# PreFair: Privately Generating Justifiably Fair Synthetic Data

David Pujol
Duke University
dpujol@cs.duke.edu

Amir Gilad
Duke University
agilad@cs.duke.edu

Ashwin Machanavajjhala
Duke University
ashwin@cs.duke.edu

## ABSTRACT

When a database is protected by Differential Privacy (DP), its usability is limited in scope. In this scenario, generating a synthetic version of the data that mimics the properties of the private data allows users to perform any operation on the synthetic data, while maintaining the privacy of the original data. Therefore, multiple works have been devoted to devising systems for DP synthetic data generation. However, such systems may preserve or even magnify properties of the data that make it unfair, rendering the synthetic data unfit for use. In this work, we present PreFair, a system that allows for DP fair synthetic data generation. PreFair extends the state-of-the-art DP data generation mechanisms by incorporating a causal fairness criterion that ensures fair synthetic data. We adapt the notion of justifiable fairness to fit the synthetic data generation scenario. We further study the problem of generating DP fair synthetic data, showing its intractability and designing algorithms that are optimal under certain assumptions. We also provide an extensive experimental evaluation, showing that PreFair generates synthetic data that is significantly fairer than the data generated by leading DP data generation mechanisms, while remaining faithful to the private data.

## 1 INTRODUCTION

Among the various privacy definitions that have been explored, Differential Privacy (DP) [13] has emerged as the most reliable. DP allows users to get answers to queries and even train Machine Learning models over private data by augmenting the true answers with noise that cloaks information about individuals in the data. Indeed, multiple works have focused on answering queries over private data [20, 22, 26, 29, 33] and training models over such data [2, 39]. However, performing queries or training models directly on the private data has several prominent drawbacks. First, the class of allowed operations is limited. For example, only aggregate queries and specific techniques of training are allowed. Second, the data itself cannot be shared or used to explain the results and verify them without spending more precious privacy budget.

These limitations and others have driven the approach of generating synthetic data under DP [3, 7, 19, 20, 23, 30–32, 34, 35, 45, 49, 52, 55, 60]. In general, these systems often work in two main steps as depicted in Figure 1. First, they measure some set of summary statistics and generate some model to represent the data that is consistent with the measured statistics. This step is done in a privacy preserving manner by infusing noise into both the statistics and the model generating step. From that point, the synthetic data is sampled directly from the model in a non-private manner. DP synthetic data provides a strong formal guarantee of privacy on the original private data while creating a synthetic dataset that retains many of the properties of the original dataset. The synthetic data may then be used in any way and shared without additional privacy risk, due to the post-processing property of DP [15].

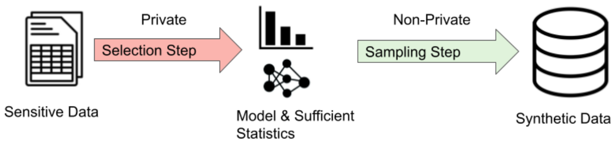


**Figure 1: Popular private synthetic data framework design.**

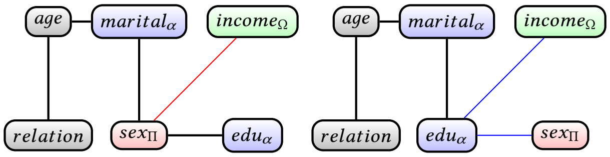
However, as indicated by [18], the synthetic data generation process may preserve or even exacerbate properties in the data that make it unfair. There has been a significant effort in recent years to both define fairness [5, 9, 10, 14, 17, 24, 25, 28, 38, 46, 54], and develop frameworks to understand and mitigate bias [16, 47, 50]. Among these, causal approaches for fairness [24, 28, 46] propose a principled approach of measuring the effect of protected attributes on the outcome attributes. These works consider the causal relationship between protected attributes, i.e., those that cannot be explicitly used in decision making, admissible attributes, i.e., those that can be explicitly used in decision making, and outcome attributes that are the prediction targets.

Prior works have proposed frameworks for pre-processing steps to repair the data and ensure causal fairness [6, 16, 46]. Notably, none of these have done so while under the constraint of DP as they assume the data is accessible.

*In this paper, we propose* PreFair (**Pr***ivat***e** *and* **Fair** *synthetic data generation), a novel framework for generating fair synthetic data that satisfies DP.* Our solution combines the state-of-the-art DP synthetic data generation system, MST [34] (shown to be the best performing mechanism in terms of minimizing the discrepancy between the private database and the generated one [51]) with the robust causal fairness definition of Justifiable Fairness [46] to generate data that ensures fairness while satisfying DP. In justifiable fairness, the data is defined to be fair if there is no directed path in the causal graph between the protected attribute and the outcome attribute that does not pass through an admissible attribute in a graphical model which describes the data. Specifically, PreFair augments the first step shown in Figure 1 to satisfy the graph properties needed for Justifiable Fairness and provides a guarantee that the data generated in the second step will be fair, under a common assumption.

Example 1. *Consider the Adult dataset [4] with the attributes age, marital-status (marital), education (edu), sex, income, and relationship. The MST system first learns the 2-way attribute marginals and then samples from them to generate a synthetic database instance.*

*Figure 2a shows the graphical representation of the database attributes, generated by MST. Here, protected attributes are denoted with a subscript $\Pi$ (and are colored red), admissible attributes are denoted with a subscript $\alpha$ (and are colored blue) and the outcome attribute with a subscript $\Omega$ (and is colored green). The edges denote attributes that have a strong correlation between them in the private database. Note that the protected attribute sex (denoted by a subscript*



**(a) Non-fair model gen. by [34]** **(b) Fair model gen. by** `PreFair`

**Figure 2: Graphical data generation models**

*$\Pi$ and colored red) has an edge to the income attribute which is the outcome. Intuitively, any synthetic database that will follow the distribution defined by this graph will have a direct dependence between the attribute sex and the attribute income, resulting in unfair data (we recognize that the graphical model is not a causal graph, but we rely on the graphical properties of justifiable fairness). On the other hand, Figure 2b shows an alternative graphical representation of the dependencies, generated by* `PreFair`*. In this graph, the path from sex to income passes through the admissible attribute education (the edges colored blue). Intuitively, in any synthetic data that will adhere to this distribution, the influence of the sex attribute on income will be mitigated by the education attribute, which is considered admissible, so, it is allowed to directly influence the outcome.*

Ensuring that `PreFair` is useful gives rise to several challenges. In particular, we tackle the following challenges: (1) adapting the definition of fairness to undirected graphical models, (2) making a connection between the graphical model and the data to guarantee that a fair model would lead to the generation of fair data, and (3) providing efficient algorithms that will generate fair data while remaining faithful to the private data and satisfying DP.

**Our contributions.**

- We propose `PreFair`, the first framework, to our knowledge, that generates synthetic data that is guaranteed to be fair and differentially private. `PreFair` combines the state-of-the-art synthetic data generation approach that satisfies DP [34] with the definition of justifiable fairness [46].
- We provide a novel model for fair data generation that relies on probabilistic graphical models and characterize the needed properties for the sampling approach to generate justifiably fair data.
- We prove that providing the optimal fair graph in our model is NP-hard even in the absence of privacy.
- Due to the intractability of the problem, we devise two solutions: (1) an algorithm that provides an optimal solution for an asymptotically large privacy budget and (2) a greedy algorithm that uses heuristics to construct the graph, while still guaranteeing fairness but possibly reducing the faithfulness to the private data.
- We perform an extensive experimental evaluation to test `PreFair` and show that our greedy approach can generate fair data while preserving DP and remaining faithful to the private data. We further show that our greedy approach does not incur major overhead in terms of performance.

## 2 PRELIMINARIES

**Data.** We assume a single table schema $\mathcal{S} = R(A_1, A_2 \dots A_d)$ where $\mathcal{A} = \{A_1, A_2 \dots A_d\}$ denotes the set of attributes $R$. Each attribute $\mathcal{A}_i$ has a finite domain $Dom(A_i)$. The full domain of $R$ is $Dom(R) = Dom(A_1) \times Dom(A_1) \times \dots Dom(A_d)$. An instance $D$ of relation $R$ is a bag whose elements are tuples in $Dom(R)$, i.e., a tuple can be written as $t = (a_1, \dots, a_d)$ where $a_i \in Dom(A_i)$. The number of tuples in $D$ is denoted as $|D| = n$. We consider synthetic databases. Given a database $D$ with schema $\mathcal{S}$ and $n$ tuples, we say that $D'$ is a synthetic copy of $D$ if $D'$ also has the schema $\mathcal{S}$ and contains $n$ tuples of the form $t = (a'_1, \dots, a'_d)$ where $a'_i \in Dom(A_i)$.

From now, we refer to the process of generating a synthetic copy of a database $D$ as synthetic data generation.

### 2.1 Renyi Differential Privacy

In this paper, we employ the notion of Renyi-DP (RDP) [37]. This is a formal model of privacy that guarantees each individual that any query computed from sensitive data would have been almost as likely as if the individual had opted out. More formally, RDP is a property of a randomized algorithm that bounds the Renyi Divergence of output distributions induced by changes in a single record. To define privacy, we begin with neighboring databases.

Definition 1 (Neighboring Databases). *Two databases $D$ and $D'$ are considered neighboring databases if $D$ and $D'$ differ in at most one row. We denote this relationship by $D' \approx D$.*

We often use the notion of neighboring databases to distinguish the impact of any particular individual's input on the output of a function. We likewise measure the maximum change in any function due to the removal or addition of one row often calling it the sensitivity of the function.

Definition 2 (Sensitivity). *Given a function $f$ the sensitivity of $f$ is $sup_{D' \approx D}|f(D) - f(D')|$ and is denoted by $\Delta f$.*

From here we can define our formal notion of Privacy, RDP.

Definition 3 (Renyi Differential Privacy). *A randomized mechanism $\mathcal{M}$ satisfies $(\alpha, \gamma)$-RDP for $\alpha \geq 1$ and $\gamma \geq 0$ if for any two neighboring databases $D$, and $D'$:*

$$D_\alpha(\mathcal{M}(D) \| \mathcal{M}(D')) \leq \gamma$$

*where $D_\alpha(\cdot\|\cdot)$ is the Renyi divergence of order $\alpha$ between two probability distributions.*

This ensures that no single individual contributes too much to the final output of the randomized algorithm by bounding the difference of the distributions when computed on neighboring databases.

All mechanisms which satisfy RDP also satisfy classic differential privacy [13]. Given privacy parameters $\alpha$ and $\gamma$ the RDP guarantee can be translated into a classical DP guarantee as follows.

Theorem 1 (RDP to DP [37]). *If a mechanism $\mathcal{M}$ satisfies $(\alpha, \gamma)$-RDP it also satisfies $\left(\gamma + \frac{\log(1/\delta)}{\alpha-1}, \delta\right)$ - DP for all $\delta \in (0, 1]$*

The parameter $\gamma$ quantifies the privacy loss. While classic DP bounds the worst-case privacy loss, RDP treats privacy loss as a random variable which allows for computing a tighter bound over privacy loss in many situations. Our mechanisms, such as Algorithm 1 (Section 4.1), use repeated calls to the Gaussian mechanism and benefit from this tighter analysis. If there are two RDP releases of the same data with two different privacy parameters the amount of privacy loss is equivalent to the sum of their privacy parameters.

Theorem 2 (RDP composition [37]). *Let $\mathcal{M}_1$ be an $(\alpha, \gamma_1)$-RDP algorithm and $\mathcal{M}_2$ be an $(\alpha, \gamma_1)$-differentially private algorithm. Then their combination defined to be $\mathcal{M}_{1,2}(x) = (\mathcal{M}_1(x), \mathcal{M}_2(x))$ is $(\alpha, \gamma_1 + \gamma_2)$-RDP.*

When designing private systems designers are required to balance utility and privacy loss over several mechanisms to ensure that a global privacy loss is not exceeded. RDP is robust to post-processing, i.e., if any computation is done on an RDP data release without access to the original data, the additional computation also satisfies RDP.

**The Gaussian mechanism [37].** This is a DP primitive for answering numerical queries and is often used to construct more complex DP mechanisms.

Definition 4 (Gaussian Mechanism). *Given a query $q$ and a database, $D$ the randomized algorithm which outputs the following query answer is $(\alpha, \alpha\frac{\Delta q^2}{2\sigma^2})$-RDP [37] for all $\alpha \geq 1$.*

$$q(D) + \mathcal{N}(0, \sigma^2)$$

where $\mathcal{N}(0, \sigma)$ denotes a sample from the Normal distribution with mean 0 and scale $\sigma$, and $\Delta q$ is the sensitivity of $q$ (Definition 2).

**The Exponential Mechanism (EM) [36].** EM is an RDP primitive for queries that outputs categorical data instead of numerical data. The exponential mechanism releases a DP version of a categorical query $Q$ by randomly sampling from its output domain $\Omega$. The probability for each individual item to be sampled is given by a pre-specified score function $f$. The score function takes in as input a dataset $D$ and an element in the domain $\omega \in \Omega$ and outputs a numeric value that measures the quality of $\omega$. Larger values indicate that $\omega$ is a better output with respect to the database and as such increases the probability of $\omega$ being selected. More specifically given a dataset $D$ the exponential mechanism samples $\omega \in \Omega$ with probability proportional to $\exp(\frac{\epsilon}{2\Delta f} \cdot f(D, \omega))$ where $\Delta f$ is the sensitivity of the scoring function $f$.

Theorem 3 (From [36]). *The Exponential Mechanism applied to the quality score function $f$ satisfies $(\alpha, \alpha\frac{(2\epsilon\Delta f)^2}{8})$- RDP. Where $\Delta f$ is the sensitivity of f*

Both the Gaussian and Exponential mechanism are DP primitives which are often used as subroutines in the design of more complex DP mechanisms.

## 2.2 Marginals-MST

Previous work [34] has proposed an approach of using a Marginals-MST for generating DP synthetic data. The Marginals-MST is based on a Bayes network that encodes the 2-way marginals between the different attributes of the database. This approach was shown to be the state-of-the-art DP mechanism for generating synthetic data [51]. In the selection step (recall Figure 1), the system generates a Marginals-MST which maximizes the pairwise mutual information over the attributes of the database. The mutual information between two attributes measures the mutual dependence between the two attributes. Two attributes that depend on each other more have higher mutual information. As such the 2-way marginals of attributes with high dependence on one another are chosen to be measured directly. Mutual information is defined as follows.

Definition 5 (Mutual Information [11]). *The mutual information between two attributes $A_i$ and $A_j$ is defined as follows.*

$$\sum_{a_i \in dom(A_i)} \sum_{a_j \in dom(A_j)} P[A_i = a_i, A_j = a_j] \log\left(\frac{P[A_i = a_i, A_j = a_j]}{P[A_i = a_i]P[A_j = a_j]}\right) \tag{1}$$

**Selection step.** The Marginals-MST selection step is a three step process each with its own split of the privacy parameter. First, it measures the 1-way marginals. It then selects a set of 2-way marginals to measure by creating a graph where the nodes represent the attributes and the edges represent 2-way marginals between attributes. The weights of each edge are set to (an estimation of) the mutual information between the two attributes. It then uses a private mechanism to generate a maximum spanning tree (Marginals-MST) over this graph by using a private version of Kruskal's algorithm [27] where the exponential mechanism is used to select edges.

Definition 6 (Marginals-MST). *Given a database $D$ over the attribute set $\mathcal{A}$ and an undirected graph $G = (V, E, w)$, where $V = \mathcal{A}$, $E$ are edges between attributes, and $w : V \times V \to \mathbb{R}^+$ returns the mutual information between every pair of attributes such that $(A_1, A_2) \notin E$ iff $w(A_1, A_2) = 0$, a Marginals-MST is a spanning tree of $G$.*

Once the marginals are selected the final graphical model is constructed using Private PGM [35] which (approximately) preserves the conditional independence properties of the Marginals-MST as well as its marginals. From here the model is sampled directly.

**Sampling step.** Once the graphical model is generated, the data is sampled in a way similar to sampling from a Bayes net. It samples attributes one at a time with probabilities dependent on its already sampled neighbors. This imposes a direction in all the edges in the Marginals-MST resulting in a probability distribution that can be encoded in a directed Bayes net. The direction of the edges is directly dependent on the order in which the attributes are sampled with edges going from nodes which were sampled first toward nodes which were sampled later.

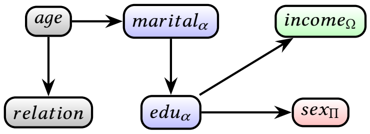


**Figure 3: Directed Marginals-MST as a result of sampling**

Example 2. *Figure 3 shows the process of sampling from the Marginals-MST shown in Figure 2b. The first attribute to be sampled is age. Its probability is dependent only on its 1-way marginal probability. Then, marital-status is sampled. Since age is already sampled, marital-status is then sampled according to the estimated 2-way marginal distribution. This process continues down the tree until all attributes are sampled.*

Likewise, this ensures that the final distribution (approximately) follows the local Markov property of Bayesian networks.

Definition 7 (Local Markov Property). *A graphical model $\mathcal{N}$ satisfies the Local Markov Property if each attribute is conditionally independent of its non-descendants given its parent values. That is more formally as follows.*

$$\forall a \in \mathcal{A} : X_a \perp\!\!\!\perp X_{\mathcal{A}\setminus\delta(a)} | X_{\Pi(a)} \tag{2}$$

Where $\delta(a)$ are the descendants of $a$ and $\Pi(a)$ is the parents of $a$. In particular, given the local Markov property, we can write the probability distribution on the attributes $\mathcal{A}$ as follows.

$$Pr[\mathcal{A}] = Pr[X_1, X_2 \dots X_{|\mathcal{A}|}] = \prod_{i=1}^{|\mathcal{A}|} = Pr[X_i|\Pi_i] \tag{3}$$

where $\Pi_i$ is the parent set of $X_i$. Given a Bayesian network a common inference task is determining conditional independence between two attributes.

A sufficient criteria to denote conditional independence is d-separation [41] a condition that can be checked directly on the graph structure. We say that if $\mathcal{G}$ is a graphical model and $Pr$ is a probability distribution then $Pr$ is Markov compatible with $\mathcal{G}$ if a d-separation in $\mathcal{G}$ implies conditional independence in $Pr$. or more formally.

$$\mathbf{X} \perp\!\!\!\perp \mathbf{Y}|_d \mathbf{Z} \rightarrow \mathbf{X} \perp\!\!\!\perp \mathbf{Y}|\mathbf{Z} \tag{4}$$

If the converse is true, we say that $Pr$ is faithful to $\mathcal{G}$. In terms of Bayesian networks, the following is true.

Theorem 4. *If $\mathcal{G}$ is a directed acyclic graph over attributes $\mathcal{A}$ and $Pr$ is a probability distribution that follows Equation* (3)*, then $Pr$ is Markov compatible with $\mathcal{G}$.*

Interventions on a Bayseian network are achieved via the **do** operator denoted do$(X = x)$. On the graph structure itself this is equivalent to setting the value of a particular node and removing all the edges between that node and its parents. There is an equivalent notion on the distribution itself which can be computed analytically.

## 2.3 Causal Fairness

Causal fairness notions [24, 28] consider the dependence between a set of protected attributes $P$ and a set of outcome decisions $O$. Protected attributes are those which are considered sensitive and unactionable. For example, race and gender are considered unactionable attributes in highly sensitive tasks such as college admissions and loan applications. The outcome attribute $O$ is the objective to predict given the other attributes.

Here, we focus on justifiable fairness [46] which considers additional admissible attributes. These attributes are considered actionable for decision making despite their possible causal links to protected attributes.

We begin by first introducing $K$ fairness and then using that to derive the notion of justifiable fairness.

Definition 8 (K-fairness [46]). *Given the disjoint set of attributes $\mathcal{A}$, protected attributes $P$, and outcome attribute $O$ we say that a mechanism $\mathcal{M}$ is $k$-fair with respect to the protected attribute if for any context $K = k$ and every outcome $O = o$ and every $P_i \in P$ the following holds:*

$$P[O = o|do(P_i = 0), do(K = k)] = P[O = o|do(P_i = 1), do(K = k)]$$

That is, conditioned on the setting of the $k$ attributes, the value of the protected attribute should not affect the final outcome.

K-fairness is usually defined with respect to the outcome of some algorithm such as a classifier but as most pre-processing techniques do, we will make the "reasonable classifier" assumption, where the conditional probability of classification is close to that same conditional probability of the training label in the training set. Therefore, we will use the outcome of a classifier and the training label interchangeably here.

Definition 9 (Justifiable Fairness [46]). *Let $\mathcal{A}$ be the set of all attributes, $P \subset \mathcal{A}$ be a set of protected attributes $A \subset \mathcal{A} \setminus P$ be a set of admissible attributes and $O$ be an outcome attribute. A mechanism $\mathcal{M}$ is justifiably fair if it is $K$-fair with respect to all supersets $K \supseteq A$*

Past work [46] notes that for a directed causal graph $\mathcal{G}$ over a probability distribution if all directed paths from a protected attribute to the outcome attribute pass through at least one admissible attribute then that probability distribution is justifiably fair.

Theorem 5 ([46]). *If all paths in a causal directed acyclic graph $\mathcal{G}$ going from protected attributes to outcome attributes go through at least one admissible variable then $\mathcal{G}$ is justifiably fair. If the probability distribution is faithful to the causal DAG, then the converse also holds.*

While this is true for data that is faithful to a causal DAG, it is also true for data that is Markov compatible with that same graphical structure.

# 3 A FRAMEWORK FOR THE GENERATION OF FAIR AND DP SYNTHETIC DATA

All of our mechanisms can be seen as imposing a distribution represented in the created Bayes net onto a particular dataset. This does not inherently preserve any causal structure nor impose any causal structure on the dataset it merely preserves a chosen set of measured statistics. In our setting, this is required as it is known that no private mechanism can preserve all possible statistics on a dataset [12] but instead, it must choose to preserve a subset of them and, even then, noise has to be infused into that subset.

## 3.1 Model For a Fair Marginals-MST

We now detail the model for a fair Marginals-MST, and relate it to the synthetic data sampled in the second step (Figure 1), and define the main problem at the center of our model.

**Causal interpretation.** Previous work on justifiable fairness [46] satisfies justifiable fairness by changing the underlying distribution to make attributes that could be causally linked to instead be independent. Fundamentally, this repair mechanism imposes independence between attributes and does not impose new casual structures. Similarly, our work can be seen as imposing an entire distribution onto a particular dataset. We generate distributions with independence properties that ensure the independence between the protected and outcome attributes when conditioned on the admissible attributes. We do not impose any given causal structures. Instead, we create data with a specific distribution from which no causality between protected and outcome attributes can be inferred.

**Fairness of a Marginals-MST.** We can adapt the notion of interventional fairness to Marginals-MST. As a first step, we need to bridge the gap between an undirected Marginals-MST and Definition 8 that assumes a DAG. Here, we consider the option of multiple outcomes rather than a single outcome attribute. This generalization is very practical for the synthetic data generation scenario, as we explain in the discussion in Section 3.2.

Proposition 1. *Let $D$ be a database over attributes $\mathcal{A}$, let $P, A, O \subseteq \mathcal{A}$ be disjoint sets representing the protected, admissible and outcome attributes respectively, and let $T$ be a directed Marginals-MST over $\mathcal{A}$. If all directed paths in $T$ going from any attribute in $P$ to any outcome attribute go through at least one attribute in $A$, then the distribution of $T$ is justifiably fair.*

This proposition leads to the definition of a fair Marginals-MST $T$ as one for which any directed Marginals-MST derived by directing the edges of $T$ satisfies the premise in Proposition 1.

Definition 10 (Fair Marginals-MST). *Given a database $D$ over attributes $\mathcal{A}$, and disjoint sets $P, A, O \subseteq \mathcal{A}$ representing the protected, admissible and outcome attributes respectively, a Marginals-MST $\mathcal{T}$ whose node set is $\mathcal{A}$ is fair if in any directed Marginals-MST obtained by directing the edges of $T$, all directed paths in $\mathcal{T}$ going from any attribute in $P$ to any outcome attribute in $O$ go through at least one attribute in $A$.*

Example 3. *Figure 2b shows an example of a fair Marginals-MST. The only path from the protected attribute (sex) goes through at least one admissible attribute (education). Regardless of any direction imposed on the edges, the path between the outcome and protected attribute remains blocked by an admissible attribute.*

We define a fair Marginals-MST as a Marginals-MST that satisfies Proposition 1, regardless of the imposed direction of edges. This ensures that a fair Marginals-MST represents a distribution that is justifiably fair regardless of the sampling order. Since Marginals-MST is a tree, we get the acyclic property ‘for free’ as there cannot be cycles in the tree, no matter the direction of the edges.

**From a fair Marginals-MST to fair synthetic data.** We now show that given a fair Marginals-MST, one can sample data which satisfies the property in Definition 9 as well, so long as the sampling approach preserves the distribution of the fair Marginals-MST.

Proposition 2. *Any synthetic database whose distribution is Markov Compatible (Equation (4)) with a directed fair Marginals-MST satisfies Justifiable Fairness (Definition 9).*

Intuitively, this proposition states that it is sufficient to find a fair Marginals-MST to ensure that the data derived from it is justifiably fair, as long as the sampling method is faithful to the distribution described by the fair tree. This is inherently true for any sampling method that preserves the distribution of a Bayes net (Equation (3)) such as the MST [34] and PrivBayes [58] sampling steps.

## 3.2 Problem Definition and Intractability

We have adapted the notion of interventional fairness to Marginals-MST and have further determined that it is sufficient to consider the fairness of the Marginals-MST to know that a synthetic database that is generated by a faithful sampling process will be fair. We can now define FDPSynth (Fair DP Synthetic Data Generation) as the problem at the heart of `PreFair`.

Definition 11 (The FDPSynth Problem). *Let $D$ be a database with attribute set $\mathcal{A}$, a set of protected, admissible and outcome attributes $P, A, O \subseteq \mathcal{A}$, such that $P, A, O$ are non-empty and pairwise disjoint. Let $G = (V, E, w)$ be a graph encoding the mutual information between every pair of attributes in $\mathcal{A}$, where $V = \mathcal{A}$ is the set of nodes, $E = V \times V$ is the set of edges, and $w : V \times V \to \mathbb{R}^+ \cup \{0\}$ is a weight function for pairs of nodes that denotes the mutual information between every pair of attributes in $D$, $w(A_1, A_2) = 0$ iff $(A_1, A_2) \notin E$ and the mutual information between $A_1$ and $A_2$ is $0$. The FDPSynth problem is to find a fair MST that maximizes the pairwise mutual information between the attributes in $\mathcal{A}$ while satisfying RDP for a given privacy parameter $\rho$.*

Example 4. *Reconsider Example 1, where the protected attribute is sex, the admissible attributes are education, marital-status and the outcome attribute is income. We assume that there is a graph where the edges represent the mutual information between these attributes in the private database. The FDPSynth problem is to generate a fair Marginals-MST that maximizes the pairwise mutual information between the attributes while satisfying RDP for a given $\rho$.*

We next give a simple existence claim, which will be implicitly used by our solutions (Section 4) to find a fair Marginals-MST.

Proposition 3. *For any instance of the FDPSynth problem, a solution always exists.*

**The need for multiple outcomes.** Previous work on justifiable fairness [46] assumes that the database contains a single outcome attribute. However, in the context of generating private synthetic data this assumption is restrictive. In the non-private setting several different synthetic data sets can be constructed for individual tasks with different outcomes. In the private setting due to Theorem 2 each release of privacy protected data comes at the cost of additional privacy leakage. As a result, it is more efficient to release a single set of synthetic data which considers multiple outcomes.

Example 5. *Reconsider Figure 2b with the attribute relation as a second outcome. This simulates the case where the datatset needs to be used for two independent prediction tasks. In the first prediction task, income is to be predicted and in the second task relation status is to be predicted. Since both relation and income are to be predicted they should both be dependent on the admissible attributes and not the protected attribute (sex).*

**NP-hardness of FDPSynth.** Here, we show that FDPSynth is intractable. We first define the decision version of the FDPSynth problem without the RDP requirement. We show that, even in the absence of the RDP requirement, the decision problem is NP-hard. In the presence of private noise every instance of the problem must have non-zero probability, therefore, the addition of privacy still allows for the graph instances that are shown in the proof.

Definition 12 (Decision version of FDPSynth). *Let $D$ be a database with attribute set $\mathcal{A}$, a set of protected, admissible, and outcome attributes $P, A, O \subseteq \mathcal{A}$, such that $P, A, O$ are non-empty and pairwise disjoint and let $k \in \mathbb{N}$. Let $G = (V, E, w)$ be a graph encoding the mutual information between every pair of attributes in $\mathcal{A}$, where $V = \mathcal{A}$ is the set of nodes, $E = V \times V$ is the set of edges,*

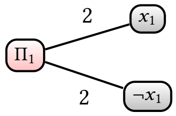


**Figure 4: Assignment Gadget**

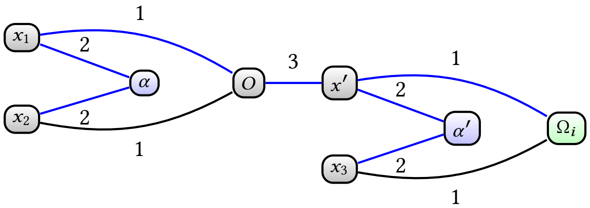


**Figure 5: 3-way OR gadget for a 3CNF clause $C_i$. The blue edges represent the maximum weight spanning tree when there is an unblocked path from a protected attribute to $x_2, x_3$.**

*and $w : V \times V \to \mathbb{R}^+ \cup \{0\}$ is a weight function for pairs of nodes that denotes the mutual information between every pair of attributes in D, $w(A_1, A_2) = 0$ iff $(A_1, A_2) \notin E$ and the mutual information between $A_1$ and $A_2$ is 0. The goal is to decide if there exists a fair Marginals-MST, $T = (V, E, w)$ of G such that $\sum_{e \in E} w(e) \geq k$.*

We now detail the NP-hardness claim and the reduction from 3-SAT leading to its proof.

Proposition 4. *The decision version of FDPSynth is NP-Hard.*

Proof Sketch. We show that, given an instance of 3-SAT, we can define an equivalent instance of the FDPSynth problem, where any maximal spanning tree will correspond to a satisfying assignment to all 3-CNF clauses. We assume without loss of generality [57] that the 3-SAT instance has no trivial clauses and, for each literal, the formula contains at least one clause with that literal and at least one clause with its negation.

**Graph construction.** We describe the construction of a graph $\mathcal{G}$ from a 3-SAT instance $\varphi$ with $n$ literals and $m$ clauses.

(1) For each literal $x_i$ in $\varphi$, create an assignment gadget (see Figure 4). This gadget contains a protected attribute $\Pi_i$ and edges from the protected attribute to two additional attributes representing the literal and its negation ($x_i$ and $\neg x_i$ respectively).

(2) For each clause $C_i$, in $\varphi$ we create a 3-way OR gadget (see Figure 5). The $\alpha$ and $\alpha'$ nodes in the OR gadget are admissible attributes and the $\Omega_i$ nodes are outcome attributes. The nodes $x_1, x_2, x_3$ are the inputs to the OR gadget and represent the literals in the corresponding clause.

(3) Each of the inputs for the OR gate $(x_1, x_2, x_3)$ are then connected to the corresponding literal in the assignment gadget using a weight 3 edge. Due to the constraints, both the literal and its negation will always connect to at least one OR gadget. E.g., the $x_1$ node in Figure 5 would be connected to the $x_1$ node in Figure 4 through a weight 3 edge[1].

We set $k = 22m + 2n$, and thus $G = (V, E, w), k$ is an instance of the FDPSynth problem. Given $\varphi$, this instance can be constructed in polynomial time and, therefore, the reduction is valid.

**Proof of equivalence.** We now prove that $\varphi$ has a satisfying assignment iff there is a Marginals-MST with weight $\geq 22m + 2n$.

($\Rightarrow$) First, we describe the construction of a fair Marginals-MST, $T$, given a satisfying assignment to $\varphi$, $\theta$.

For each assignment gadget, add to $T$ only the edge corresponding to the literal set to $False$ by $\theta$. That is, $\theta(x_j) = False$ implies that $(\Pi_i, x_j)$ is included in $T$. For example, in Figure 4, if $\theta(x_1) = True$, the edge $(\Pi_1, \neg x_1)$ would be added to $T$. All the weight 3 edge connecting the assignment gadgets and OR gadgets will always be added to $T$. Then, for each OR gadget we add to $T$ edges such that any input which is set to $False$ has its path to $\Omega$ blocked by the admissible attributes. In Figure 5, the blue edges are added to $T$ when $\theta(x_1) = True$ and $\theta(x_2) = \theta(x_3) = False$. Note that any subtree of maximum weight (13) in the OR gadget can have at most two inputs whose path to $\Omega$ is blocked by an admissible attribute (these correspond to the literals set to $False$ by $\theta$).

We will now show that $T$ is a fair Marginals-MST. By Definition 10, we need to show that in any directed Marginals-MST, all directed paths in $T$ starting from $\Pi_i$ and ending at $\Omega_i$, go through an admissible node $\alpha$ or $\alpha'$. Since $\theta$ is a satisfying assignment, at most two of the inputs to each OR gadget have an unblocked path to a protected attribute (those set to $False$ by $\theta$). Since a maximum weight subtree of an OR gadget can block at most two paths from the inputs to $\Omega$, a set of edges can always be chosen such that all paths between protected and outcome attributes are blocked by admissible attributes. Since this holds for undirected paths it also holds for any direction given to those edges. For instance, the blue edges in Figure 5 form a fair Marginals-MST when $x_1$ is not connected to a protected attribute. This example is a valid setting in **any** assignment where $x_1$ is set to $True$ since $x_1$ has a direct path to $\Omega$ and the paths from $x_2, x_3$ are blocked by admissible attributes.

We now show that $T$ has a weight of $22m + 2n$. This weight arises from choosing the tree of maximum weight (13) from each OR gadget (so far, the weight is $13m$) and a weight 3 edge connecting the OR gadgets to the assignment gadget for each literal in the clause. Each clause has three such edges (so far, the weight is $13m + 3 \cdot 3m$). Then, one edge is taken from each assignment gadget of weight 2. Thus, the overall weight is $13m + 3 \cdot 3m + 2n = 22m + 2n$. This is the maximum weight for any possible Marginals-MST since we chose all the maximum subtree for each of the gadgets.

($\Leftarrow$) Assume that $G$ has a fair Marginals-MST, $T$, with weight $\geq 22m + 2n$. We show how to convert $T$ to an assignment, $\theta$, such that $\theta(\varphi) = True$.

To infer a satisfying assignment, $\theta$, from $T$, one only needs to look at the edges from the assignment gadgets. The edge that is taken in each assignment gadget defines which literal is set to $False$, i.e., if $(x_i, \Pi_i) \in E(T)$ then $\theta(x_i) = False$. For example, in Figure 4 if the edge between $\neg x_1$ and $\Pi_1$ is chosen that means $\theta(\neg x_1) = False$ and $\theta(x_1) = True$.

We now show that this assignment satisfies $\varphi$. For this, it is enough to show that any OR clause has at least one literal that is assigned to $True$ by $\theta$. Since $T$ is a fair Marginals-MST, the paths from the protected attributes to the outcome attributes are blocked

[1] Edges that are not explicitly listed in the construction have weight 0 and can be ignored.

by admissible attributes in the OR gadgets. Each of the OR gadgets can block two of the paths from the inputs to the outcome attribute. Therefore, each OR gadget must have exactly one input that has an unblocked path to the outcome attribute. Since $T$ is fair, that attribute does not have unblocked paths to a protected attribute (this input is set to $True$ by $\theta$). Since this must be true for every OR gadget and each gadget corresponds to one clause, $\theta$ assigns at least one literal to $True$ in each clause and, hence, is a satisfying assignment for $\varphi$. In Figure 5, since $T$ is fair, only $x_1$ has an unblocked path to $\Omega_i$, and this corresponds to $\theta(x_1) = True$. This same subtree is consistent with any assignment $\theta$ such that $\theta(x_1) = True$. $\theta(x_2)$ and $\theta(x_3)$ are not restricted by this gadget.

□

In order to connect the problem on an abstract graph to a database, we note that there always exists a database where the mutual information between attributes results in the structures of the assignment gadget and the 3-way OR gadget.

Lemma 1. *There exists a database $D$ whose attributes have the mutual information relationship that results in the graph of Figure 5.*

## 4 COMPUTING A FAIR MARGINALS-MST

Here we propose both an exponential time optimal solution as well as a linear time greedy algorithm. It is important to note when designing each of these mechanisms, we first designed a non-private solution to the fair Marginals-MST problem then adapt them to satisfy RDP. Even if an optimal non-private solution is adapted it does not ensure that the resulting private solution is also optimal.

### 4.1 Exponential Asymptotically Optimal Algorithm

Here we present an exponential time solution for finding the optimal tree. The non-private version of this solution acts similarly to Dijkstra's algorithm. We take in as input the set of measured 1-way marginals, and the RDP privacy parameter $\rho$. In addition, the mechanism takes as input the sets of admissible, protected, and outcome attributes. We estimate all of the 2-way marginals using Private-PGM [35]. From there in line 4 we measure the L1 error of each of the 2-way marginals (a sensitivity 1 estimate of the mutual information) using the Gaussian mechanism and set each edge weight to the error of its corresponding marginal. In line 5 we create a priority queue where we will store partial Marginals-MSTs which will be sorted by their current weight in ascending order. In lines 6-7 we set the weight of each edge to be the maximum measurement minus the measurement for that pair. This ensures that the attribute pairs with the highest mutual information have the lowest weight and reduces the problem to that of finding the lowest weight tree. This way the partial tree with the lowest weight will always remain on the top of the queue. In order to seed the queue in line 8, we add the partial tree where no nodes are connected with weight 0. From there (lines 9-14) at each time step we take the top partial tree from the queue and add to the queue all the possible trees with one additional edge that does not violate Proposition 1. We continue this process until the first complete Marginals-MST is on the top of the queue. Since the queue is sorted by weight and adding any edge only increases the weight, the first complete Marginals-MST to be found, by definition, must be the minimum weight tree.

**Algorithm 1:** Exponential-`PreFair`

```
input  : Database D, set of admissible attributes A, set of
         protected attributes P, set of outcome attribute O,
         measurements of 1-way marginals $log$, and a
         privacy parameter $\rho$
output : Set of $(i, j)$ pairs to measure $\mathcal{C}$
1  Use Private-PGM[35] to estimate all 2-way marginals $\bar{M}_{i,j}$
     from $log$
2  Let $r$ be the number of 2 way marginals
3  Let $\sigma = \sqrt{\frac{\rho}{2r}}$
4  Compute noisy measurements of $L_1$ error between the
     estimated 2-way marginal and real 2-way marginal for all
     $i, j$ pairs. $q_{i,j} = \|M_{i,j}(D) - \bar{M}_{i,j}\|$ using the Gaussian
     Mechanism with scale $\sigma$
5  Initialize empty priority queue $\mathcal{Q}$ sorted in ascending order
6  Let $q_{max}$ be the maximum measurement.
7  Set the weight of each edge (i,j) to $q_{max} - q_{i,j}$
8  Add the graph $G = (\mathcal{A}, \varnothing)$ to $\mathcal{Q}$
9  while $\mathcal{Q}$ is non-empty do
10     Let $G = (\mathcal{A}, \mathcal{C})$ be the first graph in $\mathcal{Q}$
11     if $G$ is a spanning tree then
12         return $G$
13     foreach edge $(i, j)$ that connects two connected
         components in G and does not create an unblocked path
         from a node in $O$ to a node in $P$ do
14         add $G + (i, j)$ to $\mathcal{Q}$
```

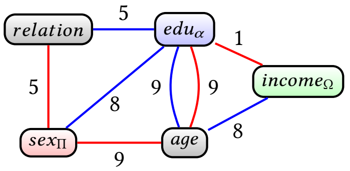


**Figure 6: Outputs of Greedy-`PreFair` (red edges) and Exponential-`PreFair` (blue edges)**

Example 6. *Consider the graph in Figure 6 on the attributes of the Adult dataset. The blue edges show the output of Exponential-`PreFair` in the absence of privacy noise. Exponential-`PreFair` iterates through all permutations of partial Marginals-MSTs, checking those with high edge weights. The result (in the absence of noise) is the optimal Marginals-MST with a total weight of 30.*

**Correctness and optimality condition.** We now detail the guarantees of Algorithm 1.

Lemma 2. *Algorithm 1 always terminates at line 12.*

Proof. Algorithm 1 iterates through all possible Marginals-MSTs as it considers all $|\mathcal{A}|^{|\mathcal{A}|}$ spanning trees in lines 13 and 14, removing those which do not satisfy Definition 10. Since a fair Marginals-MST always exists (see Proposition 3), Algorithm 1 will find it and will terminate at line 12. □

Proposition 5. *Given a private database $D$ over a set of attributes $\mathcal{A}$, a set of admissible attributes $A \subseteq \mathcal{A}$, a set of protected attributes $P \subseteq \mathcal{A}$, outcome attribute $O \subseteq \mathcal{A}$, measurements of 1-way marginals of $\mathcal{A}$ over $D$, log, and a privacy parameter $\rho$, the following holds:*

1. *Algorithm 1 satisfies $(\alpha, \rho\alpha) - RDP$ for all $\alpha \geq 1$.*
2. *Algorithm 1 outputs a fair Marginals-MST.*

While Algorithm 1 always outputs a fair Marginals-MST in all cases it is only guaranteed to maximize the mutual information in the case without any privacy preserving noise. Adding privacy preserving noise to the optimal non-private mechanism does not always result in the optimal private mechanism.

Proposition 6. *In the absence of privacy preserving noise (as $\rho \to \infty$), Algorithm 1 outputs the fair Marginals-MST with maximum mutual information.*

**Complexity and privacy budget discussion.** This solution can be both slow and privacy intensive. First, there is an exponential number of possible spanning trees and, in the worst case, considering all of them is prohibitively expensive, resulting in the following time complexity.

Proposition 7. *The time complexity of Algorithm 1 is $O\left(|\mathcal{A}|^{|\mathcal{A}|-2}\right)$.*

In terms of privacy budget, since there can be a possible exponential number of additions to the partial solution queue using the exponential mechanism at each time step is not practical. Instead, the mechanism measures all of the edges using the Gaussian mechanism and repeatedly uses those values when necessary. While better this still requires a larger split of the privacy budget than the greedy solution. Specifically, there are $\frac{|\mathcal{A}|(|\mathcal{A}|-1)}{2}$ calls to the Gaussian mechanism in the algorithm. The repeated calls to the Gaussian mechanism requires that the privacy budget be split into smaller parts and as such results in higher noise in these measurements. This often leads to poor running time performance which we measure in Section 5.

## 4.2 Greedy Algorithm

We now introduce an alternative solution. We slightly change the selection step of MST [34] to only consider a subset of possible edges. This will always result in a fair Marginals-MST but may not be optimal in terms of mutual information. However, this solution runs in linear time with respect to the number of attributes. Furthermore, we show that this solution is optimal in certain special cases.

We can adapt the greedy selection step from the MST mechanism in order to generate a fair Marginals-MST by restricting the set of possible neighbors of outcome attributes. We simply restrict all the neighbors of outcome attributes to be either other outcome attributes or admissible attributes. This ensures that any path to the outcome attributes must pass through at least one admissible attribute in all cases. In practice, this involves doing the private Kruskal's algorithm (described in [34]) for finding a Marginals-MST but deleting all the edges from outcome attributes to attributes that are neither outcomes nor admissible. This algorithm is shown in full in Algorithm 2. The blue line shows the modification to the original algorithm.

**Algorithm 2:** Greedy-PreFair

```
input  : D, A, P, O, log, $\rho$ ;              // defined in Algorithm 1
output : Set of $(i, j)$ pairs to measure $\mathcal{C}$
1  Use Private-PGM [35] to estimate all 2-way marginals $\bar{M}_{i,j}$
     from log
2  Compute $L_1$ error between the estimated 2-way marginal
     and real 2-way marginal for all $i, j$ pairs.
     $q_{i,j} = \|M_{i,j}(D) - \bar{M}_{i,j}\|$(an approximation of mutual
     information)
3  Let $G = (\mathcal{A}, \mathcal{C})$ be the graph where attributes are vertices
     and edges are pairs of attributes.
   /* Addition to ensure fairness                                  */
4  Remove the edges $(o, i)$ where $o \in O$ and $i \notin O \cup A$
5  Let $r = |\mathcal{A}|$
6  Let $\epsilon = \sqrt{\frac{8\rho}{r-1}}$
7  for $k = 1 to r - 1$ do
8      Let $S$ be the set of all pairs $(i, j)$ where $i$ and $j$ are in
         different connected components of $G$.
9      Select pair $(i, j)$ via the exponential mechanism with
         score function $q_{i,j}$ on set $S$ with privacy parameter $\epsilon$.
10     add $(i, j)$ to $\mathcal{C}$
11 return $\mathcal{C}$
```

Algorithm 2 takes as input the database, the measured one-way marginals and the RDP privacy parameter $\rho$. Additionally, it takes in as input the set of admissible and protected variables. We then construct a Marginals-MST over a complete graph where the nodes are attributes and the edges represent 2-way marginals between attributes and their weights are low sensitivity estimates of the mutual information between the two. In order to get this estimation in line 2 we first estimate the 2-way marginals from the 1-way marginals using Private-PGM [35] then set each edge's weights to be the L1 error of those estimates. We then do the private version of Kruskal's algorithm. In line 4, we first remove all the edges from outcome attributes to inadmissible attributes as adding any of these nodes would result in a violation of justifiable fairness. Then at each time step (lines 7-9) we select an edge to add to the partial maximum spanning tree from the set of edges that would connect two disjoint connected components. To maintain privacy, this selection is done via the exponential mechanism. After $|\mathcal{A}| - 1$ rounds of this process, the resulting edges will result in a private estimation of a minimum spanning tree over this graph. Once this MST is selected they are measured directly using the Gaussian mechanism and are then used to generate the synthetic data.

Example 7. *Reconsider the graph in Figure 6. The red edges demonstrate the output of Greedy-*PreFair *in the absence of privacy noise. The mechanism greedily takes the highest weight edge so long as that edge is not between an attribute in $O$ and an attribute not in $O \cup A$. In this example it starts by taking the edge between sex and age, then takes the edge between age and education followed by the edge between sex and relation and finally the edge between education and income. This results in a final edge weight of* 24 *which is slightly less than the optimal* 30 *however it still results in a fair Marginals-MST.*

The proof that Algorithm 2 satisfies RDP is identical to the proof that the original MST algorithm satisfies DP [34]. Furthermore, since Algorithm 2 only allows attributes in $O$ to be adjacent to attributes in $A$ regardless of the sampling order each $O$ is independent of all other nodes given the attributes in $A$ and, therefore, generates a fair Marginals-MST. As such, we get the following.

Proposition 8. *Given a private database $D$, a set of admissible attributes $A$, a set of protected attributes $P$, outcome attribute $O$, measurements of 1-way marginals of the attributes of $D$, log, and a privacy parameter $\rho$, the following holds:*

*(1) Algorithm 2 satisfies $(\alpha, \rho\alpha)$ – RDP for all $\alpha \geq 1$.*
*(2) Algorithm 2 outputs a fair Marginals-MST.*

**Optimality for the saturated case.** Algorithm 2 is not optimal (in the absence of noise) in the general case as there are cases where the greedy solution results in a fair Marginals-MST with sub-optimal total mutual information. There is, however, a special case for which Algorithm 2 is optimal. When all attributes are either admissible protected or outcomes (i.e., $A \uplus O \uplus P = \mathcal{A}$), we say that the problem instance is *saturated.* The saturated case corresponds to the cases where the user has complete knowledge as to which attributes may or may not be used for decision making and there is no ambiguity among any of the attributes. In the saturated case, we note that in any fair Marginals-MST, all edges from outcome attributes must either come from other outcome attributes or admissible attributes.

Proposition 9. *Given a database $D$ over attributes $\mathcal{A}$, and disjoint sets $P, A, O \subseteq \mathcal{A}$ representing the protected, admissible, and outcome attributes respectively, a Marginals-MST $T$ whose node set is $\mathcal{A}$ is fair if all the neighbors of nodes in the set $O$ are in the set $A \cup O$. If the problem is saturated the converse is also true.*

This means that when the problem is saturated finding the optimal fair Marginals-MST is equivalent to finding the MST on the subgraph containing no edges from attributes in $P$ to attributes not in $P$ or $A$. Therefore, using a traditional MST algorithm such as Kruskal's algorithm [27] on this subgraph results in an optimal Marginals-MST. Thus, using Algorithm 2 yields an optimal fair Marginals-MST.

**Complexity and privacy budget discussion.** Algorithm 2 is efficient with respect to both privacy budget and computation time. Unlike Algorithm 1 which uses a quadratic number of calls to the Gaussian Mechanism Algorithm 2 calls the exponential mechanism only a linear number of times. This results in significantly more privacy budget being allocated to each individual call of the exponential mechanism resulting in lower noise. Algorithm 2 also has a significantly improved run time. Since Algorithm 2 only selects $|\mathcal{A}| - 1$ edges using the exponential mechanism it only requires a linear amount of time to run. This is a dramatic improvement over Algorithm 1 which is exponential.

Proposition 10. *The time complexity of Algorithm 2 is $O(|\mathcal{A}|)$.*

## 5 EXPERIMENTS

We present experiments that evaluate the efficacy of the proposed mechanisms. In particular, we explore the following main questions.

(1) **Q1**: What is the error overhead incurred by Algorithms 1 and 2 compared to the standard approach without the fairness condition?
(2) **Q2:** How effective are Algorithms 1 and 2 at reducing unfairness in downstream classification tasks?
(3) **Q3:** What is the performance of Algorithms 1 and 2 compared to the baseline that does not consider fairness?

**Implementation Details.** We have implemented our system in Python 3.8 based on the existing MST approach [34].

All values shown are averaged over 10 instances of synthetic data.

### 5.1 Experimental Setup

We next detail the settings of the experiments including the datasets used, the approaches that were examined, and the measures that were employed.

**Datasets.** We test our mechanisms on three datasets, each composed of a single table.

- **Adult [4]:** This dataset contains 14 attributes and 48843 tuples. It contains individual' census data. The objective is to predict if an individual's income is above or below 50k.
- **Compas [43]:** This dataset contains 8 attributes and 6173 tuples. It contains the criminal history, jail and prison time, demographics and Compas risk scores for defendants from Broward County from 2013 and 2014. The objective is to predict if an individual will commit a crime again within 2 years.
- **Census KDD [4, 44, 51]**: This dataset contains 41 attributes and 299,285 tuples. It contains information from the 1994 and 1995 Population Survey from the US census. The objective is to predict if an individual's income is above or below 50k.

To prepare the data for each task, we transformed categorical data into equivalent numerical data by mapping each domain value to a unique integer. Continuous data is instead treated as discrete categorical data with each unique value being its own category (see discussion on discretization in Section 7). We have listed the set of protected, admissible, and outcome variables in Table 1.

**Baselines and examined approaches.** For baselines, we compare against the original versions of MST [34]. For our mechanisms, we use adapted versions of each of these baselines, Exponential-`PreFair` (Algorithm 1) and Greedy-`PreFair` (Algorithm 2).

Each mechanism is used with 3 different settings of $\epsilon = [0.1, 1, 10]$.

We found the optimal RDP parameters and converted back to traditional DP using Theorem 1. Each mechanism is set to generate a dataset of the same size as the original dataset. All the measurements are done over 10 different samples of synthetic data for each mechanism.

We generated learned models using Tensorflow [1] for MLP models and scikit-learn [42] for linear regression as well as random forests. Each model was trained using one set of synthetic data and was tested against the underlying true data.

**Data quality.** In order to measure the overall quality of the data, we consider the same set of metrics as past benchmarks [51]. Each of these metrics are intended to measure a different quality of the data that should be preserved in the synthetic data.

(1) **Individual Attribute Distribution:** Here we measure the similarity of the 1-way marginals between the synthetic and original Data. To do so we measure the Total Variation Distance

**Table 1: Dataset Attributes Division**

| Dataset | Division | Attributes |
|---|---|---|
| Adult | Protected | Sex, race, native country |
| | Outcome | Income |
| | Admissible | Workclass, education, occupation, capital-gain, capital-loss, and hours per week |
| Compas | Protected | Sex, race |
| | Outcome | Two Year Recidivism |
| | Admissible | Number of Priors, misdemeanor/felony |
| Census KDD | Protected | Sex, race |
| | Outcome | Income |
| | Admissible | Workclass, Industry, Occupation, Education . . . |

**Table 2: Fairness measures**

| | |
|---|---|
| DP | $Pr(O=1\|S=1) - Pr(O=1\|S=0)$ |
| TPRB | $Pr(O=1\|S=1,Y=1) - Pr(O=1\|S=0,Y=1)$ |
| TNRB | $Pr(O=0\|S=1,Y=0) - Pr(O=0\|S=0,Y=0)$ |
| CDP | $\mathbb{E}_A(Pr(O=1\|S=1,A=a) - Pr(O=1\|S=0,A=a))$ |
| CTPRB | $\mathbb{E}_A[Pr(O=1\|S=1,Y=1,A=a) - Pr(O=1\|S=0,Y=1,A=a)]$ |
| CTNRB | $\mathbb{E}_A[Pr(O=0\|S=1,Y=0,A=a) - Pr(O=0\|S=0,Y=0,A=a)]$ |

(TVD) between a vectorized version of the 1-way marginals for both the synthetic and original data. The value shown is the average of the total variation distances for each 1-way marginal. We show the distribution over 10 samples of synthetic data generation.

(2) **Pairwise Attribute Distribution:** We extend the methodology of 1-way marginals to instead measure the distributions of 2-way marginals as well. As before we measure the TVD of the 2-way marginals vectors of the original and synthetic data and show the average over all 2-way marginals.

(3) **Pairwise Correlation Similarity:** As in previous work [51] we measure the Cramer's V with bias correction measure between each pair of attributes. Cramer's V, which lies in the range $[0,1]$, is a measure of correlation between two attributes with higher values signifying a larger correlation. We measure the difference between the measures in the synthetic and original data and show the average of the differences across all pairs of attributes, a measure we call the average correlation difference (ACD).

(4) **Downstream Classification Accuracy:** Here we use the synthetic data to train several classifiers and use it to classify the original data. We report the accuracy $\frac{TP+TN}{TP+FP+FN+TN}$ for a model trained on each of the 10 output samples.

**Fairness metrics.** We also measure the impact that training on the fairness constrained synthetic data has on downstream classification fairness. For each of these measures, we trained a MLP, Random Forrest, and Linear Regression models on the synthetic data and measured commonly used empirical indicators of fairness.

In order to establish our results, we employ many of the same measures used in [46].

(1) **Demographic Parity [14, 25]:** Demographic Parity measures the difference in the rate at which each class is classified as having an income of higher than $50k$.

(2) **True Positive Rate Balance [9, 48]:** This measures the overall rate of true positives across both the privileged and non-privileged group.

(3) **True Negative Rate Balance [9, 21]:** This measures the overall rate of true negatives across both the privileged and non-privileged group.

(4) **Conditional Measures [10]:** We measure the expected value of the above metrics conditioned on the admissible attributes. This more directly captures the notion of justifiable fairness as the admissible variables are directly allowed be used regardless of their discriminatory effects. However, discrimination within groups with identical admissible attributes is prohibited. This measures the discrimination within each group with identical admissible attributes. These measures are abbreviated as their non-conditional counterparts with an additional $C$ prior.

## 5.2 Experimental Results

We first give an overview of all aspects of the results, using Table 3, and then give an in-depth analysis of the quality and fairness results, using Figures 7 and 8.

**Overview of the performance Greedy-Marginals-MST.** Table 3 gives an overview of the experimental results for Greedy-PreFair (Algorithm 2) compared to MST across all datasets with a singular privacy budget of $\epsilon = 1$. Across all datasets, the cost of fairness is relatively low and does not significantly hinder accuracy.

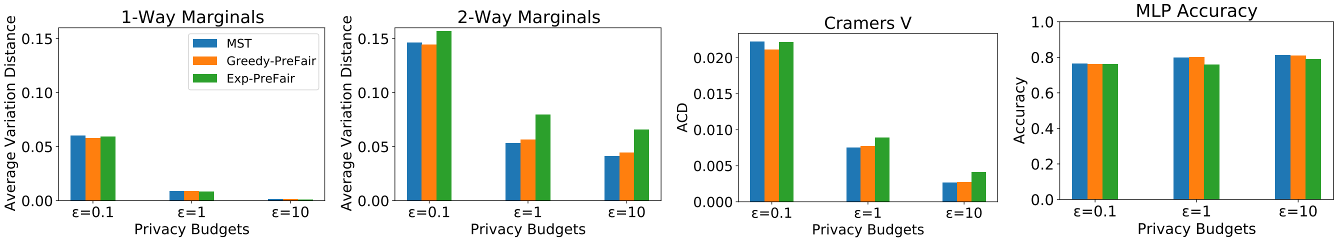
In both the Adult and KDD datasets, Greedy-PreFair incurs no more than a 2% increase in error, on both 1-way and 2-way marginals, due to the additional fairness constraints. Greedy-PreFair incurs a larger error on the Compas dataset, on the 2-way marginals, incurring an almost 20% increase in error compared to MST. This is a result of the greater inequities that exist in the Compas dataset. In order for PreFair to satisfy the fairness constraint, the mechanism generates a distribution that has a greater deviation than the fairer datasets. Since each of the mechanisms only differ in how they measure the 2-way marginals this greater deviation is only represented in the 2-way marginals. This is similarly reflected in the downstream MLP accuracy as Greedy-PreFair incurs a relatively small penalty in accuracy compared to MST for both the Adult and KDD datasets but incurs a larger penalty for the Compas dataset.

In terms of fairness metrics, Greedy-PreFair yields better results on all of the fairness measures regardless of the dataset. Since Greedy-PreFair ensures independence when conditioned on the admissible attributes the conditional fairness metrics are significantly reduced. This also impacts the non-conditional measures as they are also reduced albeit by a smaller amount. While all the measures improved by a similar percentage since both the adult and KDD datasets had little disparity originally, the overall change is less significant than the change in the Compas dataset, which has a significantly higher base rate of disparity. We show the significance of these changes in Figure 8.

We have also measured the overall runtime for Greedy-PreFair compared to MST. Greedy-PreFair performs slightly better than MST in most cases and the improvement increases with the increase in the size of the dataset (Table 3). This is a result of the greedy optimization where Greedy-PreFair considers fewer options than

**Table 3: Overview of Greedy-PreFair's (Algorithm 2) Performance against MST. All values are the percent of Greedy-PreFair's performance to MST's. Data quality values use $\epsilon = 1$ and ML values use the MLP model. Green cells show the favorable scenarios for Greedy-PreFair.**

| | Quality Measures | | | | Fairness Measures | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Dataset** | **TVD 1-Way** | **TVD 2-Way** | **ACD** | **Accuracy** | **DP** | **TPRB** | **TNRB** | **CDP** | **CTPRB** | **CTNRB** | **Runtime** |
| **Adult** | 100.5% | 106.0 % | 131.6 % | 99.8% | 64.4% | 67.9% | 10.2% | 64.8% | 25.7% | 11.7% | 92.9% |
| **Compas** | 96.5% | 119.8% | 103.3% | 86.2% | 66.2% | 70.3% | 73.0% | 50.0% | 50.5% | 53.1% | 97.8% |
| **KDD** | 101.4% | 1.00.8% | 1.04.3 % | 99.6% | 62.5 % | 81.8 % | 24.4% | 22.5% | 31.8% | 24.4% | 92.8% |



**(a) 1-Way Marginals** **(b) 2-Way Marginals** **(c) Cramers V Correlation** **(d) MLP Accuracy**

**Figure 7: Data Quality Measures of the Synthetic Data For Adult**

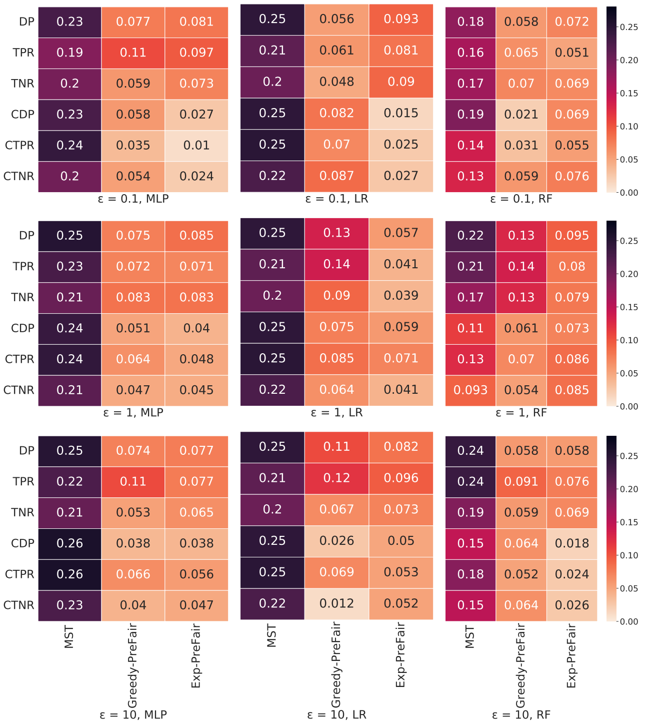


**Figure 8: Fairness measurements for the Compas dataset.**

MST for neighbors of outcome attributes. We measured the runtime of Exponential-PreFair as well. Even on small datasets it takes 10 times longer than either MST or Greedy-PreFair and is prohibitively expensive to run on a large dataset[2].

**Data quality.** Figure 7 shows a detailed breakdown of the quality measures across different privacy budgets and both Greedy and Exponential - PreFair. Greedy-PreFair performs similarly to MST in all cases. In Figure 7a we can see that in the worst case when $\epsilon = 0.1$ Greedy-PreFair has only a 1% increase in error. In Figure 7b we see that this increases to a 7% increase in error across 2-way marginals when compared to MST. While Exponential-PreFair (Algorithm 1) performed similarly to MST on 1-way marginals (Figure 7a), it performed poorly on 2-way marginals (Figure 7b). At $\epsilon = 0.1$ Exponential-PreFair sees a 7% increase but when the privacy budget increases to $\epsilon = 10$ that the increase in error is as large as 60%. Despite the additional optimization in choosing a Marginals-MST, Exponential-PreFair invokes the Gaussian mechanism significantly more often than either MST or Greedy-PreFair. This results in less privacy budget for each invocation and thus more noise per measurement leading to more error. In Figure 7c we can see that the patterns are similar to the 2-way marginals. Greedy-PreFair incurs a small 1% increase in error when the Exponential-PreFair sees a 56% increase in error at $\epsilon = 10$. Figure 7d shows the downstream classification accuracy of MLP models. The results on the Random Forrest models and linear regression models are largely similar and have been omitted from the figure.

The difference in classification accuracy between MST and Greedy-PreFair never exceeds 1% even across all classifiers and privacy budgets. Exponential-PreFair performs similarly in most cases but performs slightly worse than the baseline in the low privacy setting. When $\epsilon = 10$, the difference between the accuracy of the MLP model trained on MST and Exponential-PreFair data is 4%.

**Fairness metrics.** Figure 8 showcases the fairness measures in more detail for the Compas dataset. Every entry in the matrix shows the median results over 10 runs for a specific approach measured by a specific associational fairness metric, where lower numbers (denoted by lighter colors) indicate better fairness. Overall, our methods decrease observed unfairness regardless of the downstream classifier used and regardless of the privacy budget. The tradeoff comes in the quality of data. While the downstream classifiers remain fair regardless of the privacy budget the overall accuracy and

[2] Exponential-PreFair has a significantly higher runtime overhead. As Exponential-PreFair runs in exponential time we were only able to measure performance on the Adult and Compas datasets. In the low privacy regime ($\epsilon = 0.1$), Exponential-PreFair can take up to 10 times longer than MST, taking on average 1344 seconds to run. In larger datasets, this overhead becomes prohibitively expensive as it scales exponentially in the number of attributes.

quality of the data decreases with increased privacy. MST shows significant disparities in both conditional and unconditional measures (> 0.15). Generating data using either of the fair methods results in a significantly fairer downstream classifier. On average, the unconditional fairness metrics of Greedy-PreFair are 45% of those seen with MST and the conditional fairness metrics are 20% of those seen with MST. This is expected, as our solutions ensures that there will be no correlation between sensitive attributes and outcome attributes when conditioned on the admissible attributes. This results in a reduction in the unconditional associational metrics while the conditioned metrics are more significantly reduced. Of the three classifiers tested, the linear regression model benefited the most from the fair synthetic data. On average, when using data from the fair mechanisms the linear regression model saw a decrease in unconditional fairness metrics to 35% of that observed when using the data from MST. The conditional fairness metrics dropped to 14% of what was observed when using data from MST.

## 6 RELATED WORK

We give a detailed review of related works. *To the best of our knowledge, this is the first work that proposes an approach for generating fair synthetic data while satisfying DP.*

**Differentially private data generation.** By releasing synthetic data one can release a large dataset that can be used for an unbounded amount of computation, all while preserving privacy. There are many methods for generating differentially private synthetic data [3, 7, 19, 20, 23, 30–32, 34, 35, 45, 49, 52, 55, 60] including methods that use low level marginals to approximate the data [34, 35, 58], and GAN based methods [23, 45, 55] among others. According to past work [51] the marginals-based approaches such as MST [34], PrivBayes [58] and MWEM-PGM [35] tend to perform best in the private setting. Kamino [19] also provides DP synthetic data generation with constraints, but focuses on denial constraints [8], rather than causality-based constraints.

**Fair data generation.** There have been many works that define fairness in different ways. In particular, our approach relies on justifiable fairness [46] which is a causal-based notion of fairness. Other works consider associational fairness measures based on some statistical measures relative to the protected attributes. Additionally conditional associational fairness measures [10] have been proposed. These measures consider a set of attributes to be conditioned on. Likewise, there have been approaches for generating non-private fair synthetic data [53, 56] GAN based approaches such as FairGAN [56] have been used to generate data that satisfies these associational measures. While GAN based approaches work well in the non-private setting, they have been shown to perform poorly in the private setting [51].

## 7 EXTENSIONS OF PREFAIR

Here we detail several possible extensions. These include expanding the techniques used in PreFair to other private synthetic data mechanisms such as PrivBayes [58] as well as extending the techniques shown here to numerical and continuous data.

**Other private data generation mechanisms.** While PreFair relied on MST [34], all of our propositions in Section 3 apply to any Bayes network for synthetic data generation which samples from a generated graphical model both private and non-private. This applies to private mechanisms beyond just MST. For instance, both PrivBayes [58] and MWEM-PGM [35] generate a graphical model in this way and sample from it directly. To adapt these mechanisms, one only needs to change the selection step, where the graphical model is generated. The selection step must be changed to either consider only admissible attributes as neighbors of outcome attributes (as in Algorithm 2) or find the optimal fair Marginals-MST (as in Algorithm 1).

**Additional data types.** MST [34] along with other marginal approaches takes discrete data as input. These techniques can handle numeric attributes by first using a discretization step to map the numeric domain into a discrete domain. This can be done using using a data independent method or a private subroutine such as PrivTree [59]. Previous work [51] has shown that a data dependent discretization can lead to a large increase in performance overall, particularly in large datasets. In these cases, there is a tradeoff between the expressiveness and performance of the dataset. A fine-grained discretization is more expressive but leads to higher noise while coarse-grained discretization loses some information but can result in a lower overall error. In Section 5, we evaluate PreFair on several datasets containing mostly categorical data and discretize numerical attributes with single-value buckets. In future work, it would be intriguing to assess the impact of different discretization approaches on the quality and performance of our system.

**Additional fairness definitions.** Justifiable fairness is only one of many different fairness definitions, both causal and non-causal [9, 10, 14, 21, 25, 48]. While we show in Section 5 that satisfying justifiable fairness also improves additional fairness metrics, there are no guarantees that they are satisfied. As such we leave it to future work to generate private synthetic data that also satisfies additional fairness definitions. Likewise, incorporating integrity constraints in PreFair is an important future work that will allow the generated data to have other desired properties. While past work [19] has incorporated such constraints into private synthetic data, those solutions do not consider fairness.

## 8 CONCLUSION

We have presented PreFair, a system for generating fair synthetic data in a DP manner. Our approach relies on a state-of-the-art DP data generation system, as well as the definition of justifiable fairness. We have formally defined the problem of generating fair synthetic data that preserves DP and showed that it is NP-hard. Bearing this in mind, we devised two algorithms that guarantee fairness and DP, and have further extended our greedy approach to other DP synthetic data generation systems. We have experimentally shown that our approach provides significant improvements in the fairness of the data while incurring low overhead in terms of faithfulness to the original private data.

## ACKNOWLEDGMENTS

This work was supported by the NSF awards IIS-1703431, IIS-1552538, IIS-2008107, and NSF SATC-2016393.

# REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/ Software available from tensorflow.org.

[2] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *SIGSAC*. 308–318.

[3] Sergül Aydöre, William Brown, Michael Kearns, Krishnaram Kenthapadi, Luca Melis, Aaron Roth, and Amaresh Ankit Siva. 2021. Differentially Private Query Release Through Adaptive Projection. In *ICML*, Vol. 139. 457–467.

[4] K. Bache and M. Lichman. 2013. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[5] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2021. Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods & Research* 50, 1 (2021), 3–44.

[6] Flávio P. Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. 2017. Optimized Pre-Processing for Discrimination Prevention. In *NIPS*. 3992–4001.

[7] Dingfan Chen, Tribhuvanesh Orekondy, and Mario Fritz. 2020. GS-WGAN: A Gradient-Sanitized Approach for Learning Differentially Private Generators. In *NIPS*.

[8] Jan Chomicki and Jerzy Marcinkowski. 2005. Minimal-change integrity maintenance using tuple deletions. *Inf. Comput.* 197, 1-2 (2005), 90–121.

[9] Alexandra Chouldechova. 2017. Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments. *Big Data* 5, 2 (2017), 153–163.

[10] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic Decision Making and the Cost of Fairness. In *SIGKDD*. ACM, 797–806.

[11] T. M. Cover and Joy A. Thomas. 2005. *Elements of information theory.* Wiley-Interscience.

[12] Irit Dinur and Kobbi Nissim. 2003. Revealing information while preserving privacy. In *PODS*. 202–210.

[13] Cynthia Dwork. 2006. Differential Privacy. In *Automata, Languages and Programming*, Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener (Eds.). Springer Berlin Heidelberg.

[14] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. 2012. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. 214–226.

[15] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* (2014).

[16] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and Removing Disparate Impact. In *SIGKDD*. 259–268.

[17] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. 2017. Fairness testing: testing software for discrimination. In *ESEC/FSE*. 498–510.

[18] Georgi Ganev, Bristena Oprisanu, and Emiliano De Cristofaro. 2022. Robin Hood and Matthew Effects: Differential Privacy Has Disparate Impact on Synthetic Data. In *ICML*, Vol. 162. 6944–6959.

[19] Chang Ge, Shubhankar Mohapatra, Xi He, and Ihab F. Ilyas. 2021. Kamino: Constraint-Aware Differentially Private Data Synthesis. *Proc. VLDB Endow.* 14, 10 (2021), 1886–1899.

[20] Moritz Hardt, Katrina Ligett, and Frank Mcsherry. [n.d.]. A Simple and Practical Algorithm for Differentially Private Data Release. In *NIPS*. Curran Associates, Inc., 2339–2347.

[21] Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. 2016. Equality of Opportunity in Supervised Learning. In *NIPS*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29.

[22] Moritz Hardt and Guy N. Rothblum. 2010. A Multiplicative Weights Mechanism for Privacy-Preserving Data Analysis. In *FOCS*. 61–70.

[23] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. 2019. PATE-GAN: Generating Synthetic Data with Differential Privacy Guarantees. In *ICLR*.

[24] Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. 2017. Avoiding Discrimination through Causal Reasoning. In *NIPS*. 656–666.

[25] Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In *ITCS*, Vol. 67. 43:1–43:23.

[26] Ios Kotsogiannis, Yuchao Tao, Xi He, Maryam Fanaeepour, Ashwin Machanavajjhala, Michael Hay, and Gerome Miklau. 2019. PrivateSQL: A Differentially Private SQL Query Engine. *Proc. VLDB Endow.* 12, 11 (2019), 1371–1384.

[27] Joseph B. Kruskal. 1956. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proc. Amer. Math. Soc.* 7, 1 (1956), 48–50.

[28] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual Fairness. In *NIPS*. 4066–4076.

[29] Chao Li and Gerome Miklau. 2013. Optimal error of query sets under the differentially-private matrix mechanism. In *ICDT*. 272–283.

[30] Haoran Li, Li Xiong, Lifan Zhang, and Xiaoqian Jiang. 2014. DPSynthesizer: Differentially Private Data Synthesizer for Privacy Preserving Data Sharing. *Proc. VLDB Endow.* 7, 13 (2014), 1677–1680.

[31] Ninghui Li, Zhikun Zhang, and Tianhao Wang. 2021. DPSyn: Experiences in the NIST Differential Privacy Data Synthesis Challenges. *CoRR* abs/2106.12949 (2021).

[32] Terrance Liu, Giuseppe Vietri, Thomas Steinke, Jonathan R. Ullman, and Zhiwei Steven Wu. 2021. Leveraging Public Data for Practical Private Query Release. In *ICML*, Vol. 139. 6968–6977.

[33] Ryan McKenna, Gerome Miklau, Michael Hay, and Ashwin Machanavajjhala. 2018. Optimizing Error of High-dimensional Statistical Queries Under Differential Privacy. *PVLDB* 11, 10 (2018).

[34] Ryan McKenna, Gerome Miklau, and Daniel Sheldon. 2021. Winning the NIST Contest: A scalable and general approach to differentially private synthetic data. *CoRR* abs/2108.04978 (2021).

[35] Ryan McKenna, Daniel Sheldon, and Gerome Miklau. 2019. Graphical-model based estimation and inference for differential privacy. In *ICML*, Vol. 97. 4435–4444.

[36] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *FOCS*. 94–103.

[37] Ilya Mironov. 2017. Rényi Differential Privacy. In *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*. IEEE Computer Society, 263–275. https://doi.org/10.1109/CSF.2017.11

[38] Razieh Nabi and Ilya Shpitser. 2018. Fair Inference on Outcomes. In *AAAI*. 1931–1940.

[39] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian J. Goodfellow, and Kunal Talwar. 2017. Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data. In *ICLR*.

[40] Judea Pearl. 1995. Causal diagrams for empirical research. *Biometrika* 82, 4 (1995), 669–688.

[41] Judea Pearl. 2009. *Causality.* Cambridge university press.

[42] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[43] ProPublica. [n.d.]. Compas recidivism risk score data and analysis. https://www.propublica.org/datastore/dataset/compas-recidivism-risk-score-data-and-analysis

[44] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. 2022. A survey on datasets for fairness-aware machine learning. *WIREs Data Mining and Knowledge Discovery* 12, 3 (mar 2022). https://doi.org/10.1002/widm.1452

[45] Lucas Rosenblatt, Xiaoyan Liu, Samira Pouyanfar, Eduardo de Leon, Anuj Desai, and Joshua Allen. 2020. Differentially Private Synthetic Data: Applied Evaluations and Enhancements. *CoRR* abs/2011.05537 (2020).

[46] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional Fairness: Causal Database Repair for Algorithmic Fairness. In *SIGMOD*. 793–810.

[47] Andrew D. Selbst, danah boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and Abstraction in Sociotechnical Systems. In *FAT*. 59–68.

[48] Camelia Simoiu, Sam Corbett-Davies, and Sharad Goel. 2017. The problem of infra-marginality in outcome tests for discrimination. *The Annals of Applied Statistics* 11, 3 (2017), 1193–1216.

[49] Joshua Snoke and Aleksandra B. Slavkovic. 2018. pMSE Mechanism: Differentially Private Synthetic Data with Maximal Distributional Similarity. In *PSD*, Vol. 11126. 138–159.

[50] Harini Suresh and John V. Guttag. 2021. A Framework for Understanding Sources of Harm throughout the Machine Learning Life Cycle. In *EAAMO*. 17:1–17:9.

[51] Yuchao Tao, Ryan McKenna, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. 2021. Benchmarking Differentially Private Synthetic Data Generation Algorithms. *CoRR* abs/2112.09238 (2021).

[52] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. 2019. DP-CGAN: Differentially Private Synthetic Data and Label Generation. In *CVPR*. 98–104.

[53] Boris van Breugel, Trent Kyono, Jeroen Berrevoets, and Mihaela van der Schaar. 2021. DECAF: Generating Fair Synthetic Data Using Causally-Aware Generative Networks. In *NIPS*. 22221–22233.

[54] Sahil Verma and Julia Rubin. 2018. In *FairWare@ICSE*. 1–7.

[55] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. 2018. Differentially Private Generative Adversarial Network. *CoRR* abs/1802.06739 (2018). arXiv:1802.06739 http://arxiv.org/abs/1802.06739

[56] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. 2019. FairGAN$^+$: Achieving Fair Data Generation and Classification through Generative Adversarial Nets. In *IEEE BigData*. 1401–1406.

[57] Ryo Yoshinaka. 2005. Higher-Order Matching in the Linear Lambda Calculus in the Absence of Constants is NP-Complete. In *Proceedings of the 16th International Conference on Term Rewriting and Applications* (Nara, Japan) *(RTA'05)*. Springer-Verlag, Berlin, Heidelberg, 235–249. https://doi.org/10.1007/978-3-540-32033-3_18

[58] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2014. PrivBayes: private data release via bayesian networks. In *SIGMOD*. 1423–1434.

[59] Jun Zhang, Xiaokui Xiao, and Xing Xie. 2016. PrivTree: A Differentially Private Algorithm for Hierarchical Decompositions. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, Fatma Özcan, Georgia Koutrika, and Sam Madden (Eds.). ACM, 155–170. https://doi.org/10.1145/2882903.2882928

[60] Zhikun Zhang, Tianhao Wang, Ninghui Li, Jean Honorio, Michael Backes, Shibo He, Jiming Chen, and Yang Zhang. 2021. PrivSyn: Differentially Private Data Synthesis. In *USENIX*. 929–946.

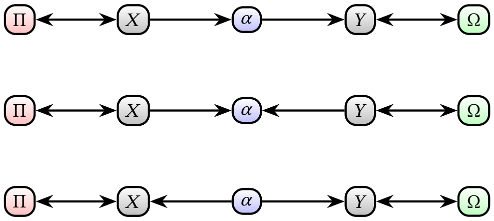


**Figure 9: Possible paths for the proof of Proposition 2.**

# A PROOFS

## A.1 Proofs for Section 3

Proof of Proposition 1. The proof is largely the same as that in past work [46]. Note that for any choice of $K \supseteq A$, if all the directed paths go through at least one attribute in $K$ then intervening on $K$, that is $do(K = k)$ then at least one of the edges in the path will always be severed. This will separate all nodes in $O$ from those in $P$ and thus any further intervention on $P$ cannot influence the distribution of $O$. □

Proof of Proposition 2. Here, we show that given a set $K \supseteq A$ that the probability of the outcome is dependent only on the intervention on $K$ and is the same regardless of the value of $p$. We rely on a fact from [40] namely the following equivalence on the do operator.

$$P(O = o|do(K = k), do(P = p)) = P(O = o|do(K = k)) \quad \text{if}(O \perp\!\!\!\perp P|K)_{\overline{PK}} \tag{5}$$

Where $\overline{PK}$ denotes the distribution derived by removing all edges pointing to attributes in $P$ and $K$.

This will allow us to show that the outcome is independent of *any intervention on* $P$ once we have already intervened on $K$,i.e., applying the *do* operator on $P$ does not change the probability. In particular, we will be able to show that $P[O = o|do(P_i = 0), do(K = k)] = P[O = o|do(P_i = 1), do(K = k)] = P[O = o|do(K = k)]$, which is the definition of K-fairness.

We also highly rely on the concept of d-separation a notion that can be directly observed on the graph. We say two attributes $X$ and $Y$ are d-connected (and thus not d-separated) conditioned on a set of attributes $Z$ if any of the following hold.

- There exists an undirected path from $X$ to $Y$ that is not blocked by $Z$
- There exists an undirected path from $X$ to $Y$ that is not blocked by a collider $a \notin Z$

Where a collider is an attribute along the undirected path where both edges in the path are directed towards that attribute.

Let $K$ be any superset of $A$ which shares no elements with $P$. Note that in any directed Marginals-MSTthere is exactly one undirected path between any attribute in $P$ and any attribute in $O$. Likewise this path is always blocked by at least one attribute in $K$.

Therefore the path from the attribute in $P$ to an attribute in $O$ must have one of the structures shown in Figure 9 where $X$ and $Y$ denote arbitrary sets of nodes along the path with arbitrary directed edges between them. Double edges denote an edge that can be in any arbitrary direction. Π denotes an attribute in $P$, Ω denotes an attribute in $O$ and $\alpha$ denotes an attribute in $A$. Note that the path must either pass through $\alpha$ (the first case), have edges that go to $\alpha$ (the second case) or have edges that originate from $\alpha$ (the third case).

In the graph $\overline{PK}$ where all edges going into attributes in $P$ and $K$ are removed each of these cases introduces a d-separation between Ω and Π when conditioned on $K$. In the first case, the edge going into $\alpha$ is removed and as such the path is severed. In the second case, both edges going to $\alpha$ are removed and the path is severed. In the third case, no edges are severed but conditioning on $\alpha$ d-separates Π and Ω since it blocks the path and is not a collider.

Since the dataset is Markov compatible with the graph then the d-separation of Ω and Π imply that the two are independent when conditioned on $\alpha$. Since they are conditionally independent we can use Equation (5) to show that $P(O = o|do(K = k), do(P = p))$ is the same regardless of the intervention on $P$. Therefore, the distribution satisfies K-fairness. Likewise since $K$ could be an arbitrary superset of $A$ then the distribution also satisfies justifiable fairness. □

Proof of Proposition 4. We show that, given an instance of 3-SAT, we can define an equivalent instance of the decision version of FDPSynth, where any maximal spanning tree will correspond to a satisfying 3-SAT assignment.

**Construction.** First we describe the construction of a graph $\mathcal{G}$ from a 3-SAT instance $\varphi$ with $m$ clauses and $n$ literals. We create an assignment gadget (Figure 4) for each literal in $\varphi$. This contains a protected attribute $\Pi_i$ and edges from the protected attribute to two additional attributes representing the literal and its negation ($x_i$ and $\neg x_i$ respectively). For each clause in $\varphi$ we create a 3-way OR gadget (Figure 5). The $\alpha$ nodes in the OR gadget are admissible attributes and the Ω nodes are outcome attributes. Each of the inputs for the 3-way or gate $(x_1, x_2, x_3)$ are then connected to the corresponding literal in the assignment gadget using a weight 3 edge. We now prove that there is a solution to the decision version of FDPSynth (Definition 12) with $k = 22m + 2n$

To build the intuition behind the 3-way OR gadget we first describe the 2-way OR gadget (Figure 10) and build upon that. This gadget takes two inputs $x_1$ and $x_2$. These may be either connected to an assignment gadget or the output of another gadget. Note that in order to make the maximum spanning tree across this substructure both edges with weight 2 must be taken and at least one edge with weight 1 must be taken resulting in a total weight of 5. We recall that we consider a literal to be set to *False* if there exists an unblocked path from a protected attribute to it. We now consider the output of the two way OR gadget $O$. This node will be considered set to *False* if there is an unblocked path between a protected attribute and $O$. For either of the possible maximum spanning trees on the 2-way OR gadget only one of the inputs can have a path to $O$ that is blocked by $\alpha$. Therefore, for either of the maximum spanning trees, if both inputs are set to *False* only one of them may be blocked by $\alpha$ and the other will have a direct edge to the output resulting in an output of *False*. As such any maximum spanning tree on the 2-way OR gadget evaluates to *True* if at least one of its inputs evaluates to *True* resulting in the same functionality as an OR gate. Similar to traditional OR gates in order to construct the 3-way OR gadget we simply apply two 2-way OR gadgets together with the only exception being that the output of the second 2-way or gadget is an outcome attribute. This ensures that if a tree over

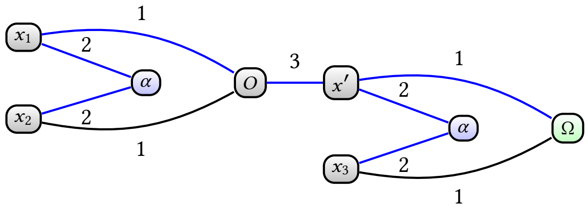


**Figure 11: Example 3-Way OR gadget. The satisfying edges are highlighted in blue**

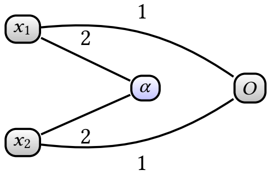


**Figure 10: 2-way OR gadget.**

the 3 way or gadget has the maximum weight (13) and satisfies Proposition 1 then it corresponds to a satisfying assignment for the corresponding clause.

iff $\varphi$ has a satisfying assignment.

**Proof.** ($\Rightarrow$) Assume that we have a fair Marginals-MST $T$ of the constructed graph $G$. If $T$ has the maximum weight of $22m + 2n$, where $m$ is the number of clauses and $n$ is the number of literals, and then the assignment to each literal can be read by looking at each assignment gadget. In $T$ only one of the edges from each assignment gadget will be kept due to the tree condition. Whichever node has an edge to the protected attribute $\Pi_i$ will be set to $False$ and the other will be set to $True$. This assignment is consistent as the literal in the assignment gadget in Figure 4 will have an edge to every 3-way OR gadgets that corresponds to clauses which have that literal. This ensures that all 3-way OR clauses which have $x_1$ will have an edge to $x_1$ in the assignment gadget and those that have $\neg x_1$ have an edge to $\neg x_1$ in the assignment gadget. Since only one edge can be taken in the assignment gadget then either $x_1$ or $\neg x_1$ can be set to $False$ but not both.

Recall that in order for the minimum spanning tree over the 3-way OR gadget to be maximum and fair at least one literal must be set to $True$. Because of this any global Marginals-MSTof the maximum weight of $13m + 5n$ ensues that each clause has at least one literal set to $True$.

($\Leftarrow$) Here we show that given a satisfying assignment to $\varphi$, $a$, we can construct a unique fair Marginals-MST on $G$. In order to show that a minimum spanning tree of $G$ always corresponds to a 3-SAT solution we will rely heavily on the following fact about MSTs: for any spanning tree on graph $G$ to be an MST all the subtrees on smaller connected components must also be a minimum spanning trees for each of those components. This allows us to reason about the smaller parts of the tree (mainly the OR gadget) without considering the rest of the tree.

Now given a satisfying assignment to $\varphi$, $a$, we can construct the corresponding fair Marginals-MST. First, select the edge that correctly assigns the attributes according to the assignment gadget. That is if the assignment sets $x_1$ to $True$ select $(\Pi\neg, x_1)$ in Figure 4 and $(\Pi, x_1)$ otherwise. Then for each 3-way OR gadget (Figure 5) take all four weight 2 edges.

In the case of the 3-way OR gadget all edges will refer to the notation of Figure 5. Since $a$ is a satisfying assignment for $\varphi$, at least one of the 3 literals $x_i$ will evaluate to $True$ . If this literal is connected to the first 2-way OR gadget , select its 1 weight edge to be connected to the output $(x_i, O)$ and the 1 weight edge from the output to the outcome node $(x', \Omega)$. If the literal is connected to the second 2-way OR gadget either weight 1 edge $(x_1, O)$ or $(x_2, O)$ from the first 2 weight or gadget can be selected and the weight 1 edge from the literal to the outcome $(x_3, \Omega)$ will be selected in the second 2-way OR gadget. We give an example of such a 3-WAY OR gadget in Figure 11 where the edges to be taken are highlighted in blue. This corresponds to the assignment $(x_1 = True, x_2 = False, x_3 = False)$. Taking these edges both maximizes the weight as it takes all the 2 weight edges and only one 1 weight edge from either 2-way or gadget. This also ensures that the resulting Marginals-MST is fair since each outcome in the 3-way OR gadget has paths all blocked by admissible attributes. In each case for any maximal spanning tree at most one path from the literals to the outcome may be blocked since only one of the weigh 1 edges can be taken.

□

Proof of Lemma 1. Here we first show that we can create a set of random variables $A, B, C, D$ which results in the graph where $B, C, D$ all have arbitrary mutual information with $A$ and there is no mutual information between any other pair of variables. From there we demonstrate that we can extend this construction to an arbitrary number of variables each with their arbitrary weight. We argue that this construction is sufficient to be able to construct a graph with the structure of the 3-way OR gadget.

Let $B, C$ and $D$ be discrete uniform random variables with domain size $n$ for some $n \in \mathbb{N}$. Let $A$ be the discrete random variable with the following probability distribution.

$$\begin{cases} A = B & \text{With Probability } x \\ A = C & \text{With Probability } y \\ A = D & \text{With Probability } z \\ A = 0 & \text{With Probability } \lambda \end{cases}$$

Where $x + y + z + \lambda = 1$. The entropy of $A$ from is as follows.

$$H(A) = \\ -\left(x\log(x/n) + y\log(y/n) + z\log(z/n) + (\lambda)log(\lambda)\right) \tag{6}$$

Likewise the conditional entropy is as follows.

$$H(A|B) = \\ -\left(x\log(x) + y\log(y/n) + z\log(z/n) + (\lambda)log(\lambda)\right) \tag{7}$$

As such the mutual information is as follows.

$$\begin{aligned} I(A;B) = H(A) - H(A|B) = \\ x\log(x) - x\log(x/n) = \\ x(\log(x) - \log(x/n)) = \\ x\log(n) \end{aligned} \tag{8}$$

The same can be done for the other random variables $B$ and $C$. It is clear that $B, C$ and $D$ have mutual information 0 since they are all uniform random variables.

The only constraint is that $x + y + z + \lambda = 1$. Given a sufficiently large $n$ we can set $x \log(n)$ to be any arbitrary value and the same goes for the other two attributes. For example if for example we would like to make the relationship between $O, x', \alpha$ and $\Omega$ shown in Figure 5 we would do the following construction.

Let $A$ take the place of $x'$, $B$ take the place of $O$, C take the place of $\alpha$ and $D$ take the place of $\Omega$. Let $n = 2^{12} = 4096, x = \frac{1}{4}, y = \frac{1}{6}, z = \frac{1}{12}$ and $\lambda = \frac{1}{2}$. First these values of $x, y, z$ and $\lambda$ satisfy the constraint as $\frac{1}{4} + \frac{1}{6} + \frac{1}{12} + \frac{1}{2} = 1$. Likewise these probabilities give us the desired mutual probabilities as $x \log(n) = \frac{1}{4}\log(4096) = 3$. Likewise $y \log(n) = 2$ and $z \log(n) = 1$ which is what we need for $A = x'$, $B = \alpha$, $C = \Omega$, and $D = O$ in Figure 5.

We can extend this to any arbitrary number of edges as well as any value of mutual information. This only requires a larger value of $n$. If we would like $\zeta$ edges with the maximum mutual information $I$ $n$ must be at least $2^{\zeta I}$

□

## A.2 Proofs for Section 4

Proof of Proposition 5(1). Using the Gaussian Mechanism (line 4) with scale $\sigma$ satisfies $(\alpha, \frac{1}{2\sigma^2})$-RDP. If we set $\sigma = \sqrt{\frac{r}{2\rho}}$ this becomes $(\alpha, \alpha\frac{\rho}{r})$-RDP. This is invoked $r$ times and is the only line that directly accesses private data thus by Theorem 2 it thus satisfies $(\alpha, \alpha\rho)$-RDP. □

Proof of Proposition 5(2). When Algorithm 1 creates incomplete spanning tree to add to the priority queue it rejects any partial spanning tree that violate Proposition 1. Therefore the resulting output satisfies Proposition 1. □

Proof of Proposition 7. Let $G$ be a graph over the input database $D$ where the nodes represent attributes and edges represent mutual information between attributes. Consider the case where each edge has uniform weight (by Lemma 1 such a database exists). In this case, the weight of any partial spanning tree is merely the number of edges in the partial tree. As such, Algorithm 1 will create every possible spanning trees before selecting any of them . There are $|\mathcal{A}|^{|\mathcal{A}|-2}$ possible spanning trees resulting in the worst case time complexity. □

Proof of Proposition 8(1). Line 8 satisfies $(\alpha, \alpha\frac{1}{8}\epsilon^2)$-RDP. This is the only line that accesses the private data. If we set $\epsilon = \sqrt{\frac{8\rho}{r-1}}$ this becomes $(\alpha, \alpha\frac{\rho}{r-1})$-RDP. This is invoked $r - 1$ times and by Theorem 2 it thus satisfies $(\alpha, \alpha\rho)$-RDP □

Proof of Proposition 8 (2). In line 4, Algorithm 2 removes any edges from nodes in $O$ that do not go to nodes in $O \cup A$. This results in a Marginals-MST where all of the neighbors of nodes in $O$ are in $O \cup A$. Any path from attributes in  to attributes in $O$ must pass through a neighbor of the attribute in $O$ and therefore must pass through an attribute in $A \cup O$. Therefore by Proposition 9 the output of Algorithm 2 is a fair Marginals-MST. □

Proof of Proposition 9. Assume that all the neighbors of nodes in $O$ are in $O \cup A$. Any path from a node in $P$ to a node in $o \in O$ must pass through at least one of $o$'s neighbors. If that neighbor is in $A$ then the path is blocked by an admissible attribute therefore satisfying Proposition 1. If that neighbor was in $O$ then the path must also go through one of its neighbors which must be in $O \cup A$. Since only nodes in $A$ are allowed to have edges to nodes outside of $O \cup A$ then the path must go through at least one of these nodes therefore blocking the path and satisfying Proposition 1.

To prove the converse we use a proof by contradiction. Assume $T$ is a fair Marginals-MST, the problem setting is saturated, and that there is an edge from a node $o \in O$ to a node $x \notin O \cup A$. By the definition of the saturated case since $x$ is not in $O \cup A$ then it must be in $P$. Since $x$ is in $P$ and is neighbor of $o$ then there exists a direction of edges ($x$->$o$) such that $o$ is directly dependent on $x$ resulting in a violation of Definition 10.

□

Proof of Proposition 10. Algorithm 2 greedily selects $|\mathcal{A}| - 1$ edges one at a time using the exponential mechanism (lines 7-9). This process is done exactly $|\mathcal{A}| - 1$ times resulting in a time complexity of $|\mathcal{A}| - 1$. □