Use Case of Counterfactual Examples: Data Augmentation

MGM Mehedi Hasan and Douglas A Talbert, Dept. of Computer Science, Tennessee Tech University

Research Problem

- Counterfactual examples (CFEs) are generally created to interpret the decision of a model. In this case, if a model makes a certain decision for an instance, the counterfactual examples of that instance reverse the decision of the model.
- The counterfactual examples can be created by craftily changing particular feature values of the instance.
- In this work, we explore other potential application areas of utilizing counterfactual examples other than model explanation.
- We are particularly interested in exploring whether counterfactual examples can be a good candidate for data augmentation. At the same time, we look for ways of validating the generated counterfactual examples.

Explanation in Machine Learning

- Explanations are critical for machine learning, which are being used to inform decisions in societally critical domains such as finance, healthcare, education, and criminal justice.
- However, most explanation methods depend on an approximation of the ML model to create an interpretable explanation.
- For example, consider a person who applied for a loan and was rejected by the loan distribution algorithm of a financial company.
- Typically, the company may provide an explanation on why the loan was rejected, for example, due to “poor credit history”.
- However, such an explanation does not help the person decide what they do should next to improve their chances of being approved in the future.
- Critically, the most important feature may not be enough to flip the decision of the algorithm, and in practice, may not even be changeable such as gender and race.

Counterfactual Explanation

- A counterfactual explanation of a prediction describes the smallest change to the feature values that changes the prediction to a predefined output.
- In interpretable machine learning, counterfactual explanations can be used to explain predictions of individual instances.
- Counterfactual examples are great way to explain the outcome of a machine learning model.

Motivation and Contribution

- In interpretable machine learning, counterfactual explanations can be used to explain predictions of individual instances.
- The counterfactual explanation method is model-agnostic, since it only works with the model inputs and output.
- The interpretation can be expressed as a summary of the differences in feature values.
- Counterfactuals are human-friendly explanations, because they are contrastive to the current instance and because they are selective, meaning they usually focus on a small number of feature changes.
- However, we found that none of the existing approaches talk about how counterfactual example can be an efficient way for data augmentation.
- In this work, we propose that counterfactual example can be a viable option foe data augmentation and we show that under different scenarios.

Existing CFEs generation Techniques

- Wachter et. al [1] proposed an approach by minimizing the following loss function:

$$L(x, x', y', \lambda) = \lambda \cdot (\hat{f}(x') - y')^2 + d(x, x')$$

- The first term is the quadratic distance between the model prediction for the counterfactual x’ and the desired outcome y’, which the user must define in advance. The second term is the distance d between the instance x to be explained and the counterfactual x’.
- The parameter λ balances the distance in prediction (first term) against the distance in feature values (second term).
- The loss is solved for a given λ and returns a counterfactual x’.
- The authors [2] suggest instead of selecting a value for λ to select a tolerance ε for how far away the prediction of the counterfactual instance is allowed to be from y’. This constraint can be written as:

$$|\hat{f}(x') - y'| \leq \epsilon$$

- To minimize this loss function, any suitable optimization algorithm can be used. In our work we follow the approach adopted by Mothilal et. al [2] to generate Counterfactual Examples.

Applied CFE generation Technique

- We apply the concept and technique introduced by Mothilal et. al [2] to generate the counterfactual examples (CFEs).
- In this case, we generate the counterfactual examples using a shallow artificial neural network (ANN) and then use those counterfactual examples in other models.

Model Choice

- We use different models to experiment with the generated counterfactual examples to make sure that those models will not have any bias to the labels of the CFEs, which are in fact generated by other model.
- At the same time, we also wanted to make sure that CFEs generated by one model is transferable to another model.

Dataset

- We consider the Adult-Income, which contains demographic, educational, and other information based on 1994 Census database and is available on the UCI machine learning repository [3].
- We obtain 8 features, namely, hours per week, education level, occupation, work class, race, age, marital status, and sex by applying the preprocessing based on a previous analysis [7].

Experiment

- We train an ANN model using the adult dataset. We randomly select 400 instances and generate maximum of 4 CFEs for each of the instances and generate the CFEs.
- In total, we got 1000 CFEs. We use this CFEs in with different fraction of the original adult dataset.

Case Study I

- We first consider the whole original adult dataset to train and test three kind of models, which are decision tree, Random Forest and Bagging.
- From the dataset, we make a 80: 20 train/test spilt. The test accuracy of different models are shown in Table I.

Table I
TEST ACCURACY OF DIFFERENT MODEL WHEN USED THE WHOLE ORIGINAL ADULT DATASET

Decision Tree	Random Forest	Bagging
78.1%	79.99%	80.99%

Case Study II

- In this case, we consider 20% of the original adult dataset to train and test three the same three kinds. Again we make a 80: 20 train/test spilt.
- The test accuracy of different models are shown in Table II

Table II
TEST ACCURACY OF DIFFERENT MODEL WHEN USED 20% OF THE ORIGINAL ADULT DATASET

Decision Tree	Random Forest	Bagging
68.18%	72.72%	68.18%

Case Study III

- We now consider 20% of the original adult dataset and the generated CFEs as the dataset to train and test those three models, which are decision tree, Random Forest and Bagging.
- From the dataset (20% of the original adult dataset and the generated CFEs), we make a 80: 20 train/test spilt. The test accuracy of different models are shown in Table III.

Table III
TEST ACCURACY OF DIFFERENT MODEL WHEN USED 20% OF THE ORIGINAL ADULT DATASET AND THE GENERATED CFES

Decision Tree	Random Forest	Bagging
68.78%	77.27%	72.72%

Discussion and Future Direction

- We use different case studies to realize the significance of CFEs as a way for data augmentation.
- If we compare case studies I and II with III, we observe that CFEs indeed can be a good alternative for data augmentation.
- In the future, we will look for ways of validating the generated counterfactual examples.
- We will explore efficiency of our proposed technique with the existing data augmentation technique.
- We will look for explanations on why different models are showing different accuracy and whether accuracy can be a good indicator to determine effective counterfactual examples.

References

[1] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. Harv. JL & Tech., 31:841, 2017.

[2] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, pages 607–617, 2020..

[3] R. Kohavi and B. Becker. UCI machine learning repository, 1996. https://archive.ics.uci.edu/ml/datasets/adult.