPoor}) = 0.15$, $p(\text{Poor}) = 0.25$, $p(\text{Moderate}) = 0.30$, $p(\text{Good}) = 0.16$, $p(\text{VeryGood}) = 0.14$. *Continuing the process one can show,* $I(\text{M(Age)};\text{Health}) \approx 0.42$

Similarly, its estimated mutual information could be calculated from the reconstructed *Age, Health*. The mutual information deviation is the absolute difference between these two.

5 Evaluation

We experimentally evaluate Aegis on real-world datasets to investigate the (a) overall efficiency and effectiveness of the framework in determining optimal masking configurations; (b) quality of reconstructed joint distributions in the presence and absence of data summaries; (c) how the number of attributes, masking configurations, and dataset size impact the estimation; and (d) scalability of the framework.

Our evaluation demonstrates that Aegis can determine an optimal masking configuration order of magnitude faster while achieving comparable predictive performance for various downstream ML tasks. Our IPF-based reconstruction of joint distributions using both available and unavailable data summaries yields superior quality compared to the sampling-based approach. The parameters (number of attributes, configurations, and rows) do not impact Aegis's effectiveness, and our approach scales nearly linearly with an increase in the number of attributes, configurations, and dataset size. Overall, we found Aegis an efficient and effective middleware framework in data-sharing ecosystems for selecting masking configurations.

5.1 Experimental Setup

5.1.1 Implementation and Setup. We implemented all algorithms in Aegis using Python 3.11.7. All experiments were performed on a server, which is equipped with Intel i9-11900K3.50GHz with 8

Table 2. Overview of benchmark dataset collection and features' statistics. Cat.:Categorical; Num.: Numerical; DS: Domain Size (for Num. Attr.); Min./Max./Avg. #categories (catgr) across Cat. attributes; Min./Max./Avg. range over Num. attributes.

Name	#Rows	#Cat.	#Num.	Min. DS	Max. DS	Avg. DS	Min. #catgr.	Avg. #catgr.	Max. #catgr.	Min. range	Avg. range	Max. range
Air Quality [2] (AQ)	5,000	1	9	4	683	141.78	4	4	4	3.0	183.78	769.00
Customer [5] (HCI)	10,000	10	15	5	10000	2428.40	2	657	3245	4.0	1.43×10^{9}	1.06×10^{10}
Income [1] (IN)	32,561	9	6	16	21648	3673.67	2	11.56	42	15.0	262826.83	1472420.00

CPU cores and 128GB RAM. Additional scalability experiments (Section 5.7) were conducted on server with AMD EPYC 7713 CPU (2.0GHz), 128 cores, 512GB RAM, and a single NVIDIA A100 GPU with 80GB memory.

5.1.2 Datasets. We used three real-world datasets from Kaggle [1, 2, 5] pertaining to environmental monitoring, e-commerce, and socioeconomic domains. The datasets are summarized in Table 2.

- **Air Quality Dataset.** [2] This data set captures environmental and demographic indicators associated with exposure to pollution. The prediction task is a four-class classification problem based on air quality labels.
- **Customer–Hotel Interaction Dataset.** [5] This data set contains features that describe user engagement and hotel characteristics. The target is a binary indicator of booking success.
- **Income Dataset.** [1] This benchmark dataset includes sensitive socioeconomic variables such as age, education level, and employment status. It supports a binary income classification task.

In addition, we also used synthetic datasets to evaluate Aegis 's scalability under controlled variations in dataset size, number of attributes, and number of masking configurations (see Section 5.7).

5.1.3 Data Summaries. For each attribute in the dataset, we construct one-dimensional histograms from which the marginal distribution of attribute values can be derived. We encode each histogram as a dictionary that maps unique discrete values to their corresponding frequencies. Further, to evaluate the impact of available summaries on joint distribution reconstruction quality, we consider two settings.

- **Aegis+1D**: In this setting, the marginal distributions of the individual attributes are available and used during reconstruction (recall case 1; Section 4.2). These histograms provide statistical guidance that improves the accuracy of estimating joint distributions.
- **Aegis-1D**: In this setting, the reconstruction is performed without access to attribute-level statistics, and we assume a uniform prior over each attribute's domain (recall Case 2; Section 4.2).

In the following, unless otherwise stated, we assume access to data summaries as the default setting.

5.1.4 Masking Configurations. To evaluate Aegis under diverse privacy scenarios, we develop a configurable masking configuration generator that applies a range of privacy-preserving transformations to the dataset. The generator supports both generalization-based masking functions (e.g., bucketization, generalization, and blurring) and attribute suppression, with parameterizable controls to vary the extent of information loss. The generator constructs multiple masking configurations by carefully assigning masking functions to attributes while ensuring the target variable remains unmodified. We used the set of 50 masking configurations across both reconstruction settings (with and without access to data summaries) to ensure a consistent basis for comparison for each dataset. In addition, we used privacy-preserving masking configuration based on k-anonymization using [48] (see Section 5.2.3)

5.1.5 Correlation Measures and Downstream Tasks. We evaluated Aegis using correlation measures from Section 3.2 including g_3, mutual information (MI), and Chi-square test (χ^2). For downstream tasks, we consider several ML models, including Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), Stochastic Gradient Descent (SGD), and Naive Bayes (NB).

5.1.6 Competitors.

Baseline. We implemented the traditional approach (recall Section 1) as a baseline that exhaustively applies each masking configuration to first obtain a masked dataset on which a downstream model is trained. This allows us to compute the gold standard accuracy, which we use as our predictive utility metric.

Mascara. We also compared to Mascara [43], a recently proposed middleware system for disclosure-compliant query answering. While Mascara is designed as a SQL rewriting system in the presence of a masking function, we adapt its utility estimator to our setting to determine optimal masking configurations.

Sampling. We also implemented a sampling-based reconstruction in Aegis that generates synthetic records by sampling from original data in accordance with masked attribute ranges. For each attribute in the masked data, we identify the possible set of values and then randomly sample from it with replacement. This procedure produces a reconstructed dataset where the distribution of attribute values matches the masked view, and the associated labels are directly inherited from the sampled original records.

5.2 Overall Efficiency and Effectiveness

5.2.1 End-to-end Efficiency. Figures 5a-5c compares the end-to-end runtime of Aegis against both Sampling, Mascara, and the baseline approach across the three datasets. We consider logistic regression, SVM, and random forest as the downstream tasks.

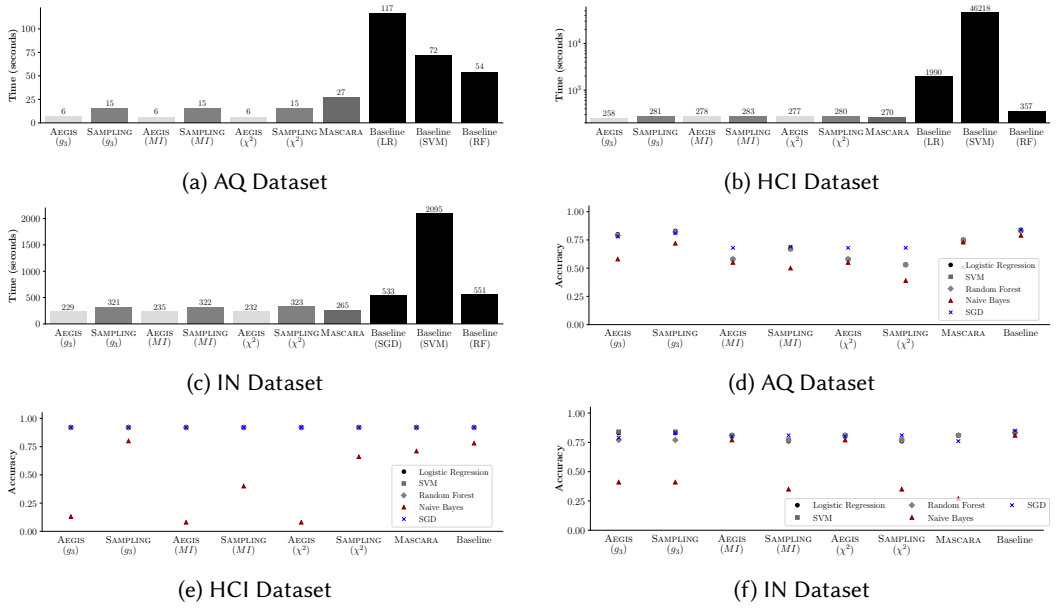
On the smaller AQ dataset (Figure 5a), Aegis completes the reconstruction and predictive-utility evaluation in just over 6*s* for each of the three measures (g_3, MI, and χ^2), yielding a 2.5× speedup over the corresponding Sampling ($\approx$ 15s) and a 4× improvement over Mascara (27*s*). In contrast, the baseline approach incurs substantially higher cost—116*s* for logistic regression, 72*s* for SVM, and 54*s* for random forest—demonstrating that Aegis can evaluate masking configurations nearly an order of magnitude faster.

On the larger HCI and IN datasets (Figures 5b and 5c), Aegis remains competitive with both Sampling and Mascara. At the same time, it outperforms the baseline by multiple orders of magnitude. For example, on HCI, the Aegis is over 7× faster for logistic regression and more than 160× for SVM. For IN Dataset, it is roughly 2× faster for both SGD (533*s*) and random forest (551*s*) and nearly 9× faster than SVM (2,095*s*).

Overall, Aegis provides an efficient framework to determine optimal masking configuration.

5.2.2 End-to-end Effectiveness. Figures 5d-5f compares the predictive utility of masked datasets (AQ, HCI, and IN) using five classifiers (logistic regression, random forest, naive Bayes, and SGD). We compare Aegis (using g_3, MI, and χ^2 measures) with Sampling, Mascara, and the baseline approach that serves as the gold standard.

As general trends, we observe that classifiers trained on masked data (determined by Aegis) achieve accuracies close to the baseline. In particular, using g_3 error as a correlation measure consistently yields the best masking configuration, with an accuracy loss of less than 3–7% for all classifiers except Naive Bayes (see discussion below). On average, Aegis achieves accuracy comparable to the Sampling approach and outperforms Mascara, which adapts KL-divergence as



(a) AQ Dataset (b) HCI Dataset

(c) IN Dataset (d) AQ Dataset

(e) HCI Dataset (f) IN Dataset

Fig. 5. Overall efficiency (a)-(c) and effectiveness (d)-(f) of AEGIS.

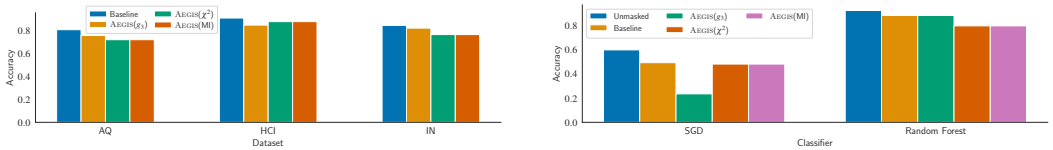
a utility metric. In addition, except for Naive Bayes, all classifiers had an average accuracy of 0.69, 0.92, and 0.8 for AQ, HCI, and IN datasets.

More specifically, on the AQ dataset (Figure 5d), we observe that for Logistic Regression (circle marker), using g_3 error as a correlation measure yielded the best accuracy for both AEGIS (0.81) and SAMPLING (0.82), compared to the baseline (0.84). For other correlation measures, both AEGIS and SAMPLING performed up to 30% worse than the baseline, with SAMPLING using χ^2 yielding the lowest accuracy of 0.53. Similar trends were observed for SVM (square marker), Random Forest (diamond marker), and SGD (cross marker). A notable exception was Naive Bayes (triangle marker), where MASCARA outperformed other approaches, achieving an accuracy of 0.73 compared to the baseline's 0.79. In general, Naive Bayes performs poorly due to its conditional independence assumption which does not align with preserving the correlation between individual features and the label while masking.

On the HCI dataset (Figure 5e), we observed that all methods performed similarly well, except again when using Naive Bayes.

On the Income dataset (Figure 5f), AEGIS using g_3 achieved an accuracy of 0.83, comparable to the baseline, for both Logistic Regression (circle marker) and SVM (square marker). For Random Forest (diamond marker), AEGIS with MI and χ^2 correlation measures yielded the best masked datasets, achieving an accuracy of 0.81 compared to the baseline's 0.83. When using SGD, both AEGIS and SAMPLING attained similar accuracies of 0.81, compared to the baseline's 0.85.

Additionally, we compared the effectiveness of AEGIS with a Neural Network classifier. We use a fully connected 4-layer MLP classifier with two hidden layers of 100 neurons each and ReLU activation, trained using the Adam optimizer for up to 200 epochs. The network processes data through one-hot encoding for categorical features and evaluates performance using a 70-30 train-test split. The results are shown in Figure 6a. We observe that AEGIS (with g_3) achieves comparable



(a) Effectiveness with Neural Network classifier

(b) Effectiveness on k-anon.-based configurations.

Fig. 6. Effectiveness of Aegis (a) with Neural Network classifier and (b) on k-anonymization-based configurations.

performance to baseline: the accuracy was 6% and 2% less for AQ and IN datasets, resp.; and was 3% less for HCI dataset when using with χ^2 and MI.

We note that, although, out initial experiments with Neural networks are encouraging, they require more extensive evaluation due to their sensitivity to architectural choices, hyperparameters, and training dynamics, which can significantly impact performance. We leave this as an important future work.

Overall, Aegis offers a model-agnostic framework to effectively determine a masking configuration that preserves key feature-label correlations required for high predictive utility, without ever training the downstream models during configuration selection.

5.2.3 Effectiveness on k-anonymization-based configurations. In these set of experiments, we studied the effectiveness of Aegis when using real-world masking configurations based on k-anonymization (setting $k = 5$). We used the AQ dataset and two classifiers: Stochastic Gradient Decent (SGD), and Random Forest (RF). Our goal was to evaluate the accuracy of the masked AQ dataset determined by Aegis and compare it the masked dataset determined by the baseline approach. As reference, we also evaluate the accuracy of unmasked dataset. The results are shown in Figure 6b.

We observe that Aegis achieves comparable performance for the two classifiers: the accuracy was 2% less for SGD (with χ^2) and was at par for RF (with g_3). These results are consistent with our observations when using synthetic masking configurations above as Aegis, given a set of masking configurations as input, aims to determine the one that best preserves feature label correlations for a given correlation measure.

5.3 Quality of Joint Distribution Reconstruction

In these experiments, we evaluate the quality of the reconstructed joint distributions produced by Aegis +1D and Aegis-1D. In both scenarios, we measure reconstruction fidelity using *Total Variation Distance (TVD)*, a standard divergence metric that quantifies the difference between two probability distributions. Formally, given two discrete distributions P and Q over the same domain Ω, TVD is defined as:

$$\mathrm{TVD}(P, Q) = \frac{1}{2} \sum_{\omega \in \Omega} |P(\omega) - Q(\omega)|$$

This value represents the maximum shift in probability mass required to transform one distribution into the other. Lower TVD values indicate a closer match between the reconstructed and ground truth distributions.

The results are shown in Figure 7, where for each of the three datasets, we report the box plots over 50 different configurations (recall Algorithm 2) for both Aegis variants and for the sampling-based reconstruction. In every case, Aegis achieves substantially lower TVD; the median TVD for Aegis+1D lies between 0.45 and 0.50 across datasets, and even for Aegis-1D, the median remains below 0.55. In contrast, the sampling-based reconstruction exhibits median errors of 0.80-0.85

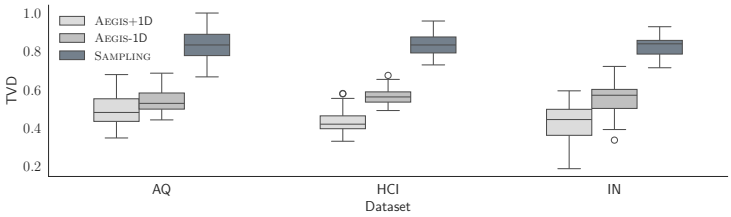
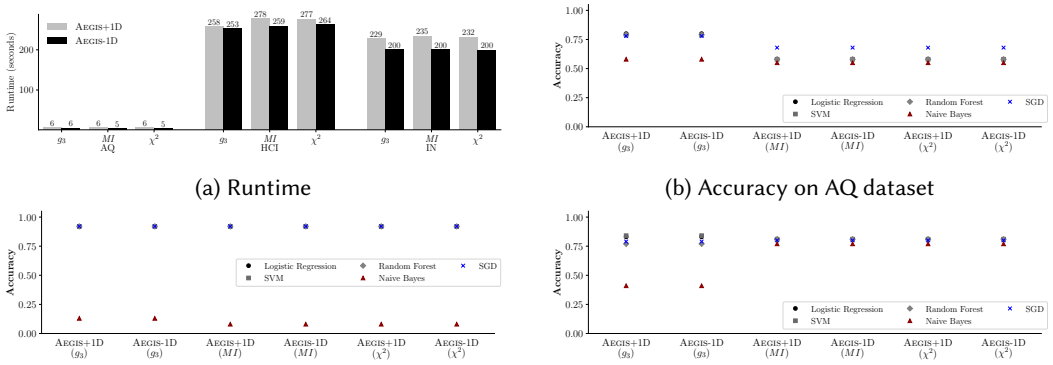


Fig. 7. Quality of joint distribution reconstruction across 50 masking configurations.



(a) Runtime

(b) Accuracy on AQ dataset

(c) Accuracy on HCI dataset

(d) Accuracy on IN dataset

Fig. 8. Impact of availability of data summaries on accuracy of downstream tasks.

and frequently reaches TVD values near 1.0, indicating almost complete distortion of the joint distribution.

Overall, these results demonstrate that AEGIS's reconstruction using Iterative Proportional Fitting and modeling constraints derived from masking functions and data summaries, when available, yields more accurate reconstructed joint distributions than the sampling-based approach. In addition, this underscores AEGIS's capability in producing model-agnostic utility estimates based on joint (feature-label) distributions for several downstream tasks in data-sharing ecosystems.

5.4 Impact of Availability of Data Summaries

We now study how the availability of statistical summaries affects the performance in terms of runtime and downstream predictive utility (accuracy). The results are shown in Figure 8, where we compare AEGIS+1D and AEGIS-1D.

We first discuss the runtime experiments (see Figure 8a) for AEGIS across the three model-agnostic measures and the three datasets. We observe that on the smaller AQ dataset, both variants complete in under 6.5s. As the dataset size grows, the runtimes increase to roughly 230 − 280s, where AEGIS-1D has a slightly better (up to 15%) efficiency. This is expected as IPF converges faster with fewer constraints to satisfy.

Figures 8b-8d show that the downstream predictive accuracy of four off-the-shelf classifiers (logistic regression, SVM, random forest, naive Bayes, and SGD) when trained on the data masked according to the configuration selected by AEGIS+1D and AEGIS-1D, for each of the three measures. In each case, AEGIS+1D yields higher accuracy than AEGIS-1D. For example, on the AQ dataset (Figure 8b), SVM accuracy improves by up to 2%. We also observe similar improvements for HCI

and IN datasets (Figures 8c and 8d) across all measures. Naive Bayes, which is most sensitive to distributional errors, benefits most from the availability of data summaries.

Overall, we conclude that incorporating data summaries adds negligible overhead and helps determine better masking configurations. Additionally, even without data summaries, Aegis still provides a robust framework to determine effective masking configurations.

5.5 Impact of Parameters

In this set of experiments, we evaluate the robustness of Aegis to changes in three key parameters: the number of attributes (Figures 9a-9c), the number of masking configurations (Figures 9d-9f), and the number of rows in the dataset (Figures 9g-9i). We measure the impact of these parameters on the three model-agnostic measures g_3, mutual information (MI), and Chi-squared statistic (χ^2). The Δ on the y-axis (note the log scale) captures the change in predictive utility due to changes in these parameters.

5.5.1 Impact of Number of Attributes. We varied the number of attributes from 25% to 100% of the dataset's features. The results are shown in Figures 9a-9c. Across all datasets and all three measures, we observe that the change in utility scores (Δ_{g_3}, Δ_{MI}, and Δ_{χ^2}) remains consistently small (below 10^{-8} for Δ_{g_3}, 10^{-6} for Δ_{MI}, and even lower for Δ_{χ^2}). The results show that Aegis remains stable in computing the predictive utility when the dimensionality of the data varies significantly.

5.5.2 Impact of Number of Configurations. Figures 9d-9f show the effect of changing the number of masking configurations from 20 to 50. We observe negligible differences in the utility scores across all three datasets. For example, Δ_{χ^2} remains below 10^{-7}, and Δ_{g_3} and Δ_{MI} show only minor fluctuations (on the order of $10^{-2} - 10^{-6}$). The experiments demonstrate that Aegis is not sensitive to the exact number of candidate masking configurations.

5.5.3 Impact of Number of Rows. Lastly, we evaluate the effect of increasing the sample size from 25% to 100% of the dataset. The results are shown in Figures 9g-9i. We observe that the changes in the utility metrics are negligible, particularly for Δ_{MI} and Δ_{χ^2}, where the differences are on the order of 10^{-10} or less. Even for Δ_{g3}, which shows the largest variation, the overall change remains small and bounded.

In sum, the impact of varying attributes, configurations, and rows is minimal, both in absolute terms and relative to the log. y-axis scale. The largest observed change (Δ_{g3} in the AQ dataset) remains below 10^{-1}, while most other variations are several orders of magnitude smaller. This demonstrates the Aegis's robustness and stability.

5.6 Impact of Generalization and Domain Size

In this set of experiments, we evaluate the impact of generalization and domain size on the robustness of Aegis 's joint–distribution reconstruction. We construct a synthetic dataset comprising numerical feature drawn from a uniform distribution and a binary class label, with 10 000 records. To study generalization, in each run we generate configurations with increasing bucket sizes, thereby varying the degree of value coarsening. Additionally, we consider runs with varying the feature's domain size by changing its number of distinct values. This design allows us to isolate and quantify how both generalization (via bucket size) and domain cardinality affect reconstruction quality. The results are shown in Figure 10.

We first discuss the impact on reconstruction time. Figure 10a shows that reconstruction time decreases as the bucket (bin) size increases with domain fixed at 200. This is because when the feature domain is partitioned into very fine-grained buckets (e.g., size = 2), IPF must reconcile a larger number of probability cells, leading to runtimes on the order of 90s. As the bucket size grows

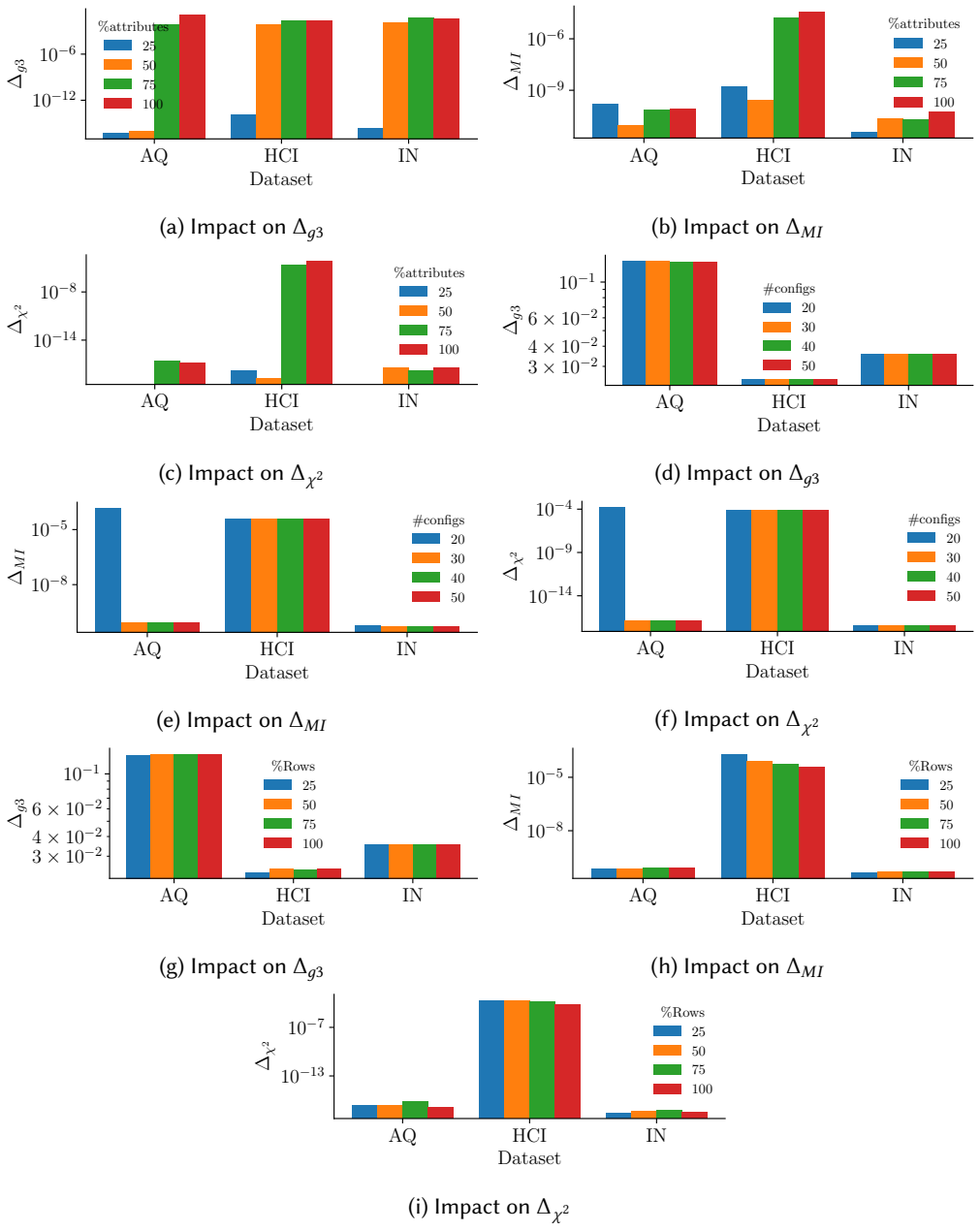
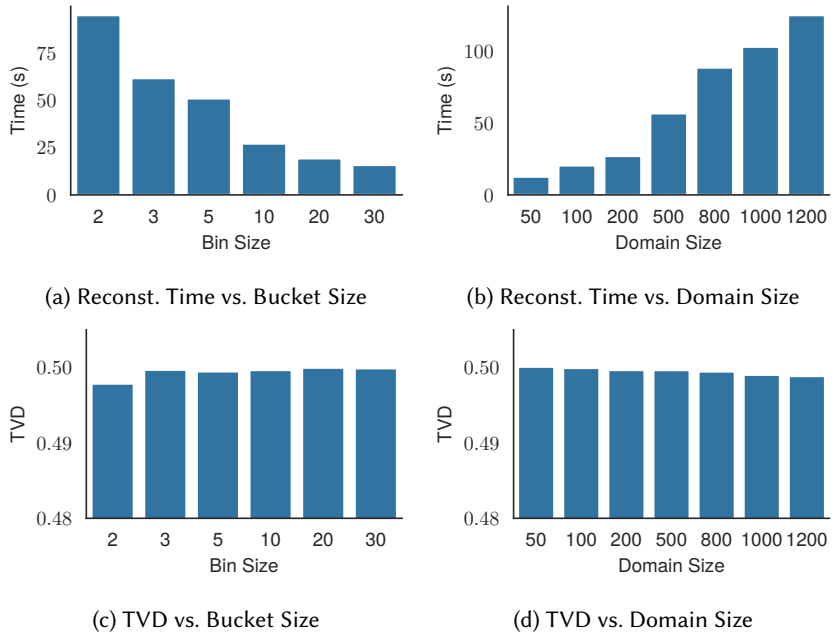


Fig. 9. Impact of parameters (# attributes (a)-(c), # configurations (d)-(e), and # rows (g)-(i)) on model-agnostic measures.

to 5 and then 10, the number of cells falls, and reconstruction time drops to approximately 50s and 25s, respectively. Beyond a bucket size of 20, the dimensionality of the contingency table is sufficiently low, and reconstruction requires much less time. Figure 10b shows the complementary effect of domain cardinality with bin size fixed at 10. With just 50 distinct values, reconstruction



(a) Reconst. Time vs. Bucket Size

(b) Reconst. Time vs. Domain Size

(c) TVD vs. Bucket Size

(d) TVD vs. Domain Size

Fig. 10. Impact of generalization and domain size on joint distribution reconstruction.

completes in under 15s; however, as we enlarge the domain to 500 values, time increases to 55s, and exceeds 120s when the domain reaches 1200 values. This nearly linear growth in runtime reflects the cost of performing IPF over ever-larger joint-distribution tables.

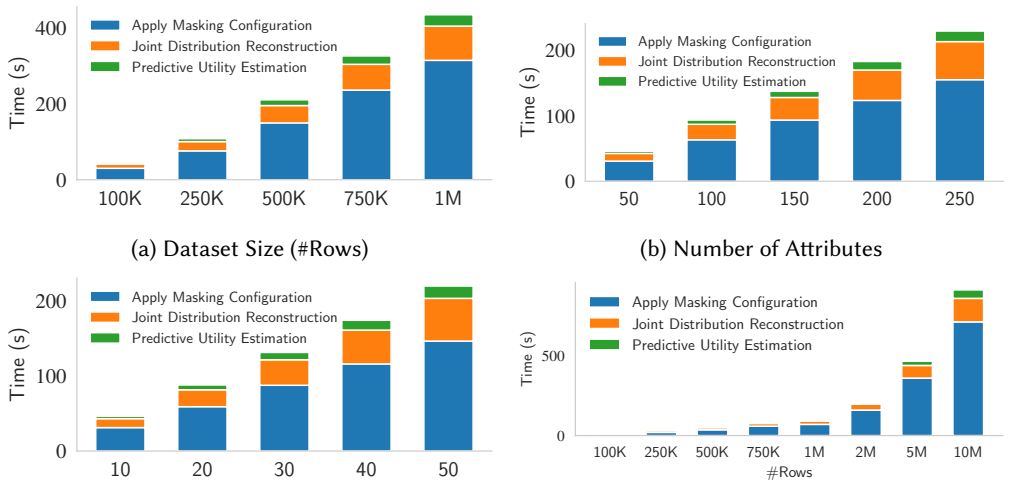
We also evaluated the impact on reconstruction quality using TVD (see above) in Figures 10c and 10d. We observe that (TVD) increases slightly as the bin size grows—indicating a marginal degradation in reconstruction fidelity when the data are more coarsely generalized. Conversely, TVD decreases as the domain size expands, reflecting improved fidelity when more distinct values are available. These TVD mirror our observations on the AQ, HCI, and IN datasets (see Section 5.3), where TVD values remain below 0.5.

Overall, increasing bin size reduces reconstruction runtime but slightly increases TVD, whereas increasing domain size raises runtime while marginally lowering TVD.

5.7 Framework Efficiency

In our last set of experiments, we study the scalability of our approach by measuring the runtime with varying numbers of rows, attributes, and masking configurations. We used a synthetic dataset (as described earlier) for these experiments, for which the results are shown in Figures 11a-11c. Each figure shows the runtime breakdown for applying the masking configuration, reconstruction, and utility estimation.

As Figure 11a shows, the runtime increases steadily with the number of rows (from 100K to 1M). We observe that applying masking configuration to compute the masked joint distribution dominates the runtime, with reconstruction and utility estimation adding less overhead. For instance, for 1M rows, the total runtime exceeds 400*s* with computing masked joint distribution accounting for more than 50%. This linear growth is expected as more rows increase the masking operation and reconstruction routing (recall Section 4.2).



(a) Dataset Size (#Rows)

(b) Number of Attributes

(c) Number of Configurations

(d) Additional scalability experiment w.r.t. #rows.

Fig. 11. Scalability Experiments

In Figure 11b, we vary the number of attributes from 50 to 250. We observe that runtime increases steadily. This is again because of the increasing cost of applying the masking configurations, which depends on the number of attributes.

We also examine the scalability with respect to the number of masking configurations (from 10-50; Figure 11c). We observe that runtime grows approximately linearly across this range, again dominated by the masking. The reconstruction cost increases linearly, although more slowly, while the utility metric evaluation remains consistently low (less than 10%).

Lastly, we also evaluated scalability on larger datasets upto 10M rows. The results are shown in Figure 11d. We note up front that these large-scale experiments were run on a more powerful hardware configuration (with GPU acceleration) than the CPU-only setup described in Section 5.1; on the CPU-only we could not scale beyond 1M rows in a reasonable time. From Figure 11d, we see similar trends in that time grows roughly linearly with the number of records. Applying the masking configuration dominates the cost, rising from under 20s at 100K rows to roughly 740s at 10M rows. Joint distribution reconstruction remains in the the order of 15–20%—and predictive utility estimation contributes only a few seconds (up to 53s) even at the largest scale. Together, these results confirm that Aegis can handle large datasets when GPU resources are available. For even larger dataset (with 1B rows or more), runtimes would clearly exceed practical budget for selecting configurations. As future work, we plan to parallelize algorithms within Aegis to further reduce end-to-end runtime and enable fast configuration selection on truly massive datasets.

Overall, Aegis demonstrates scalable performance across dataset size, dimensionality, and a number of masking configurations.

6 Related Work

We now discuss how ideas presented in this paper relate to prior works, which can be broadly categorized into:

Data Masking. Data Masking is a well-studied topic in the privacy literature. Beyond modifying or suppressing identifiers, anonymization methods including k-Anonymity, l-Diversity, and

t-Closeness [33, 36, 52] are based on the concept of equivalence classes and group records to satisfy privacy thresholds. The original data can be anonymized using these techniques, each distorting correlations to varying degrees. Differential privacy [20] based masking techniques offer strong privacy guarantees. However, noise often degrades feature-label dependencies, leading to utility-aware variants [35, 37]. Graph and relational anonymization techniques [18, 25] have also been studied to protect link structure [39, 53]. In general, masking often involves masking functions like generalization, suppression, perturbation, tokenization, and shuffling to obscure raw values while aiming to retain aggregate statistics. Our work complements all of these masking techniques—Aegis can incorporate any k-anonymity, differential privacy, structural anonymization, or masking operation to evaluate and select the configuration that best preserves predictive utility in a model-agnostic way.

Data Correlation. Maintaining feature–label and inter-feature dependencies is critical for downstream analytical tasks. Key measures include the Chi-Square Test for categorical independence; Mutual Information for capturing linear and nonlinear associations; Functional-Dependency Errors (G1, G2, G3) [29] for quantifying violations of $X \rightarrow Y$; Correlation Matrices and Partial Correlation, for multivariate and conditional relationships; KL/Jensen-Shannon divergence measures, for distributional shifts. These metrics evaluate the effect of a given transformation but do not guide the *selection* of transformations that minimize correlation distortion. Aegis can incorporate model agnostic measures based on joint distributions, which allows selecting the masking configuration that best preserves predictive utility without ever training a downstream model. An interesting direction of future work is to extend our framework of other correlation measures as well.

Balancing Privacy and Utility. Optimizing data utility under privacy requirement is a major challenge in using anonymized data for ML [17, 34, 50]. [46] delves into the fundamental tradeoff between privacy and utility when publishing data for counting queries. [25] tackles the problem of anonymizing data with the least information loss before releasing it. [53] focuses on publishing sparse, multidimensional data, like web search logs. [51] quantitatively evaluated the trade-off between privacy protection and data utility of synthetic data than traditional anonymization methods. [37] advocates adopting differential privacy frameworks for publishing aggregated data. [35] have studied balancing privacy protection and data utility when publishing anonymized microdata. In the context of our setting, Aegis is not itself a privacy mechanism and treats privacy as binary, i.e., given a set of privacy-guaranteed candidate configurations, it selects the one (without access to raw dataset and using available summaries or none) that maximizes feature–label correlation under the chosen correlation metric.

Data Reconstruction. Reconstruction of distribution has also been studied in various context. For example, Mosaic [41] utilizes IPF to re-weight samples for answering population queries when the sampling mechanism is unknown. [9] address the problem of signal reconstruction in databases and leverage IPF. When initial solutions yield negative values. [49] employs IPF within an entropy-maximization framework to estimate co-occurrence probabilities for keyword-query expansions. [38] presents a framework to address the empty-answer problem in databases by estimating joint distributions from marginal data, facilitating the generation of probable answers when exact matches are absent. In our work, we leverage IPF to reconstruct joint distributions for computing predictive utility deviation. The key difference lies in modeling constraints that are specific to our setting.

Data-sharing Ecosystems. Finally, our work also intersects with research on data-sharing ecosystems, e.g. [3, 4, 6–8, 11, 12, 19, 24, 40, 54]. Whereas those efforts predominantly explore the underlying infrastructure, system architectures, and policy trade-offs that enable secure, scalable data

exchange, Aegis offers a complementary, task-agnostic framework that allows data publishers and providers to share anonymized datasets without sacrificing predictive utility.

7 Conclusion

We introduced Aegis, a middleware framework designed to identify optimal masking configurations for machine learning datasets to preserve their predictive utility without requiring the ML pipeline execution. In data-sharing ecosystems where raw data is anonymized and inaccessible, Aegis serves as an "advisor" to data providers, recommending masking strategies that maximize utility in a task-agnostic manner. The framework builds on the understanding that predictive utility could be estimated using model-agnostic measures like correlation or association. Aegis is compatible with any utility estimator that relies on joint distribution estimation and is instantiated with well-established techniques like Mutual Information, Chi-Square, and functional dependency-based measures. We studied computational challenges in designing Aegis. A central computational challenge in Aegis lies in estimating joint distributions. We formalized this as a constrained optimization problem and adapted Iterative Proportional Fitting (IPF) to solve it efficiently. We evaluated Aegis on real-world datasets and downstream models, demonstrating that it achieves comparable predictive accuracy to exhaustive pipeline while offering significant speed-ups—often by order of magnitude. Our experiments also showed that different algorithms within Aegis are highly effective and scalable.

As ongoing work, we explore how Aegis can integrate correlation measures beyond those based on attribute-label joint distributions. In the future, we aim to design masking functions that jointly preserve predictive utility and privacy, explore how masking strategies interact with Neural Networks, LLMs or foundational models when applied to tabular data, and parallelize Aegis to scale to massive datasets.

Acknowledgments

This work of Kaustubh Beedkar is supported by Anusandhan National Research Foundation project ANRF/ECRG/2024/005781/ENS. The work of Fatemeh Ramezani Khozestani, Sohrab Namazi Nia, and Senjuti Basu Roy are supported by the following funding agencies: (1) National Science Foundation award number(s): 1942913, 1814595 (2) Office of Naval Research award number(s): N000141812838, N000142112966, N000142412466.

References

[1] [n. d.]. Adult Census Income Dataset. https://www.kaggle.com/datasets/priyamchoksi/adult-census-income-dataset.
[2] [n. d.]. Air Quality and Pollution Assessment. https://www.kaggle.com/datasets/mujtabamatin/air-quality-and-pollution-assessment.
[3] [n. d.]. AWS Data Exchange. https://aws.amazon.com/data-exchange/.
[4] [n. d.]. Dawex. https://www.dawex.com/en/data-exchange-solution/.
[5] [n. d.]. Expedia Hotel Recommendations. https://www.kaggle.com/c/expedia-hotel-recommendations/data.
[6] [n. d.]. Gaia-X. https://gaia-x.eu/gaia-x-framework/.
[7] [n. d.]. Indian Urban Data Exchange. https://iudx.org.in/.
[8] [n. d.]. Snowflake Data Market Place. https://www.snowflake.com/en/product/features/marketplace/.
[9] Abolfazl Asudeh, Azade Nazi, Jees Augustine, Saravanan Thirumuruganathan, Nan Zhang, Gautam Das, and Divesh Srivastava. 2018. Leveraging similarity joins for signal reconstruction. *Proc. VLDB Endow.* 11, 10 (June 2018), 1276–1288. https://doi.org/10.14778/3231751.3231752
[10] Roberto Battiti. 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks* 5, 4 (1994), 537–550.
[11] Lennart Behme, Sainyam Galhotra, Kaustubh Beedkar, and Volker Markl. 2024. Fainder: A Fast and Accurate Index for Distribution-Aware Dataset Search. *Proc. VLDB Endow.* 17, 11 (July 2024), 3269–3282. https://doi.org/10.14778/3681954.3681999

[12] Lennart Behme, Leonard Geißler, Pratham Agrawal, Emil Badura, Benjamin Ueber, Kaustubh Beedkar, and Volker Markl. 2025. Finding What You're Looking For: A Distribution-Aware Dataset Search Engine in Action. In *Companion of the 2025 International Conference on Management of Data* (Berlin, Germany) *(SIGMOD/PODS '25)*. Association for Computing Machinery, New York, NY, USA, 39–42. https://doi.org/10.1145/3722212.3725104

[13] Åke Björck. 1990. Least squares methods. *Handbook of numerical analysis* 1 (1990), 465–652.

[14] California Legislature. 2020. California Consumer Privacy Act (CCPA). https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?lawCode=CIV&division=3.&title=1.81.5.&part=4.&chapter=&article=. https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?lawCode=CIV&division=3.&title=1.81.5.&part=4.&chapter=&article= California Civil Code §§ 1798.100 - 1798.199.

[15] Xingguang Chen and Sibo Wang. 2021. Efficient approximate algorithms for empirical entropy and mutual information. In *Proceedings of the 2021 ACM SIGMOD international conference on Management of data*. 274–286.

[16] William G Cochran. 1952. The $\chi 2$ test of goodness of fit. *The Annals of mathematical statistics* (1952), 315–345.

[17] Graham Cormode, Cecilia M Procopiuc, Entong Shen, Divesh Srivastava, and Ting Yu. 2013. Empirical privacy and empirical utility of anonymized data. In *2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW)*. IEEE, 77–82.

[18] Graham Cormode, Divesh Srivastava, Ting Yu, and Qing Zhang. 2008. Anonymizing bipartite graph data using safe groupings. *Proc. VLDB Endow.* 1, 1 (Aug. 2008), 833–844. https://doi.org/10.14778/1453856.1453947

[19] DataSHIELD Consortium. 2023. DataSHIELD - Secure Bioscience Collaboration Without Data Sharing. https://www.datashield.org/. Accessed: 2025-04-17.

[20] Cynthia Dwork. 2006. Differential privacy. In *International colloquium on automata, languages, and programming*. Springer, 1–12.

[21] Khaled El Emam and Fida Kamal Dankar. 2008. Protecting privacy using k-anonymity. *Journal of the American Medical Informatics Association* 15, 5 (2008), 627–637.

[22] European Union. 2016. General Data Protection Regulation (GDPR). https://eur-lex.europa.eu/eli/reg/2016/679/oj. https://eur-lex.europa.eu/eli/reg/2016/679/oj EU Regulation 2016/679.

[23] François Fleuret. 2004. Fast binary feature selection with conditional mutual information. *Journal of Machine learning research* 5, 9 (2004).

[24] GAIA-X Initiative. 2023. GAIA-X: A Federated and Secure Data Infrastructure for Europe. https://www.gaia-x.eu/. Accessed: 2025-04-17.

[25] Gabriel Ghinita, Panagiotis Karras, Panos Kalnis, and Nikos Mamoulis. 2007. Fast data anonymization with low information loss. In *Proceedings of the 33rd international conference on Very large data bases*. 758–769.

[26] Jiawei Han, Micheline Kamber, and Jian Pei. 2012. Data mining concepts and techniques third edition. *University of Illinois at Urbana-Champaign Micheline Kamber Jian Pei Simon Fraser University* (2012).

[27] Jinjie Huang, Yunze Cai, and Xiaoming Xu. 2007. A hybrid genetic algorithm for feature selection wrapper based on mutual information. *Pattern Recognition Letters* 28, 13 (2007), 1825–1844. https://doi.org/10.1016/j.patrec.2007.05.011

[28] Martin Idel. 2016. A review of matrix scaling and Sinkhorn's normal form for matrices and positive maps. *arXiv preprint arXiv:1609.06349* (2016).

[29] Jyrki Kivinen and Heikki Mannila. 1995. Approximate inference of functional dependencies from relations. *Theoretical Computer Science* 149, 1 (1995), 129–149. https://doi.org/10.1016/0304-3975(95)00028-U Fourth International Conference on Database Theory (ICDT '92).

[30] J Kruithof. 1937. Calculation of telephone traffic. *De Ingenieur* 52, 8 (1937), E15–E25.

[31] Nojun Kwak and Chong-Ho Choi. 2002. Input feature selection by mutual information based on Parzen window. *IEEE transactions on pattern analysis and machine intelligence* 24, 12 (2002), 1667–1671.

[32] Marie Le Guilly, Jean-Marc Petit, and Vasile-Marian Scuturici. 2020. Evaluating classification feasibility using functional dependencies. *Transactions on Large-Scale Data-and Knowledge-Centered Systems XLIV: Special Issue on Data Management–Principles, Technologies, and Applications* (2020), 132–159.

[33] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. 2007. t-Closeness: Privacy Beyond k-Anonymity and ℓ-Diversity. *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE)* (2007).

[34] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. 2022. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th international conference on data engineering (ICDE)*. IEEE, 965–978.

[35] Tiancheng Li and Ninghui Li. 2009. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 517–526.

[36] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. 2006. L-diversity: Privacy Beyond k-Anonymity. *Proceedings of the 22nd International Conference on Data Engineering (ICDE)* (2006).

[37] Gerome Miklau. 2022. Negotiating Privacy/Utility Trade-Offs under differential privacy. *Santa Clara, CA* (2022).

[38] Davide Mottin, Alice Marascu, Senjuti Basu Roy, Gautam Das, Themis Palpanas, and Yannis Velegrakis. 2013. A probabilistic optimization framework for the empty-answer problem. *Proc. VLDB Endow.* 6, 14 (Sept. 2013), 1762–1773. https://doi.org/10.14778/2556549.2556560

[39] M. Ercan Nergiz, Maurizio Atzori, and Christopher W. Clifton. 2007. Hiding the Presence of Individuals from Shared Databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. 665–676. https://doi.org/10.1145/1247480.1247554

[40] OpenMined Community. 2023. OpenMined - Building Technology for Privacy-Preserving AI. https://www.openmined.org/. Accessed: 2025-04-17.

[41] Laurel J. Orr, Samuel K. Ainsworth, Kevin G. Jamieson, Walter Cai, Magdalena Balazinska, and Dan Suciu. 2020. Mosaic: A Sample-Based Database System for Open World Query Processing. In *10th Conference on Innovative Data Systems Research, CIDR 2020, Amsterdam, The Netherlands, January 12-15, 2020, Online Proceedings*. www.cidrdb.org. http://cidrdb.org/cidr2020/papers/p26-Orr-cidr20.pdf

[42] Cláudia Pascoal, M Rosário Oliveira, António Pacheco, and Rui Valadas. 2017. Theoretical evaluation of feature selection methods based on mutual information. *Neurocomputing* 226 (2017), 168–181.

[43] Rudi Poepsel-Lemaitre, Kaustubh Beedkar, and Volker Markl. 2024. Disclosure-Compliant Query Answering. *Proc. ACM Manag. Data* 2, 6, Article 233 (2024), 28 pages.

[44] Insurance Portability and Accountability Act. 2012. Guidance regarding methods for de-identification of protected health information in accordance with the health insurance portability and accountability act (HIPAA) privacy rule. *Human Health Services: Washington, DC, USA* (2012).

[45] Prabhakar Raghavan and Clark D Tompson. 1987. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica* 7, 4 (1987), 365–374.

[46] Vibhor Rastogi, Dan Suciu, and Sungho Hong. 2007. The boundary between privacy and utility in data publishing. In *Proceedings of the 33rd international conference on Very large data bases*. 531–542.

[47] Ludger Rüschendorf. 1995. Convergence of the iterative proportional fitting procedure. *The Annals of Statistics* (1995), 1160–1174.

[48] Judith Sáinz-Pardo Díaz and Álvaro López García. 2024. An Open Source Python Library for Anonymizing Sensitive Data. *Scientific data* 11, 1 (2024), 1289.

[49] Nikos Sarkas, Nilesh Bansal, Gautam Das, and Nick Koudas. 2009. Measure-driven keyword-query expansion. *Proc. VLDB Endow.* 2, 1 (Aug. 2009), 121–132. https://doi.org/10.14778/1687627.1687642

[50] Jordi Soria-Comas, Josep Domingo-Ferrer, David Sánchez, and Sergio Martínez. 2014. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. *The VLDB Journal* 23, 5 (2014), 771–794.

[51] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. 2022. Synthetic data–anonymisation groundhog day. In *31st USENIX Security Symposium (USENIX Security 22)*. 1451–1468.

[52] Latanya Sweeney. 2002. k-anonymity: A model for protecting privacy. *International journal of uncertainty, fuzziness and knowledge-based systems* 10, 05 (2002), 557–570.

[53] Manolis Terrovitis, John Liagouris, Nikos Mamoulis, and Spiros Skiadopoulos. 2012. Privacy preservation by disassociation. *VLDB* (2012).

[54] Jonas Traub, Zoi Kaoudi, Jorge-Arnulfo Quiané-Ruiz, and Volker Markl. 2021. Agora: Bringing Together Datasets, Algorithms, Models and More in a Unified Ecosystem [Vision]. *SIGMOD Rec.* 49, 4 (March 2021), 6–11. https://doi.org/10.1145/3456859.3456861

[55] U.S. Department of Health and Human Services. 1996. Health Insurance Portability and Accountability Act (HIPAA). https://www.hhs.gov/hipaa/index.html. https://www.hhs.gov/hipaa/index.html HIPAA Privacy, Security, and Breach Notification Rules.

[56] Jorge R Vergara and Pablo A Estévez. 2014. A review of feature selection methods based on mutual information. *Neural computing and applications* 24 (2014), 175–186.

[57] Laurence A Wolsey and George L Nemhauser. 1999. *Integer and combinatorial optimization*. John Wiley & Sons.

Received April 2025; revised July 2025; accepted August 2025