

Guide benchmarking

This guide is intended for those who plan to use the benchmarking of the pysatl-cpd project to analyze algorithms for finding change points.

Installation

Clone repository:

```
git clone https://github.com/PySATL/pysatl-cpd.git
```

Linux

Go to repository folder and run installation script:

```
cd pysatl-cpd
chmod +x scripts/install_user_linux.sh
./install_user_linux.sh
```

Windows

Go to repository folder and run installation script

```
Set-Location pysatl-cpd
./scripts/install_user_windows.ps1
```

Data Generation

Available distributions

Распределение	Название	Параметры
Нормальное	normal	mean, variance
Экспоненциальное	exponential	rate
Вейбулла	weibull	shape, scale
Равномерное	uniform	min, max
Бета	beta	alpha, beta
Гамма	gamma	alpha, beta
t-Стюдента	t	n

Распределение	Название	Параметры
Логнормальное	lognorm	s
Многомерное нормальное	multivariate_normal	mean, в виде списка-строки, например "[0.5, 2.0]"

How to configure?

To generate a test time series, create a new configuration file inside the `pysatl_cpd/examples/configs` directory. This file defines the segments that will be concatenated in order to create the final time series.

Structure of the config file:

```
- name: config_name
  distributions:
    - type: dist1
      length: length1
      parameters:
        parameter1_1: value1_1
        parameter1_2: value1_2
    - type: dist2
      length: length2
      parameters:
        parameter2_1: value2_1
        parameter2_2: value2_2
    # ... you can add more distribution segments here
```

Fields:

- `name`: A unique name for your configuration.
- `distributions`: A list of data segments to be generated. Each item in the list is a segment.
- `type`: The distribution type for the segment (e.g., normal, uniform).
- `length`: The length (number of data points) for this segment.
- `parameters`: The parameters required by the chosen distribution type (e.g., mean and variance for a normal distribution).

Note: The available distribution types and their parameters must match the options listed in the table above. Please refer to it for a complete list of supported distributions and their required parameters.

Config example

```
- name: example
  distributions:
    - type: exponential
      length: 200
      parameters:
```

```

    rate: 2.0
- type: beta
  length: 200
  parameters:
    alpha: 1.0
    beta: 5.0
- type: uniform
  length: 200
  parameters:
    min: 0
    max: 0.5

```

Algorithm configure

Experiment run

Run example in the main directory:

```
poetry run python example.py
```

Troubleshooting

Import error: cannot import matplotlib

If you saw a similar error when running the script:

```

Package operations: 13 installs, 0 updates, 0 removals

- Installing pyqt5-qt5 (5.15.17): Failed

| Unable to find installation candidates for pyqt5-qt5 (5.15.17)
| This is likely not a Poetry issue.
|
| - 3 candidate(s) were identified for the package
| - 3 wheel(s) were skipped as your project's environment does not support
|
| Solutions:
| Make sure the lockfile is up-to-date. You can try one of the following;
|
| 1. Regenerate lockfile: poetry lock --no-cache --regenerate
| 2. Update package      : poetry update --no-cache pyqt5-qt5

```

And then you get this error:

```

PS C:\Users\79787\Downloads\Test\pysatl-cpd> poetry run python example.py
Traceback (most recent call last):
  File "C:\Users\79787\Downloads\Test\pysatl-cpd\example.py", line 5, in <module>
    from benchmarking.steps.data_generation_step.data_handlers.generators.cpd_generator import CpdGenerator
  File "C:\Users\79787\Downloads\Test\pysatl-cpd\benchmarking\steps\data_generation_step\data_handlers\generator.py", line 17, in <module>
    from pysatl_cpd.generator.generator import ScipyDatasetGenerator
  File "C:\Users\79787\Downloads\Test\pysatl-cpd\pysatl_cpd\generator\generator.py", line 10, in <module>
    from .saver import DatasetSaver
  File "C:\Users\79787\Downloads\Test\pysatl-cpd\pysatl_cpd\generator\saver.py", line 4, in <module>
    import matplotlib.pyplot as plt
ModuleNotFoundError: No module named 'matplotlib'

```

Then try installing a lower version of the package:

```
poetry add pyqt5-qt5==5.15.2
```