Counterfactual Explanation Manual; a ProM Plugin

Mahnaz Sadat Qafari

Process and Data Science Department, RWTH Aachen University
Lehrstuhl fur Informatik 9 52074 Aachen, Germany
m.s.qafari@pads.rwth-aachen.de

Abstract. Organizations are interested in detecting the reason behind the mediocre performance of their processes. They can immensely benefit from the explanations that not just explain the reason of the defect but also are actionable prescriptions. In this paper, we introduce *counterfactual explanations* plugin in ProM that is dedicated to providing such explanations in the process instance level.

Keywords: Process mining · counterfactual explanation · process instance level.

1 Introduction

Suppose in a loan company the process of validating the application of one of the customers took too long and we want to see if this delay could have been prevented had more resources been assigned to that case. Such retrospective questions are known as counterfactual questions. The *counterfactual explanations* (or an *explanation* for short) is a ProM plugin that is dedicated to answer such questions in processes.

A counterfactual explanation can be defined as an imaginary instance as close as possible to the what that has happened in reality which would have resulted in a different outcome. An explanation for an instance should have the following properties:

- It should be as close as possible to the what that has happened in reality.
- It should result in a different outcome.
- It should be simple and easy to understand as it is meant to be used by humans. Therefore those explanations with fewer features with different values from the reality are more desirable.

Apart from the above properties, an explanation should be customized and applicable for a specific situation. And it is easy to imagine that different explanations would be applicable for two identical situations with the same undesirable outcome (because of the individual differences). Therefor, this plugin provides a diverse set of explanations so that the user (which is associated to an instance)

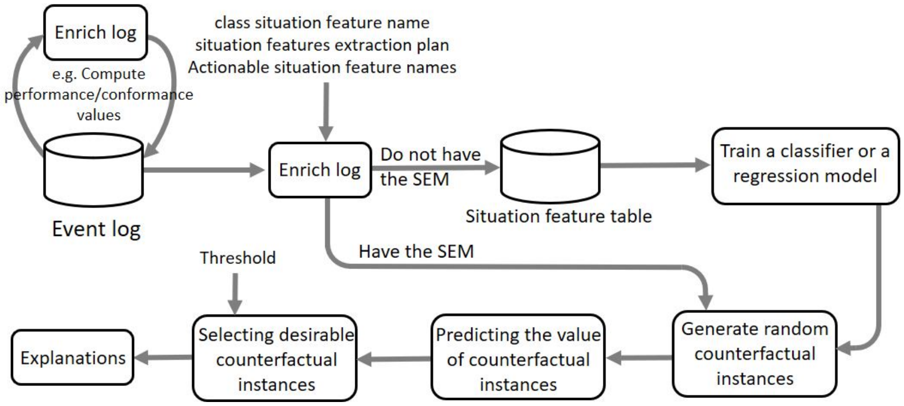

Enrich log
e.g. Compute performance/conformance values
Event log
class situation feature name
situation features extraction plan
Actionable situation feature names
Enrich log
Do not have the SEM
Situation feature table
Train a classifier or a regression model
Have the SEM
Threshold
Explanations
Selecting desirable counterfactual instances
Predicting the value of counterfactual instances
Generate random counterfactual instances

Fig. 1. An overview of the the method that is used to provide the set of diverse explanations.

can select the most suitable one for her/his individual case. This set of explanations is diverse; i.e., its explanations differ from each other in the subset of features that disagree with the given instance.

Figure 1 shows an overview of the method that is used to provide the set of diverse explanations. In the first step, we enrich the event log by adding several derivative features to the events and process instances of the event log. The user provides a target feature, a set of descriptive features which he/she believes may have a causal effect on the target feature, a set of actionable features which is a subset of descriptive features that can be modified by the user, and a threshold for desirable values of the target feature (or a set of values in the case of nominative target feature). He/she also identifies a (an undesirable) instance that needs to be explained. The output of the plugin is a set of explanations that helps the user to understand the reason for the process outcome for the selected instance. This instance is presented as a set of feature name feature values and called *given instance*. Then, a large number of random instances, which we call then *possible instances*, similar to the given instance are generated. In half of the generated instances, the values of the actionable features are selected randomly from their domain and in another half from their distribution. Please refer to [1] for more information on underlying concepts and methods.

Running example

Download the event log provided in here (itlog.xes), let's call this event log *itlog*. Also, download the file named "sem.txt", we will need it later on. This event log includes 1000 traces of a company that implements software for its customers. The implementation of the software for the case with case-id 1000 took 577 person days which is too long. The customer believes it should have taken at most 500 person days. Now we want to explain to him why it took more than 500 person days, whether it could have been shorter and what should change in future orders to have implementation activity taking less than 500 person days.

2 Step by Step Manual

The inputs of the plugin are the event log, the Petri net model, and the result of replaying the event log on the given Petri net model. These inputs are used to enrich the event log by adding event and trace level derivative attributes to the event log. The output of the plugin is a set of diverse explanations for the given instance. To use this plugin two steps are required, data extraction and explanation generation. Here we explain the setting needed for each of these steps separately.

Let's start!

Download the latest version of the nightly built of ProM from here. Run "PackageManager.bat" and add "counterfactual explanations" plugin to ProM. Run "ProM.bat" and import the itlog.
Use *inductive miner* with the default setting to discover the Petri-net. Now, use the "replay a log on Petri-net for conformance analysis" for the replay results. Please use the "event name" as the classifier. Now use these three as the input of the "counterfactual explanations" plugin. The UI of the plugin (current version) would be as shown in Figure 2

! To use this plugin, all the descriptive features and the target feature must be numerical.

2.1 Data Extraction

The UI for the data extraction setting is shown in Figure 2. The required setting is as follows:

1. *Situation type*. Situation type is determining the type of the target feature and consequently the part of the process instance that the data would be extracted from. Three values for a situation type are possible which are trace, event, or choice.

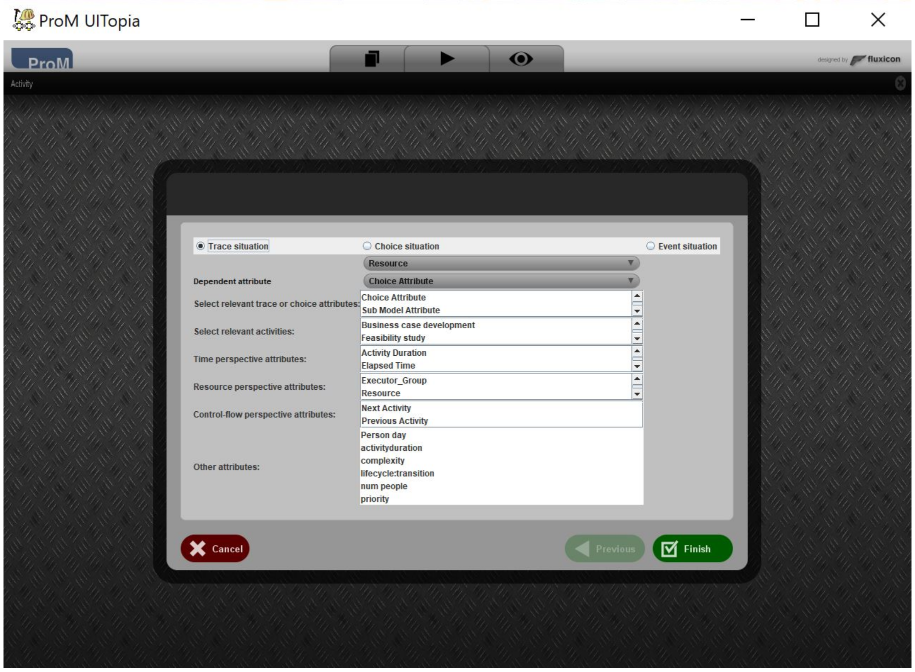
2. *Target feature*. The selection of the target feature depends on the type of situation that has been selected in the previous step.
   - Trace situation. The first drop-down list is deactivated and the second one can be used to select one of the trace level attributes.
   - Choice situation. The Petri net of the process is visualized in a window where one of the choice places can be selected as the target feature.
   - Event situation. Using the first drop-down list the set of activities that the class feature belongs to can be selected. The set of activities can be selected based on their resource, duration, timestamp, or activity name. Then using the second drop-down list can be used to select one of the event level attributes as the target feature.
3. *Trace level descriptive features*. If the data includes descriptive trace level attributes, then we need to select the activity names using the *select relevant activities* list. Please note that, if you select the *choice attribute*, the Petri net model of the process would be demonstrated on a window and you can select some of the choice places on it. The choice made in these choice places will be considered as descriptive features. Similarly, if you select *Sub-model attribute*, you have to select a sub-model of the Petri net model of the process by selecting some of its transitions. The duration of the selected sub-model will be considered as one of the descriptive features.
4. *Activities to consider*. If the data includes descriptive event level attributes, then we need to select the relevant activity names using the *select relevant activities* list.
5. *Event level descriptive features*. Select the descriptive feature names using the last four lists of the UI (Figure 2). The attribute names are categorized into four main groups: time perspective, resource perspective, control-flow perspective, and other attributes.

To create the data table, click on *finish* button.

Data table setting

Do the following steps:

1. select event situation radio button.
2. in the first drop-down list select “implementation”,
3. in the second drop-down list select “person day”,
4. in the “Activities to consider” list (the second list) select “feasibility study”.
5. in the “other attribute” list (the last list) select “complexity”, “person day”, “num people” and “priority”. By now, the UI should look like the one in Figure 4.
6. Click on the “finish button.



● Trace situation ○ Choice situation ○ Event situation

Resource

Dependent attribute: Choice Attribute

Select relevant trace or choice attributes: Choice Attribute, Sub Model Attribute

Select relevant activities: Business case development, Feasibility study

Time perspective attributes: Activity Duration, Elapsed Time

Resource perspective attributes: Executor_Group, Resource

Control-flow perspective attributes: Next Activity, Previous Activity

Other attributes: Person day, activityduration, complexity, lifecycle:transition, num people, priority

Fig. 2. The UI of the plugin for the setting of the feature table.

2.2 Counterfactual Explanation setting

The UI for selecting the instance that needs to be explained (given instance) and other needed settings are shown in Figure 3. The required settings are as follows:

- *select the instance.* The instance that needs to be explained can be selected using the top drop-down list. Here each instance is demonstrated with its case-id and the value for its target feature. For example, 1000[577] refers to the instance in which the case-id is 1000 and the target feature is 577.
- *Threshold.* Using this slider, you can select the threshold of the acceptable values for the target feature.
- *Is lower desirable.* This checkbox indicates if you want the value of the target feature for the counterfactual instances generated for the given instance to be lower or bigger than the given threshold in the previous step.
- *Optimization.* Using this check box, you indicate if you want to use optimization to generate counterfactual instances as close as possible to the given instance.
- *Method.* We need to model the data to be able to predict the value of the target features for the generated counterfactual instances. The following methods for modeling the data are supported by the plugin:

  - *Structural Equation Model (SEM).* If you have access to the SEM of the data, then you can upload it and use it for the prediction the value of the target feature for the counterfactual instances.
  - *Neural network.* For the neural network, you need to input the learning rate, momentum, Training time (number of epochs), and the hidden layers (for example if you want to have two hidden layers the first one with 16 and the second one with 8 nodes, then the input should be "16,8").
  - *Locally weighted learning.* In this method, we use locally weighted regression to model the data. For that, you need to set the number of neighbors as well as the shape of the weighting kernel. The kernel can be either linear, epanechnikov, tricube, inverse, or gaussian.
  - *Regression tree.* In this case, the maximum depth of the tree and the minimum number of instances per leaf have to be set.
- *Actionable attributes.* Not all the attributes are actionable. Using this drop-down list, you can select the set of attributes that you want to be changed. Those that are not selected will remain the same as the given instance in all the generated counterfactual instances.

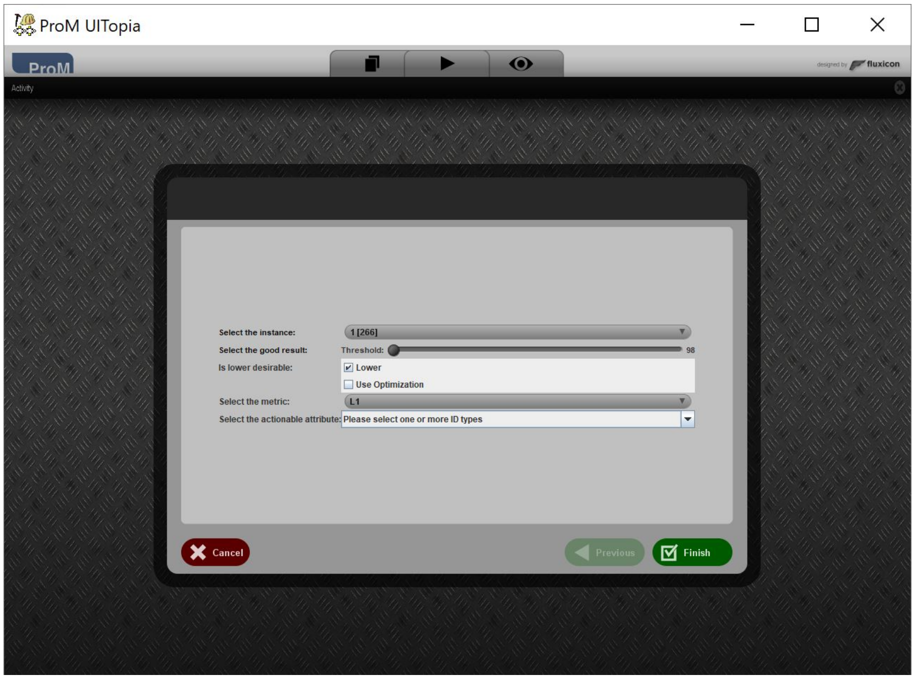
Counterfactual explanation generation setting

Do the following steps:

1. in the first drop-down list select the given instance as the one with case-id 1000 that it's implementation took 577 person days (i.e., the one denoted as 1000[577]),
2. using the slider set the threshold to 500 (or some number as close as possible to it),
3. check the "lower" checkbox to indicate that the values lower than 500 are desirable,
4. check the "use optimization" checkbox [optional],
5. in the drop-down list for method select "sem" and then upload "sem.txt",
6. in the last checklist select "feasibility-study-person-day", "feasibility-study-complexity", "feasibility-study-priority", and "feasibility-study-num-people". The final UI should look like the one in Figure 5.
7. Click on the "finish" button.

How to interpret the outcome The outcome of this plugin is a set of diverse explanations for the given instance. They are in the following general form:

Set X_1 to x_1 instead of x'_1, and set X_2 to x_2 instead of x'_2 then the value of Y would have been y.



Select the instance: 1 [266]
Select the good result: Threshold: 98
Is lower desirable: ☑ Lower
☐ Use Optimization
Select the metric: L1
Select the actionable attribute: Please select one or more ID types

Fig. 3. The UI of the plugin for the setting of the given instance and other settings needed for generating the diverse counterfactual explanations.

These explanations provide actionable information for the process owner as well as the customer on how to reach the desired outcome in the future. They exactly mention what has to be changed to receive a different outcome. Also, because of the variety of explanations, the user can select the one that fits best his/her condition. Plus, the difference between the provided explanations and the given instance is illustrated using plots. The following metrics have been considered:

- Difference between the target feature in the given instance and explanation.
- The number of features with different values in the given instance and each explanation.
- The sum of differences between all features in the given instance and each explanation. Please note that in this case the feature values have not been normalized.

In the end, all the provided explanations are summarized in a table. In this table, each column is corresponding to a feature and each row is corresponding to a sample. Those features whose values are bigger than the one in the given instance are in red, and those with a smaller value are in green.

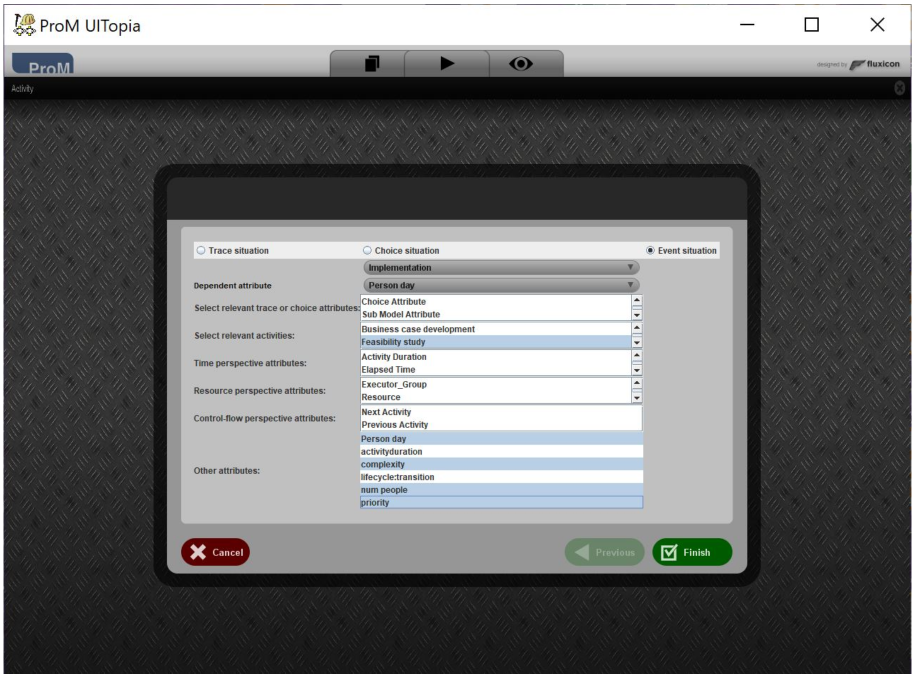
Generated explanations

You can find the generated explanations in Figures 6, 7, 8, 9, and 10.

References

1. Qafari, Mahnaz Sadat, and Wil MP van der Aalst. "Case level counterfactual reasoning in process mining." In International Conference on Advanced Information Systems Engineering, pp. 55-63. Springer, Cham, 2021.

Appendix



○ Trace situation ○ Choice situation ● Event situation

Implementation

Dependent attribute: Person day

Select relevant trace or choice attributes: Choice Attribute, Sub Model Attribute

Select relevant activities: Business case development, Feasibility study

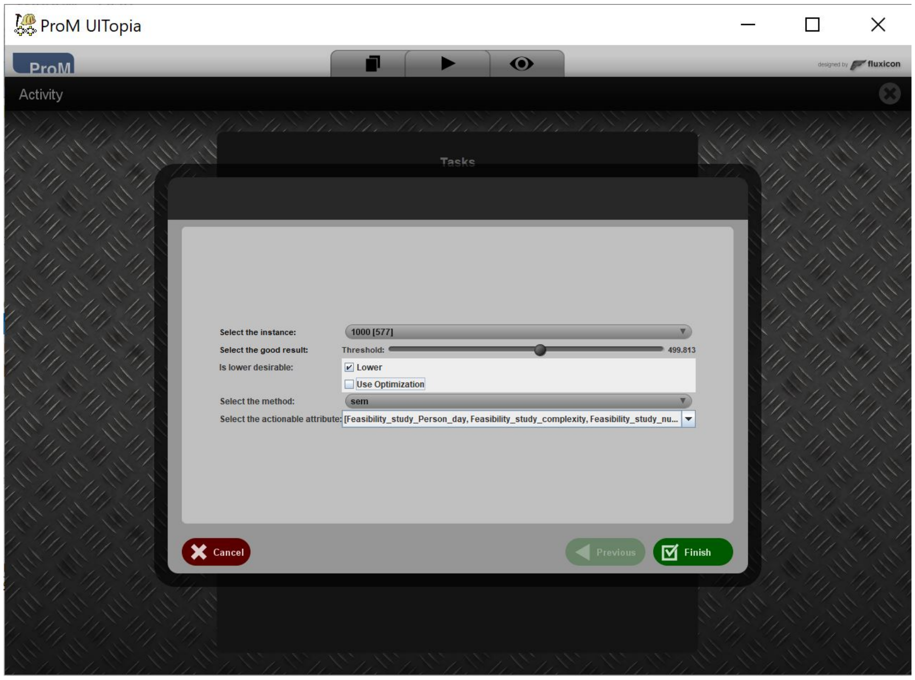
Time perspective attributes: Activity Duration, Elapsed Time

Resource perspective attributes: Executor_Group, Resource

Control-flow perspective attributes: Next Activity, Previous Activity

Other attributes: Person day, activityduration, complexity, lifecycle:transition, num people, priority

Fig. 4. The UI of the plugin after setting the target and descriptive features for extracting the data table from the event log.



Select the instance: 1000 [577]

Select the good result: Threshold: 499.813

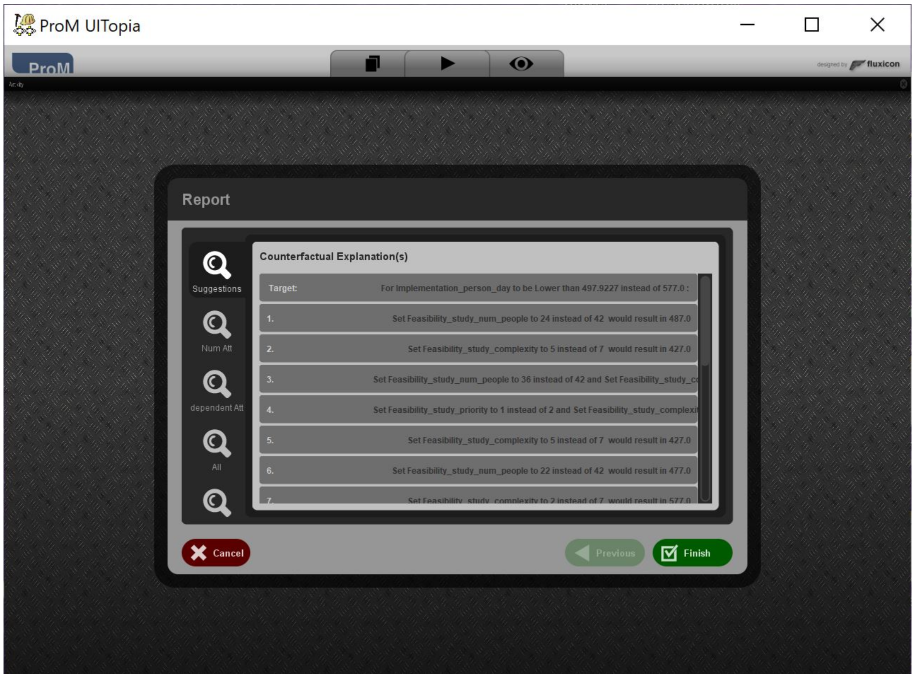
Is lower desirable: ☑ Lower

☐ Use Optimization

Select the method: sem

Select the actionable attribute: [Feasibility_study_Person_day, Feasibility_study_complexity, Feasibility_study_nu...

Fig. 5. The UI of the plugin after the setting for counterfactual explanation generation.



Counterfactual Explanation(s)

Target: For Implementation_person_day to be Lower than 497.9227 instead of 577.0 :

1. Set Feasibility_study_num_people to 24 instead of 42 would result in 487.0

2. Set Feasibility_study_complexity to 5 instead of 7 would result in 427.0

3. Set Feasibility_study_num_people to 36 instead of 42 and Set Feasibility_study_c

4. Set Feasibility_study_priority to 1 instead of 2 and Set Feasibility_study_complexit

5. Set Feasibility_study_complexity to 5 instead of 7 would result in 427.0

6. Set Feasibility_study_num_people to 22 instead of 42 would result in 477.0

7. Set Feasibility_study_complexity to 2 instead of 7 would result in 577.0

Fig. 6. Provided explanations in text format.

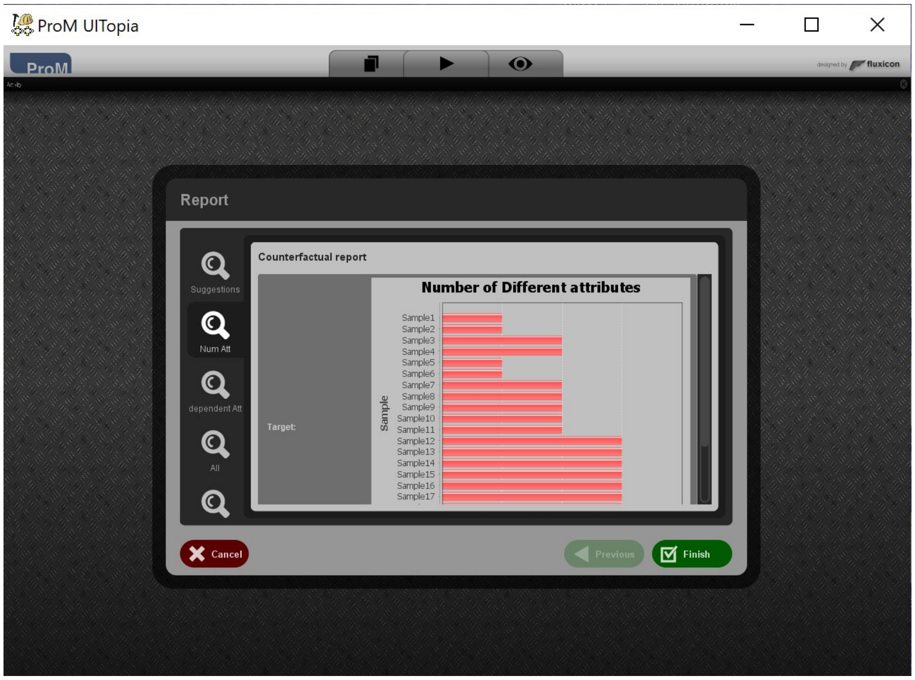


Fig. 7. The number of descriptive features with different values in the given instance (instance 1000[577]) and each provided explanation.

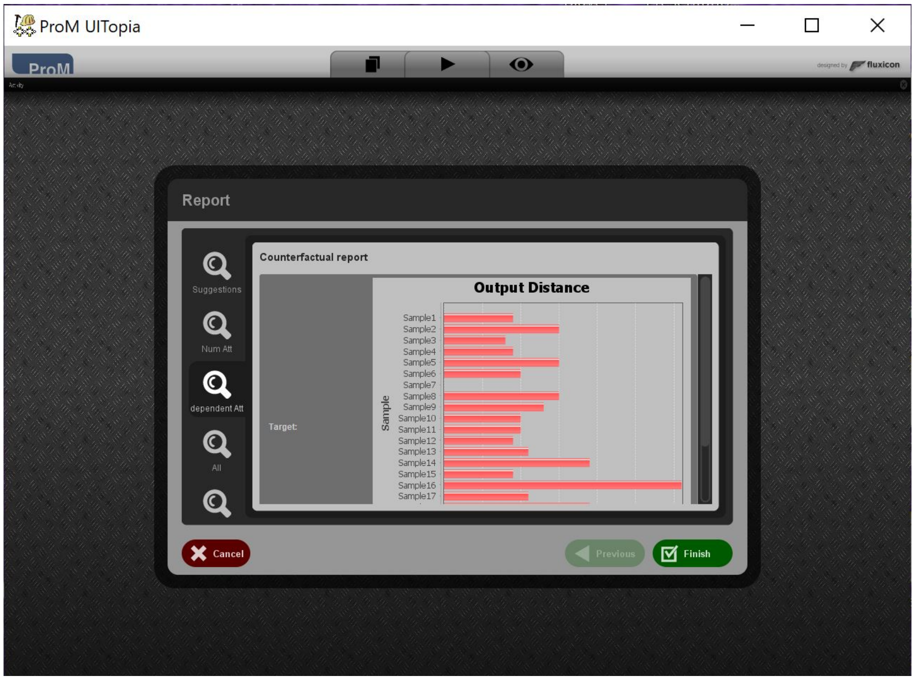


Fig. 8. The L_1 distance between the target feature in the given instance (instance 1000[577]) and each provided explanation.

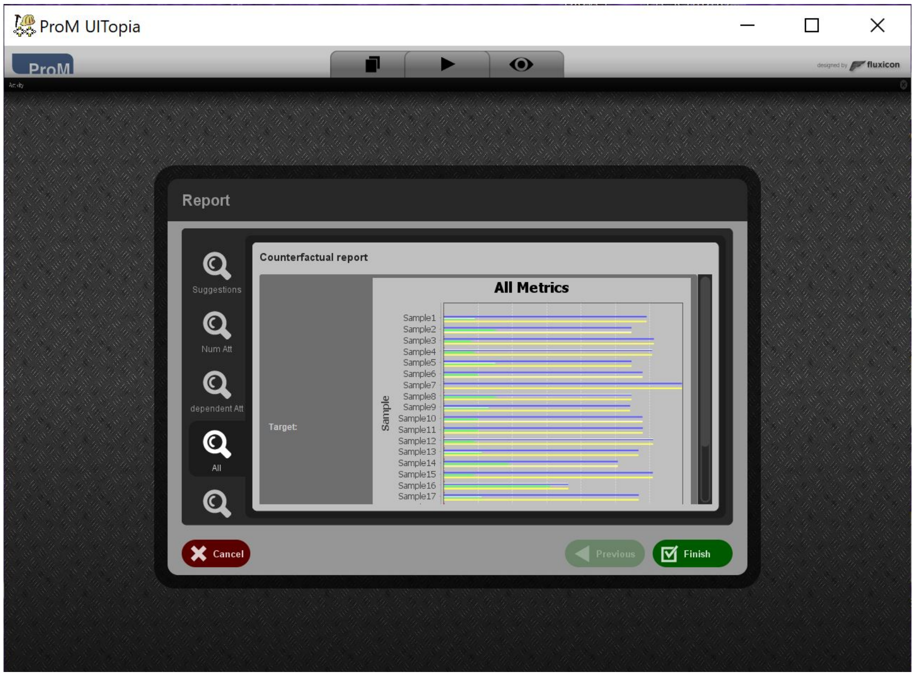
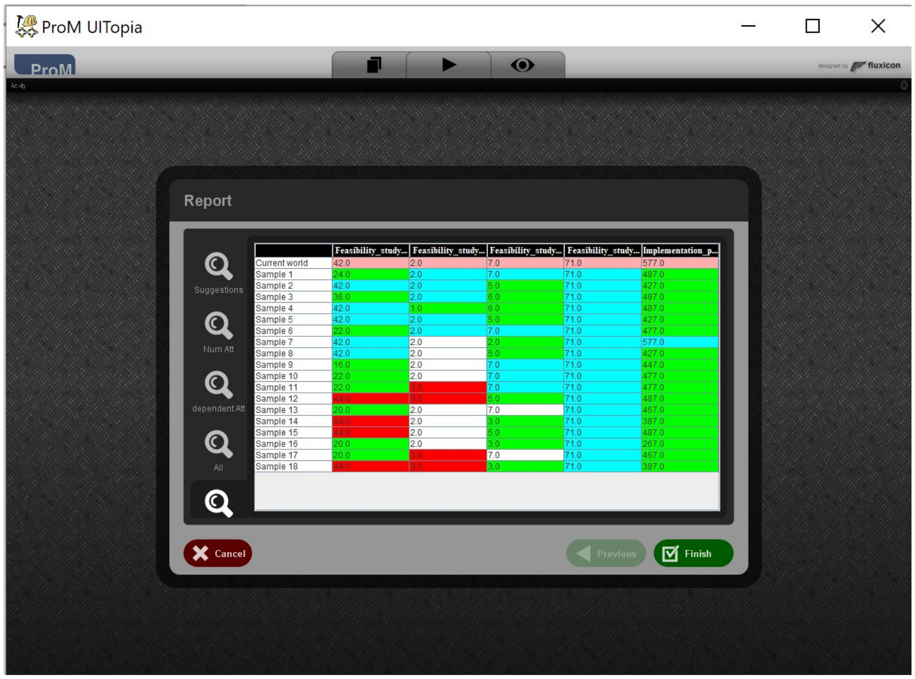


Fig. 9. Comparing the given instance (instance 1000[577]) and each provided explanation using different metrics including L_1 and L_2 distance of the target features,L_1 distance between all features, and number of different descriptive features.



	Feasibility_study...	Feasibility_study...	Feasibility_study...	Feasibility_study...	Implementation_p...
Current world	42.0	2.0	7.0	71.0	577.0
Sample 1	24.0	2.0	7.0	71.0	487.0
Sample 2	42.0	2.0	5.0	71.0	427.0
Sample 3	36.0	2.0	6.0	71.0	497.0
Sample 4	42.0	1.0	6.0	71.0	487.0
Sample 5	42.0	2.0	5.0	71.0	427.0
Sample 6	22.0	2.0	7.0	71.0	477.0
Sample 7	42.0	2.0	2.0	71.0	577.0
Sample 8	42.0	2.0	5.0	71.0	427.0
Sample 9	16.0	2.0	7.0	71.0	447.0
Sample 10	22.0	2.0	7.0	71.0	477.0
Sample 11	22.0	3.0	7.0	71.0	477.0
Sample 12	44.0	3.0	5.0	71.0	487.0
Sample 13	20.0	2.0	7.0	71.0	467.0
Sample 14	44.0	2.0	3.0	71.0	387.0
Sample 15	44.0	2.0	5.0	71.0	487.0
Sample 16	20.0	2.0	3.0	71.0	267.0
Sample 17	20.0	3.0	7.0	71.0	467.0
Sample 18	44.0	3.0	3.0	71.0	387.0

Fig. 10. A summary of all the provided explanations in a table format. In this table, each column is a feature and each row is a sample. Those features with a bigger value than the one in the given instance are in red, and those with a smaller value are in green.