As mentioned in **??**, counterfactual generation is notorious for their lack of a standardised evaluation procedure. Nonetheless, we attempt to address our research questions with the following experiments.

Experiment 1: Model Selection

Before comparing models, it is important to reduce the number of possible models that *can* be compared. Especially, the evolutionary generator has a number of free parameters. These range from structural configurations to general hyperparameters. In terms of operators, we introduced 4 initiators, 3 selectors, 3 crossers, 2 mutators and 3 recombiners. Hence, comparing all possible eolutionary operator combinations requires to test a total of 216 different models. Furthermore, each model has hyperparameters, we have to define, too. Therefore, the first set of experiments are dedicated to choose among a subset of operator combinations and subsequently select appropriate hyperparameters.

First, we compute all possible configurations, without changing any hyperparameter. To avoid confusion, we refer to each unique operator combination as a model-configuration. For instance, one model-configuration would consist of **[a SamplingBasedInitiator, an ElitismSelector, a OnePointCrosser, SamplingBasedMutator and a FittestSurvivorRecombiner]**. We refer to a specific model-configuration in terms of its abbreviated operators. For instance, the earlier example is denoted as **[SBI-ES-OPC-SBM-FSR]**.

Afterwards, we explore the hyperparameters of the model. We start with the termination point. Hence, we want to explore the effects of the iterative cycles that each evolutionary alogirthm will run for. The goal is to find a stopping criterion which yields reasonably good counterfactuals, while reducing the computation time. We will only consider the number of iterative cycles as a stopping criterion. We refer to each different criterion as termination point. We can choose the termination point by inspecting how the average population viablity evolves across each cycle. We keep every other experimental setting as established beforehand.

We determine an appropriate number of individuals we generate in every iterative cycle and a population size. We test both together, as they are dependent on each other. We keep every other experimental setting as before and only experiment on the model-configurations selected prior. Our goal is to find the optimal ratio between children generated and population size.

For determining the mutation rate for every mutation type, we choose the best evolutionary algorithm and run the configuration with 6 rates from 0 to

0.5 in steps of 0.1. We omit everything beyond 0.5 to preserve information about the parent. For instsnce, if we use a change rate of 0.9, we mutate 90% of the genes the child inherited. This would defeat the purpose of evolving better counterfactuals through breeding. We use the termination point established in the prior experiment. We keep every other experimental setting as established beforehand.

After, executing all preliminary experiments we choose the evolutionary generators and compare them with all baseline models in all subsequent experiments.

Experiment 2: Model Comparison

First, we assess the viability of **[a number of]** models. For this purpose, we sample **[10]** factuals and use the models to generate **[50]** counterfactuals. We determine the **[mean]** viability across the counterfactuals. With this experiment, we show that a model which optimizes quality criteria of counterfactuals produces better results than models, which do not. Hence, we expect the evolutionary algorithm to perform best, as it can directly optimize multiple viability criterions. In the following we list all models, we are going to compare:

RNG A Random-Search Generator , which generates random values and acts as a baseline.

CBG A Casebased-Search Generator , which samples from process instances within the training set

EVO A SBI-ES-OPC-SBM-FSR Generator , which optimizes viability using principles of evolution.

In accordance with *RQ1-H1* and *RQ1-H2* we expect the SBI-ES-OPC-SBM-FSR Generator to perform best among these baselines, when it comes to viability.

Experiment 3: Comparing with alternative Literature

The model comparison is not enough to establish the validity of our solution, as defined proposed the viability measure ourselves. Therefore, we also assess each model based on the evaluation criterions of an alternative work. More precisely, we quantify the viability of our models using the metrics employed by Hsieh et al. Hence, we measure the sparsity by computing the average Levenshtein difference and proximity using the L2-Norm. Furthermore, we compute the average intra-list-diversity and plausibility as well as the models capability of changing the prediction to a desired one.

Similar to Hsieh et al., we only focus on the *activities* that are generated by each model and its accompaniying *resource* event-attribute. For diversity and plausibility we remain close to the original evaluation protocol by Hsieh et al. as we will also treat each counterfactual trace sequence as a symbol. Hence, a sequeunce ABC is treated as a completely different symbol than $ABCD$.

The goal is to show that models, which optimise viability criterions, perform better, even if viability is assessed differently as stated in *RQ2-H1* of our research question (**??**).

Experiment 4: Qualitative Assessment

For the last assessment, we follow Hsieh et al.'s procedure of assessing the models qualitatively. We use the dataset as the authors do. **[FOR XIXI: Should I use the exact same examples?]** However, as we focus on outcome prediction, we attempt to answer one of two questions:

1. *what would I have had to change to prevent the cancellation/rejection of the loan application process*

2. *what would I have had to change to get cancelled/rejected of the loan application process*

The goal is to show, that the results are viable despite not having a standardized protocol to measure their viability.