Methodology

The tests are divided into three groups:

- Quality assurance: used to ensure that implementation works.
- The performance: used to measure time and memory complexity.
- The quality: used to evaluate the results.

Quality assurance tests are run on the part of examples while other tests are run on all examples.

Examples

First, we should select the ontologies that will be used for evaluation. After selecting ontologies for testing, we should select certain examples to measure the progress of the model. Following cases should be present in the dataset:

- one or more assertions are inconsistent: `VegeterianPizza and (hasBase some ThinAndCrispyBase) and (hasTopping some MeatTopping)`
- when all existing assertions are inconsistent: `Parmense and hasTopping some NutTopping`
- the initial individual contains "individual - P - object" and the target class "individual - not P - object": `VegeterianPizza and hasTopping some MeatTopping` while `VegeterianPizza` has explicit: `not (hasTopping some MeatTopping)`
- the initial individual contains "individual - not P - object" and the target class has "individual - P - object": `MeatyPizza and not (hasTopping some MeatTopping)` while `MeatyPizza` contains `(hasTopping some MeatTopping)`.

For each type of cases, there should be a number of examples and they will be split into "training" and "test": the former will be used in development of algorithms, the latter in final evaluation.

Metrics

Performance metrics

Allow us to evaluate how usable the algorithm and its implemention are.

Quality metrics

- Proximity: the distance between the initial individual and generated counterfactuals. We use dissimilarity for calculating proximity.
- Sparcity: amount of assertions in the individual should be changed in order to get the desired result. Sparcity may be part of proximity but setting it separately helps better

evaluate the performance of the algorithm.

- Diversity: distance between generated counterfactuals. $det(A), A_{i,j} = \frac{1}{1+dist(x_i,x_j)}$ [1].
- Number of explored counterfactuals, of valid counterfactuals, of axioms in the ontology. (helps to project the ontology size).

The used distance is dissimilarity between the individuals.
In our case, sparcity is already embedded into proximity but I think it's important to display it separately too.

Quality assurance

Comparing generated counterfactuals with a manually prepared list of expected counterfactuals. For example, to change $MeatyPizza$ with $hasTopping\ some\ MeatTopping$ into $VegetarianPizza$, expected explanations would be $hasTopping\ some\ VegetarianTopping$ but not removing all assertions.

References

1.
```
author = {Mothilal, Ramaravind K. and Sharma, Amit and Tan, Chenhao},
title = {Explaining Machine Learning Classifiers through Diverse
Counterfactual Explanations},
year = {2020},
isbn = {9781450369367},
publisher = {Association for Computing Machinery},
address = {New York, NY, USA},
url = {https://doi.org/10.1145/3351095.3372850},
doi = {10.1145/3351095.3372850},
booktitle = {Proceedings of the 2020 Conference on Fairness, Accountability,
and Transparency},
pages = {607–617},
numpages = {11},
location = {Barcelona, Spain},
series = {FAT* '20}
}
```