mutalyzer-mutator

Release 0.2.0

Mihai Lefter

Jun 01, 2021

CONTENTS:

Mutate a sequence according to a list of variant models.

CHAPTER

ONE

INSTALLATION

The software is distributed via PyPI, it can be installed with `pip`:

```
pip install mutalyzer-mutator
```

1.1 From source

The source is hosted on GitHub, to install the latest development version, use the following commands.

```
git clone https://github.com/mutalyzer/mutator.git
cd mutator
pip install .
```

CHAPTER

TWO

USAGE

The `mutate()` function provides an interface to mutate a sequence according to a list of variants. A dictionary with the reference ids as keys and their sequences as values should be provided as input. The reference with the `reference` key is the one to be mutate according to the variants model list, the second input of the `mutate()` function.

```python
from mutalyzer_mutator import mutate

sequences = {"reference": "AAGG", "OTHER_REF": "AATTAA"}

variants = [
    # 2_2delinsOTHER_REF:2_4
    {
        "type": "deletion_insertion",
        "source": "reference",
        "location": {
            "type": "range",
            "start": {"type": "point", "position": 2},
            "end": {"type": "point", "position": 2},
        },
        "inserted": [
            {"sequence": "CC", "source": "description"},
            {
                "source": {"id": "OTHER_REF"},
                "location": {
                    "type": "range",
                    "start": {"type": "point", "position": 2},
                    "end": {"type": "point", "position": 4},
                },
            },
        ],
    }
]

observed = mutate(sequences, variants)  # observed = 'AACCTTGG'
```

CHAPTER

THREE

API DOCUMENTATION

3.1 Mutator

Module to mutate sequences based on a variants list.

Assumptions for which no check is performed:

- Only `deletion insertion` operations.
- Only exact locations, i.e., no uncertainties such as *10+?*.
- Locations are zero-based right-open with `start > end`.
- There is no overlapping between variants locations.

Notes:

- If any of the above is not met, the result will be bogus.
- There can be empty inserted lists.

`mutalyzer_mutator.mutator.`**`mutate`**`(`*sequences*, *variants*`)`

Mutate the reference sequence under `sequences["reference"]` according to the provided variants operations.

Parameters

- **`sequences`** (*`dict`*) – Sequences dictionary.
- **`variants`** (*`list`*) – Operations list.

Returns Mutated sequence.

Return type str

3.2 Util

Various util functions.

The following code is adapted from biopython 1.77:

- https://github.com/biopython/biopython/blob/biopython-177/Bio/Seq.py
- https://github.com/biopython/biopython/blob/biopython-177/Bio/Data/IUPACData.py

Notes:

- The alphabet check was removed.
- No previous custom errors are raised any longer.

`mutalyzer_mutator.util.`**`complement`**(*sequence*)

Complement the `sequence`.

```
>>> sequence = 'CCCCCGATAG'
>>> complement(sequence)
'GGGGGCTATC'
```

You can use mix DNA with RNA sequences.

```
>>> sequence = 'CCCCCaTuAGD'
>>> complement(sequence)
'GGGGGuAaTCH'
```

Also, you can use mixed case sequences.

```
>>> sequence = 'CCCCCgatA-GD'
>>> complement(sequence)
'GGGGGcuaT-CH'
```

Note that in the above example, the ambiguous character `D` denotes `G`, `A` or `T` so its complement is `H` (for `C`, `T` or `A`).

Parameters **sequence** (*str*) – Input sequence.

Returns Complemented sequence.

Return type str

`mutalyzer_mutator.util.`**`reverse_complement`**(*sequence*)

Reverse complement the `sequence`.

```
>>> sequence = 'CCCCCGATAGNR'
>>> reverse_complement(sequence)
'YNCTATCGGGGG'
```

Note that in the above example, since `R` = `G` or `A`, its complement is `Y` (which denotes `C` or `T`).

You can use mix DNA with RNA sequences.

```
>>> sequence = 'CCCCCaTuAGD'
>>> reverse_complement(sequence)
'HCTaAuGGGGG'
```

You can of course used mixed case sequences,

```
>>> sequence = 'CCCCCgatA-G'
>>> reverse_complement(sequence)
'C-TaucGGGGG'
```

Parameters **sequence** (*str*) – Input sequence.

Returns Reverse complemented sequence.

Return type str

PYTHON MODULE INDEX

INDEX