




1 TopSearch: a Python package for topographical 2 analysis of machine learning models and physical 3 systems

4 **Luke Dicks** ¹ and **Edward O. Pyzer-Knapp** ¹ 

5 ¹ IBM Research Europe, Hartree Centre, Daresbury, United Kingdom  Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Chris Vernon](#)  

Reviewers:

- [@ml-evs](#)
- [@lbi59](#)

Submitted: 03 April 2024

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

6 Summary

7 Machine learning (ML) is now ubiquitous in all scientific fields, but there remains a significant
8 challenge to understanding and explaining model performance ([Angelov et al., 2021](#); [Zhang et
al., 2021](#)). Therefore, there is increasing interest in applying methods from other scientific
9 disciplines (e.g. physics and biology) to improve the performance and explainability of machine
10 learning algorithms ([Hassabis et al., 2017](#); [Karniadakis et al., 2021](#)). One methodology that has
11 proved useful to understand machine learning performance is the energy landscape framework
12 from chemical physics ([Wales, 2003](#)).

13
14 The energy landscape framework is a set of algorithms that map the topography of continuous
15 surfaces by their stationary points. The topography is encoded as a weighted graph ([Noé
& Fischer, 2008](#)) and in application to potential energy surfaces all physical properties of
16 a system can be extracted from this graph ([Swinburne & Wales, 2020](#)). Examples of the
17 methodology applied to potential energy surfaces explain physical phenomena for proteins
18 ([Röder et al., 2019](#)), small molecules ([Matysik et al., 2021](#)), atomic clusters ([Csányi et al.,
2023](#)) and crystalline solids ([Pracht et al., 2023](#)).

19
20 Since the energy landscape framework is applicable to any given continuous surface, the
21 methodology can also be applied to a wide range of machine learning algorithms through
22 the corresponding loss function surface. Fitting of a machine learning model usually aims to
23 locate low-valued or diverse solutions, and an understanding of the solution space topography
24 explains model reproducibility and performance. Leveraging the energy landscape framework
25 the performance and reliability of neural networks ([Niroomand et al., 2022](#)), Gaussian processes
26 ([Niroomand et al., 2023](#)) and clustering algorithms ([Dicks & Wales, 2022, 2023](#); [Wu et al.,
2023](#)) has been explored. Moreover, it has been used to explain the effect of dataset roughness
27 on ML model performance ([Dicks et al., 2024](#)). A tutorial review of different applications is
28 given in [Niroomand et al. \(2024\)](#).

31 Statement of need

32 The topsearch Python package provides a rapid prototyping software for application of the
33 energy landscape framework. It contains the functionality to be used for both potential energy
34 surfaces and the loss function surfaces of varied machine learning models.

35 There is limited software for explicitly analysing the topography of loss function surfaces. These
36 surfaces are considered implicitly when optimising an ML model through local minimisation,
37 but none attempt to capture global topographical features of the parameter space. There
38 is significantly more software for analysing potential energy surfaces, the majority of which
39 approximate topographical features indirectly. Popular examples that aim to explore diverse
40 regions of the surface through enhanced sampling are PyEMMA ([Scherer et al., 2015](#)) and

41 large molecular simulation suites such as LAMMPS (Thompson et al., 2022), GROMACS
42 (Abraham et al., 2015), and AMBER (Case et al., 2023) the simulations of which can be
43 simplified using PLUMED (Tribello et al., 2014). Explicit location of topographical features,
44 such as stationary points, is more common in quantum chemistry and can be performed by
45 software such as VTST (Henkelman, 2018), PASTA (Kundu et al., 2018), PyMCD (Lee et
46 al., 2023) and ORCA (Neese et al., 2020). The explicit computation of topography using the
47 energy landscape framework has several advantages for application to machine learning and
48 none of the above software contains all the required functionality.

49 Current leading tools for applying the energy landscape framework are the suite of FORTRAN
50 programs: GMIN (D. J. Wales, 2024a) OPTIM (D. J. Wales, 2024b) and PATHSAMPLE (D. J.
51 Wales, 2024c). This software implements almost all functionality described within the energy
52 landscape literature and, being written in a compiled language, is highly performant. Whilst a
53 clear choice for production work where performance is critical, it is not without limitations
54 for rapid prototyping. The user requires a detailed understanding of, and to pass information
55 between, three large distinct pieces of software. There is a Python wrapper, pylfl (Niroomand,
56 2023), which simplifies their use, but does not remove the limitation of multiple programs that
57 all require a detailed understanding. Furthermore, the software suite contains limited support
58 for machine learning models, and addition of new models is challenging and time-consuming
59 due to a lack of implementations of ML libraries in FORTRAN. Therefore, there is a need for a
60 single software that performs the energy landscape framework for both ML and physics, which
61 integrates seamlessly with ML libraries, thus enabling rapid prototyping in this domain.

62 topsearch replaces the functionality of the FORTRAN software suite in a single software
63 package, reducing the need for data transfer and subsequent parameterisation and setup. The
64 package, written entirely in Python, contains additional novel functionality for machine learning,
65 and due to the prevalence of Python in machine learning further new models can be included
66 quickly and easily. Furthermore, the implementation is significantly shorter, containing less
67 than a hundredth of the lines of code; enabling faster developer onboarding.

68 Applications

69 The Github repository (<https://github.com/IBM/topography-searcher>) contains examples for
70 varied applications, which are listed in turn below.

- 71 ■ `example_function` - This folder contains examples for mapping the surface topography
72 of an arbitrary function. The examples provide an introduction to the methodology, and
73 illustrate the major code functionality. Application to two-dimensional functions allows
74 direct visualisation of the surfaces, which makes clear the topographical analysis.
- 75 ■ `dataset_roughness` - Illustration of the novel code application to quantify dataset
76 roughness (Dicks et al., 2024). This analysis can uniquely explain and predict ML
77 regression performance both globally and locally, even in the absence of training data.

78 Atomic and molecular systems require significant additional functionality. However, the
79 examples illustrate that the scripts remain remarkably similar.

- 80 ■ `atomic` - An example that performs exploration of the potential energy surface of a small
81 atomic cluster.
- 82 ■ `molecular` - These examples illustrate how to explore the potential energy surfaces of
83 small molecules using quantum chemistry.

84 This list of examples does not form an exhaustive set of use cases. Previous applications of
85 this methodology, which will also be possible using topsearch, are protein and nucleic acids
86 potential energy surfaces and Gaussian process, neural network and clustering loss function
87 surfaces. Moreover, there are many additional machine learning models that could be analysed,
88 and the Python implementation allows for their rapid inclusion.

89 Conclusions

90 The topsearch Python package fulfils the need for a rapid prototyping and analysis tool for
91 the energy landscape framework that can be applied to both physics and machine learning
92 models. This software is significantly more lightweight than existing solutions; a large reduction
93 in code and integration in a single piece of software ensures the Python implementation is
94 significantly easier to develop. Moreover, the package provides a simpler interface for accessing
95 the functionality, and in tandem with detailed examples, results in a shallower learning curve
96 for use within diverse applications. Lastly, the software is unique in the amount of machine
97 learning models that can be explored and and can easily be extended with existing Python
98 implementations. Our aim is that this software package will aid diverse researchers from
99 computer science to chemistry by providing a simple solution for application of the energy
100 landscape framework.

101 Acknowledgements

102 LD and EOP-K would like to acknowledge the financial support of the Hartree National Centre
103 for Digital Innovation – a collaboration between the Science and Technology Facilities Council
104 and IBM. The authors would also like to thank Nicholas Williams, Matthew Wilson, Nicolas
105 Galichet and Vlad Cărare for their helpful feedback as early users of the package.

106 References

- 107 Abraham, M. J., Murtola, T., Schulz, R., Páll, S., Smith, J. C., Hess, B., & Lindahl, E. (2015).
108 GROMACS: High performance molecular simulations through multi-level parallelism from
109 laptops to supercomputers. *SoftwareX*, *1*, 19–25. <https://doi.org/10.1016/j.softx.2015.06.001>
110
- 111 Angelov, P. P., Soares, E. A., Jiang, R., Arnold, N. I., & Atkinson, P. M. (2021). Explainable
112 artificial intelligence: an analytical review. *Wiley Interdiscip. Rev.: Data Min. Knowl.*
113 *Discov.*, *11*, e1424. <https://doi.org/10.1002/widm.1424>
- 114 Case, D. A., Aktulga, H. M., Belfon, K., Ben-Shalom, I. Y., Berryman, J. T., Brozell, S. R.,
115 Cerutti, D. S., III, T. E. C., Cisneros, G. A., Cruzeiro, V. W. D., Darden, T. A., Forouzes, N.,
116 Giambasu, G., Giese, T., Gilson, M. K., Gohlke, H., Goetz, A. W., Harris, J., Izadi, S.,
117 ... Kollman, P. A. (2023). *AMBER 2023*. <https://ambermd.org/doc12/Amber23.pdf>
- 118 Csányi, G., Morgan, J. W. R., & Wales, D. J. (2023). Global analysis of energy landscapes
119 for materials modeling: a test case for C60. *J. Chem. Phys.*, *159*, 104107. <https://doi.org/10.1063/5.0167857>
120
- 121 D. J. Wales. (2024a). *GMIN: A program for basin-hopping global optimisation, basin-sampling,*
122 *and parallel tempering*. <https://www-wales.ch.cam.ac.uk/GMIN/>
- 123 D. J. Wales. (2024b). *OPTIM: A program for geometry optimisation and pathway calculations*.
124 <http://www-wales.ch.cam.ac.uk/OPTIM/>
- 125 D. J. Wales. (2024c). *PATHSAMPLE: A program for generating connected stationary point*
126 *databases and extracting global kinetics*. <http://www-wales.ch.cam.ac.uk/PATHSAMPLE/>
- 127 Dicks, L., Graff, D. E., Jordan, K. E., Coley, C. W., & Pyzer-Knapp, E. O. (2024). A
128 physics-inspired approach to the understanding of molecular representations and models.
129 *Mol. Syst. Des. Eng.* <https://doi.org/10.1039/D3ME00189J>
- 130 Dicks, L., & Wales, D. J. (2022). Elucidating the solution structure of the *K*-means cost
131 function using energy landscape theory. *J. Chem. Phys.*, *156*, 054109. <https://doi.org/10.1063/5.0078793>
132

- 133 Dicks, L., & Wales, D. J. (2023). Evolution of K -means solution landscapes with the addition
134 of dataset outliers and a robust clustering comparison measure for their analysis. *arXiv*.
135 <https://doi.org/10.48550/arXiv.2306.14346>
- 136 Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired
137 artificial intelligence. *Neuron*, *95*, 245–258. <https://doi.org/10.1016/j.neuron.2017.06.011>
- 138 Henkelman, G. (2018). *VTST tools*. <https://vtstools.readthedocs.io/en/latest/index.html>
- 139 Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021).
140 Physics-informed machine learning. *Nat. Rev. Phys.*, *3*, 422–440. <https://doi.org/10.1038/s42254-021-00314-5>
141
- 142 Kundu, S., Bhattacharjee, S., Lee, S.-C., & Jain, M. (2018). PASTA: Python Algorithms for
143 Searching Transition stAtes. *Comput. Phys. Commun.*, *233*, 261–268. <https://doi.org/10.1016/j.cpc.2018.06.026>
144
- 145 Lee, K., Kim, J. H., & Kim, W. Y. (2023). pyMCD: Python package for searching transition
146 states via the multicoordinate driven method. *Comput. Phys. Commun.*, *291*, 108831.
147 <https://doi.org/10.1016/j.cpc.2023.108831>
- 148 Matysik, S. C., Wales, D. J., & Jenkins, S. J. (2021). Rotational dynamics of desorption:
149 methane and ethane at stepped and kinked platinum surfaces. *J. Phys. Chem. C*, *125*,
150 27938–27948. <https://doi.org/10.1021/acs.jpcc.1c09120>
- 151 Neese, F., Wennmohs, F., Becker, U., & Riplinger, C. (2020). The ORCA quantum chemistry
152 program package. *J. Chem. Phys.*, *152*, 224108. <https://doi.org/10.1063/5.0004608>
- 153 Niroomand, M. P. (2023). *pylfl*. <https://pypi.org/project/pylfl/>
- 154 Niroomand, M. P., Cafolla, C. T., Morgan, J. W. R., & Wales, D. J. (2022). Characterising
155 the area under the curve loss function landscape. *Mach. Learn.: Sci. Tech.*, *3*, 015019.
156 <https://doi.org/10.1088/2632-2153/ac49a9>
- 157 Niroomand, M. P., Dicks, L., Pyzer-Knapp, E. O., & Wales, D. J. (2023). Physics inspired
158 approaches to understanding Gaussian processes. *arXiv*. <https://doi.org/10.48550/arXiv.2305.10748>
159
- 160 Niroomand, M. P., Dicks, L., Pyzer-Knapp, E. O., & Wales, D. J. (2024). Insights into
161 machine learning models from chemical physics: an energy landscapes approach (EL for
162 ML). *Digital Discovery*. <https://doi.org/10.1039/D3DD00204G>
- 163 Noé, F., & Fischer, S. (2008). Transition networks for modelling the kinetics of conformational
164 change in macromolecules. *Curr. Opin. Struct. Biol.*, *18*, 154–162. <https://doi.org/10.1016/j.sbi.2008.01.008>
165
- 166 Pracht, P., Morgan, J. W. R., & Wales, D. J. (2023). Exploring energy landscapes for
167 solid-state systems with variable cells at the extended tight-binding level. *J. Chem. Phys.*,
168 *159*, 064801. <https://doi.org/10.1063/5.0159367>
- 169 Röder, K., Joseph, J. A., Husic, B. E., & Wales, D. J. (2019). Energy landscapes for
170 proteins: from single funnels to multifunctional systems. *Adv. Theory Simul.*, *2*, 1800175.
171 <https://doi.org/10.1002/adts.201800175>
- 172 Scherer, M. K., Trendelkamp-Schroer, B., Paul, F., Pérez-Hernández, G., Hoffmann, M.,
173 Plattner, N., Wehmeyer, C., Prinz, J.-H., & Noé, F. (2015). PyEMMA 2: A software
174 package for estimation, validation, and analysis of Markov models. *J. Chem. Theory
175 Comput.*, *11*, 5525–5542. <https://doi.org/10.1021/acs.jctc.5b00743>
- 176 Swinburne, T. D., & Wales, D. J. (2020). Defining, calculating and converging observables
177 of a kinetic transition network. *J. Chem. Theory Comput.*, *16*, 2661–2679. <https://doi.org/10.1021/acs.jctc.9b01211>
178

- 179 Thompson, A. P., Aktulga, H. M., Berger, R., Bolintineanu, D. S., Brown, W. M., Crozier, P.
180 S., Veld, P. J. in 't, Kohlmeyer, A., Moore, S. G., Nguyen, T. D., Shan, R., Stevens, M. J.,
181 Tranchida, J., Trott, C., & Plimpton, S. J. (2022). LAMMPS – a flexible simulation tool
182 for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput.*
183 *Phys. Commun.*, 271, 10817. <https://doi.org/10.1016/j.cpc.2021.108171>
- 184 Tribello, G. A., Bonomi, M., Branduardi, D., Camilloni, C., & Bussi, G. (2014). PLUMED
185 2: New feathers for an old bird. *Comput. Phys. Commun.*, 185, 604–613. <https://doi.org/10.1016/j.cpc.2013.09.018>
- 187 Wales, D. J. (2003). *Energy Landscapes*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511721724>
- 189 Wu, Y., Dicks, L., & Wales, D. J. (2023). Archetypal solution spaces for clustering gene
190 expression datasets in identification of cancer subtypes. *arXiv*. <https://doi.org/10.48550/arXiv.2305.17279>
- 192 Zhang, Y., Tiño, P., Leonardis, A., & Tang, K. (2021). A survey on neural network
193 interpretability. *IEEE Trans. Emerg. Top. Comput. Intell.*, 5, 726–742. <https://doi.org/10.1109/TETCI.2021.3100641>
- 194

DRAFT