# Improving Counterfactual Explanations for Time Series Classifications Models in Healthcare Settings

**Tina Han**
American Express
Phoenix, AZ
tinahan789@gmail.edu

**Pedram Akbarian**
The University of Texas at Austin
Austin, TX
akbarian@utexas.edu

**Joydeep Ghosh**
The University of Texas at Austin
Austin, TX
jghosh@cognitivescale.com

**Jette Henderson**
CognitiveScale
Austin, TX
jhenderson@cognitivescale.com

## Abstract

Explanations of machine learning models' decisions can help build trust between a model and a user, as well as identify and isolate unexpected model behavior. Time series data, abundant in medical applications, and their associated classifiers pose a particularly difficult explainability problem due to the inherent feature dependency that results in complex modeling decisions and assumptions. Counterfactual explanations for a given time series tells the user how the input to the model needs to change in order to receive a different class prediction from the classifier. While a few methods for generating counterfactual explanations for time series have been proposed, the needs of simplicity and plausibilty have been overlooked. In this paper, we propose an easily understood method to generate realistic counterfactual explanations for any black box time series model. Our method, Shapelet-Guided Realistic Counterfactual Explanation Generation for Black-Box Time Series Classifiers (SGRCEG), grounds the search for counterfactual explanations in shapelets, which are discriminatory subsequences in time series. SGRCEG greedily constructs counterfactual explanations based on shapelets. Additionally, SGRCEG also employs a realism check, so the likelihood of producing a counterfactual that is not plausible is minimized. Using SGRCEG, model developers as well as medical practitioners can better understand the decisions of their models.

Time series classification has widespread applications in the medical field including anomaly detection and assisting in diagnosis. Especially in such high stakes situations, there has been growing awareness that model building and the ability to explain model decisions must co-exist. The dependent nature of time series features as well as the complexity of the models used to learn patterns from them, however, make it particularly difficult to build digestible explanations. For tabular data, counterfactual explanations offer an intuitive way of explaining classifier decisions. Counterfactual explanations tell the user how a given input to the model needs to change in order to receive a different class prediction from the classifier [13]. For the counterfactual explanation to be useful, it needs to be close to the original input and realistic (i.e., in distribution).

Counterfactual explanations for time series data and their respective classifiers are more complicated. While the definition of counterfactual explanations for time series is identical, the goal is to find a similar time series to the input time series but one that is different enough to be assigned a different label by the classifier, time series counterfactual explanations are inherently more complex due to the dependent structure of their features. While a few methods for generating counterfactual explanations

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

for time series have been proposed, our goal is to offer an improved counterfactual explanation generation method so the results are tailored to medical practitioners' needs.

To answer the specific challenges posed by time series and their associated models, we propose a two step method that first grounds the search for counterfactual explanations in key subsequences of a given set of time series, creating a common language with which to discuss these explanations, and then second, generates realistic counterfactual explanations for any black box model using these key subsequences. The method, Shapelet-Guided Realistic Counterfactual Explanation Generation for Black-Box Time Series Classifiers (SGRCEG, pronounced "sugar keg"), uses shapelet classification to identify subsequences that are globally representative of a model's classification decision. Figure 1 shows one time series from each class from the ECG200 data set [4] with shapelet sequences highlighted. Looking at Figure 1, a practitioner can focus on which windows, defined by shapelets, the model is using to decide to which class a time series belongs.

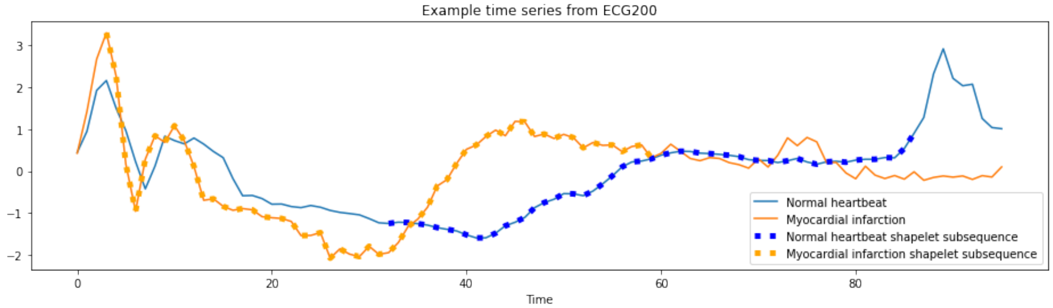

Figure 1: Example time series from each class in the ECG 200 with corresponding shapelet highlighted. [4]

Once the key subsequences are identified, given a time series of interest and a dictionary of the start of shapelet sequences and their lengths, SGRCEG sequentially replaces the window defined by shapelet sequence in that time series with the aligned sequence from a "close" time series with the desired predicted label until the label changes. SGRCEG also employs a realism check, so the likelihood of producing a counterfactual that is not plausible is minimized. Using SGRCEG, a practitioner could examine multiple time series, and by focusing on counterfactual explanations generated within these key subsequences within each time series, they can quickly gain a better understanding of model behavior. Furthermore, SGRCEG can provide actionable direction with respect to the robustness of the model. For example, if counterfactual explanations are generated with a short substitution, this may be an indication the model is too sensitive, and the model may require tuning or an expanded training set. Finally, SGRCEG is suited for any univariate time series classification model and does not assume any knowledge of the true labels or model type.

In the following sections, we first cover related methods for finding time series classification counterfactual explanations. We then introduce our method and show how it can be used to generate counterfactual explanations for healthcare time series. Finally, we outline areas for future research.

## 1 Related work

While there have been a few methods proposed to generate counterfactual explanations for time series [8, 9], we focus on the two most relevant frameworks to our work, CoMTE, a method proposed by [1], and an instance-based method put forth by [6], for the sake of brevity. CoMTE is a model agnostic method that works with multivariate time series. CoMTE greedily replaces entire univariate sequences of the the time series of interest with univariate sequences from a "distractor," a sample from the training set which is as similar as possible to the queried time series but receives a different predicted label. This method does not use information about influential subsequences. In the univariate case, CoMTE collapses to find the closest time series in the other class. While this does guarantee a plausible time series, it may not produce the closest counterfactual explanation, and it does not offer succinct insight into what the key differences are between the two time series.

A similar method proposed by [6], which we will call the Native Guide method, works on univariate time series. They find the “native guide” or “nearest unlike neighbor” (NUN) which is equivalent to the distractor from [1], then use it to replace parts of the queried time series until the label changes. While their experiments use a neural network and associated class activation maps, they note their approach is model agnostic and can be used in combination with any vector-based feature importance method (e.g., Shap values). Using local feature importance methods to identify key subsequences adds a layer of complexity to their framework that is not practical for a setting like healthcare where efficiency is key. The authors of [6] raise concerns about the interpretability of shapelets. However [5], [7], and [14] show shapelets help practitioners understand model behavior. Additionally, shapelets offer a global view of important subsequences, which we believe could help ground analysis for practitioners. They offer insight into what subsequences are actually informative and thus assist in ignoring random noise. Shapelets also have the added advantage of being easy for layman to understand, as their goal, finding “shapes” that are present in one class and not in another, is intuitive.

A reviewer alerted us to the parallel work of [2] published in August of 2022.One key difference between SGRCEG, which was developed primarily during summer of 2021 while authors Han and Akbarian were interns at Cognitive Scale, and SETS is SETS is for multivariate time series while SGRCEG is focused on univariate time series. Additionally, SGRCEG is more flexible in that it requires fewer predefined parameters, particularly in the generation of the shapelet dictionary. SETS uses a shapelet algorithm that requires predetermining the length of the shapelets, whereas we specifically chose an algorithm which does not. The shapelet algorithm choice may limit interpretability, since the length of important sequences is likely unknown. [2] also includes an additional step which eliminates shapelet candidates based on a distance threshold, requiring another predetermined parameter. To find a nearest unlike neighbor, SETS uses KNN, so the user must also determine an appropriate $k$, and lacks the flexibility that SGRCEG offers with the ability to use any similarity measure for finding NUNs. Finally, SETS uses three metrics to determine plausibility: isolation forest (IF), one class support vector machine (OC-SVM), and local outlier factor (LOF) similar to [6] in a post-hoc fashion rather than being built into the algorithm. SGRCEG uses LOF in the algorithm itself, and could be changed to another outlier detection method.

## 2 Method

Let $T_i = \{t_1, t_2, ..., t_m\}, i = 1, ..., n$, be a univariate time series of length $m$ with associated label $y_i = l$, $l = 1, ..., k$, where $k$ is the number of classes.[1] Denote $D = \{T_1, T_2, ..., T_n\}$ as a set of time series, and $D_{tr} = \{T_k\}_{k=1}^{n_{tr}}$ and and $D_{te} = \{T_k\}_{k=1}^{n_{te}}$ are the training and test sets, respectively, where $D_{tr} \cap D_{te} = \emptyset$ and $D_{tr} \cup D_{te} = D$.

Suppose we have a queried time series, $T_q \in D$, for which we want to find a counterfactual explanation, and a classifier $C$ such that $C : T_q \mapsto \hat{y}_q$ where $\hat{y}_q$ is the predicted class label given to $T_q$ by $C$. Then the counterfactual explanation, $T_{cf}$, is a time series such that $C : T_{cf} \mapsto \hat{y}'$, where $\hat{y}'$ is the desired class label. Ideally, $T_{cf}$ is similar to $T_q$ so that the change is as minimal as possible while still being realistic.

**Algorithm 1** Counterfactual Explanations Generation Method

```
Require: $T_q \in D, D_{Tr} = \{T_1, T_2, ..., T_n\}$, classifier $C$, shapelet set $S$
  $T_{cf}$ =None
  $T_{NUN}$ =min $d(X_q, X_t), X_t \in D_{Tr}$ where $C(T_{NUN}) \neq C(T_q)$
  for $(s_{start}, s_{length})$ in $S$ do
      for $i = 1 : s_{length}$ do
          $T_c = (T_q[: s_{start}], T_{NUN}[s_{start} : s_{start} + i], T_q[s_{start} + i :])$   ▷ Candidate explanation
          if $C(T_q) \neq C(T_c)$ then
              $T_{cf} = T_c$
              Return $T_{cf}$
  if $T_{cf}$ is None then
      Return $T_{NUN}$
```

[1] For this paper, we assume all time series are of the same length, and leave it to future work to adapt it to variable length time series.

The first step of the SGRCEG framework is to identify global subsequence locations that are informative of the classifier's predictions and present these to the user. Given a classifier $C$ and time series set $D$, we extract these subsequence locations using shapelet analysis. Once the classifier is built, we save the classifier's labels for each time series. Then using those labels build a library of shapelets. Shapelets are subsequences that are present in one class, but not other classes. We use `sktime` [10] to extract a library of shapelets, which does not have the user predetermine the shapelet length. We sort the extracted shapelets by their respective information gain, and construct $S$, a sorted list of tuples $(s_{start}, s_{length})$ of the starting index of the shapelet sequence and the length of the sequence.

SGRCEG then presents the shapelets to the user so they are familiar with where the counterfactual explanations will be generated, but due to space constraints, we cannot show all of them. Figure 1 demonstrates a condensed version of our idea. It shows one time series per each class from the ECG200 data set [4] with a shapelet sequence highlighted. Practitioners can examine these subsequences to gain an understanding of which windows are important to the classifier.

Having identified where in the time series sequences to look for counterfactual explanations, the user can then query SGRCEG for counterfactual explanations. The counterfactual explanation generation process is detailed in Algorithm 1. For a given queried time series $T_q$, SGRCEG first finds the nearest unlike neighbor (NUN), $T_{NUN}$ where $T_{NUN}$ is closest under some distance $d(\cdot,\cdot)$ where $C(T_{NUN}) \neq C(T_q)$. We tested dynamic time warping (DTW) and Euclidean distance for our similarity measure, but any appropriate measure or algorithm can be used. DTW is resistant to the "stretching"/"warping" of time series that other measures (e.g., Euclidean) can be sensitive to [12].

Since there is not one single measure that is the best for all time series data, SGRCEG includes this flexibility to account for one's prior knowledge of the data.

Then, beginning at position $s_{start}$ for the shapelet with the highest information gain, SGRCEG substitutes values from $T_{NUN}$ into $T_q$ for a maximum subsequence of length $s_{length}$. At each substitution we checked if the classifier assigns the perturbed version of $T_q$, called $T_c$, the desired label. If $T_c$'s prediction does not change after the end of the shapelet sequence is reached, SGRCEG repeats the process with the shapelet with the next highest information gain, and so on until $C(T_c) = y'$ or have exhausted all shapelets. If all shapelets are exhausted and no plausible counterfactual explanation is found, we use the NUN as the counterfactual explanation.

Finally, SGRCEG checks if the counterfactual explanation candidate is in distribution with a local outlier factor model (LOF), which is an algorithm that uses density-based outlier detection to give a score to an observation indicating if it is an outlier [3]. We chose LOF as our plausibility check, but this could be replaced with another outlier detection method. SGRCEG rejects a counterfactual candidate if it is not in distribution. Thus, it makes sense to use a less sensitive method for outlier detection, which ideally strikes a balance between plausibility and sparseness.

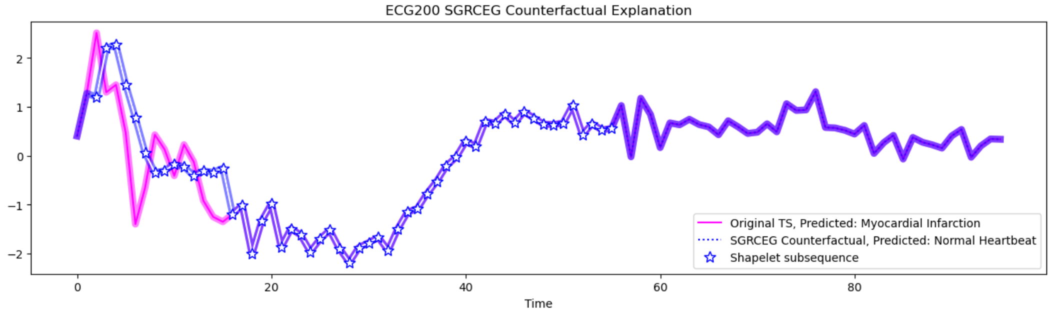

Figure 2: Queried time series (pink) with SGRCEG generated counterfactual explanation (blue) and shapelet sequence highlighted (stars) using the ECG200 data set[4].

Table 1: Results for ECG200 data set. The % Valid column is the percentage of times a method generated a counterfactual explanation.

| Method | % Valid | LOF mean (sd) | Substitution length mean (sd) |
|---|---|---|---|
| Native Guide (Shap) | 83% | 0.20 ( 0.10 ) | 29.69 ( 35.96 ) |
| SGRCEG | 90% | 0.15 ( 0.15 ) | 23.39 ( 28.57 ) |
| Separated SGRCEG | 86% | 0.19 ( 0.11 ) | 27.02 ( 31.53 ) |

## 3 Electrocardiogram Case Study

To explore the effectiveness of SGRCEG, we use the ECG200 dataset from [4], which contains 200 electrocardiogram time series of length 96, and has two classes: a normal heartbeat and a myocardial infarction event. We use the predefined training and test set splits that each contain 100 time series. We train a k-nearest neighbors classifier from [11] with Euclidean distance using the predefined training set. We compare SGRCEG with the Native Guide method proposed by [6] where the feature importance vectors are Shap values from the NUNs.

The results of these experiments are summarized in Table 1. We used two versions of SGRCEG: one where we used any shapelet associated with any class (SGRCEG) and one where we used shapelets of the opposite class to inform substitution (Separate SGRCEG). Overall, SGRCEG outperforms the Native Guide method and Separated SGRCEG, successfully finding more valid counterfactual explanations that are more succinct. Native Guide (Shap) does return counterfactual explanations that are marginally more realistic on average (i.e., higher LOF mean), but this is not surprising given the NUN is returned more frequently, which are by definition more realistic.

Figure 2 shows a queried time series that was predicted to have a myocardial infarction (pink), where SGRCEG looks for counterfactual explanations (blue stars), and the generated SGRCEG counterfactual explanation (blue dots). Looking at these results, a user can first identify which time window is important for receiving this prediction (i.e., the shapelet sequence) and how the time series needs to change to receive a different prediction. Since the explanations are succinct and grounded within a window the user expects, SGRCEG allows a user to quickly examine multiple queried time series in order to gain understanding about model behavior.

## 4 Conclusions & Future Work

We have presented a novel way to generate plausible and interpretable counterfactual explanations for time series data. In a real world example, we found SGRCEG typically produces counterfactual explanations that are more succinct and realistic when compared to a competing method. As this is a new area of research, there are still many paths for future work. It would be interesting to optimize where to find the ideal index at which to start the substitution, and it is also worth exploring different shapelet methods.

There are also other methods for determining whether or not a sample may be in distribution, which still needs to be explored in SGRCEG. Another interesting research direction would be to align time series or match time stamps so SGRCEG could work with time series of different lengths.

## References

[1] Emre Ates, Burak Aksar, Vitus J Leung, and Ayse K Coskun. Counterfactual explanations for multivariate time series. In *2021 International Conference on Applied Artificial Intelligence (ICAPAI)*, pages 1–8. IEEE, 2021.

[2] Omar Bahri, Soukaina Filali Boubrahimi, and Shah Muhammad Hamdi. Shapelet-based counterfactual explanations for multivariate time series, 2022.

[3] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: Identifying density-based local outliers. *SIGMOD Rec.*, 29(2):93–104, may 2000.

[4] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The UCR time series classification archive, July 2015.

[5] Ziqiang Cheng, Yang Yang, Wei Wang, Wenjie Hu, Yueting Zhuang, and Guojie Song. Time2Graph: Revisiting time series modeling with dynamic shapelets. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3617–3624, Apr. 2020.

[6] Eoin Delaney, Derek Greene, and Mark T Keane. Instance-based counterfactual explanations for time series classification. In *International Conference on Case-Based Reasoning*, pages 32–47. Springer, 2021.

[7] Riccardo Guidotti and Anna Monreale. Designing shapelets for interpretable data-agnostic classification. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '21, page 532–542, New York, NY, USA, 2021. Association for Computing Machinery.

[8] Riccardo Guidotti, Anna Monreale, Francesco Spinnato, Dino Pedreschi, and Fosca Giannotti. Explaining any time series classifier. In *2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI)*, pages 167–176, 2020.

[9] Isak Karlsson, Jonathan Rebane, Panagiotis Papapetrou, and Aristides Gionis. Explainable time series tweaking via irreversible and reversible temporal transformations. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 207–216, 2018.

[10] Markus Löning, Anthony J. Bagnall, Sajaysurya Ganesh, Viktor Kazakov, Jason Lines, and Franz J. Király. sktime: A unified interface for machine learning with time series. *CoRR*, abs/1909.07872, 2019.

[11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[12] Stan Salvador and Philip Chan. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580, 2007.

[13] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31(2), 2017.

[14] Zhengzheng Xing, Jian Pei, Philip S. Yu, and Ke Wang. *Extracting Interpretable Features for Early Classification on Time Series*, pages 247–258.