Secure Counterfactual Explanations in a Two-party Setting

1st M. Molhoek
Datascience
Toegepast Natuurwetenschappelijk Onderzoek (TNO)
Den Haag, Netherlands
msmolhoek@gmail.com

2nd J. Van Laanen
Datascience
Technical University Eindhoven
Eindhoven, Netherlands
jorisvanlaanen@icloud.com

***Abstract*—When multiple parties want to learn from each others' data, but do not want to share this data becuase it is privacy sensitive, using a federated trained Machine Learning (ML) model is a good option. Explanation of the results are essential to use and and therefore trust the outcome of this trained model. However, explanations reveal sensitive information which is not allowed when using privacy sensitive data. In this paper, we introduce a novel approach generating Counterfactual Explanations (CFEs) in a secure way utilising synthetic data. A CFE provides an example data point, that with the smallest change to the original feature values provides a different outcome. Thereby showcasing what needs to change for a different output. In our case two parties owning different features of the same persons jointly train a ML model. In this setting, one party owns one feature and the other party owns multiple features including the target feature, both data must remain confidential to the other party. A CFE is created by first securely generating vertical distributed synthetic data with the aid of a Split Neural Network (Split-NN). We show that the distributed synthetic data maintain characteristics of the original data in the cases where the predictability is high, and do not reveal sensitive information when under an Attribute Inference Attack. Secondly, synthetic Counterfactuals (CFs) are generated and ranked using secure Multi-Party Computation. The ranking is based on the optimization of a selection of distance metrics from the CFE with respect to the original event. The outcome of the CFE can be revealed to both parties. In this way we provide a complete privacy-preserving pipeline to explain a federated trained ML model on vertically partitioned data.**

***Index Terms*—Explainable AI, Privacy Enhancing Technologies, Counterfactuals, Synthetic Data Generation, Federated Learning, Multi-party Computation**

I. Introduction

Digitization of personal or confidential data has increased tremendously in the current world. The data is often stored at different locations, containing similar events, but different features. This is called vertically partitioned data. It can be undesirable to share vertically partitioned data amongst all parties due to privacy or confidentiality concerns as this reveals sensitive information. However, analysis on these datasets could greatly benefit decision-making and the quality of services. Let's take for example data on diabetes patients; The hospital has information on blood pressure and the GP has information on family history on diabetes. To determine novel approaches that benefit large amounts of diabetes patients, analysis on extensive data is essential and can improve healthcare in general. Since the datasets contains sensitive personal information, the datasets cannot be shared between the GP and the hospital. Another example is the detection of fraudulent transactions across banks and financial institutions.

There is an increase in the use of Machine Learning (ML) models to improve decision-making. However, standard ML models cannot be used when parties have privacy sensitive data. Several methods exist to train federated models on vertically partitioned data in a privacy friendly manner. There are models based on Federated Learning (FL), Multi-Party Computation (MPC), or a combination of both [1]. FL and MPC are both techniques that enable collaboration on sensitive data while maintaining privacy [2].

However, after training a ML model it remains challenging to understand how the model has come to its prediction. Therefore explainability is desired to validate results and increase trust in the model. In the example of the diabetes patients one might be interested in the question: why is the advice to increase exercise the best action for this particular patient? Or in the case of fraud detection; why is a given transaction marked as being fraudulent? For this reason, the field of Explainable Artificial Intelligence (XAI) has gained traction in academic literature [3]. Current approaches in XAI rely heavily on the model input and associated output [4]. The input and the model's output is used to calculate and provide the best explanations. Therefore, when dealing with sensitive vertically distributed data, explainability introduces additional privacy challenges. It is not allowed to reveal the sensitive input data nor the features of that specific event, to explain the outcome to all parties. One promising technique in XAI is Counterfactual Explanations (CFEs). CFEs provide a data point which is very similar to the original event but has a different outcome. This data point in itself contains sensitive information and may therefore not be revealed. As well as there can be attribute inference and explanation linkage attacks performed on these explanations to acquire on specific persons [5]. Therefore an explanation must protect all sensitive information whilst still being informative. For a two-party setting it is vital both parties do not infer more private information on persons in their data. To the authors knowledge there is very limited research on providing privacy-friendly explanations in a secure

ML pipeline involving vertically distributed data.

A. Contributions of the Paper

To create a complete secure pipeline to reveal relevant CFEs we face several technical challenges: 1) How to create representative synthetic data on vertically partitioned data, and 2) How to find the best synthetic CFEs around a local data point in a secure way? For our problem definition we have two parties, of which one party holds one (sensitive) feature. Our Secure Counterfactual Explanations (SCFEs) protocol's main contributions are:

1) Introducing a method for creating synthetic data on vertically distributed data in a two-party using Federated Learning.
2) Proposing a method for calculating secure counterfactual explanations for a single query from synthetic data using cryptography. Enabling the parties expertise to use prior knowledge resulting in feasible counterfactuals.
3) Proofing the integration of the above methods in a single secure pipeline on an open-source dataset (California Housing data).

II. Background Knowledge

In this work a diverse set of state-of-the art techniques is used in the field of Privacy Enhancing Technologies (PETs) and in a novel way combined with explainability of black-box models. Therefore, the authors deem it necessary to elaborate on background knowledge in the areas of PETs and explainability. PETs are technologies that enable users to extract value from data whilst protecting the privacy of individuals in the data. In the following sections the PETs used in this paper are explained, followed by the used XAI technique.

A. Federated Learning

In the field of FL the central model weights are updated based on local updates at the clients. Various methods can be used in a vertical setting. Li et al. provided a comparison of FL algorithms [6] and applicability in a vertical setting as well as availability of open-source code. A special form of FL is the Split Neural Network (Split-NN) proposed by Vepakoma et al. [7], which relies on sharing intermediate representations of the input data rather than updates on the model weights. Essentially partitioning the model itself over multiple parties. The results are especially encouraging in terms of increased communication efficiency and flexibility. The feature maps of the last intermediate layer are combined and shared with a central server that completes the neural network with additional layers and the output layer. All clients have a similar output structure of neural network layers. Several options exist to combine the intermediate feature maps for the central model, for example: averaging, max pooling, summation, multiplication or concatenation [8].

B. Synthetic Data Generation

There have been many advances in the field of creating synthetic data, a lookalike of the original data, over the last decade. The goal of synthetic data is to maintain as much of the properties of the original data whilst simultaneously protecting the privacy and/or sensitive information. There is a trade-off between these two objectives. Several standard statistical methods are used like Random Oversampling [9] and Gaussian Mixture Models [10]. However, there have been important breakthroughs using deep learning models. Generative Adverserial Networks (GAN) are generally outperforming the more standard approaches in capturing the underlying patterns which results in an increased quality of synthetic data as well as protecting the sensitive information [11].

C. Multi-Party Computation

Secure multi-party computation (MPC) is a field of cryptography that enables multiple parties to jointly compute a function on their private inputs, while maintaining the privacy of these inputs from the other parties [12]. In this paper, Shamir's Secret Sharing (SSS) threshold scheme is used in particular.

SSS can be used as a building block for constructing MPC protocols. With this method, each party first uses Shamir Secret Sharing to share their private input, then the shares are used in the computation, and finally the output is reconstructed [13]. In this way, each party only learns the output of the computation, and nothing else about the other parties' inputs.

Another MPC technique utilized in our study is Homomorphic Encryption (HE) which converts data in ciphertext to create a Private Set Intersection (PSI). PSI enables secure computation of the intersection between private sets, providing parties the ability to determine shared elements without revealing individual set contents [14]. This step ensures that both parties can contribute the features that corresponds to a particular sample without learning any other information on the samples they don't have in common.

D. Counterfactual Explanations

In the field of XAI, CFs have risen to the attention. Besides methods as SHAP [15] and LIME [16] which aim to explain how a decision was made, CFs aim to explain why [17] and what needs to change in order to obtain a different model outcome. E.g. why does this diabetes patient gets treatment X prescribed and not Y? CFs can be rated based on their proximity, sparsity, diversity, plausibility and feasibility to the original query [18]. Proximity means the CF should match the original Factual as closely as possible. The CF should be sparse which results in a minimum difference in features to make it easier to interpret the output. The explanations should preferably be diverse such that the user can choose from multiple CFs to interpret the outcome. Plausibility adheres to the required action being actionable e.g. the person cannot change gender. Lastly, feasibility is closely related and focuses more on likelihood of existence of the CF. Various algorithms optimize these aspects and most use distance metrics (see

section IV-A4) to rank and optimize the CF candidates [19]. Some open-source methods to rank CFs are DICE [20], CARLA [21] and MACE [22], which all use versions of iterative loss optimization.

III. Related Work

A. Distributed Vertical Synthetic Data Generation

To the authors' knowledge work on synthetic data generation for vertically distributed tabular data is limited. The main focus in literature is on horizontally distributed tabular. A distributed method for tabular data horizontally partitioned using federated learning is proposed by Zhao et al. [23] called Fed-TGAN with promising result in similarity and training time. The work of Tajeddine et al. [24] proposed a method using DP-SDG and MPC with differential privacy to train a probabilistic generative model in a secure way for vertical partitioned data. However, results on the quality of the synthetically generated data is not provided. Therefor, there is a need for methods to generate vertical synthetic tabular data.

B. Secure Explanations in a Vertical Distributed Setting

Research on the subject of secure multi-party explanations for black-box models trained on vertically distributed data is limited. Previous work on the topic includes Secure Local Foil trees by Veugen et al. [25] that introduces a method to explain a sample by a secure decision tree. The work suggests to securely generate synthetic data by defining a standard numerical distribution around the query. The quality of the synthetic data highly depends on the similarity of the real feature's distribution to a normal distribution. From the Local Foil Trees a rule-based explanation can be extracted to explain a query. Wang et al. [26] studied the explainability of vertical federated models calculating Shapley values in a two-party setting. When the host party calculates feature importance from the guest party, this reveals sensitive information from the guest features. Therefore Wang et al. propose a method to calculate a unified importance value for guest features showing that the experiments are robust and provide informative results. Shapely values focus on explaining the *how* of the outcome, CFEs clarify the *why* of the outcome. The work of Chen et al. [27] introduces an Explainable Vertical Federated Learning framework that minimizes the distributions of CF samples to the query with the Kullback–Leibler (KL) divergence method. The goal of this framework is to filter abundant features to improve model predictions and provide insight in feature importance. When CF candidates are revealed, they reveal the sensitive input data. To the authors perspective basing the feature importance on one query provides a local result and may not equal the global feature importance.

Our study focuses on generating realistic synthetic CF candidates which can be revealed to explain a single query. Our framework optimizes CFEs based on several distance metrics calculated securely with the help of MPC. This approach ensures that CFEs remain vertically distributed throughout computation and do not unintentionally reveal input data or explanations to collaborating parties. As most calculations for the distance metrics are done locally computational time is lowered.

IV. Approach and Evaluation Method

A. Workflow Generating Synthetic Counterfactuals

Our SCFE approach explains the Factual by providing synthetic CFs which shows which features changed to get another output. Our approach is secure in every step. All steps are explained in detail in the following sections.

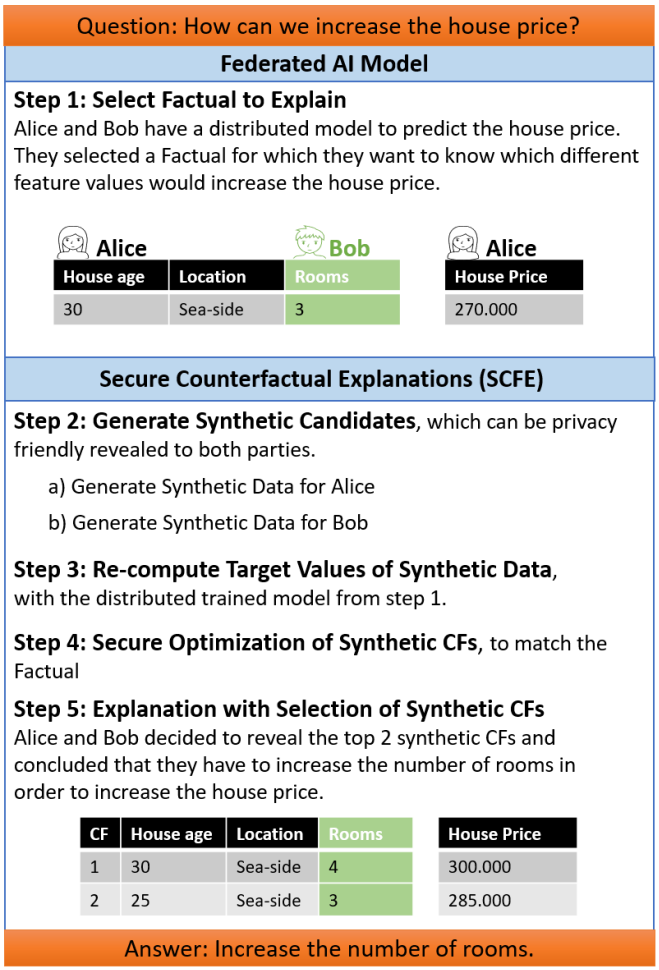


Fig. 1: Schematic overview of steps to calculate Secure CFEs with an example on Housing Data. All steps are explained in detail in the following sections.

1) Step 1: Select Factual to Explain: In this first step, the data owners decide together which data point they want to have explained (the Factual). Define the task of the model's explanation depending on the goal to either classify an event or estimate its value (regression) differently. E.g. why is my event class 1 and not class 0? Or in case of regression; why is my event not 10-15% higher or 10-15% lower?

2) Step 2: Generate Synthetic Candidates: Our method uses the Split-NN from Vepakomma et al [7] in a new way to generate synthetic data on vertically distributed datasets which can be revealed without showing sensitive information. The

method is able to make use of the recent breakthroughs in applying GAN for synthetic data generation. Our set-up has two data-owners, where Alice has most features including the target feature and Bob has one feature. Via the following three steps vertical synthetic data is generated:

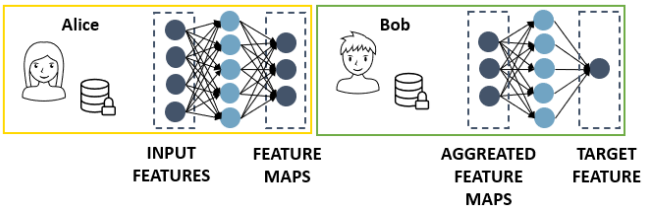


Fig. 2: Schematic overview of proposed vertical Split-NN synthetic data generation approach.

1) Generate synthetic data for only Alice's data locally. We used an open-source method from SDV based on Tabular Conditional GANs [28].
2) Train a Split-NN using the original data of Alice as features to predict the feature owned by Bob, which is the target.
3) Provide the synthetic data from Alice to the trained Split-NN from step 2, to generate the synthetic feature for Bob as an output.

Now all parties have a synthetic version of their original data maintaining cross-party correlations and similarities.

Both parties can put constraints on the synthetic data by checking the validation and feasibility. They can validate whether the created synthetic data lies in the desired outcome range and if the data is feasible (or even plausible) for the event to reach the synthetic samples values. With PSI both parties can securely align input samples before continuing to step 2.

3) Step 3: Re-compute Target Values: To determine why the model generated a certain output, all the classes or continuous values of the original and synthetic data need to be computed with the vertical Split-NN to be explained. Using the same Split-NN ensures that the explanations are relevant to that particular model.

4) Step 4: Secure optimization of Synthetic CFs: To determine the best CFs, first the individual losses are provided by the calculation of the $L0$, $L1$ and $L2$ distances between all synthetic candidates and the selected Factual. In Protocol (1) the distance metrics are calculated in a secure way as the sum of the distances per feature and the ranking is calculated with MPC.

The loss of an individual CF candidate is calculated by equation (1). The U represents the set of parties, α the weight of the $L0$ loss, β the weight of the $L1$ loss, and γ the weight of the $L2$ loss. The $L0$ represents the amount of features that are changed. The $L1$ represents the element-wise distance summed between all CF's features and the sample's features. The $L2$ is the pair-wise distance between all CF's features and the sample's features squared. The $L2$ norm is squared to aggregate it with the other distance metrics and to save precious calculation time via the MPC protocol as parties do not have to jointly calculate the square root. In our experiments we stated that α , β and γ are equal as it is decided that all distances are equally important while calculating the CFs. The users can tweak the constants to their specific requirements. The categorical features are one-hot-encoded and are only taken into account once to calculate the loss.

$$\begin{aligned} loss(\boldsymbol{\delta}_x) &= \sum_{u \in U} loss_u(\boldsymbol{\delta}_x) \\ &= \sum_{u \in U} (\alpha ||\boldsymbol{\delta}_x||_0 + \beta ||\boldsymbol{\delta}_x||_1 + \gamma ||\boldsymbol{\delta}_x||_2^2)_u. \end{aligned} \tag{1}$$

$$\begin{aligned} ||\boldsymbol{\delta}_x||_0 = \lim_{z \to 0^+} \sum_{p \in P} \delta_{x_p}^z, \qquad ||\boldsymbol{\delta}_x||_1 = \sum_{p \in P} \delta_{x_p}, \\ ||\boldsymbol{\delta}_x||_2^2 = \sum_{p \in P} \delta_{x_p}^2. \end{aligned} \tag{2}$$

The MPC protocol is used to rank the CF candidates and reveal the top K indices (see protocol (1). For this purpose the SSS scheme is used from the MPyC library [29]. Using the MPyC library compared to normal un-encrypted code does not affect the correctness of the outcome, as was verified.

Protocol 1 Secure top-k counterfactual candidate ranking

Require: Secret shared losses $[S]$, counterfactual candidates N, parties U, top K number of CFs
Ensure: List containing indices from K smallest counterfactual losses

$L \leftarrow \emptyset$
$Q \leftarrow \emptyset$
for i=0,. . ., N **do** ▷ Aggregate loss for counterfactual candidate i
    $L \leftarrow L \cup \sum_{u \in U} [s_{u,i}]$
end for
[max_loss] = mpc.max(L)
for $k = 0 \ldots, K$ **do** ▷ Reveal indices of K smallest losses
    [idx], [loss] ← mpc.argmin(L)
    idx ← mpc.output([idx])
    $L.idx \leftarrow$ [max_loss]
    $Q \leftarrow Q \cup$ idx
end for
return Q

5) Step 5: Explanation with Selection of Synthetic CFs: After all synthetic candidates are ranked based on their distances to the selected event, the users can decide what will be revealed. In this paper the assumption is that the users want to see a certain amount (10) of synthetic CFs explaining the Factual. The more synthetic samples are revealed, the more general statistics are revealed about the features. Also with the synthetic data the parties can perform reconstruction attacks on the original data. However, there will always remain plausible deniability for the outcome. Therefore, it is advised to limit the

revealed information as much as possible. If desired users can also reveal only a summary of the synthetic CFs e.g. averages, maxima or minima.

B. Privacy

The approach is designed in such a way that both Alice and Bob can not learn any new sensitive information during the process as only abstract representations or encrypted values are shared. Even if Alice and/or Bob are honest but curious parties they cannot perform relevant reconstruction attacks during the process. Only when parties reveal synthetic CFs additional information is learned by both parties. Therefore, an Attribute Inference Attack is performed ([30]) in section V-D to analyse privacy leakage from revealed synthetic CFs. This attack simulates whether it is possible to gain knowledge on a feature owned by the other party for a specific sample.

C. Data

For the experiments the publicly available California Housing data set is used. This data set consists of 8 numerical features, one categorical feature and 20.640 samples. What the dataset is mostly used for is to predict the median house value. The data contains information of a block on the household (population, income, total residents), characteristics of the houses (total rooms, total bedrooms, housing median age), location (longitude, latitude, ocean proximity) and the house price (median house value). The dataset is scaled between [0,1] for continuous features and categorical features are one-hot-encoded in binary features. This results in an equal influence of all features to the optimization function in equation 1.

D. Evaluation Metrics

Our method is validated after each step following commonly used evaluation metrics for Synthetic Data Generation (SDG) and CFs. Multiple experiments were run to test stability of the models and evaluate computational time.

1) Evaluation Synthetic Data: The quality of the generated synthetic data is tested with utility metrics, which are selected based on increasing dimensionality. The original and synthetic data are compared on the following three aspects:

1) The comparison of the distributions of one feature measured with the Hellinger Distance (HD) with the focus on the distributed generated feature of Bob. For every $j \in \{1, .., d\}$, the difference between original feature X_j and synthetic feature X'_j is defined as:
$$HD = \frac{1}{\sqrt{2}} \cdot \| \sqrt{\hat{p}_{X_j}} - \sqrt{\hat{p}_{X'_j}} \|_2 \tag{3}$$
with probability distributions $P = (\hat{p}1, ..., \hat{p}k)$ divided in 30 bins. In case of categorical features the bins are the categories. Results are reported in $1 - HD$, which means 1 is the maximum and 0 the minimum similarity.
2) The correlation between two features, measured with a Pearson correlation matrix measured on all numerical features. The Pearson coefficient ρ is calculated by:
$$\rho_{X,X'_j} = \frac{covXX'_j}{\sigma_X \sigma_{X'_j}} \tag{4}$$
with cov the covariance, σ the standard deviation of both X and synthetic X_j. 1 is the maximum covariance and 0 means no covariance.
3) A Support Vector Machine regression task evaluates the multi-feature similarity. The target feature is 'Median House Value' and reported are the R-squared (R^2), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

The experiments are executed using three different features as distributed feature for Bob. Namely *population*, *median income* and *housing median age*. These features are selected based on varying distribution and correlation with other features. We hypothesized that a higher correlation to other features results in more realistic synthetic feature of Bob. *population* and *median income* are highly correlated with other features. *Housing median age* has lower correlations to other features, as can be seen in Figure 3.

2) Evaluation Counterfactuals: Our method is evaluated by comparing two sets of CFs. The first is generated in a one-party setting using the original data and the same distance optimization as presented in Protocol (1). The second is generated according to our novel secure protocol (1). Validity of the CFs is addressed by comparing the computed outcome with the desired situation. E.g. we want the outcome to increase by 10-15$\%$ and select only the CFs adhering to that. Feasibility and plausibility can be largely tackled by the data owners as they can put constraints on the generated CFs as mentioned in step 1. E.g. a house with a $20m^2$ cannot have 10 rooms. The CFs are evaluated on their distance to the Factual. The $L1$ distance is most distinguishable distance of the three distance metrics and matches the Factual as closely as possible by element-wise comparison representing the average change. $L1$ is defined in equation 1. For evaluation, the distances of all features between the Factual and CF are calculated and specifically for distributed feature's the $L1$ to evaluate the effect of the distributed synthesis step. To be able to make a robust comparison, the metrics are visualised by iterating over a 100 different Factual samples and taking the top 10 CF candidates. The results are shown in a violin plot as was done by Carla et al [21]. Moreover, all experiments are performed on normalized data sets.

3) Evaluation Privacy: After the synthetic CFs are revealed an honest but curious (HBC) party A can try to gain information on the distributed feature of party B. An HBC party means the party follows the prescribed protocol, but tries to learn additional information from the other party. Party A can train a ML model on the revealed synthetic CFs to predict the distributed feature from party B. After training, party A uses the model on her own original data to predict the distributed feature. If party A is able to correctly predict the feature's value, there is privacy leak. Otherwise privacy is indeed preserved. For this purpose Support Vector Regression (SVR) models are trained on synthetic data and tested on all original data. The robustness of the various models is tested with Cross Validation Scores (CVS). There are four scenarios in which 10, 100, 1000 or all synthetic data samples

Feature	Distributed	Central Synthetic
Population	0.06	0.04
Median Income	0.17	0.08
Housing Median Age	0.38	0.13

TABLE I: The Hellinger Distance for Central Synthetic and the three distributed Synthetic data sets on their respective features compared to the Original. HD has a maximum value of 1, which means no symmetry and a minimum of 0 which means exactly equal probability distributions.

are revealed. Both R^2 scores and MAEs are reported. As additional analysis the amount of samples which have been predicted exactly correct are calculated as well. Keep in mind that in general the more synthetic samples are revealed the more general statistics can be concluded with certainty e.g. means, minimum and maximum values, and distributions. Therefore it is always advised to reveal as little information as possible.

V. Results

A. Vertically Generated Synthetic Data

Utilizing the SDV open-source library, we implemented the CTGAN to generate (baseline) synthetic data for Alice. The model trained for 2500 epochs on all data generating a synthetic data set of equal size (20608). When compared to the Original data, this baseline already shows a decrease in similarity. Although our primary focus is not on the evaluation of open-source methods, we present these results to ensure a fair assessment of our approach in generating distributed synthetic features.

For ease of reading, we name the baseline synthetic data set Central Synthetic and the three synthetic data sets Population, Median Income and Median Housing Age with distributed features *population*, *median income* and *median housing age* respectively.

The prediction with the Split-NN to generate the feature *population* has an R^2 of 0.84 at 20 epochs. Followed by *median income* with an R^2 of 0.74 at 20 epochs. However, Housing Median Age has the lowest R^2 value of 0.3 at 20 epochs, indicates the predictions for 'Housing Median Age have the highest error. To begin our analysis, we illustrate the Hellinger Distances (HD) in Table I between the Original data and, the Central and Distributed Synthetic data sets. Distributions of these features can be found in Appendix VIII-A. Stability experiments calculating R^2 for test and train sets running up to 20 epochs, show that for a balance in under-fitting and over-fitting 10 epochs is an overall good match.

For Population the similarity to Original is highest, as the HD is lowest, followed by Median Income. Housing Median Age has a high HD with a poor similarity to the Original with a high HD. This was expected by the low R^2.

Secondly, Figure 3 visualises the correlations of the distributed features for the original and synthetic data sets. Central Synthetic overall has a slight decrease in correlation compared to Original. Population has a high similarity in correlations to Original and a (small) increased correlation compared to Central Synthetic, its baseline. Median Income has a small increased correlation with respect to Central Synthetic and is therefore closer to Original. However, Housing Median Age shows a clear deviation with higher artificial correlations generated during the synthesis step as these features are evidently used to predict and generate the distributed feature.

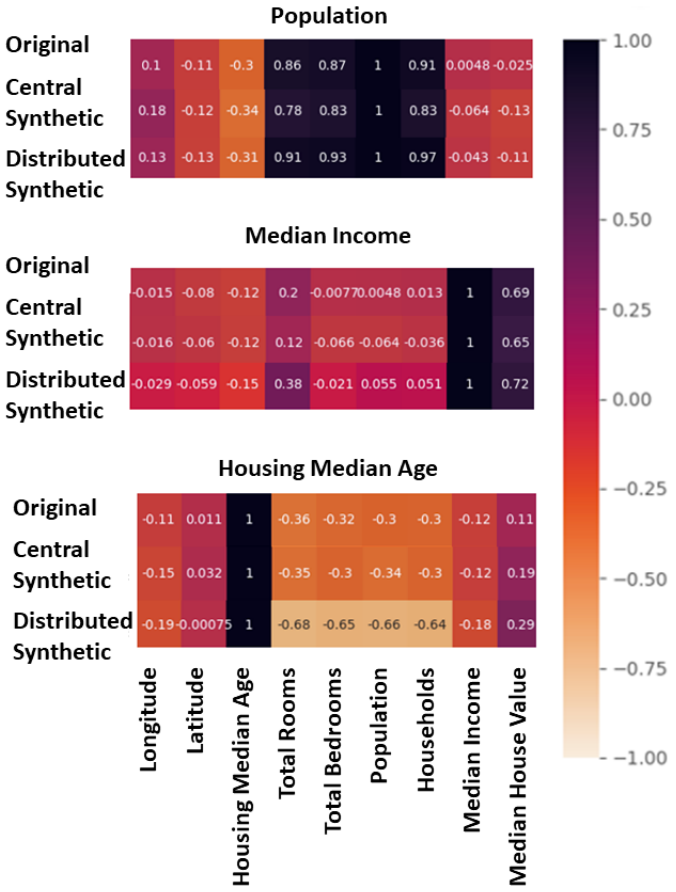


Fig. 3: Correlation plots for the features *population*, *median income* and *housing median age* for Original and synthetic data sets. The maximum positive correlation is 1 and the maximum negative correlation -1.

Thirdly, five SVM models are trained to predict the 'Median House Value' using Original data as the test set for all models. Of which the CVS is calculated during training and tested on the Original data by R^2, MAE and RMSE. As expected Original scores highest, followed by Central Synthetic, already losing prediction value. Population and Median Income data sets score reasonable R^2, MAE and RMSE in Table II as they maintain prediction value and multi-feature correlations with low errors and double the loss increase by Central Synthetic. For Housing Median Age the negative R^2 and the high MAE and RMSE indicate there is no prediction value for the target 'House Median Value' of the Original data at all. The high CVS for all distributed Synthetic data sets indicate that the outliers in Original are less generated during synthesis which results in increased prediction accuracy.

Data	Train	Test		
	CVS	R^2	MAE	RMSE
Original	0.72 ±0.01	0.719	0.089	0.125
Central synthetic	0.64 ±0.01	0.655	0.097	0.139
Population	0.77±0.00	0.504	0.116	0.166
Median Income	0.91 ±0.01	0.529	0.117	0.163
Housing Median Age	0.94±0.00	-6.184	0.526	0.623

TABLE II: Five Support Vector Machine models are trained on Original, synthetic data sets predicting the 'Median House Value' of the Original data. The Cross Validation Scores for the trained models have 5 cross sections showing standard deviations. The R^2, MAE and RMSE error are tested on Original data.

Data	epoch	R^2	MAE	MSE
Original	10	0.676	0.093	0.0184
	20	0.683	0.091	0.0180
Central Synthetic	10	0.648	0.095	0.0195
	20	0.675	0.092	0.0181
Population	10	0.629	0.103	0.0197
	20	0.628	0.103	0.0197
Housing Median Age	10	0.582	0.107	0.0221
	20	0.548	0.112	0.0239

TABLE III: Target re-computation on trained Split-NN model for Original, Central Synthetic, Population and Housing Median Age for 10 and 20 epochs in R^2, MAE, and MSE.

Moreover, while generating the distributed feature the target value of Central Synthetic is used instating a correlation with it. Furthermore, specifically for Housing Median Age there are incorrect artificial correlations generated during synthesis.

Overall our approach for a vertical synthesis on the Median Income and Population show a high similarity with the Original data with an acceptable loss and therefore are deemed successful. However, the results on Housing Median Age are unsuccessful. Which shows that our approach relies on the predictability of the distributed feature, which can be tested up-front by measuring the R^2.

For the next steps only the results of Population and Housing Median Age are shown. Following an 'successful' and 'unsuccessful' data synthesis.

B. Compute Target Values

The target values of the Original and Synthetic data sets are re-computed with the same trained Split-NN on the Original data. This is required as the CFE need to explain the predictions made by that model.

The target predictions in Original score only 10-15% higher than on Population in Table III. Housing Median Age scores lower, as expected, however due to the multi-feature correlations to the target the prediction is still relatively close to the Original prediction.

The stability experiments of R^2 for train and test data sets show the results are robust. For further experiments the results of the 10 epoch run is used as there is a reasonable trade-off with respect to overfitting and under fitting.

Data	Median
Original	0.355
Central synthetic	0.336
Population	0.334
Housing Median Age	0.349

TABLE IV: Medians of L1 distances CF for Original, Central Synthetic, Population and Housing Median Age.

C. Secure Counterfactuals

In the final step the original CFs in a one party setting are compared to the synthetic CFs in a two party setting. The ML network is a 3-layer neural network with a regression task on the target 'Median House Value'. The CFs are selected to be 10-15 % higher than the Factual. A random selection of 100 Factuals is picked for which the top 10 candidates L1 distribution is shown in Figure 4. The $L0$ and $L2$ distances for all features can be found in the Appendix VIII-B.

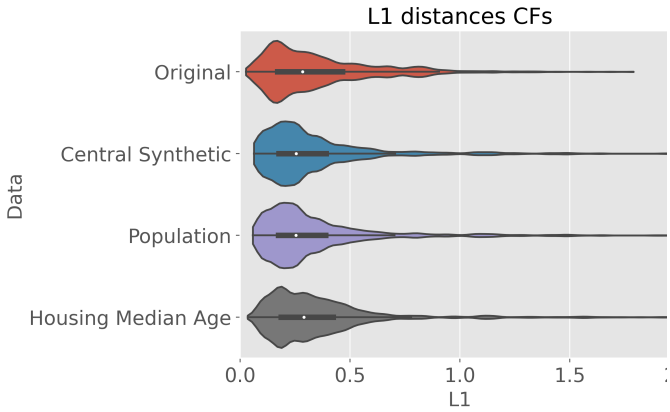


Fig. 4: L1 distribution calculated with the top 10 CFs for Original, Central Synthetic, Population and Housing Median Age data sets for 100 random Factuals. The white dots indicate the medians, and the black boxes indicate the interquartile ranges with a maximum distance of 10.

The L1 distances of the Original and Synthetic data sets have a comparable low median seen in Table IV. The Original data has more CFs spread between 0,5 and 1 as there are less outliers present in the Synthetic data sets. Comparing Population and Housing Median Age with their baseline Central Synthetic data shows that Population is comparable to Central Synthetic, however for Housing Median Age the median increased and seemingly has a distributions closer to Original. When zooming in on the CFs distribution of the distributed generated synthetic features Population (5) and Housing Median Age (6), it is explained why Housing Median Age has a higher mean compared to the Central Synthetic baseline and is more comparable to Original.

For the feature *population* the L1 median distance of the Central Synthetic and Population is lower by 5-15% than the Original data in Table V. As the Synthetic data generates less outliers and averages the events. Which is evident from the distribution plots in Figure 5 as well.

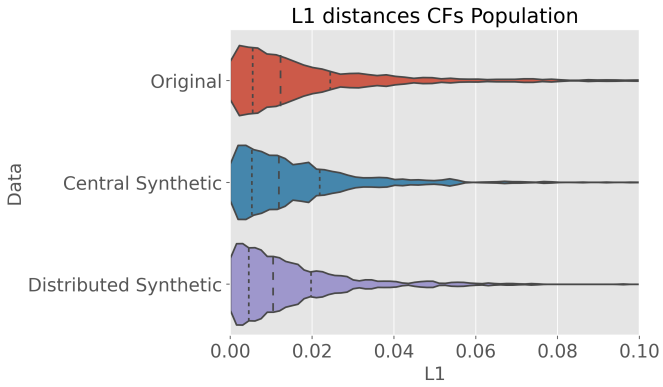


Fig. 5: L1 distribution of the feature *population* in Original, Central Synthetic and Population data sets from the top 10 CFs for 100 Factuals. The three black dotted lines indicate the quarterlies

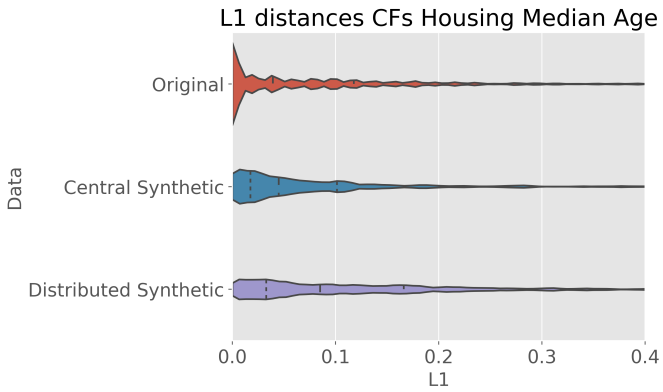


Fig. 6: L1 distribution of the feature *housing median age* in Original, Central Synthetic and Housing Median Age data sets from the top 10 CFs for 100 Factuals. The three black dotted lines indicate the quarterlies.

For Housing Median Age the median increases by 210% in Table V. The distribution plot in Figure (6) shows a wide spread for both the Central Synthetic and even more in Housing Median Age. Instead the Original data shows most CFs are equal for the feature *housing median age* to the Factual. Therefore the synthetic CFs are hardly comparable to the Original data CFs for this specific feature. This incorrectly artificially created spread results in a higher $L1$ mean and distribution in Figure 4.

D. Attribute Inference Attack

The trained SVR models on the synthetic samples show a low Cross Validation Score for *population* in all scenarios in Table VI. Therefore, the trained models do not have prediction power and it is not possible to infer attribute information from the original data. This is verified by the low R^2 values on the test data. The models for Housing Median Age scenarios with more then 10 samples are trained properly, with R^2 values >0.7. However, when tested on the original data the results are still poor. This shows that it is not possible to infer information on the feature *housing median age* either.

Data	Population	Housing Median Age
Original	0.0122	0.0392
Central synthetic	0.0117	0.0453
Distributed synthetic	0.0104	0.0851

TABLE V: Medians of L1 distance CF distributions for features *population* and *housing median age* of the Original, Central Synthetic and Population or Housing Median Age data sets respectively.

Data	Samples	CVS±Std	R^2	MAE	Nr
Population	10	-6.56+6.79	-0.07	0.21	0
	100	-8.76±9.15	0.18	0.18	0
	1000	-4.55±1.84	0.24	0.18	0
	All data	-5.06±0.24	0.25	0.17	0
Housing Median Age	10	-2.58±3.08	-0.11	0.02	0
	100	0.72±0.09	-1.46	0.04	0
	1000	0.83±0.02	-4.05	0.07	0
	All data	0.86±0.01	-3.56	0.07	0

TABLE VI: Attribute inference attack on distributed features *population* and *housing median age*. The Cross Validation Scores for 5 trained models have 5 showing standard deviations. Results on the original data are provided in R^2 and MAE. Nr is the number of exactly correct predicted samples.

VI. Conclusion

Our novel SCFE approach successfully shows that it is possible to securely create realistic synthetic explanations in a two-party setting.

The synthetic data generation method on vertical distributed data enables parties to jointly generate synthetic data when there is a high prediction accuracy. Our method can generate a synthetic feature for both numerical and categorical features.

The SCFE Protocol enables parties to collaboratively rank and optimize CFs in a secure way. Our optimization Protocol is flexible; the α, β and γ in Protocol 1 can be adjusted to the requirements of the involved parties.

For the final step, the data-owners can decide what they need and want to reveal. The top k synthetic CFs can be shared to explain the Factual. The data-owners can decide to reveal summaries of the results e.g. the average increase or decrease of features. It is advised to reveal as limited as possible, to minimise security risk.

To conclude, we have introduced a completely secure approach to calculate, optimize and reveal CFE in a two party setting.

VII. Limitations and Future Work

As the vertical synthetic data generation did not provide realistic results for distributed features with a low prediction accuracy, further research should focus on improving vertically partitioned synthetic data generation for all scenarios. The authors are developing new statistical and deep learning methods to improve the quality of the vertical distributed synthetic data.

Differential Privacy [31] can be added to the synthetic data generation steps to provide even more protection against reconstruction attacks on the model.

This method can be extended to a setting with more features for Bob by repeating steps 2 and 3 in section IV-A. And to multiple data-owners by adding a second data-owner next to Alice who consequently holds the prior generated distributed feature. It is expected the accuracy of synthetic data will decrease in both scenarios.

Lastly, reduction of calculation time is needed to use this approach in practical applications.

REFERENCES

[1] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, and B. Thorne, "Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption," 2017. [Online]. Available: https://arxiv.org/abs/1711.10677

[2] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konečný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, M. Raykova, H. Qi, D. Ramage, R. Raskar, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao, "Advances and open problems in federated learning," 2019. [Online]. Available: https://arxiv.org/abs/1912.04977

[3] F. K. Došilović, M. Brčić, and N. Hlupić, "Explainable artificial intelligence: A survey," in *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*. IEEE, 2018, pp. 0210–0215.

[4] J. M. Schoenborn and K.-D. Althoff, "Recent trends in xai: A broad overview on current approaches, methodologies and interactions." in *ICCBR Workshops*, 2019, pp. 51–60.

[5] S. Goethals, K. Sörensen, and D. Martens, "The privacy issue of counterfactual explanations: Explanation linkage attacks," *ACM Transactions on Intelligent Systems and Technology*, vol. 14, 07 2023.

[6] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," *IEEE Xplore*, 2021.

[7] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," *arXiv preprint arXiv:1812.00564*, 2018.

[8] I. Ceballos, V. Sharma, E. Mugica, A. Singh, A. Roman, P. Vepakomma, and R. Raskar, "Splitnn-driven vertical partitioning," *CoRR*, vol. abs/2008.04137, 2020. [Online]. Available: https://arxiv.org/abs/2008.04137

[9] G. Wei, W. Mu, Y. Song, and J. Dou, "An improved and random synthetic minority oversampling technique for imbalanced data," *Knowledge-Based Systems*, vol. 248, p. 108839, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705122004002

[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, jun 2002. [Online]. Available: https://doi.org/10.16132Fjair.953

[11] Figueira, Alvaro, and B. Vaz, "Survey on synthetic data generation, evaluation methods and gans," *Mathematics 2022*, vol. 10, no. 15, 2022.

[12] H. Zhong, Y. Sang, Y. Zhang, and Z. Xi, "Secure multi-party computation on blockchain: An overview," in *Parallel Architectures, Algorithms and Programming: 10th International Symposium, PAAP 2019, Guangzhou, China, December 12–14, 2019, Revised Selected Papers 10*. Springer, 2020, pp. 452–460.

[13] E. Dawson and D. Donovan, "The breadth of shamir's secret-sharing scheme," *Computers & Security*, vol. 13, no. 1, pp. 69–78, 1994.

[14] C. Hazay and M. Venkitasubramaniam, "Scalable multi-party private set-intersection," in *IACR international workshop on public key cryptography*. Springer, 2017, pp. 175–203.

[15] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[16] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?": Explaining the predictions of any classifier," *CoRR*, vol. abs/1602.04938, 2016. [Online]. Available: http://arxiv.org/abs/1602.04938

[17] S. Verma, J. P. Dickerson, and K. Hines, "Counterfactual explanations for machine learning: A review," *CoRR*, vol. abs/2010.10596, 2020. [Online]. Available: https://arxiv.org/abs/2010.10596

[18] Y.-L. Chou, C. Moreira, P. Bruza, C. Ouyang, and J. Jorge, "Counterfactuals and causability in explainable artificial intelligence: Theory, algorithms, and applications," *Information Fusion*, vol. 81, pp. 59–83, 2022.

[19] S. Wachter, B. D. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the GDPR," *CoRR*, vol. abs/1711.00399, 2017. [Online]. Available: http://arxiv.org/abs/1711.00399

[20] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. ACM, jan 2020.

[21] M. Pawelczyk, S. Bielawski, J. v. d. Heuvel, T. Richter, and G. Kasneci, "Carla: a python library to benchmark algorithmic recourse and counterfactual explanation algorithms," *arXiv preprint arXiv:2108.00783*, 2021.

[22] A.-H. Karimi, B. Schölkopf, and I. Valera, "Algorithmic recourse: from counterfactual explanations to interventions," 2020. [Online]. Available: https://arxiv.org/abs/2002.06278

[23] Z. Zhao, A. Kunar, R. Birke, and L. Y. Chen, "Fed-tgan: Federated learning framework for synthesizing tabular data," *cs.LG 2021*, 2021.

[24] R. Tajeddine, J. Jälkö, S. Kaski, and A. Honkela, "Privacy-preserving data sharing on vertically partitioned data," *cs.LG*, 2022.

[25] T. Veugen, B. Kamphorst, and M. Marcus, "Privacy-preserving contrastive explanations with local foil trees," *arXiv preprint CSCML*, 2022. [Online]. Available: https://eprint.iacr.org/2022/360

[26] W. G., "Interpret federated learning with shapley values," *arXiv preprint arXiv:1905.04519*, 2019.

[27] P. Chen, X. Du, Z. Lu, J. Wu, and P. C. Hung, "An explainable vertical federated learning for data-oriented artificial intelligence systems," *Journal of Systems Architecture, 126:102474.*, 2022.

[28] J. Rivers, A. Nelson, and L. Williams, "Synthetic data generation with sdv."

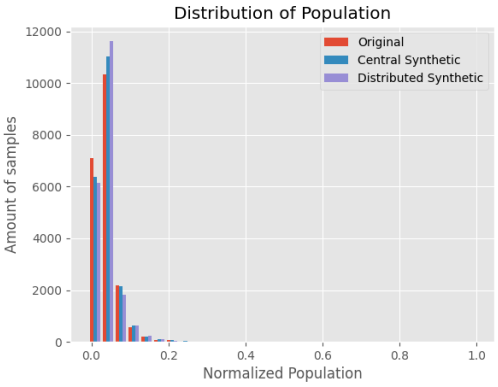
[29] B. Schoenmakers, "Mpyc—secure multiparty computation in python," in *the Theory and Practice of Multiparty Computation (TPMPC) 2018 workshop*, 2018.

[30] B. Zi Hao Zhao, A. Agrawal, C. Coburn, H. Jameel Asghar, R. Bhaskar, M. A. Kaafar, D. Webb, , and P. Dickinson, "On the (in)feasibility of attribute inference attacks on machine learning models," *IEEE*, pp. 232–251, 2021.

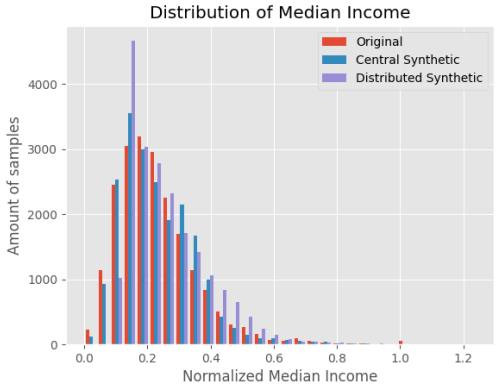
[31] Bellovin, S. M, Dutta, P. K, and N. Reitinger, "Privacy and synthetic datasets," *Stan. Tech. L. Rev.*, vol. 22, p. 1, 2019.

VIII. Appendix

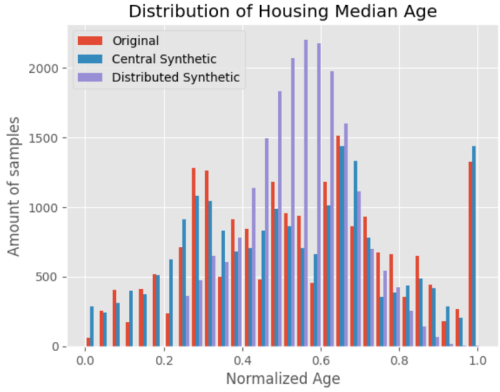
A. Distributions of Distributed Features



(a) Feature *population*



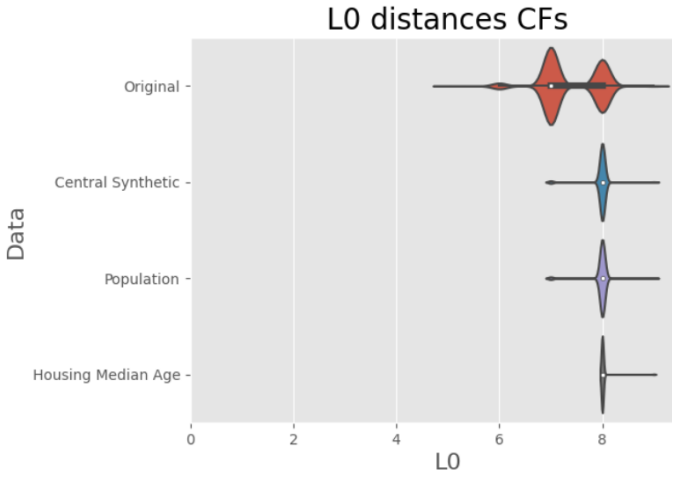
(b) Feature *median income*



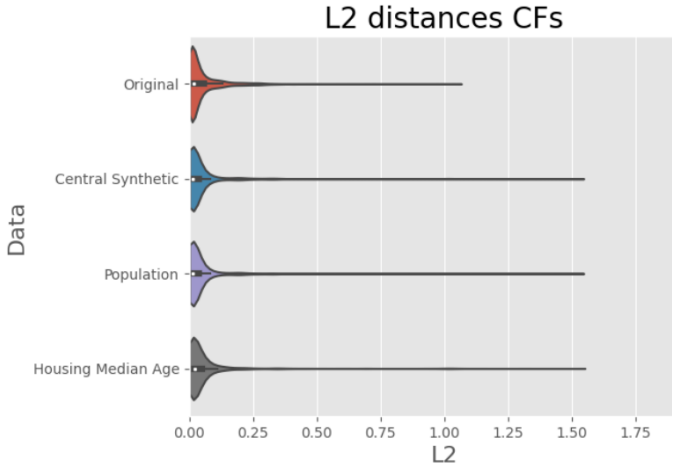
(c) Feature *housing median age*

Fig. 7: Distributions of Original, Central Synthetic and Distributed Synthetic data for the features *population*, *median income* and *housing median age*. The distribution is divided into 30 bins. Population is most similar to Original data followed by Median Income which overall has higher values. Housing Median Age does not match the Original data.

B. L0 and L2 Distance Distributions



(a) L0 distance



(b) L2 distance

Fig. 8: $L0$ and $L2$ distance distributions for Original, Central Synthetic, Population and Housing Median Age data sets from the top 10 Counterfactuals for 100 random Factuals.