Documentation Choquet classifier

Daniel Ebensberger

November 7, 2022

Contents

The Choquet classifier is an algorithm for binary classification problems and was designed in the context of supervised learning. Due to its definition, which is based on Choquistic regression, it is particularly suitable for attributes with numerically scaled ranges of values, which assume monotonicity in the sense of "larger values are better values". The implementation is compatible with the library `scikit-learn` and can therefore be used in various algorithms such as grid search, which is implemented in this library.

1 Installation

The Choquet classifier is implemented in Python 3.9 and has been tested. Installation is done via following command:

```
    pip install choquet-classifier-glenscalai
```

2 Interface

To use the Choquet classifier in projects or pipelines, the class `ChoquetClassifier` from module `choquet_classifier` has to be imported. The interface design is following the `scikit-learn` API and presents three functions:

ChoquetClassifier(additivity=2, regularization=None)

The Choquet classifier is instantiated via a constructor with two hyperparameters: order k of additivity and regularization. The default value of k is set to 2 since this provides a sufficient tradeoff between flexibility and complexity as experiments have shown.Regarding the regularization parameter, it reflects the strength of L1-regularization during the fitting process and will be set to 0.001 if not set otherwise.

fit(X, y)

The function `fit` computes the feature transformation and the capacity function for a set of sample data `X` and the associated class labels `y`. `X` is an array-like data structure with the dimensions (`n_samples`, `n_features`), where `n_samples` is the number of sample data and `n_features` is the number of attributes. y is an array-like data structure with the dimensions (`n_samples`,), where `n_samples` must be equal to the number of sample data of `X`. The class labels y can be integer numeric values. The classes can be described by integer numeric values as well as textually. The respective smaller class with respect to the relation < or the lexicographic order is internally assigned to the negative class and the larger class to the positive class.

predict(X)

The predict function determines the classes for a set of test data X. X is an array-like data structure with the dimensions (n_samples, n_features), where n_samples is the number of sample data and n_features is the number of attributes. The value of n_features must be the same as the value used to calculate the parameters of the Choquet classifier. In addition, the fit function must have been called before predict is applied.

3 Examples

3.1 Default example

Use the constructor and the function fit to initialize the Choquet classifier for a given data set.

```
>>> from choquet_classifier import ChoquetClassifier
>>> X = [[1, 3, 2],
...      [1, 0, 3]]
>>> y = [1, 0]
>>> cc = ChoquetClassifier()
>>> cc.fit(X, y)
```

Use the function predict to classify samples.

```
>>> Z = [[1, 1, 2],
...      [2, 1, 3]]
>>> cc.predict(Z)
array([0, 0])
```

3.2 Example with hyper-parameters

```
>>> from choquet_classifier import ChoquetClassifier
>>> X = [[1, 3, 2],
...      [1, 0, 3]]
>>> y = [1, 0]
>>> cc = ChoquetClassifier(additivity=3, regularization=1)
>>> cc.fit(X, y)
```

Again, the function `predict` can be used to classify samples. Note the different output compared to the first example.

```
>>> Z = [[1, 1, 2],
...      [2, 1, 3]]
>>> sc.predict(Z)
array([0, 1])
```

3.3 Example with different class labels

The classes do not have to be labeled with 0 and 1; any integer numbers or strings may be used instead. The smaller label in terms of the relation or lexicographic ordering is given to the negative class, whereas the other label is assigned to the positive class.

The first example contains the class labels 2 and 1. Label 2 is assigned to the positive class and label 1 is assigned to the negative class since $2 > 1$.

```
>>> from choquet_classifier import ChoquetClassifier
>>> X = [[1, 3, 2],
...      [1, 0, 3]]
>>> y = [2, 1]
>>> cc = ChoquetClassifier()
>>> cc.fit(X, y)
>>> Z = [[1, 1, 2],
...      [2, 1, 3]]
>>> sc.predict(Z)
array([1, 1])
```

The second example contains the class labels ’one’ and ’two’. Label ’one’ is assigned to the negative class and label ’two’ is assigned to the positive class because ’one’ comes lexicographically first.

```
>>> from choquet_classifier import ChoquetClassifier
>>> X = [[1, 3, 2],
...      [1, 0, 3]]
>>> y = [’two’, ’one’]
>>> cc = ChoquetClassifier()
>>> cc.fit(X, y)
>>> Z = [[1, 1, 2],
...      [2, 1, 3]]
>>> sc.predict(Z)
array([’one’, ’one’])
```

List of Figures