API Rest Functions and Parameter Specification**:**

-Function: **DicePublic**

# **params:** string representing a JSON object containing the following fields:

* “**instances**”: array of arrays, where each one represents a row with the feature values of an instance including the target class. However, when using TensorFlow or PyTorch as backend, only one instance is supported, so an array containing a single array representing the values must be passed.
* "**backend**": a string containing the backend of the prediction model. The supported values are: “sklearn” (Scikit-learn), “TF1” (TensorFlow 1.0), “TF2” (TensorFlow 2.0), “PYT” (PyTorch).
* “**method**”: The method used for counterfactual generation. The supported methods are: “random” (random sampling), “genetic” (genetic algorithms), “kdtrees”.
* "**cont\_features**": array of strings containing the name of the continuous features. Features not included here are considered categorical.
* "**desired\_class**": integer representing the index of the desired counterfactual class, or “opposite” in the case of binary classification.
* **“features\_to\_vary”**: Either a string “all” or a list of strings representing the feature names to vary.
* "**num\_cfs**":number of counterfactuals to be generated for each instance.
* “**permitted\_range”** (optional): JSON object with feature names as keys and permitted range in array as values.
* “**continuous\_features\_precision”** (optional): JSON object with feature names as keys and precisions as values.
* **data\_name** (optional): name of the dataset.

Example of params.json:

{

"backend": "sklearn",

"cont\_features": [“Height”, “Weight”],

"continuous\_features\_precision": {“Height”: 1, “Weight”:3},

"data\_name": "datasetName",

"desired\_class": 0,

“instances”: [ [X1, X2, …, Xn], [Y1, Y2, …, Yn]],

"method": "random",

"num\_cfs": 3,

"permitted\_range": {"Height": [ 0, 250]}

}

}

# **model**: the trained prediction model given as a file. The extension must match the backend being used i.e. a **.pkl** file for Scikit-learn (check joblib for reference), **.pt** for PyTorch or **.h5** for TensorFlow models.

# **data:** pandas DataFrame containing the training data given as a .pkl file. The target class must be the last column of the DataFrame.

**Command Example:**

curl -F "params=<params.json" -F "model=@model.pkl" -F "data=@traindata.pkl" -H "Authorization: Bearer %TOKEN%" <http://localhost:5444/DicePublic>

**Return:** JSON Object containing the generated counterfactuals and metadata of the explanation.

-Function: **DicePrivate**

# **params:** string representing a JSON object containing the following fields:

* “**instance**”: JSON object representing the instance of interest with attribute names as keys, and feature values as values.
* "**backend**": a string containing the backend of the prediction model. Currently, the only supported backend for private data is “TF2” (TensorFlow 2.0).
* “**method**”: The method used for counterfactual generation. The supported methods for private data are: “random” (random sampling) and “genetic” (genetic algorithms).
* "**features”**: JSON Object with feature names as keys and arrays containing the ranges of continuous features, or strings with the categories for categorical features.
* **“features\_to\_vary”**: Either a string “all” or a list of strings representing the feature names to vary.
* "**desired\_class**": integer representing the index of the desired counterfactual class, or “opposite” in the case of binary classification.
* "**num\_cfs**":number of counterfactuals to be generated for each instance.
* “**outcome\_name**”**:** name of the target column.
* “**type\_and \_precision**” (optional): JSON object with continuous feature names as keys. If the feature is of type int, the value should be the string “int”. If the feature is of type float, an array of two values is expected, containing the string “float”, and the precision.
* “**mad**” (optional): JSON with feature names as keys and corresponding Median Absolute Deviation.
* **“data\_name”** (optional): name of the dataset.

Example of params.json:

{

"backend": "TF2",

"data\_name": "datasetName",

"desired\_class": 0,

"features”: {“Gender”:[“male”, ”female”], "Height": [ 0, 250], “Weight”:[ 0, 250]}

“instance”: {“Gender”:”Female”, “Height”:100, “Weight”:100},

"method": "random",

"num\_cfs": 3,

"outcome\_name": “Target”,

“type\_and\_precision”: {“Height”: [“float”,1], “Weight”: ”int”}

}

# **model**: the trained prediction model given as a .pkl file. The model must have a TensorFlow 2.0 backend.

**Command Example:**

curl -F "params=<params.json" -F "model=@model.pkl" -H "Authorization: Bearer %TOKEN%" <http://localhost:5444/DicePrivate>