Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Software update

Update 2.0 to "ViroCon: A software to compute multivariate extremes using the environmental contour method"

Andreas F. Haselsteiner *, Kai-Lukas Windmeier, Lena Ströer, Klaus-Dieter Thoben

A R T I C L E   I N F O

Code metadata

Current Code version	*2.0.13*
Permanent link to code/ repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-21-00160
Legal Code License	*MIT*
Code Versioning system used	*git*
Software Code Language used	*Python*
Compilation requirements, Operating environments & dependencies	-
If available Link to developer documentation/ manual	https://virocon.readthedocs.io
Support email for questions	a.haselsteiner@uni-bremen.de

Software metadata

Current software version	*virocon does not have an executable*
Permanent link to executables of this version	–
Legal Software License	–
Computing platform/ Operating System	–
Installation requirements & dependencies	–
If available Link to user manual — if formally published include a reference to the publication in the reference list	–
Support email for questions	–

Free text.

This update to virocon introduces a heavily changed software architecture and major new functionality. The original software "ViroCon" was released in 2018 and comprised two Python software packages: one package titled "viroconcom" that contained the statistical methods and one package titled "viroconweb", which provided a web-based graphical user interface. We stopped maintaining "viroconweb" and renamed "viroconcom" to "virocon" (Fig. 1). In version 2.0 of virocon we changed the workflow for defining a joint model structure and for estimating its parameters based on a dataset ("fitting"; Listing 1). This rewrite allows users to freely define parametric dependence function. Additionally, parameter estimation techniques can be selected more systematically: By default, marginal distribution parameters are estimated using maximum likelihood estimation and dependence function parameters are estimated using least squares, however, other fitting methods can be selected by passing an optional argument to the "fit" function. Lastly, version 2.0 provides some commonly used, predefined joint model structures: The wind speed–wave height and the wave height–wave period models recommended in DNV's guideline on environmental conditions [1] as well as the wind speed–wave height and wave height–wave period models proposed by Haselsteiner et al. (2020) [2]. Code from viroconcom 1.X is not compatible with virocon 2.0.

We also updated virocon's documentation, which is now available at https://virocon.readthedocs.io . The documentation includes a comprehensive user guide with many examples of how to estimate joint distributions and how to compute contours. These examples are also available as Python scripts in the GitHub
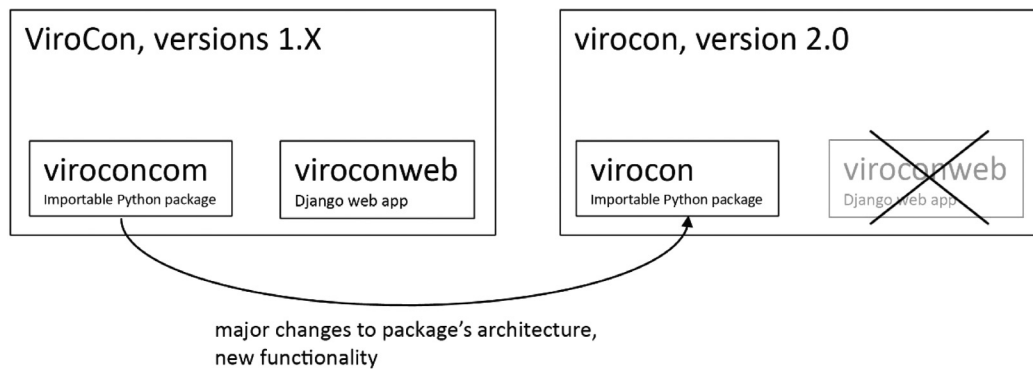
Fig. 1. Sketch showing how virocon changed from version 1.X to version 2.0. The web application "viroconweb" is not maintained anymore such that we now call the importable Python package "virocon" instead of "viroconcom".

```python
"""
Brief example that computes a sea state contour.
"""
import matplotlib.pyplot as plt
from virocon import (read_ec_benchmark_dataset, get_OMAE2020_Hs_Tz,
    GlobalHierarchicalModel, IFORMContour, plot_2D_contour)

# Load sea state measurements.
data = read_ec_benchmark_dataset("datasets/ec-benchmark_dataset_A.txt")

# Define the structure of the joint distribution model.
dist_descriptions, fit_descriptions, semantics = get_OMAE2020_Hs_Tz()
model = GlobalHierarchicalModel(dist_descriptions)

# Estimate the model's parameter values (fitting).
model.fit(data, fit_descriptions=fit_descriptions)

# Compute an IFORM contour with a return period of 50 years.
tr = 50 # Return period in years.
ts = 1 # Sea state duration in hours.
alpha = 1 / (tr * 365.25 * 24 / ts)
contour = IFORMContour(model, alpha)

# Plot the contour.
plot_2D_contour(contour, data, semantics=semantics, swap_axis=True)
plt.show()
```

Listing 1: Example of how an IFORM environmental contour can be computed in virocon version 2.0. In this example the model structure that was proposed at the OMAE 2020 conference [2] is used by calling the function get_OMAE2020_Hs_Tz().

repository's folder "examples". A full API documentation is also available.

Listing 1 presents a brief example that highlights the new syntax of virocon 2.0 as well as the use of predefined joint models. Running this Python script will fit a joint distribution to a wind speed–wave height dataset, compute an IFORM environmental contour with a return period of 50 years and finally plot this contour. The resulting plot is shown in Fig. 2.

In summary, virocon 2.0 comprises three major changes compared to virocon 1.X:

1. The software architecture has been rewritten from the ground up such that defining model structures and fitting parameters are two distinct steps: Models are created by instancing a GlobalHierarchicalModel object, its parameters are fitted by calling the model's fit() function.

2. Dependence functions can be freely defined now (instead of choosing from a given set of functions).

3. The new predefined.py module contains commonly used joint model structures that can be imported and readily be used.

Declaration of competing interest
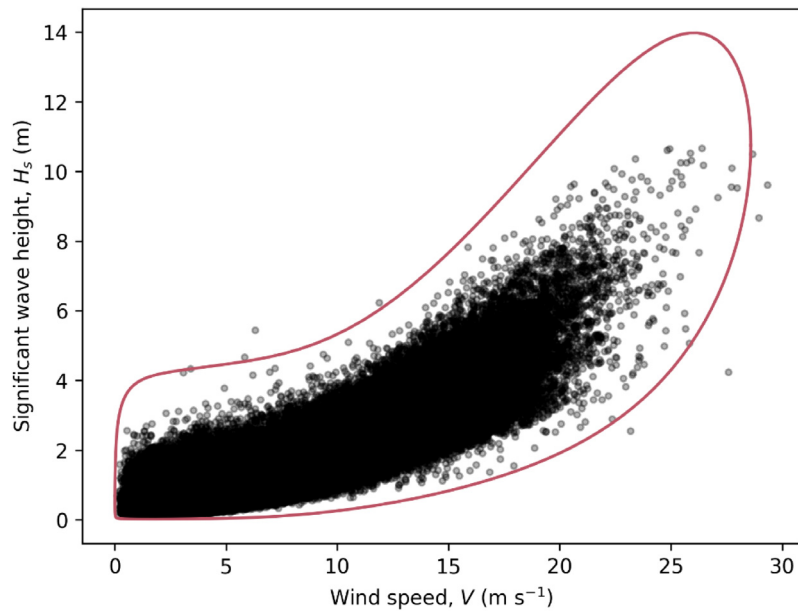
Acknowledgments

Fig. 2. Environmental contour computed with virocon version 2.0 by running the code shown in Listing 1.

References

[1] DNV GL. Recommended practice DNVGL-RP-C205: Environmental conditions and environmental loads. 2017.

[2] Haselsteiner AF, Sander A, Ohlendorf J-H, Thoben K-D. Global hierarchical models for wind and wave contours: physical interpretations of the dependence functions. In: Proc. 39th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2020). 2020, http://dx.doi.org/10.1115/OMAE2020-18668.