Case for Support for sktime: a toolkit for machine learning with time series

`sktime`[1] provides an easy-to-use, flexible and modular open-source Python framework for a wide range of time series machine learning tasks [2]. It offers scikit-learn compatible algorithms and model composition tools, with the goal to make the ecosystem more usable and interoperable as a whole. We build and support an open, diverse and self-governing community. We welcome new contributors from academia and industry through instructive documentation, mentoring and workshops. `sktime` was initially conceived by researchers at the University of East Anglia (UEA) and University College London (UCL) as part of the Alan Turing Institute's Tools, Practices and Systems (TPS) programme and has grown to become a central part of the Python ecosystem for time series analysis. This project will support the ongoing maintenance and development of `sktime` whilst helping us extend the user base to important areas of scientific research within the EPSRC remit.

1 Investigators Previous Research Track Record

Anthony Bagnall (AJB) is a Professor within the School of Computing Sciences at UEA. His main research interest is in the field of time series data mining, with a specific focus on time series classification (TSC). He was PI on EPSRC grant "The Collective of Transform Ensembles (COTE) for Time Series Classification" [EP/M015807/1] [3], the direct precursor of this proposal that lead to the development of the current state of the art in the field, the Hierarchical Vote Collective of Transform Ensembles (HIVE-COTE) [4, 1] and a widely cited benchmark study [5]. He leads the time series machine learning group in CMP which currently consists of four faculty members and six PhD students. His work in TSC led to the invitation to develop the `sktime` toolkit with UK Research and Innovation Council funded TPS prgramme [EP/T001569/1]. He has contributed to a range of applications for TSC, including: detecting electric devices in bags [DASA grant ACC106973]; finding bogus call centres based on call activity [EPSRC iCASE award]; detecting forged spirits from spectra [BBSRC iCASE award]; and insect classification from reconstructed audio. He maintains the data archives for both univariate and multivariate time series classification[2] in collaboration with the University of California, Riverside (UCR) and is a community council member and core developer for `sktime`.

Markus Löning (ML) is a data scientist who has recently completed his PhD at UCL. His research focuses on machine learning and software engineering. He is the lead developer of `sktime`. He has worked on applied problems in collaboration with different industry partners (Shell, BMW, AstraZeneca) and third-sector organisations (Great Ormond St Hospital, Rothamsted Research). Markus will be employed part time on this project to lead the collaboration with the UCL team on Work Package (WP) 3. He will continue to contribute to other aspects of `sktime` in his own time.

Franz Király (FK) is the founder of `sktime` and a principal data scientist at Shell. He also has a research focus on quality of data science, assurance of algorithms, and software engineering research for AI toolboxes. In addition to his continued involvement in all aspects of developing `sktime`, he will manage the interaction with project partner Shell and be responsible for coordinating and extending `sktime` into further industrial settings.

Maintenance and Community Building Contributors

James Hetherington is Director of the UCL Advanced Research Computing (ARC) Centre, with previous roles as Chief Data Science Advisor to the UK's Joint Biosecurity Centre, Director of Digital Research Infrastructure at UKRI, and Director of Research Engineering at the Turing. He will oversee ARC's involvement in WP1 and WP3 and assist with Turing liaison.

Jonathan Cooper is Head of Research Software Engineering at UCL, leading a group of around 30 research software engineers and data scientists. He will coordinate ARC contributions to `sktime` development on in order to integrate `sktime` with research projects throughout UCL, and oversee the link with CHIMERA.

[1] https://www.sktime.org/en/latest/
[2] https://www.timeseriesclassification.com

Kirstie Whitaker leads the Tools, Practices and Systems project at the Turing and has extensive experience building and nurturing inclusive open source communities, particularly through The Turing Way and Brain Imaging Data Structure (BIDS) communities. Her role will be to co-ordinate the Turing contribution on WP1 and WP2 and help integrate `sktime` with other Turing projects.
Martin Walter is a senior data scientist in the Data and Analytics group of Mercedes-Benz, an `sktime` community council member and a core contributor. He will contribute to ongoing maintenance and development on WP1 and WP2, assist in industry outreach and look for more internal uses of `sktime`.
Leo Tsaprounis is a senior data scientist in the Consumer Healthcare division of GlaxoSmithKline and an `sktime` contributor. He will continue to contribute, promote `sktime` within the company and seek out mechanisms to formalise their support of `sktime`.

Academic collaborators for improved functionality

Geoff Webb is the Director of the Monash University Centre for Data Science. His group have contributed extensively to `sktime` and will continue to make valuable contributions to the toolkit on WP1 and WP2.
Eamonn Keogh is a distinguished professor at UCR and has been a long term collaborator with AJB [7, 9]. His group will support this proposal by providing expert guidance on time series anomaly detection in WP2.

Scientific collaborators for improved scientific workflow

Christina Pagel is a professor of operational research at University College London and co-director of the EPSRC Collaborative Healthcare Innovation through Mathematics, EngineeRing and AI (CHIMERA) project [EP/T017791/1]. She will contribute to WP3 to help integrate `sktime` into CHIMERA projects.
Samiran Ray is a Consultant at the Paediatric Intensive Care Medicine unit at the Great Ormond Street Hospital and a member of the CHIMERA team. He will advise on the cases study using GOSH data.
Rik Henson is an MRC Programme Leader, University of Cambridge Professor, and Deputy Director of the MRC Cognition and Brain Sciences Unit. He is also current President of the British Neuroscience Association. He will help advise on M/EEG classification on WP3.
Saber Sami is a Senior Lecturer in Dementia Research at the Norwich Medical School, UEA and has two decades of work in brain electrophysiology and imaging.
Louis Renoult is an Associate Professor in Psychology at UEA. He has accumulated more than 15 years of experience using EEG in cognitive neuroscience research and has set-up the first EEG lab at UEA.
Thomas Sambrook is a Lecturer in Psychology at UEA and studies EEG signatures of reinforcement learning using monetary gambles and primary reinforcers such as food and electric shock.

References

[1] M. Middlehurst, J. Large, M. Flynn, J. Lines, A. Bostrom and A. Bagnall. HIVE-COTE 2.0: a new meta ensemble for time series classification. *Machine Learning*, online first, 2021.

[2] M. Löning, A. Bagnall, S. Ganesh, V. Kazakov, J. Lines, F. Király. sktime: A Unified Interface for Machine Learning with Time Series. *Workshop on Systems for ML at NeurIPS 2019*.

[3] A. Bagnall, J. Lines, J. Hills, and A. Bostrom. Time-series classification with COTE: The collective of transformation-based ensembles. *IEEE Trans. on Knowl. and Data Eng.*, 27:2522–2535, 2015.

[4] J. Lines, S. Taylor, and A. Bagnall. Time series classification with HIVE-COTE: The hierarchical vote collective of transformation-based ensembles. *ACM Trans. Knowl. Disc. from Data*, 12(5), 2018.

[5] A. Bagnall, J. Lines, J. Hills, A. Bostrom, J. Large and E. Keogh The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Min. and Knowl. Disc.*, 31(3):606–660, 2017.

[6] J. Large, J. Lines, and A. Bagnall. A probabilistic classifier ensemble weighting scheme based on cross validated accuracy estimates. *Data Min. and Knowl. Disc.*, 33(6):1674—1709, 2019.

[7] H. Dau, A. Bagnall, K. Kamgar, M. Yeh, Y. Zhu, S. Gharghabi, C. Ratanamahatana, A. Chotirat, and E. Keogh. The UCR time series archive. *IEEE/CAA Journal of Auto. Sinica*, 6(6):1293–1305, 2019.

[8] A. Dempster, F. Petitjean, and G. Webb. ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels. *Data Min. and Knowl. Disc.*, 34: 1–42, 2020.

[9] S. Gharghabi, S. Imani, A. Bagnall, A. Darvishzadeh, and E. Keogh. Matrix Profile XII: MPDist: A novel time series distance measure to allow data mining in more challenging scenarios. In *proc. of 18th IEEE ICDM*, 2018.

2 Description of proposed research and its context

Techniques for learning from time series have been developed in a wide range of disciplines, including: statistics; machine learning; signal processing; econometrics; and finance. Each discipline has a favoured set of tools and accepted workflows. Despite similarity between tasks, the development and evaluation of algorithms has traditionally been siloed. Moreover, tasks are frequently reduced from one type of problem (e.g. forecasting) to another (e.g. regression, classification or clustering) and this commonly happens without reference to the state-of-the-art algorithms developed specifically for the new task. In recent years, machine learning frameworks such as scikit-learn have become essential infrastructure of modern data science. They have become the principal tool for practitioners and central components in scientific, commercial and industrial applications. But despite the ubiquity of time series data, no such framework exists for machine learning with time series.

`sktime` provides this unified framework to make the existing ecosystem of tools more usable and interoperable as a whole. Frameworks like `sktime` not only offer reusable functionality, but also provide overall structure to application code. They capture common design decisions and distil them into reusable templates that practitioners can copy. This reduces the number of decisions practitioners must take and allows them to focus on application specifics. Not only can practitioners write software faster as a result, but applications will have a similar structure. They will be more consistent, more reusable and easier to maintain. In addition, `sktime` aims to link communities of practitioners, researchers and domain experts. We provide a common platform to define and formalise multiple time series tasks such as forecasting, classification, clustering, regression, annotation, anomaly detection and change point detection as well as reduction approaches between them. Since its inception in 2018, `sktime` has become an established toolkit for time series analysis used world-wide by academics and industry alike. Whilst demand is steadily growing, `sktime` is increasingly facing bottlenecks in its maintenance activities. It has been without any dedicated funding since 2019 and its operations are currently entirely driven by volunteers.

This project will allow `sktime` to continue to sustain and grow its operations, by providing dedicated maintenance resources. It will also allow us to further enhance sktime's functionality and have impact on new scientific and industrial user communities.

2.1 Aims and objectives

The project aims are to maintain, improve, and widen participation in the development and maintenance of `sktime`. Our objectives, achieved through three work packages (WP), are as follows.

1. Maintenance and community building (WP1) to improve the process and speed of conducting essential maintenance activities and widen participation in the maintenance of sktime.
2. Extend functionality (WP2) to oversee and steer the development of new state-of-the-art functionality by the wider `sktime` community.
3. Enhance scientific workflows (WP3) by applying `sktime` to problems arising in two specific research communities within the EPSRC remit.

2.2 Governance of sktime

We have put in place an infrastructure to support and grow a friendly and diverse user and developer community. sktime's community organisation is described on our website[3]. The community council is responsible for technical leadership and project management. The council is structured to represent sktime's three main communities: the academy, industry, and early-career data scientists. We hold meetings open to everyone fortnightly and publish minutes on a dedicated repository. Members of the council are elected by the group of core developers, of which there are currently 13 active members. Core developers take on key responsibilities in the maintenance and development of `sktime`, including resolving issues, managing pull requests and implementing new features. The community team contains contributors wanting to help with outreach, community engagement, communications and media. We have adopted a clear and enforceable code of conduct based on the Turing Way[4], with multiple designated points of contact for

[3] `https://www.sktime.org/en/latest/governance.html`

[4] `https://github.com/alan-turing-institute/the-turing-way`

incident reporting. Finally, some contributors are code owners, with specific responsibilities for particular algorithms, normally ones proposed by themselves in original research. They are asked to review and approve any modifications to code they are responsible for.
The team engage with the academic community through presenting research talks, publishing papers describing novel research conducted with `sktime` in relevant workshops [2], conferences [15] and journals [1]. Industry links are coordinated by council member Franz Király. We communicate to the wider Python community through presentations at PyData, EuroPython] and SciPy events. We onboard new users and contributors through mentoring programmes such as Google of Summer of Code, Major League Hacking and Outreachy in addition to our own mentoring programme. Since 2019, we have had 18 mentees joining our community, seven of whom have gone on to become core developers.
A community-driven approach will always be at the heart of `sktime`. Funding through this project will allow us to make our operations more robust, establish deeper links within specific user communities, and make the project more sustainable by widening participation in the development process.

2.3 Evidence of established and growing demand for sktime

`sktime` has gained significant traction and interest is growing rapidly. It has become the most developed and comprehensive open-source toolkit for machine learning with time series. At the time of writing, `sktime` has been downloaded over a million times. On GitHub, it has over 110 contributors, 230 dependent projects, 650 forks and 4.5K stars (a star indicates a user likes or wants to follow a repository). `sktime` rose from 200 stars in October 2019 to 2.3K in October 2020, and then to 4.5K in September 2021. For context, scikit-learn had 2.6K stars in May 2014, which rose to 5.2K stars in May 2015. scikit-learn has maintained growth of around 5-10K stars per annum. We envisage mirroring this growth. Demand for `sktime` is further evidenced by the fact a PyData tutorial introducing `sktime` has been viewed approximately 70K times[5].
We already have a large community of users across a range of disciplines. `sktime` has been used by researchers across the globe, and has gained traction with the academic time series machine learning community. We have had contributions from over 10 research groups based in UK, Ireland, France, Spain, Germany, USA, Brazil and Australia. Early-career researchers, including dozens of undergraduate and postgraduate students, have used and contributed to the toolkit. `sktime` is embedded in Computer Science and Data Science degrees at UEA, and doctoral training at UCL. We use it as a vehicle to teach about open-source software development and time series machine learning on projects and dissertations. UEA are launching a new MSc module based around the toolkit. Through collaboration with UCL and their Centre for Advanced Research Computing (ARC), we will also further incorporate `sktime` in a range of research and teaching activities. We have also begun the process of actively reaching out to specific scientific communities through our 2021 Google Summer of Code project focusing on using `sktime` for Electroencephalography (EEG) classification. A core objective of this project is continuing this process of establishing new collaborations, educating and enabling scientific communities under the EPSRC remit that are less likely to actively seek out open-source projects.
Industry and the public sector has also adopted `sktime` in production. Commercial platform solutions such as Alteryx use `sktime` as a core commodity component. `sktime` is also receiving an increasing number of contributions from industry and public-sector data scientists, up to and including contributions that are embedded in company-internal development and deployment processes, such as from GlaxoSmithKline, Mercedes Benz, Shell, and the US Farm Credit Administration. We have recently enabled support of `sktime` through Open Collective and GitHub Sponsors, and will continue to promote this funding mechanism.

3 Programme and methodology

This project will allow `sktime` to meet this increasing demand. We will maintain and develop the codebase, run outreach events, increase collaborations in academia and industry, improve the usability and interoperability of the time series analysis ecosystem, build specific applications and new user communities focused on medical and healthcare topics through dedicated companion packages. Our three work packages map directly onto our three objectives.

[5] https://www.youtube.com/watch?v=wqQKFu41FIw&t=14s

WP1: Maintenance and Community Building

We currently have over 200 open issues and 60 open pull requests (PRs). Dealing with the regular queries, bugs and feature requests requires a significant amount of effort from the core developers, all of whom work or study full time. As a consequence, our response time is sometimes slow, and issues and PRs are not always given the attention they deserve. WP1 will help improve the maintenance and sustainability of `sktime`.

The research software engineer (RSE) on this grant will be given a central role in the maintenance and development of `sktime`, and enable its move towards being a professional grade software tool. The RSE will have a range of key roles: they will be responsible for triaging and responding quickly to new issues and pull requests, prioritising issues where appropriate, taking on basic project management responsibilities, and delegating issues to core developers or code owners to review; they will make a valuable contribution to the software design and full integration of existing and new modules through propagating changes in overarching design throughout the toolkit; they will take a crucial role throughout our deliberation process as outlined in sktime's governance; they will lead discussion at developer meetings and liase with core developers to help find consensus on controversial topics; and they will help organise and run annual hackathons held in collaboration with our project partners.

`sktime` already provides consistent interfaces for a number of Python libraries for time series analysis, including scikit-learn, statsmodels, tslearn, tsfresh and fbprophet. We collaborate with the maintainers of these libraries and will continue working towards defining standard interfaces for different learning tasks, with the aim of improving usability and interoperability of the ecosystem as a whole. The RSE will help implement a more standardised and portable approach to unit testing. This will enable other libraries and closely-aligned companion packages such as `sktime-dl` to be developed in parallel, while ensuring interface compliance with `sktime`.

A recent `sktime` docsprint highlighted significant areas to further improve sktime's usability. The RSE will lead this improvement and provide guidance in the form of tutorials, how-to-guides and PR reviews to further improve our documentation standard. In addition, the RSE will help improve the existing user guides and work with developers to extend them to include new functionality introduced in WP2.

We will hold a series of events to help widen participation in the development and maintenance in academia, industry and the wider Python community. Academic participation will be encouraged through continued research collaborations, conference tutorials and publications of new results that showcase the functionality of `sktime`. The UEA and UCL teams will work together to construct a postgraduate module in time series analysis using `sktime` and embed this within BSc and MSc courses at both universities. Current `sktime` contributors in our three industry partners will propagate the usage of `sktime` within their respective organisations. Industry applications will provide valuable feedback. The requirements and design of new features will be formalised through consultation with our industry partners. Their public support will encourage other industry supporters to become involved. They will provide feedback from using `sktime` in production in industry, guidance on new features and priorities and software domain expertise. Annual hackathons will help forge links between project partners and widen engagement. Finally, participation in the wider community will be promoted by extending our mentorship scheme and summer internships as well as tutorials at conferences such as PyData, EuroPython and SciPy.

WP2: Extend Functionality

WP2 involves oversight of the broadening of the functionality of `sktime` to impact current and new user communities. The scope of the functional improvements is ambitious, but we envisage that the majority of the code will be provided by the open-source community. The primary role of the RSE, AJB, ML and FK in this process will be to oversee the quality control and code design compliance, provide API design solution where necessary, and help coordinate parallel development efforts. `sktime` is structured into modules that relate to the learning task.

The **forecasting module** has been widely adopted by a range of practitioners. It forms the basis for the involvement of Mercedes-Benz and Shell. Developing a range of forecasting tools within an easy-to-use and modular API is a significant challenge. The forecasting module will be extended to include better support for multivariate forecasting and new functionality for hierarchical time series and panel/longitudinal data. Rather than implementing every algorithm from scratch, we will interface to existing libraries wherever

possible.
The **classification module** in `sktime` contains the very latest TSC algorithms that are unavailable anywhere else in one toolkit. This includes recent contributions from UEA [1], Oxford University [14], Monash [8], University College Dublin [17] and Humboldt University of Berlin [20]. `sktime` has become the de facto standard for developing and comparing TSC algorithms. The priority with the classification module will be to improve the code efficiency (for example, through better threading of algorithms) and adapting the algorithms to handle alternative classification use cases: multivariate TSC [19] is a less developed field than univariate classification. Little is known about the best way to handle unequal length series. Missing values in time series require different treatment to those in standard classification.
The **transformation module** contains functionality used by all the other modules. Many TSC algorithms are a pipeline combination of bespoke time series transformations and standard scikit-learn classifier. For example, the shapelet transform [13], signatures [14] and Mr-SEQL [17] all follow this design pattern. `sktime` provides concrete implementations of these algorithms, but also allows the user to configure their own pipeline composition of transformer and classifiers. We will review and revise the structure of this module to make it more usable across different learning tasks.
The **clustering module** was recently introduced, extending scikit-learn clustering functionality to time series distance functions. There are a host of bespoke transformation-based clustering algorithms (for example, u-shapelets [24] and k-shapes [18]) we do not yet have. We will work with the current contributors to help implement efficient versions of the state-of-the-art for time series clustering and encourage researchers in the field to contribute their new algorithms in `sktime`.
The **regression module** is at an early stage of development. Time series regression (TSR) is most commonly encountered as a reduction of a forecasting problem: a window is run along a time series and the target variable is the next time point. There is another type of TSR problem, where each case is independent and the target is an extrinsic variable [22]. This TSR scenario has not been systematically studied. Preliminary work [22] has highlighted a gap in machine learning research in this area. We will work with our contributors at Monash to set up a clear workflow for reproducible research into TSR algorithms. TSC algorithms can often be naively adapted for TSR through a change of loss function. We will provide standard TSR versions of the best TSC algorithms. This will link the forecasting and classification modules through regression.
The **annotation module** contains prototype functionality for anomaly detection that is in the design stage. Anomaly detection in time series has been popular of late (for example, see [16]), but it has recently been claimed that much of the experimentation and evaluation in the field is flawed [25]. We will join the efforts of researchers at UCR to place this research on a sounder experimental footing. Our implementation will be based around a 2021 SIGKDD time series anomaly detection competition[6]. We will collaborate with the organiser of the competition, Eamonn Keogh, to port the top performing algorithms into `sktime`. We will demonstrate the benefit of the unified approach to deliver more effective research workflows through the application of the techniques to problems arising in WP3.
The new **change point detection module** will be based on an existing package hosted by the Turing[7] and the ruptures toolkit [23]. One approach to change point detection is to reduce the problem to clustering or anomaly detection through windowing. `sktime` is designed to make this type of benchmarking simple, and we will compare standard techniques to these benchmarks. We will also include recently proposed algorithms such as CLASP [21] through collaboration with the algorithm inventor.
Deep learning has been applied to each task addressed by `sktime`. We aim to link deep learning and traditional machine learning research strands through `sktime`. Our approach to enabling deep learning functionality is to have a companion package, `sktime-dl`[8], that integrates `sktime` and TensorFlow. `sktime-dl` includes the time series classification and regression algorithms evaluated in the experimental research [12] and the current state-of-the-art for classification, InceptionTime [11]. However, the package is much less developed than the core library and contains no forecasting specific approaches. We will enhance the functionality of `sktime-dl` by integrating with PyTorch and JAX. We will include common neural network architectures from the literature for forecasting, clustering, anomaly and change point detection. We will advance the sustainability of `sktime-dl` by improving the continuous integration.

[6] `https://forums.hexagon-ml.com/t/multi-dataset-time-series-anomaly-detection/591/124`
[7] `https://github.com/alan-turing-institute/TCPDBench`
[8] `https://github.com/sktime/sktime-dl`

WP3: Enhancing Scientific Workflows

WP3 is concerned with impacting scientific communities that fall under the EPSRC remit to enhance their workflow by using `sktime`. This deepening of the reach of `sktime` will be achieved through collaborations with domain experts in two fields: signal processing for magnetoencephalography and electroencephalography (M/EEG) analysis; and exploratory analysis of data from healthcare technologies. These projects will demonstrate the utility and potential for impact of `sktime` and guide its future development priorities.

EEG records electrical activity in the brain using a series electrodes placed on the scalp. EEG equipment is relatively cheap and portable and is currently one of the most widely used non-invasive brain imaging tools in neuroscience and hospitals. In research, EEG time series are used in a wide range of fields, including medicine (e.g. diagnosis of epilepsy or the early detection of dementia), computer science (e.g. brain computer interfacing (BCI) and human activity recognition) and psychology (e.g. the study of cognitive development). At the heart of many EEG related research questions is the problem of building a predictive regression or classification problem. This can be diagnostic (does the EEG recording of a patient indicate they have dementia?), descriptive (can we tell from the EEG recording whether an individual is moving their left or right arm?), or cognitive (is the subject looking at a picture of a face or random noise?). Each field has a range of related tasks and experimental structures, and each has a different default methodology. For example, Riemannian approaches are popular in BCI [26] but not commonly used in psychology or medical research. We will coordinate a team to collaborate on EEG analysis. UEA are advertising a 2022 interdisciplinary PhD position in EEG classification, focussing primarily on BCI. Our team of experts in using EEG for medicine and psychology will help us understand and implement the standard workflows adopted in the different fields. An `sktime` 2021 Google of Summer of Code (GSoC) project launched `sktime-neuro`[9], a companion package for `sktime` that integrates the MNE package[10] and interfaces to the BIDS data standard[11]. Using our experience with `sktime`, and with the help of further GSoC projects, we will make `sktime-neuro` a collaborative mechanism and community hub for the scientific exploration of EEG/MEG. We will promote reproducible research and, by providing standard pipelines and access to state of the art algorithms, facilitate more effective and efficient research workflows. To demonstrate this improved workflow, we will conduct a specific case study on using MEG to detect early onset dementia. The MRC Cognition and Brain Sciences Unit of the University of Cambridge have accumulated a unique database of resting-state MEG data from approximately 150 patients with Mild Cognitive Impairment (MCI) – a potential prodromal stage of dementia – plus over a 150 age- and sex-matched controls. Moreover, to study the effects of healthy ageing, they also have the same data on a normative sample of nearly 700 adults from the CamCAN project[12]. There is already a data-sharing agreement between the universities of Cambridge and East Anglia, so the data can be used to showcase `sktime-neuro`.

The Collaborative Healthcare Innovation through Mathematics, EngineeRing and AI (CHIMERA) project [EP/T017791/1] is a collaborative hub based at UCL and partnered with the Turing, Great Ormond Street Hospital (GOSH) and University College London Hospitals NHS Foundation Trust. CHIMERA is concerned with analysing patients' physiological data. Obtaining external access to hospital data is notoriously difficult. It is becoming more common for hospitals to employ in house data scientists. We wish to develop a new user base of hospital data scientists willing to share findings and code rather than data. We believe the rapid exploration and modelling of clinical data and ideas sharing between institutions will lead to faster discovery of new best clinical practice and a better understanding of the impact of medical interventions. To help meet this long-term goal, we will work with our partners in ARC and the Turing to embed `sktime` within the projects involved in CHIMERA. The interaction with ARC, CHIMERA and the Turing will be led by named researcher Markus Löning. This will involve mentoring PhD students working in CHIMERA and discussions with investigators on problem definitions. We will demonstrate the utility of `sktime` for the project through a case study using data from GOSH Intensive Care from a Turing data study pilot study that Markus and Prof. Pagel participated in. The data is multivariate time series with measurements for vital body functions (heart rate, blood pressure, breathing rate, etc), a range of (anonymised) demographic data, and extubation (the removal of an endotracheal tube) and reintubation timing. Ex-

[9] https://github.com/sktime/sktime-neuro
[10] https://github.com/mne-tools/mne-python
[11] https://bids.neuroimaging.io/
[12] https://www.cam-can.org/

ploration of this data offers the opportunity to improve clinical practice and patient outcomes. Through collaboration with the GOSH clinicians we will look at tasks derived from this data such as predicting whether extubation will be successful.

The purpose of both the MEG and GOSH case studies is to showcase how `sktime` can reduce the development time for analysis of time series data whilst giving access to the very latest algorithmic advances. They will also serve to demonstrate the breadth of functionality in `sktime`: each dataset can provide classification, regression, clustering, anomaly detection and change point detection problems.

The domain expert collaborators on WP3 will be encouraged to engage in the hackathons which will be held annually. This will facilitate a bidirectional knowledge transfer: the future sktime development agenda will be influenced by the scientific user bases requirements, and the latest algorithmic enhancements will be showcased to those who can make good use of them.

4 National Importance

Time series occur in all areas of scientific endeavour, and this proposal has the potential for impact across EPSRC and UKRI priority areas. `sktime` has gained significant traction, and has been widely adopted in both forecasting and machine learning time series research. Support by the National Institute for Data Science and Artificial Intelligence will ensure that the impact of our research is seen and our toolkit is promoted. Our project helps meets several of the prosperity outcomes identified in the UKRI latest strategic plan, including productive nation (P1: Introduce the next generation of innovative and disruptive technologies and P2: Ensure affordable solutions for national needs), healthy nation (H3: Optimise diagnosis and treatment) and connected nation (C1: Enable a competitive, data driven economy and C3: Deliver intelligent technologies and systems). This proposal helps support the national strategic need of maintaining a unique world leading research into time series machine learning.

References

[11] H. Fawaz et al. InceptionTime: Finding AlexNet for time series classification. *Data Min. and Knowl. Disc.*, 34:1936–1962, 2020.

[12] H. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. Muller. Deep learning for time series classification: a review. *Data Min. and Knowl. Disc.*, 33(4):917–963, 2019.

[13] J. Hills, J. Lines, E. Baranauskas, J. Mapp, and A. Bagnall. Classification of time series by shapelet transformation. *Data Min. and Knowl. Disc.*, 28(4):851–881, 2014.

[14] P. Kidger et al. Deep signature transforms. In *proc. NeurIPS*, volume 32, 2019.

[15] M. Middlehurst, J. Large, G. Cawley, and A. Bagnall. The temporal dictionary ensemble (TDE) classifier for time series classification. In *proc. ECML-PKDD*, 2020.

[16] T. Nakamura, Imamura M, R. Mercer, and E. Keogh. MERLIN: Parameter-free discovery of arbitrary length anomalies in massive time series archives. In *IEEE ICDM*, 2020.

[17] T. Le Nguyen et. al. Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. *Data Min. and Knowl. Disc.*, 33(4):1183–1222, 2019.

[18] John Paparrizos and Luis Gravano. Fast and accurate time-series clustering. *ACM Transactions on Database Systems (TODS)*, 42(2):1–49, 2017.

[19] A. Pasos Ruiz, M. Flynn, J. Large, M. Middlehurst, and A. Bagnall. The great multivariate time series classification bake off: a review and experimental evaluation. *Data Min. and Knowl. Disc.*, Online First, 2020.

[20] P. Schäfer. The BOSS is concerned with time series classification in the presence of noise. *Data Min. and Knowl. Disc.*, 29(6):1505–1530, 2015.

[21] P. Schafer, A. Ermshaus, and U. Leser. "clasp" - time series segmentation. In *proceedings of ACM International Conference on Information and Knowledge Management*, 2021.

[22] C. Tan, C. Bergmeir, F. Petitjean, and G. Webb. Time series extrinsic regression. *ArXiv e-prints*, ArXiv:2006.12672, 2020.

[23] C. Truong, L. Oudre, and N. Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167, 2020.

[24] Liudmila Ulanova, Nurjahan Begum, and Eamonn Keogh. Scalable clustering of time series with u-shapelets. In *Proceedings of the 2015 SIAM international conference on data mining*, pages 900–908. SIAM, 2015.

[25] R. Wu and E. Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *ArXiv e-prints*, ArXiv:2009.13807, 2021.

[26] F. Yger, M. Berar, and F. Lotte. Riemannian approaches in brain-computer interfaces: a review. *EEE Trans.on Neural Sys. and Rehab. Eng.*, 167, 2017.