Understanding Survival Models through Counterfactual Explanations

Abdallah Alabdallah[1*], Jakub Jakubowski[2*], Sepideh Pashami[1], Szymon Bobek[3], Mattias Ohlsson[1], Thorsteinn Rögnvaldsson[1], and Grzegorz J. Nalepa[3]

[1] Center for Applied Intelligent Systems Research (CAISR)
Halmstad University, Halmstad, Sweden
[2] Department of Applied Computer Science
AGH University of Science and Technology, Krakow, Poland
[3] Faculty of Physics, Astronomy and Applied Computer Science, Institute of Applied Computer Science, and Jagiellonian Human-Centered AI Lab (JAHCAI), and Mark Kac Center for Complex Systems Research
Jagiellonian University, Kraków, Poland

Abstract. The development of black-box survival models has created a need for methods that explain their outputs, just as in the case of traditional machine learning methods. Survival models usually predict functions rather than point estimates. This special nature of their output makes it more difficult to explain their operation. We propose a method to generate plausible counterfactual explanations for survival models. The method supports two options that handle the special nature of survival models' output. One option relies on the Survival Scores, which are based on the area under the survival function, which is more suitable for proportional hazard models. The other one relies on Survival Patterns in the predictions of the survival model, which represent groups that are significantly different from the survival perspective. This guarantees an intuitive well-defined change from one risk group (Survival Pattern) to another and can handle more realistic cases where the proportional hazard assumption does not hold. The method uses a Particle Swarm Optimization algorithm to optimize a loss function to achieve four objectives: the desired change in the target, proximity to the explained example, likelihood, and the actionability of the counterfactual example. Two predictive maintenance datasets and one medical dataset are used to illustrate the results in different settings. The results show that our method produces plausible counterfactuals, which increase the understanding of black-box survival models.

Keywords: Survival Analysis, Explainable Artificial Intelligence, Survival Patterns, Counterfactual Explanations

[*] These authors contributed equally to this work

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

1 Introduction

Survival models are a special type of machine learning models, which predict the probability of survival over time. This group of models is widely used in the healthcare domain but has also been applied to predictive maintenance tasks [3, 5,27,32]. Complex machine learning tasks, involving high dimensionality of data, often require advanced machine learning models, which are not interpretable by humans, to achieve satisfactory performance. Safety critical domains, like the two mentioned above, usually require the explanations of model's decisions [23].

Explainability in AI refers to the ability to understand and interpret how a machine learning model makes decisions or predictions. Depending on the type of task and data, different explanation methods can be applied. Our work focuses on counterfactual explanations, which is a local explainability method that answers the question of what should change in the input to observe a different output. In this paper, we present how counterfactual explanations can be used to explain survival models. In contrast to classification and regression, the output of a survival model typically includes a survival function or a hazard function, representing the probability of survival over time, which is more difficult to explain.

In this work, we propose a method for generating counterfactual explanations for survival models that supports two options in terms of the definition of the change in output. The first option depends on the Survival Score, that is, the area under the survival function, to search for counterfactual examples that would change this score by a predefined value. In the second option, we use Survival Patterns [1] that represent the survival behaviors of groups in the population significantly different from each other.

Based on such Survival Patterns, we search for counterfactual examples that would change the predicted survival functions of subjects to predefined patterns. Other important aspects, that we consider in our research, are the plausibility and actionability of counterfactual explanations. In this work, we utilize an outlier detection model to drive counterfactual explanations close to the data distribution. We also added a special term to the loss function to handle categorical variables. Lastly, the actionability of the example is controlled by masking features that cannot be changed in practice. Restricting some features from changing can cause, in some cases, the target Survival Pattern to be not reachable. However, our method generates counterfactual examples that are closest to the target pattern. To the best of our knowledge, this is the first work that utilizes both survival scores and survival patterns to generate plausible and actionable counterfactual examples for survival models. A full implementation of our method is available on our GitHub repository [*].

[*] https://github.com/abdoush/SurvCounterfactual

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

2 Related Works

2.1 Survival Analysis Background

Survival models are a type of statistical and machine learning models that aim at modeling time to an event e.g., the machine failure or the patient's death. One of the problems in survival analysis is the presence of censoring. Some subjects will experience an event during the study, but others will survive beyond the study, which are called censored cases. Survival models usually predict functions, that is, survival or hazard functions. The survival function is the probability of surviving beyond a certain time t; i.e., the failure time T is greater than t: $S(t) = P(T > t)$. The Cox proportional hazards model (CPH) [6] is the first model to predict individualized hazard functions that depend on the individual's features $\mathbf{x}$. The CPH model assumes that hazards between subjects are proportional and independent of time. As a result, the survival curves of subjects do not intersect leading to unique area-under-curve for different survival curves, which is rarely the case in real life. To address these issues, machine learning models for survival analysis have been developed, such as Randon Survival Forests (RSF) [12], Survival Support Vector Machine [26] or deep learning approaches [2, 14, 20]. In contrast to CPH, these models are able to learn nonlinear relations between features and support nonproportional hazards. They usually offer improved performance in terms of accuracy but lack explanatory insight into their predictions.

2.2 Explainable AI and Counterfactual Explanations

The CPH model, due to its linearity, can be considered inherently explainable, which is a strong advantage in safety-critical domains. More complex machine learning models for survival analysis, require external explanations methods. Many such methods were proposed for classification or regression and extended to survival analysis. SurvLIME [16] extended the LIME [28] method, where it approximates the survival model locally with a CPH model. SurvSHAP(t) [18] extended SHAP [21] to explain survival functions that can capture time-dependent variable effects. SurvSHAP [1] used a proxy-based approach to explain the survival model with SHAP, using the patterns found in the output of the survival model to build the proxy model.

A promising XAI method is Counterfactual Explanations (CE), which belong to the family of example-based explanations aiming to find a 'similar' observation to the one we are explaining, but with a different model prediction. CE was originally proposed in [31], where the authors suggested creating explanations by minimizing the objective function consisting of two terms: the distance to the target output and the distance between the original observation and the counterfactual explanation. A major issue with this approach is that unrealistic explanations might be created. To deal with this problem, the distance between the explanation and the observed data can be minimized [7], a model which estimates the likelihood of point belonging to a data distribution, e.g. Autoencoder,

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

can be used [8], or a generative model can be employed to generate candidates for explanations [24].

In [17], the authors propose a method for generating counterfactual explanations for survival models using the mean survival time as the target. Our method, while sharing similarities, enhances this approach by incorporating additional terms in the optimization to improve the likelihood and actionability of generated counterfactual examples. Moreover, our alternative option utilizes Survival Patterns which identify distinct risk groups based on the entire survival function rather than mean-time alone. This adaptation accommodates both proportional and non-proportional hazard models and facilitates specifying meaningful target changes between survival groups.

3 Research Methods

Our method is model-agnostic in the sense that it depends only on the output of the survival model. Furthermore, the method does not require access to the training data after the model is trained. We use the Particle Swarm Optimization (PSO) algorithm to optimize our objective function to meet four criteria: 1) achieving the desired change in the target output, 2) minimizing the change to the input, 3) the plausibility of the counterfactual example, and 4) the actionability of the counterfactual example. Our method provides two options with

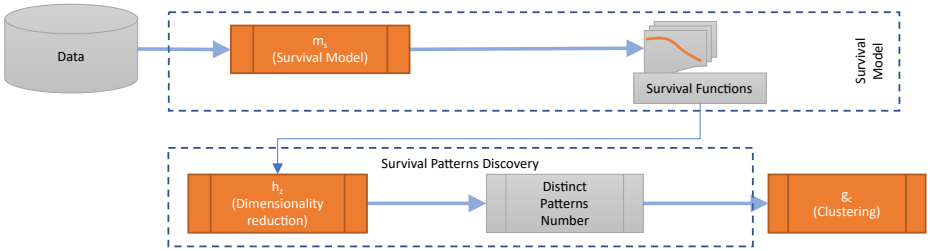


Fig. 1: Patterns discovery Workflow

respect to the first criterion, i.e. achieving the desired change in the target output. The first option relies on the Survival Score, representing the area under the survival function (mean survival time). This treats the survival problem as a regression task, aiming to find a counterfactual that achieves a specified change to the survival score, detailed in Section 3.1. While effective for proportional hazards with non-intersecting survival curves, it has limitations in cases of non-proportional hazards, where distinct survival behaviors can yield the same area under the curve.

To address nonproportional hazards, our second method option employs Survival Patterns. This involves grouping survival curves into distinct and significantly different Survival Patterns, each representing curves with similar surviv-

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

ability. Determining the optimal number of patterns and clustering curves in a lower-dimensional space transforms the problem into a classification task. The objective is to find a counterfactual example that changes the predicted survival curve to a predefined Survival Pattern. Further details on survival patterns are provided in Section 3.1.

3.1 Generation of Counterfactual Explanations

The objective function consists of three weighted terms. The purpose of each term is to fulfill one of the criteria described earlier.

$$\mathcal{L} = \alpha\mathcal{L}_y + \beta\mathcal{L}_x + \gamma\mathcal{L}_{LL} \tag{1}$$

The first term, $\mathcal{L}_y$, induces the desired target change, the second term, $\mathcal{L}x$, promotes proximity between the counterfactual and the original input, and the third term, $\mathcal{L}_{LL}$, ensures the counterfactual's likelihood. The last criterion, which is the actionability of the generated example, is realized by constraining the selected input feature from changing based on the domain expert knowledge. The following subsections provide details about each part of the objective function.

Counterfactual Explanations with Survival Scores The survival score y, calculated as the area under the survival curve, represents the mean survival time of a subject. We reframe our explanation task as determining the necessary changes in observed features X to increase the predicted survival score by Y. These counterfactual explanations reveal the feature adjustments that can positively impact the expected survival time. In practical applications, these explanations guide adjustments to extend the machine's or patient's life. The term $\mathcal{L}_y$ in the objective function associated with the target value of the survival score $\mathcal{L}_y = |\mathbf{y} + \delta - \hat{\mathbf{y}}|$, where $\mathbf{y}$ is the original survival score, δ is the required change in the survival score, $\hat{\mathbf{y}}$ is the survival score of the counterfactual candidate, and $|.|$ indicates the absolute value.

Counterfactual Explanations with Survival Patterns Survival models predict survival curves of different shapes and levels. Similarities between predicted curves reflect similarities in input subjects that can define risk groups that are significantly different from the survival perspective. Each of these risk groups has its own shape and level of survival function, which is called a Survival Pattern [1]. Such patterns can be used to explain the behavior of the survival model. Our method constructs the counterfactual example by finding the minimum change to the input features, which changes the prediction of the survival model from a source Survival Pattern to a specific target one.

Following what has been done in [1], Survival Patterns are discovered as follows. Let $X \in \mathbb{R}^{n\times p}$ be the input of the survival model m_s, and $S \in \mathbb{R}^{n\times m}$ where $S = m_s(X)$ be the output survival functions, where n is the number of examples, p is the number of features, and m is the number of timesteps in the predicted survival functions. The algorithm has three steps; see Figure 1:

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

- **Lower Dimensional Representation**: Survival functions, which are discrete one-dimensional signals of probabilities over m time steps, $S \in \mathbb{R}^{n \times m}$, are transformed into a lower dimensional space using the function h_z. This results in $Z \in \mathbb{R}^{n \times r}$ where r is the number of dimensions of the new space (Z-space). In this work, we used Principal Components Analysis (PCA) and chose the number of components r, which maintains an explained variance over 99%.
- **Finding the Number of Survival Patterns**: Using the lower dimensional representation Z, the algorithm iteratively clusters the curves into $k \in \{2, 3, ..., K_{max}\}$ clusters using the k-means clustering algorithm. At each iteration, pair-wise comparisons between the resulting clusters are performed based on the log-rank test [25], which is a statistical test to assess the statistical difference between two groups from a survival point of view. The k^* that is selected is the largest k that achieves the maximum percentage of significantly different groups.
- **Survival Patterns Prediction Model**: In this step, a k-means clustering model g_c is fitted on the Z features, using the optimal number of Survival Patterns k^*.

The final Survival Patterns prediction model f, which will be used in the search for counterfactual examples, is composed of three models: $f(\mathbf{x}) = (g_c \circ h_z \circ m_s)(\mathbf{x})$. Based on the function f and the clusters' centers in the Z-space $c_i : i \in \{0, ..., k^* - 1\}$, it is possible to compute the distance between the survival curve of the proposed counterfactual $\mathbf{x}_{cf}$ and the survival curve of the center of the target pattern c_t in the Z-space, which will be used as the target part $\mathcal{L}_y$ of the loss function:

$$\mathcal{L}_y = \mathbb{1}((f(\mathbf{x}_{cf}) \neq t) \| (h_z \circ m_s)(\mathbf{x}_{cf}) - c_t \|_2 \tag{2}$$

where t is the desired target Survival Pattern, $\|.\|_2$ is the L_2 norm, and $\mathbb{1}(.)$ is the indicator function. The use of the indicator function will block the effect of this part of the loss function once the counterfactual crosses the boundaries of the target pattern.

Minimal Change to the Features The second term of the objective function $\mathcal{L}_x(\mathbf{x}_{cf}) = \|\mathbf{x} - \mathbf{x}_{cf}\|_p$ aims at minimizing the distance between the original example $\mathbf{x}$ and the generated counterfactual $\mathbf{x}_{cf}$, where p is the order of the L^p-norm. In this work, we used L^1-norm to encourage sparsity in the difference between the explained example and the counterfactual one.

Likelihood of Counterfactual Explanations To ensure the plausibility of counterfactual explanations, we utilize an Autoencoder model (AE) fitted to the training data. The model is used to determine the reconstruction error between the counterfactual candidate and its reconstructed version by the AE, which we call the anomaly score $\mathcal{L}_{AE} = \text{ReLU}(\|\mathbf{x}_{cf} - \mathbf{x}'_{cf}\|_p - A_t)$, where p is the order of the L^p-norm, x'_{cf} is the output of the AE model, and A_t is the anomaly threshold.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

The threshold A_t is estimated based on the residuals of the test data set using the formula $Q3 + 1.5 * IQR$ where IQR is the interquartile range and $Q3$ is the upper quartile. A higher anomaly score indicates a candidate's deviation from the original data distribution. While autoencoders have been used for unlikely counterfactuals, our contribution involves introducing an anomaly threshold A_t with the ReLU function. If the reconstruction error is below this threshold, the loss term $\mathcal{L}_{AE}$ becomes zero, aiming to halt its impact when the counterfactual is sufficiently likely. This approach can result in counterfactual examples closer to the original subject. We also included an additional term $\mathcal{L}_{ohe}$ that ensures that the generated one-hot-encoded features have valid codes. These two terms, $\mathcal{L}_{AE}$ and $\mathcal{L}_{ohe}$, constitute our Likelihood Loss (LL), $\mathcal{L}_{LL} = \mathcal{L}_{AE} + \mathcal{L}_{ohe}$, that is responsible for determining the plausibility of the counterfactual.

Actionable Counterfactual Explanations Counterfactual explanations achieve actionability by employing a boolean vector, provided by a domain expert, to mask uncontrollable features. Features designated by this vector remain unchanged, focusing the optimization algorithm on modifying other features to meet the objective. While constraints on input features may limit achieving the exact desired objective, our method aims to find the nearest attainable target in such cases.

3.2 Particle Swarm Optimization

Particle Swarm Optimization [15] (PSO) is a nonlinear function optimization method inspired by bird flocking behavior. In this research, PSO is employed to minimize the objective function for generating counterfactual explanations. Using N randomly initialized particles in the search space, each particle evaluates the objective function at its position. Particle positions are adjusted based on both individual and neighboring experiences, facilitating effective exploration and convergence to a near-optimal solution. The algorithm's performance can be enhanced by adjusting hyperparameters, a process detailed in the results section.

3.3 Datasets Description

Turbofan engine Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) is a NASA-developed software for turbofan engine simulation. Saxena et al. [29] used C-MAPSS to create a dataset modeling engine degradation and failure, comprising four subsets with varying complexity. This analysis focuses on a subset with one operating mode and two failure modes, involving 100 engines. To align with survival analysis, we truncated the dataset after the 300^{th} observation, marking these as no-event observations and removing subsequent data. The original 21 sensor measurements were reduced to 13 by eliminating redundant features based on the correlation coefficient.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

Predictive Maintenance Dataset (PM) This is a dataset publicly available from [11]. The dataset contains information about machines that are provided by four providers labeled `Provider1` to `Provider4` and operated by three teams labeled `TeamA`, `TeamB`, and `TeamC`. The dataset also contains information about operating conditions (pressure, moisture, and temperature) measured through sensors. The aim is to study the lifetime of these machines under the aforementioned operating conditions.

Flchain Dataset [9] This dataset is a publicly available medical dataset aimed at studying whether the Free Light Chain (FLC) assay is a good predictor of the survival probability of patients. For the sake of visualization, we only considered the three most important features, `age`, Σ`FLC` (which is the summation of the `kappa` and `lambda` features in the original dataset), and `creatinine`.

4 Results and Discussion

In this section, we conduct experiments on three datasets described in Section 3.3 to demonstrate the effectiveness of the proposed methods in generating counterfactual explanations. We explore various approaches tailored to different types of data and tasks.

At first, we conduct a comparison and hyperparameter tuning of Particle Swarm Optimization (PSO) and Simulated Annealing (SA) algorithms. Our findings indicate that utilizing PSO improves explanation generation across all three datasets.

The first experiment utilizes a Turbofan engine dataset, showcasing the generation of counterfactuals based on Survival Scores. In the second experiment, we used our method with Survival Patterns applied to the PM dataset, which has categorical features. The third experiment highlights the model's behavior when the target Survival Pattern is unattainable due to the presence of unactionable features. We employ the Flchain dataset that has the `Age` feature, which is naturally unactionable.

It is worth noting that in the first two experiments, we compared the results generated with and without using the LL term in the loss function. In the Survival Score option, not using the LL term makes our method similar to the method proposed in [17], which makes the first experiment a direct comparison between the two works. However, the comparison is indirect in the second experiment as we employ the Survival Patterns option which is not supported by [17].

4.1 Particle Swarm Optimization vs. Simulated Annealing

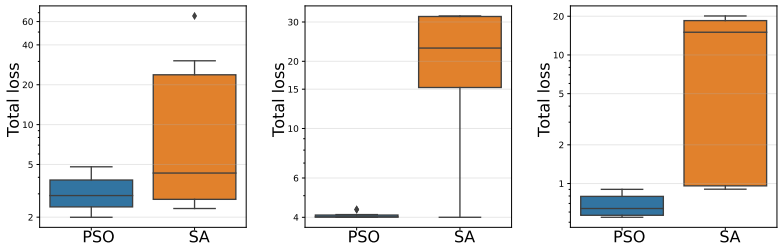
In this section, we compare the convergence of the PSO algorithm with the SA algorithm, optimizing both and assessing convergence based on the final loss. For the PSO algorithm, key hyperparameters include the number of particles, cognitive coefficient (c1), social coefficient (c2), and inertia weight (w). First, we

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

set c1, c2, and w based on empirically validated values [4], that is, $c_1 = 1.49618$, $c_2 = 1.49618$, and $w = 0.7298$, and optimized the number of particles to minimize computation time. Subsequently, we used these values to optimize c1, c2, and w for each dataset via a random search for 1000 iterations. We performed similar optimization for SA algorithm hyperparameters (T start, T end, iterations, and step), with the final values listed in Table 1. The PSO algorithm

Table 1: PSO and SA Optimized hyperparameters.

	PSO				SA			
Dataset	particles	c1	c2	w	T start	T end	iterations	step
CMAPSS	900	1.780533	1.911480	0.247112	0.203940	0.001783	750	0.039156
PM	5000	0.641856	1.125175	0.013541	0.264709	0.010108	680	0.285497
FLCHAIN	200	0.119708	1.473350	0.222749	0.415107	0.028302	870	0.180351



(a) CMAPSS (b) PM (c) FLCHAIN

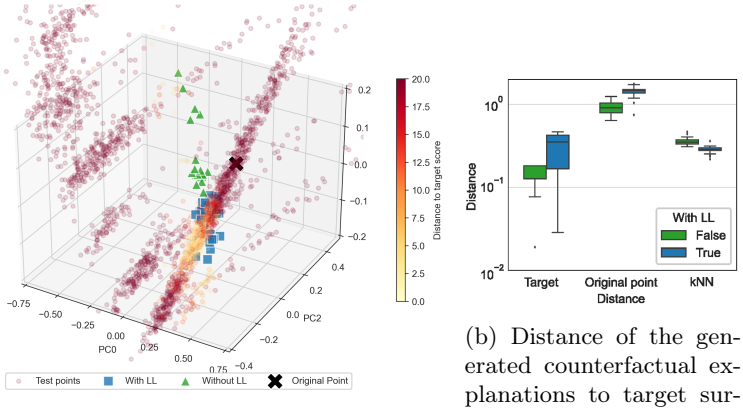
Fig. 2: Loss comparison between PSO and SA on the three datasets

showed better convergence which is evident in the final loss distributions in Figure 2. This superior performance of the PSO algorithm can be attributed to the fast computation of the loss function (0.13s per step) that allowed us to use a large number of particles in the PSO algorithm. Based on the previous results, we continue with the PSO algorithm in the following sections.

4.2 Survival-Scores-based Counterfactual Explanations

This experiment illustrates the Survival Scores approach on the turbofan engine dataset. We employed the RSF model to predict the survival probability for each unit. Conducting an investigation, we randomly selected one unit, predicting its survival probability after 200 cycles. Our objective was to identify the required changes in feature values to increase its survival score by 30%. To gather statistics on the generated counterfactual examples, we conducted the experiment 20 times, both with and without utilizing the LL loss.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28



(a) A 3D PCA projection of the CMAPSS dataset with counterfactual explanations generated for a randomly selected point

(b) Distance of the generated counterfactual explanations to target survival score, original observation, and five nearest neighbors from dataset

Fig. 3: PM dataset results of counterfactual explanations with and without LL.

In Figure 3a, a 3D PCA projection illustrates the dataset. The colors of the test observations denote proximity to the target survival score in counterfactual explanations. Desirable counterfactuals should be near the yellow region and relatively close to the original data. Counterfactuals without LL mostly lie outside the original distribution, making them less informative. Those with LL are closer to the desired region, suggesting higher explanatory validity.

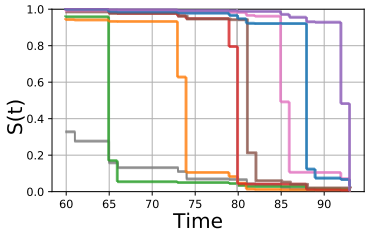
In Figure 3b, distances of counterfactual explanations to the target survival score, the original point, and the five nearest neighbors are presented. Including the LL Loss increased the distances to the original point and target survival score. However, the difference in the target score is negligible in terms of explanation validity, given its higher magnitude (10^2). Importantly, our goal was achieved as LL Loss inclusion resulted in explanations with improved proximity to the original data, measured by the proximity to the five nearest neighbors from the original points closest to the desired target score.

4.3 Survival-Patterns-based Counterfactual Examples

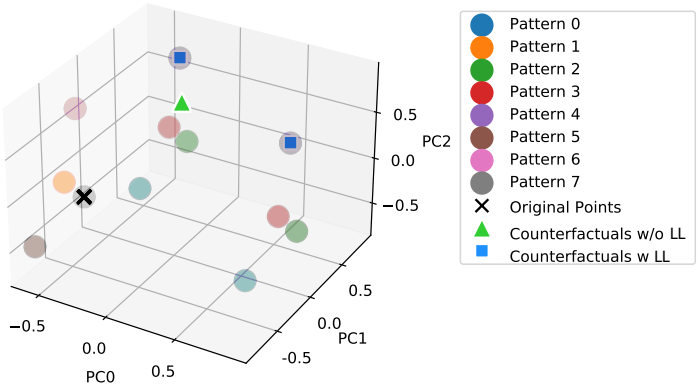
This experiment aims to showcase the use of Survival Patterns in the generation of counterfactual examples. Particularly, we illustrate that our method is capable of handling one-hot-encoded categorical features.

We used the RSF model trained on the PM dataset. Our method recognized eight Survival Patterns in the prediction of the RSF model, as shown in Figure 4a. It is worth noting that the numbers associated with the patterns do not reflect any kind of order; rather, they are assigned by the clustering algorithm.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28



(a) Survival Patterns



(b) A 3D PCA projection of the dataset

Fig. 4: PM dataset results of counterfactual explanations with and without LL.

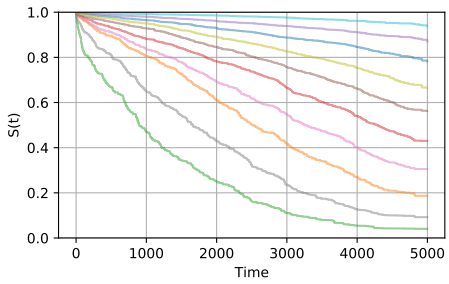
We chose all the data points in Pattern 7 (The worst Survival Pattern) as the source pattern and generated counterfactual examples setting the target Pattern to Pattern 4 (The best Survival Pattern). For each point, we generated two counterfactual examples with and without using the Likelihood Loss. Figure 4b shows a three-dimensional PCA projection of the data colored with their respective Survival Patterns, with the counterfactual examples with and without LL. It is worth noting that each circle in the PCA plot represents many points very close to each other and belongs to a specific combination of categorical values. This means that any point far from these circles would have an invalid categorical value. Although the counterfactual examples without LL correctly changed the model's decision to the target pattern, they are unrealistic and far from the data distribution. While the counterfactual examples with LL are very close to the data distribution of the target pattern. In fact, the change from Pattern 7 to Pattern 4 requires only changing the categorical features from `TeamC` to `TeamA` or `TeamB` and from `Provider3` to `Provider2`. This is what the algorithm did using the Likelihood Loss, which enabled it to generate examples with a valid one-hot encoding, shown as blue squares in Figure 4b. Without Likelihood Loss,

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

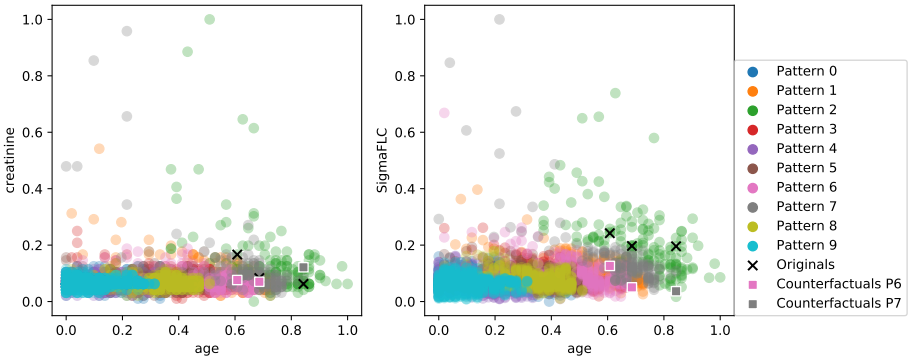
unrealistic examples with invalid one-hot encoding were generated, shown as a green triangle in Figure 4b.

4.4 Actionability of Counterfactual Explanations

In this experiment, we show an example of actionable counterfactual explanations. This is done by restricting the changes in some features. This will also show a case where the predefined target Survival Pattern cannot be reached because of this restriction. RSF model is trained on the Flchain dataset, where our method identified ten Survival Patterns as shown in Figure 5a. We chose three source examples from the worst Survival Pattern (pattern 2) and set the target pattern to the best Survival Pattern (pattern 9).



(a) Survival Patterns



(b) 2D projections of the dataset with different features on the y axis.

Fig. 5: Actionable counterfactual explanations masking the Age feature.

To generate actionable counterfactuals, we applied a mask to disallow the Age feature change. This condition made the target pattern unreachable. Our method relies on reaching the target pattern by minimizing the distance to the center of that pattern in the Z-space as shown in Equation 2. This will get the

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

counterfactual to the nearest-to-target pattern it can reach. Figure 5b shows two 2D projections of the data points colored with their respective patterns, the three selected source points (marked as ×s) from the source pattern (Pattern 2, colored in green) and the target pattern (Pattern 9, colored in cyan). However, because the target pattern is not reachable without moving along the Age feature, the method generated counterfactuals as close as possible to the target pattern. The respective counterfactuals are colored based on the patterns that they were able to reach (in this case, Patterns 6 and 7).

5 Conclusion

In this paper, we presented a method of generating plausible Counterfactual Explanations for black-box survival models. The proposed method finds the nearest plausible point to the explained observation that changes the output of the model. That is by changing the survival pattern or survival score of the studied example while maintaining the plausibility of the counterfactual example by minimizing the reconstruction loss of an Autoencoder model trained on the original data. The actionability is also guaranteed by restricting the changes in certain features.

We validated our method on three publicly available datasets. We generated counterfactual explanations for selected observations with and without the inclusion of Likelihood Loss. The results showed that not using the plausibility constraint can result in unlikely explanations. We also observed that restricting the change in some features can make the target pattern unattainable in some cases. However, in such a case, our method generates counterfactual explanations that are closest to the target pattern.

This work proposed a promising direction for explaining survival models using counterfactual explanations, as they can be easily interpreted by humans. A potential future work on this topic is to generate multiple diverse counterfactual explanations for a single subject. This is an important issue, which is a subject of research in Counterfactual Explanations [22] and can be used to strengthen the applicability of our method.

Acknowledgements

This research was funded by the CHIST-ERA XPM project, CHISTERA-19-XAI-012, and the CAISR+ project funded by the Swedish Knowledge Foundation. Project XPM is supported by the National Science Centre, Poland (2020/02/Y/ST6/00070), under CHIST-ERA IV program, which has received funding from the EU Horizon 2020 Research and Innovation Programme, under Grant Agreement no 857925.

References

1. Alabdallah, A., Pashami, S., Rögnvaldsson, T. & Ohlsson, M. SurvSHAP: A Proxy-Based Algorithm for Explaining Survival Models with SHAP. *2022 IEEE 9th Inter-*

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

national Conference On Data Science And Advanced Analytics (DSAA). pp. 1-10, doi: 10.1109/DSAA54385.2022.10032392 (2022)
2. Alabdallah, A. & Ohlsson, M. & Pashami, S & Rögnvaldsson, T. The Concordance Index decomposition: A Measure for a Deeper Understanding of Survival Prediction Models, *Artificial Intelligence in Medicine*. **148**, 102781, doi: 10.1016/j.artmed.2024.102781 (2024)
3. Alabdallah, A. & Rognvaldsson, T. & Fan, Y. & Pashami, S., & Ohlsson, M. Discovering Premature Replacements in Predictive Maintenance Time-to-Event Data. In PHM Society Asia-Pacific Conference (Vol. 4, No. 1), doi: 10.36001/phmap.2023.v4i1.3609 (2023)
4. Altarabichi, MG & Nowaczyk, S. & Pashami, S. & Sheikholharam Mashhadi, P. Fast Genetic Algorithm for feature selection — A qualitative approximation approach. Expert Systems with Applications vol. 211, doi: 10.1016/j.eswa.2022.118528 (2023)
5. Chen, C., Liu, Y., Wang, S., Sun, X., Di Cairano-Gilfedder, C., Titmus, S. & Syntetos, A. Predictive maintenance using cox proportional hazard deep learning. *Advanced Engineering Informatics*. **44** pp. 101054 (2020)
6. Cox, D. Regression Models and Life-Tables. *Journal Of The Royal Statistical Society. Series B (Methodological)*. **34**, 187-220 (1972)
7. Dandl, S., Molnar, C., Binder, M. & Bischl, B. Multi-Objective Counterfactual Explanations. *Parallel Problem Solving From Nature – PPSN XVI*. pp. 448-469 (2020)
8. Dhurandhar, A., Chen, P., Luss, R., Tu, C., Ting, P., Shanmugam, K. & Das, P. Explanations based on the Missing: Towards Contrastive Explanations with Pertinent Negatives. *Advances In Neural Information Processing Systems*. **31** (2018)
9. Dispenzieri, A., Katzmann, J., Kyle, R., Larson, D., Therneau, T., Colby, C., Clark, R., Mead, G., Kumar, S., Melton III, L. & Rajkumar, S. Use of nonclonal serum immunoglobulin free light chains to predict overall survival in the general population. *Mayo Clin Proc*. **87**, 517-23 (2012)
10. Eberhart, R. C. & Shi, Y. Comparing inertia weights and constriction factors in particle swarm optimization. Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512), La Jolla, CA, USA, 2000, pp. 84-88 vol.1, doi: 10.1109/CEC.2000.870279. (2000)
11. Fotso, S. & Others PySurvival: Open source package for Survival Analysis modeling. , https://www.pysurvival.io/
12. Ishwaran, H., Kogalur, U., Blackstone, E. & Lauer, M. Random survival forests. *Ann. Appl. Stat.*. **2**, 841-860 (2008,9)
13. Kaplan, E. & Meier, P. Nonparametric Estimation from Incomplete Observations. *Journal Of The American Statistical Association*. **53**, 457-481 (1958)
14. Katzman, J., Shaham, U., Cloninger, A., Bates, J., Jiang, T. & Kluger, Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network.. *BMC Medical Research Methodology*. **18**, 24 (2018)
15. Kennedy, J. & Eberhart, R. Particle swarm optimization. *Proceedings Of ICNN'95 - International Conference On Neural Networks*. **4** pp. 1942-1948 vol.4 (1995)
16. Kovalev, M., Utkin, L. & Kasimov, E. SurvLIME: A method for explaining machine learning survival models. *Knowledge-Based Systems*. **203** pp. 106164 (2020)
17. Kovalev, M., Utkin, L., Coolen, F. & Konstantinov, A. Counterfactual Explanation of Machine Learning Survival Models. *Informatica*. **32**, 817-847 (2021,1)
18. Krzyziński, M., Spytek, M., Baniecki, H. & Biecek, P. SurvSHAP(t): Time-dependent explanations of machine learning survival models. *Knowledge-Based Systems*. **262** pp. 110234 (2023)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28

19. Lang, J., Giese, M., Ilg, W. & Otte, S. Generating Sparse Counterfactual Explanations For Multivariate Time Series. (arXiv,2022)
20. Lee, C., Zame, W., Yoon, J. & Schaar, M. DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks. *Proceedings Of The AAAI Conference On Artificial Intelligence*. **32** (2018)
21. Lundberg, S. & Lee, S. A Unified Approach to Interpreting Model Predictions. *Advances In Neural Information Processing Systems 30*. pp. 4765-4774 (2017)
22. Mothilal, R., Sharma, A. & Tan, C. Explaining machine learning classifiers through diverse counterfactual explanations. *Proceedings Of The 2020 Conference On Fairness, Accountability, And Transparency*. pp. 607-617 (2020)
23. Pashami, S. et al, Explainable Predictive Maintenance. arXiv:2306.05120 [cs.AI], (2023).
24. Pawelczyk, M., Broelemann, K. & Kasneci, G. Learning Model-Agnostic Counterfactual Explanations for Tabular Data. *Proceedings Of The Web Conference 2020*. pp. 3126-3132 (2020)
25. Peto, R. & Peto, J. Asymptotically Efficient Rank Invariant Test Procedures. *Journal Of The Royal Statistical Society. Series A (General)*. **135**, 185-207 (1972)
26. Pölsterl, S., Navab, N. & Katouzian, A. Fast Training of Support Vector Machines for Survival Analysis. *Machine Learning And Knowledge Discovery In Databases*. pp. 243-259 (2015)
27. Rahat, M., Kharazian, Z., Mashhadi, P.S., Rögnvaldsson, T. and Choudhury, S. Bridging the Gap: A Comparative Analysis of Regressive Remaining Useful Life Prediction and Survival Analysis Methods for Predictive Maintenance. In PHMAP Conference (Vol. 4, No. 1), doi: 10.36001/phmap.2023.v4i1.3646 (2023).
28. Ribeiro, M., Singh, S. & Guestrin, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings Of The 22nd ACM SIGKDD*. pp. 1135-1144 (2016)
29. Saxena, A., Goebel, K., Simon, D. & Eklund, N. Damage propagation modeling for aircraft engine run-to-failure simulation. *2008 International Conference On Prognostics And Health Management*. pp. 1-9 (2008)
30. Van Looveren, A. & Klaise, J. Interpretable Counterfactual Explanations Guided by Prototypes. ECML PKDD 2021. pp. 650-665 (2021)
31. Wachter, S., Mittelstadt, B. & Russell, C. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR. *Harvard Journal Of Law & Technology*. (2017)
32. Yang, Z., Kanniainen, J., Krogerus, T. & Emmert-Streib, F. Prognostic modeling of predictive maintenance with survival analysis for mobile work equipment. *Scientific Reports*. **12** (2022,5)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_28