Data-driven Equation Discovery of Ocean Mesoscale Closures

Thomas Bolton[1], **Laure Zanna**[1,2]

[1]Department of Physics, University of Oxford, Oxford OX1 3PU, United Kingdom.
[2]Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, USA

Key Points:

- We present two machine learning algorithms for ocean mesoscale parameterizations.
- We discover closed-form equations for eddy momentum, temperature and energy parameterizations.
- Deep learning closure is more stable than closed-form equations when implemented in an ocean model.

Corresponding author: Laure Zanna, `laure.zanna@nyu.edu`

Abstract

The resolution of climate models is limited by computational cost. Therefore, we must rely on parameterizations to represent processes occurring below the scale resolved by the models. Here, we focus on parameterizations of ocean mesoscale eddies and employ machine learning (ML), namely relevance vector machines (RVM) and convolutional neural networks (CNN), to derive computationally efficient parameterizations from data, which are interpretable and/or encapsulate physics. In particular, we demonstrate the usefulness of the RVM algorithm to reveal closed-form equations for eddy parameterizations with embedded conservation laws. When implemented in an idealized ocean model, all parameterizations improve the statistics of the coarse-resolution simulation. The CNN is more stable than the RVM such that its skill in reproducing the high-resolution simulation is higher than the other schemes; however, the RVM scheme is interpretable. This work shows the potential for new physics-constrained interpretable ML turbulence parameterizations for use in ocean climate models.

Plain Language Summary

The complexity of numerical models used for future climate projections is limited by their computational cost. Many key processes, such as ocean eddies, are not adequately resolved and must be approximated using parameterizations. However, parameterizations are often imperfect and reduce the accuracy of the simulations. Machine learning is now opening new avenues to improve climate simulations by extracting such parameterizations directly from data, rather than using idealized theories as typically done. We show that efficient modern machine learning algorithms can accurately represent the physics of ocean eddies, be constrained by physical laws, and can be interpretable. Our results simultaneously open the door to the discovery of new physics from data and the improvement of climate simulations.

1 Introduction

Turbulent processes are critical components of the climate system and influence the circulation of both the ocean and atmosphere. For example, ocean mesoscale eddies, which are turbulent features of scale 10-100 km, dominate the oceanic kinetic energy reservoir (Ferrari & Wunsch, 2009) and are key for the lateral and vertical transport of tracers, such as heat, carbon, and oxygen. These turbulent processes occur on scales that are below the resolution of typical global climate models, which is roughly 25 km-100 km (IPCC, 2013). Therefore, the effects of these turbulent processes on the large-scale must be approximated.

These approximations, called parameterizations or closures, are often developed using idealized theories of the bulk effect of the subgrid process on the large scale (Warner, 2010). This approach has been used for many decades but is not necessarily optimal as it neglects certain physical effects. Imperfections in current parameterizations and missing physics in climate models introduce significant biases in simulations and considerable uncertainty in anthropogenic climate change projections (IPCC, 2013). For example, current parameterizations of ocean eddies target the effect of i) buoyancy fluxes by removing large-scale available potential energy (Gent & Mcwilliams, 1990), and ii) momentum fluxes using viscous closures which dissipate momentum (Zanna et al., 2020).

While improving certain properties of the flow (Danabasoglu et al., 1994), eddy parameterizations are missing key energy pathways such as the conversion of available potential energy into subgrid kinetic energy, or the backscatter of kinetic energy to the large-scale flow (Jansen et al., 2015; Zanna et al., 2017; Bachman, 2019). In addition, these parameterizations spuriously dissipate kinetic energy (Jansen & Held, 2014; Kjellsson

& Zanna, 2017). These imperfect representations of ocean eddy physics in models can affect the strength and variability of large-scale ocean currents and ocean heat uptake (Zanna et al., 2017; Kuhlbrodt & Gregory, 2012). Increasing resolution can reduce some of these biases; however, due to the computational expense of running turbulence-resolving simulations, subgrid parameterizations will be in demand for several decades.

Recently, the advent of machine learning (ML) has given rise to a new class of data-driven parameterizations. Studies rely on ML to optimally tune parameters of existing closures (Schneider et al., 2017; Ling et al., 2016). This approach, while useful, still neglects the missing physics not encapsulated in the current parameterizations. Instead, several studies have shown the promise of new ML parameterizations of subgrid processes in the atmosphere (Gentine et al., 2018; Rasp et al., 2018; O'Gorman & Dwyer, 2018; Brenowitz & Bretherton, 2018) and ocean (Bolton & Zanna, 2019). However, this new class of ML parameterizations often uses black-box algorithms (e.g., neural networks) such that the laws of physics are not necessarily respected unless imposed (Beucler et al., 2019; Ling et al., 2016), and interpreting the data-driven parameterization becomes intractable.

Here, we propose a complementary or alternative route to both the traditional physics-driven bulk approach and the ML-black box approach of deep learning. We use ML to discover closed-form equations for mesoscale eddy parameterizations for coarse-resolution ocean models using high-resolution model data. Given some spatio-temporal dataset of the subgrid eddy forcing, we uncover an equation that could have produced that dataset (Rudy et al., 2017; Zhang & Lin, 2018). This approach has the following advantages over more complex methods such as convolutional neural networks: the end result is significantly easier to interpret physically, the computational cost of implementation is lower, and training time of the algorithm is also lower. Data-driven discovery of equations has been extensively used to reveal known-equations, such as Burger's or Navier-Stokes' equations (Kutz, 2017). However, unlike in these studies, we use the algorithm to discover unknown equations for the subgrid eddy forcings.

2 Data and Methods

2.1 Training Data and Coarse-Graining

We use a primitive equation model, MITgcm (J. Marshall et al., 1997), to generate high-resolution data and construct new eddy momentum, temperature and energy parameterizations. We run highly-idealized double-gyre eddy-resolving barotropic and baroclinic simulations in a square-domain. The simulations use a beta-plane approximation, free-slip boundary conditions on lateral walls and no-slip boundary condition on the bottom, and a constant surface zonal wind forcing. These simulations are designed to create highly turbulent flow regimes, with mesoscale eddies shedding from the jet extension.

The barotropic model has a single layer of depth 500 m and length 3840 km, similar to Cooper and Zanna (2015). We spin-up the model from rest for 10 years, at a spatial resolution of 3.75 km. The baroclinic model comprises of 15 vertical levels, with a total depth of 3600 m. Due to the increased computational cost of running the baroclinic simulation compared to the barotropic model, we decreased the domain size from 3840 km in length to 1920 km, with a spatial resolution of 7.5 km. The meridional temperature gradient is imposed via surface restoring to a linear profile. We spin-up the baroclinic model for 100 years and then run for a further 10 years for data collection. Further details about the simulations are given in the Supplementary Information (SI, S1).

After spin-up, we select 1000 time-slices of model output, with 4 days between each time-slice. We remove information at small-scales by applying a horizontal Gaussian filter of width 30 km, and then coarse-grain to a 30 km grid, which is denoted by $\overline{(..)}$ (Bolton

& Zanna, 2019) (SI, S2). The subgrid eddy momentum and temperature forcing terms, for each vertical level, are then defined by

$$\mathbf{S_u} = \begin{pmatrix} S_x \\ S_y \end{pmatrix} = (\overline{\mathbf{u}} \cdot \overline{\nabla})\overline{\mathbf{u}} - \overline{(\mathbf{u} \cdot \nabla)\mathbf{u}}, \tag{1}$$

$$S_T = (\overline{\mathbf{u}} \cdot \overline{\nabla})\overline{T} - \overline{(\mathbf{u} \cdot \nabla)T}, \tag{2}$$

respectively. Here ∇ is the horizontal 2D gradient operator, T is the temperature, and the horizontal velocity $\overline{\mathbf{u}} = (\overline{u}, \overline{v})$. These terms reflect the turbulent nonlinear terms truncated in coarse-resolution models which need to be parameterized (Berloff, 2005; Mana & Zanna, 2014). At every grid-point for every time-slice, we both i) calculate the target eddy forcing, i.e, Eqs. (1) and (2), and ii) construct a library of diverse functions which are necessary for the RVM method described below and are relevant to the process being parameterized.

2.2 Data-Driven Algorithms

Relevance Vector Machine. Here, we employ the sparse Bayesian regression method used in Zhang and Lin (2018) based on relevance vector machines (RVM) (Tipping, 2001) to reveal new eddy parameterizations. RVM is a regression technique that assumes Gaussian prior distributions for each regression weight (Bishop, 2006). The width of the Gaussian prior of each regression weight provides a measure of uncertainty of that regression weight. The method relies on a library of functions, which can comprise of any function such as products or derivatives of relevant quantities defined as basis functions (e.g., velocity shears, temperature shears). The sparse regression is applied iteratively to the library of functions, and then a pruning of the library of functions is carried out by discarding the functions with an uncertainty higher than a pre-specified threshold (Zhang & Lin, 2018). This uncertainty threshold, δ, is the only parameter that requires setting in the Zhang and Lin (2018) method. The algorithm finishes when the uncertainty measures of each regression weight stop changing from iteration to iteration. We found the Zhang and Lin (2018) method to be more robust than the sequential threshold ridge regression (STRidge) of Rudy et al. (2017). For example, using data to discover the known 2D advection-diffusion equations, we found that STRidge required substantially more data for training than the RVM method, STRidge has a large number of tunable hyperparameters which substantially influenced the discovered equation compared to the RVM method which has only one hyperparameter. In addition, unlike STRidge, Zhang and Lin (2018) method provided an error associated with the weights discovered. Given these tests were performed on known equations in which we knew the answers, we opted for the use of Zhang and Lin (2018) RVM method to discover unknown parameterizations.

At every grid-point for every time-slice from the MITgcm coarse-grained output (described above) we construct a library of diverse functions, ϕ_i, which are derived from a set of basis functions relevant to the process being parameterized. We build the library from the filtered velocities $\overline{u}$, $\overline{v}$, and $\overline{T}$ using up to second-order for both spatial derivatives and polynomial products, mainly due to memory limitations. The basis of functions used for the momentum and temperature eddy parameterizations differ and will be discussed in the next section. We normalized each function individually such that they have zero mean and unit variance. We use 50% of the 1000 time-slices for training and the other 50% for validation. For both the eddy momentum and temperature forcing, we impose a physical constraint for global conservation. To do so, we only specify library functions that can be written as the divergence of a flux (or as the divergence of a tensor $\mathbf{T}$ for the eddy momentum forcing, i.e. $\overline{\nabla} \cdot \mathbf{T}$), such that with the appropriate boundary conditions there is no net input of momentum or temperature.

We then apply the iterative RVM algorithm to prune the library of functions and construct the final equation for the subgrid forcing (independently for S_x, S_y and S_T) as a linear sum of the functions, ϕ_i, each weighted by the regression coefficient, w_i. We

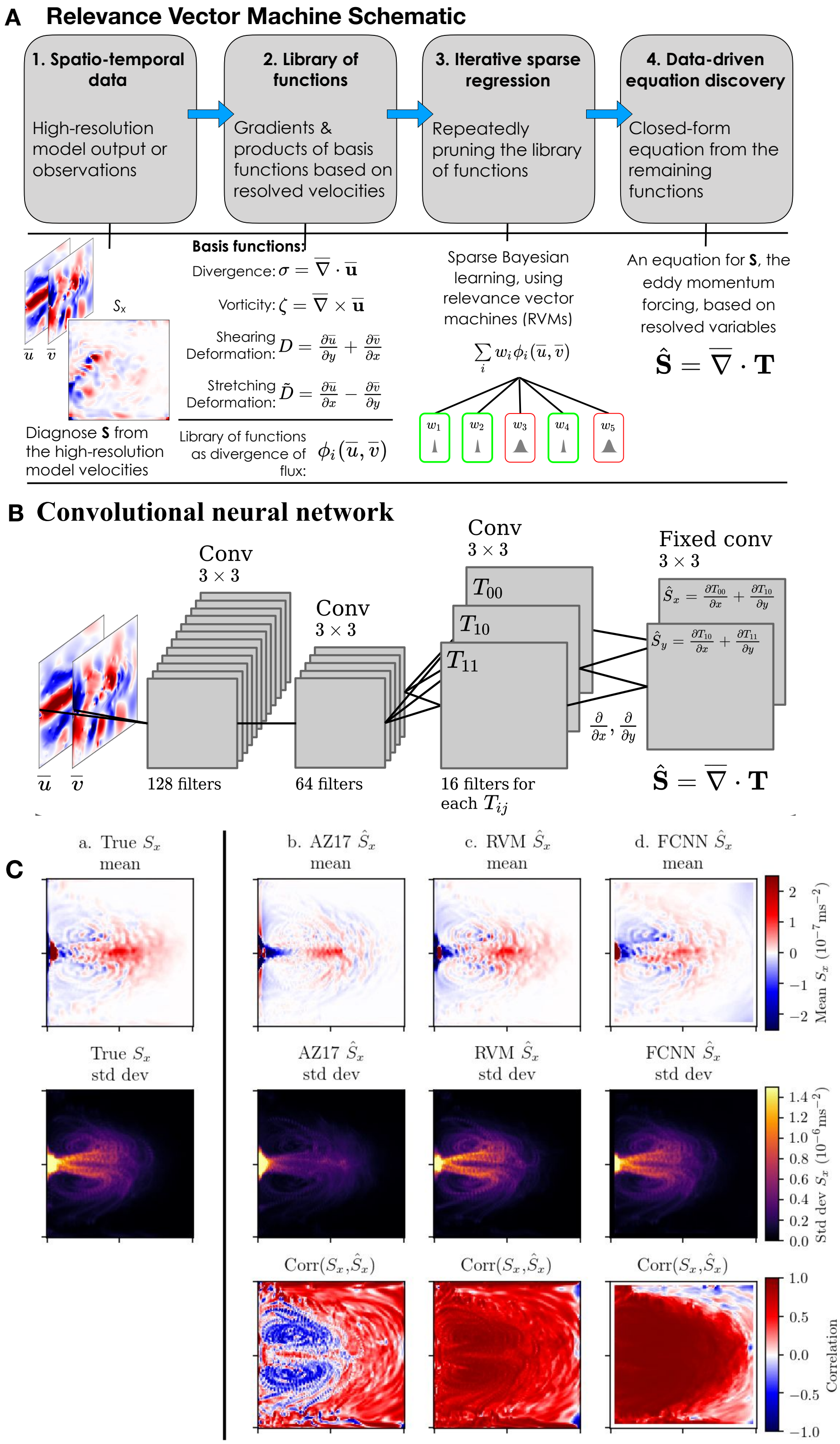


Figure 1. A) Illustration of the RVM procedure; B) Schematic of the architecture of the physics-constrained fully-convolutional neural network (FCNN); C) Offline validation of the subgrid momentum forcing from the barotropic simulations for three parameterizations, denoted as $\hat{\mathbf{S}}$ – the physics-driven $\hat{\mathbf{S}}^{AZ}$, $\hat{\mathbf{S}}^{BT}$ revealed by the RVM algorithm (Eq. 5), and the FCNN – against the diagnosed forcing from high-resolution data, $\mathbf{S}$. Top Row shows the mean $[ms^{-2}]$, Middle Row the Standard Deviation $[ms^{-2}]$, and the Bottom Row the Pearson correlation of the zonal component of the eddy momentum forcing, S_x and $\hat{S}_x$ (the meridional component is shown in SI). The x- and y-axis are longitude and latitude, respectively; the extent is 3840 km in each direction.

estimate the performance of the final equation by calculating the R^2 coefficient of determination using the withheld validation data. The full process of discovery with RVM is illustrated in Fig. 1A. Further details can be found in the SI.

Convolutional neural network. We are using a fully-convolutional neural network (FCNN) on the high-resolution validation data (the truth). The authors have already shown that CNNs are powerful at parameterizing mesoscale eddy momentum forcing and can generalize very well to different regimes, in particular to different dynamical regions and different turbulent regimes (Bolton & Zanna, 2019). Other studies have shown the success of neural network in representing turbulent closures from large-eddy simulations (Maulik & San, 2017; Ling et al., 2016; Wang et al., 2020), though none have been implemented in a forced-dissipative model as of yet. The FCNN used here (Fig. 1B and SI) is trained using the same barotropic model data as for the RVM expression, with the velocity components, $\overline{u}$ and $\overline{v}$ as inputs. There are four convolution layers simultaneously predicting both components of the eddy momentum forcing. The architecture of the FCNN is physically-constrained (Beucler et al., 2019) such that the activation maps (i.e. the results) of the third convolution layer represent the elements of a symmetric eddy stress tensor $\mathbf{T}$. The final convolution layer then takes the spatial derivatives of the eddy stress tensor elements, using fixed filters representing central-difference stencils, forming predictions S_x and S_y. By physically-constraining the architecture to form the elements of a symmetric eddy stress tensor, global momentum and vorticity conservation can be achieved. The hyperparameters of the architecture, such as the number of convolution layers and the number of filters, were chosen by experimenting with numerous configurations and examining the impact of the R^2 coefficient on the validation data, as commonly done. We do not use bias parameters in any of the convolution layers. The details of the FCNN architecture are in the SI for full reproducibility of the results.

2.3 Numerical Model for Implementation

The RVM and FCNN parameterizations are implemented in an idealized ocean model. Implementation of the FCNN into a Fortran code (e.g., MITgcm) is non-trivial, therefore we opt to implement the parameterizations using Python since it was used to train and save the FCNN. The Python-based idealized ocean model is a shallow water model, which bears many resemblances to the MITgcm primitive equation barotropic model, including the horizontal velocities and sea surface height as prognostic variables, a double-gyre configuration with a constant wind forcing, and an idealized bathymetry. The parameterizations are implemented into a 30 km resolution version of the idealized shallow-water Python model, which was span-up from rest for 10 years, and then run for an additional 10 years for analysis. Further details are available in the SI.

3 Data-Driven Equation-Discovery for Mesoscale Eddies

Improved parameterizations of mesoscale eddy momentum, temperature and energy are crucial to improving the transport of tracers, as well as countering the energy deficit caused by scale-truncation, and viscous and diffusion parameterizations within coarser-resolution models. To derive new data-driven closures, we use the data generated from idealized eddy-resolving barotropic and baroclinic simulations, with horizontal resolutions of 3.5 km and 7.5 km respectively, which emulate western boundary currents and their jet extensions at mid-latitudes (Methods). Our target is to parameterize eddy momentum (sec. 3.1) and temperature fluxes, and an eddy prognostic equation (sec. 3.2) for coarser-resolution models, here chosen to be of 30 km horizontal resolution (eddy-permitting), similar to CMIP-class eddy permitting models. We will extract the subgrid forcing using the RVM algorithm.

3.1 Discovering Eddy Momentum Parameterizations

For constructing the library of functions to reveal expressions for the eddy momentum forcing, we write the spatial derivatives of the velocity field using the following basis functions

$$\zeta = \overline{v}_x - \overline{u}_y, \qquad \sigma = \overline{u}_x + \overline{v}_y, \tag{3a}$$
$$D = \overline{u}_y + \overline{v}_x, \qquad \tilde{D} = \overline{u}_x - \overline{v}_y, \tag{3b}$$

where the short-hands $()_{x,y} \equiv \frac{\partial}{\partial x,y}$ are used for spatial derivatives, ζ is the relative vorticity, σ is the divergence, and D and $\tilde{D}$ are the shearing and stretching deformation of the flow field, respectively. We chose to write the library of functions using this basis because i) initially our data-driven discovery method was automatically forming Eq. 3a-b, when given only velocity components and their derivatives, without a priori knowledge, and ii) the dynamical quantities defined by Eq. 3 are relevant to turbulent eddy parameterizations (Smagorinsky, 1963; Pope, 1975). The RVM algorithm, therefore, revealed an improved basis in which to write the library of functions. Finding the optimal physical basis is important to identify the key dynamical components from which to construct parameterizations in general, as well as helping with physics-discovery from data.

We separately apply the RVM algorithm to data from the barotropic and baroclinic model. The predicted subgrid momentum forcing is denoted by $\hat{\mathbf{S}}_{\mathbf{u}} = \left(\hat{S}_x, \hat{S}_y\right)$. We performed an extensive sensitivity analysis to the sole hyperparameter, the threshold δ, of the method (SI, S5). At low threshold values, the RVM algorithm selects a single function, namely the gradients of enstrophy $(\zeta^2)_x$ and $(\zeta^2)_y$ for predictions of S_x and S_y, respectively, which captures $\sim$20% of the variance. As the pruning threshold increases, there is a large increase from $\sim$20% to $\sim$50% variance captured, with the number of functions only increasing from 1 to 3 for both S_x and S_y. The expression revealed by the RVM is then given by

$$\hat{\mathbf{S}}_{\mathbf{u}}^{BT} = \begin{pmatrix} w_0(\zeta^2)_x - w_1(\zeta D)_x + w_2(\zeta\tilde{D})_y \\ w_3(\zeta^2)_y + w_4(\zeta D)_y + w_5(\zeta\tilde{D})_x \end{pmatrix}, \tag{4}$$

where $w_0 = -4.096 \times 10^8$, $w_1 = -5.483 \times 10^8$, $w_2 = -4.384 \times 10^8$, $w_3 = -4.100 \times 10^8$, $w_4 = -6.332 \times 10^8$, $w_5 = -4.815 \times 10^8$, with units of m^2. Each coefficient has an uncertainty estimate which is on the order of a few percent, and never exceeds 10%. The uncertainty associated with each weigth is not listed as it is always smaller than the coefficient of variation used (see below) for each parameterization discovered. The zonal and meridional components of the predicted RVM expression capture 55.6% and 50.6% of the variance, respectively. Adding six more functions would increase the R^2 value to up to 80% but increasing the complexity of the expression (SI, Eqs. 12-13).

To quantify the differences between the regression coefficients, we use the coefficient of variation (i.e. relative standard deviation), which provides a standardized measure of the dispersion of a probability distribution. For the regression coefficients w_i above, the coefficient of variation is 14.2%. We therefore decide to write the regression coefficients as approximately equal, i.e., $w_i \approx \kappa_{BT} = -4.87 \times 10^8\ \mathrm{m}^2$, with an average error of 14.2%. Using this approximation, we can then re-write Eq. 4 as

$$\hat{\mathbf{S}}_{\mathbf{u}}^{BT} \approx \kappa_{BT} \overline{\nabla} \cdot \begin{pmatrix} \zeta^2 - \zeta D & \zeta\tilde{D} \\ \zeta\tilde{D} & \zeta^2 + \zeta D \end{pmatrix}. \tag{5}$$

The expression now has a single scalar as a tunable parameter, κ_{BT}, which determines the ‘strength’ of the parameterization. The expression depends only on the spatial derivatives of the vorticity and deformation terms, and is similar to the parameterization developed by Anstey and Zanna (2017) (see below). In addition, the tensor found is symmetric, despite separately applying the RVM algorithm to the zonal and meridional components of the eddy momentum forcing and without imposing symmetry as a constraint (unlike for the FCNN).

We perform the same procedure using data from the baroclinic model. We provide the RVM algorithm with data from multiple vertical layers at once. As for the barotropic model, a significant increase in the R^2 occurs when three functions are retained, capturing over 40% of the variance. Here, the RVM algorithm constructs the same eddy momentum forcing from the barotropic model (Eq. 4), albeit with different values for the regression coefficients. A second increase in the R^2 occurs for larger values of the threshold parameter where five functions are retained, capturing approximately 70% of the variance (Eq. 5 in SI). We proceed to calculate the average of the regression coefficients and found a mean value $\kappa_{BC} = -8.723 \times 10^8$ m^2, with a coefficient of variation of 9.8%. Due to the relatively low coefficient of variation, we again assume that all regression coefficients are approximately equal to κ_{BC}, such that the RVM expression can be approximated as

$$\hat{\mathbf{S}}_{\mathbf{u}}^{BC} \approx \kappa_{BC}\overline{\nabla} \cdot \begin{pmatrix} -\zeta D & \zeta\tilde{D} \\ \zeta\tilde{D} & \zeta D \end{pmatrix} + \mathbf{I}\frac{1}{2}\kappa_{BC}\overline{\nabla}(\zeta^2 + D^2 + \tilde{D}^2), \tag{6}$$

for each vertical layer. The baroclinic expression depends only on the spatial derivatives of the shearing deformation, the stretching deformation, and the vorticity. Like the barotropic expression, the tensor is symmetric. The baroclinic expression can be written as the barotropic expression plus the gradient of the squared deformation terms: $\hat{\mathbf{S}}_{\mathbf{u}}^{BC} = 2\hat{\mathbf{S}}_{\mathbf{u}}^{BT} + \mathbf{I}\frac{1}{2}\kappa_{BC}\overline{\nabla}(D^2 + \tilde{D}^2)$.

For the physical interpretation of the discovered parameterizations, we rely on previous studies (Pope, 1975; Meneveau & Katz, 2000; Nadiga, 2008; Mana & Zanna, 2014; Anstey & Zanna, 2017). The RVM expressions discovered encapsulate the tensor form that a Reynolds stress could take assuming frame invariance and symmetry in a 2D flow based on Pope (1975). However, not surprisingly, the RVM did not discover the standard viscous stress tensor (also proposed in Pope (1975) framework), given that we are mainly learning quasi-geostrophic effects rather than 3D turbulence. Both the expressions for $\hat{\mathbf{S}}_{\mathbf{u}}^{BT}$ and for $\hat{\mathbf{S}}_{\mathbf{u}}^{BC}$ contain within them the recently proposed deformation-based momentum parameterization of Anstey and Zanna (2017), referred to as AZ17, and defined by

$$\hat{\mathbf{S}}_{\mathbf{u}}^{AZ17} = \kappa_{AZ17}\overline{\nabla} \cdot \begin{pmatrix} -\zeta D & \zeta\tilde{D} \\ \zeta\tilde{D} & \zeta D \end{pmatrix}, \tag{7}$$

therefore, $\hat{\mathbf{S}}_{\mathbf{u}}^{BT} = \mathbf{S}_{\mathbf{u}}^{AZ17} + \kappa_{BT}\overline{\nabla}\zeta^2$. AZ17 is also related to the Pope (1975) tensors, see further discussion in AZ17. Using data from a MITgcm baroclinic simulations, AZ17 diagnosed a value of κ_{AZ17} and found it to be on the order of -5×10^8 m^2, similar to the value of κ_{BC}. AZ17 is known to capture up-gradient momentum fluxes, and to conserve kinetic energy. The parameterizations, $\mathbf{S}^{AZ17}$, $\hat{\mathbf{S}}_{\mathbf{u}}^{BT}$ and $\hat{\mathbf{S}}_{\mathbf{u}}^{BC}$ can be related the non-linear gradient model (Meneveau & Katz, 2000; Nadiga, 2008), though comprising of additional terms (AZ17). The non-linear gradient model, which is derived as a Taylor expansion of the filtered nonlinear stresses, has shown promise in a range turbulent flows applications. This class of parameterizations, based on the deformation tensor of Pope (1975), has also been shown to generalize to different dynamical regimes and scales within a range of eddying resolution (AZ17; Mana & Zanna, 2014; Zanna et al., 2017). The vorticity-contribution of each $\hat{\mathbf{S}}_{\mathbf{u}}^{BT}$ and $\hat{\mathbf{S}}_{\mathbf{u}}^{BC}$ is identical to that of $\hat{\mathbf{S}}_{\mathbf{u}}^{AZ17}$ (SI, S4). However, $\hat{\mathbf{S}}_{\mathbf{u}}^{BT}$ and $\hat{\mathbf{S}}_{\mathbf{u}}^{BC}$ lead to a net source or sink of kinetic energy, which depends on the divergence of the flow (or the potential energy of the system; Eq. 11, SI). Therefore, the RVM expressions capture processes not included in currently-implemented eddy parameterizations and have revealed new parameterizations for energy pathways between reservoirs.

Before implementing the parameterizations in an ocean model, we test their performance offline with the validation data within the barotropic model (Fig. 1C). We compare $\hat{\mathbf{S}}_{\mathbf{u}}^{BT}$, $\hat{\mathbf{S}}_{\mathbf{u}}^{AZ}$, and the FCNN trained using the same barotropic model data as for the RVM expression, with the velocity components, $\overline{u}$ and $\overline{v}$ as inputs (Fig. 1B and SI).

In the time-mean maps of $\hat{S}_x$ (Fig. 1C, top row), the RVM expression most accurately captures the spatial patterns of the high-resolution model. The FCNN also captures the majority of the spatial patterns of the true time-mean but exhibits a negative bias in the eastern part of the domain. The AZ17 parameterization loosely captures the negative values near the western boundary and positive values in the interior, but struggles to capture the finer small-scale patterns of the true time-mean. Similar results hold for the standard deviation (middle row): the RVM expression and FCNN reproduce the true standard deviation almost exactly, with differences only visible close to the western boundary. Whereas the AZ17 standard deviation underestimates the true standard deviation by 50% in the ocean interior. The higher-order moments, skewness and kurtosis (SI, S8), are also best captured by the RVM expression and FCNN, which outperform the AZ17 expression. In terms of predictive skill, measured by the correlation between the parameterized term and the true subgrid forcing (bottom row), the FCNN captures almost all of the variance in the vicinity of the jet, but this high skill is not consistent across the domain, particularly near the eastern boundary. The predictive skill of the RVM expression is not as high as the FCNN within the jet region, but is significantly more consistent across the domain, with fewer patches of zero or negative correlation. AZ17 performs poorly in a significant part of the domain. The amount of data for training the RVM could be reduced by half without deteriorating the results, this is not the case for the FCNN. Performance of the baroclinic momentum expression from RVM can be found in SI. Overall, the ML parameterizations perform well in offline validation, compared to a physics-based scheme.

3.2 Discovering Eddy Temperature and Energy Forcing

We apply the same procedure to find the eddy temperature forcing, defined by Eq. 2 as a flux, using data from the baroclinic model. The basis functions for the eddy temperature forcing are based on derivatives of momentum and temperature. For a given threshold parameter, the R^2 reaches 54.3% with only 4 functions, resulting in the following expression for the predicted subgrid temperature forcing:

$$\hat{S}_T = w_0(\overline{u}_x\overline{u}_z)_y + w_1(\overline{v}_x\overline{v}_z)_y - w_2(\overline{u}_y\overline{u}_z)_x - w_3(\overline{v}_y\overline{v}_z)_x, \tag{8}$$

with the following values for the regression coefficients $w_0 = 1.573$, $w_1 = 1.495$, $w_2 = 1.518$, $w_3 = 1.504$, which have units of 10^8 Cms. The mean coefficient value is $\kappa_T = 1.523\times10^8$ Cms with a coefficient of variation of 1.7%. Approximating all the regression coefficients as being equal to the mean, with an average error of 1.7%, yields the following expression

$$\hat{S}_T \quad = \quad \kappa_T\overline{\nabla}\cdot\begin{pmatrix}-\overline{u}_y\overline{u}_z - \overline{v}_y\overline{v}_z\\ \overline{u}_x\overline{u}_z + \overline{v}_x\overline{v}_z\end{pmatrix}. \tag{9}$$

The zonally-averaged offline diagnostics for the upper ocean, below the mixed-layer, show that the RVM expression, $\hat{S}_T$, captures the pattern of the mean and standard deviation of the true S_T, however, it underestimates the variance by approximately 50% (Fig. 2). The correlation between $\hat{S}_T$ (the prediction) and S_T (the true forcing) is vertically uniform with a value of 0.6. However, near the northern boundary of the domain, the RVM does not capture the pattern nor the amplitude of the true S_T.

The revealed expression is tied to vertical variations in velocity, which is a reflection of the eddy heat fluxes impacting the density field. The dependence of Eq. 9 on vertical variability can be examined by assuming that thermal wind balance holds for the

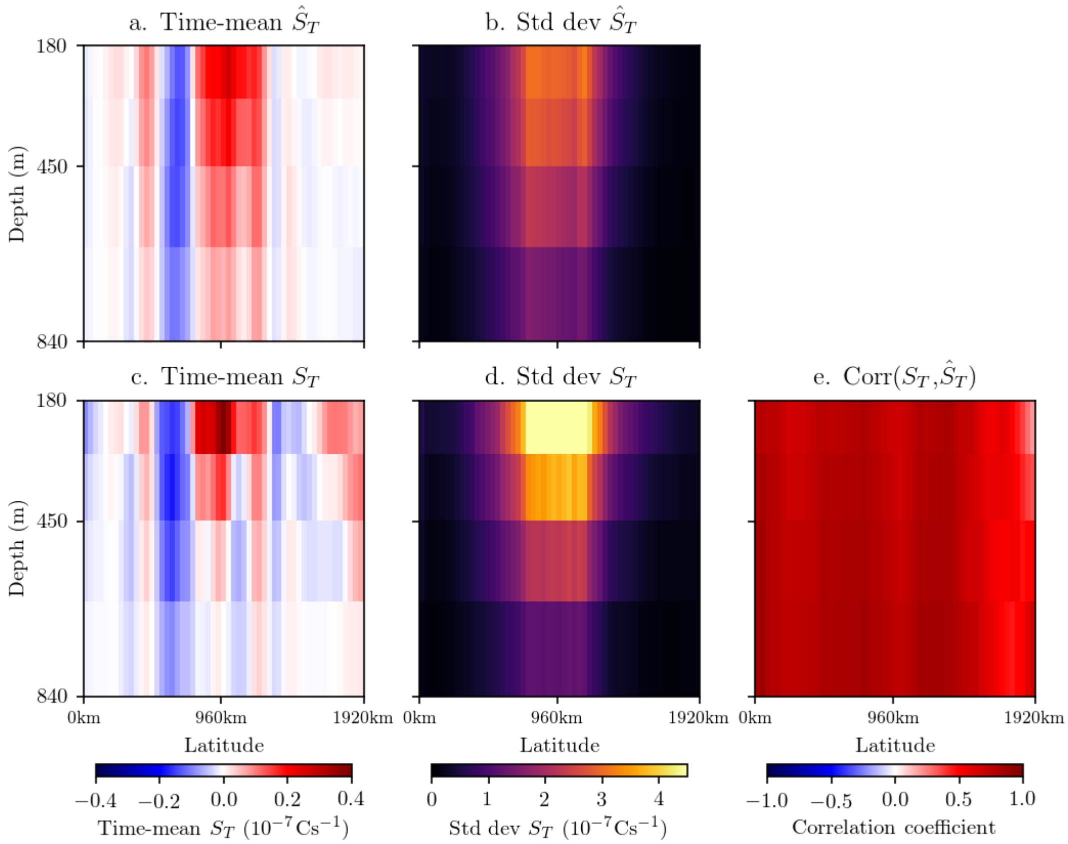


Figure 2. Validation, using the baroclinic model data, of the zonally-averaged predicted, $\hat{S}_T$ (Eq. 9; panels c, d); against the diagnosed eddy temperature forcing, S_T (Eq. 2; panels a, b), as a function of latitude and depth for the mean and standard deviation. Correlation between the prediction and the diagnosed forcing (panel e).

mesoscale variability. Using a linear equation of state, we can rewrite Eq. 9 as

$$\hat{S}_T \quad = \quad -\frac{\kappa_T g\alpha}{f}\overline{\nabla}\cdot\Big[\begin{pmatrix}\overline{v}_y & -\overline{u}_y\\ -\overline{v}_x & \overline{u}_x\end{pmatrix}\overline{\nabla}\;\overline{T}\Big], \tag{10}$$

where α is the thermal expansion coefficient, g is gravity, f is the Coriolis parameter. The coefficient $\kappa_T g\alpha/f$ has units of m^2, similarly to the coefficient for the momentum parameterization. The eddy temperature flux is now dependent on the lateral temperature gradient, modulated by lateral velocity gradients. We can further reformulate the predicted eddy temperature forcing, using the residual-mean formulation (Ferrari & Plumb, 2003; D. P. Marshall et al., 2012; Greatbatch & Lamb, 1990), into a vertical flux of horizontal momentum with a magnitude that depends on the velocity gradient (Eq. 15, SI) – the flux can be up- or down-gradient.

To further improve the energetics of the model, an additional prognostic equation for the eddy energy can be solved to account for all sources and sinks of energy within the system. However, the prognostic eddy energy equation is unknown and must therefore be constructed (Cessi, 2008; Eden & Greatbatch, 2008; D. P. Marshall & Adcroft, 2010; Jansen et al., 2015; Mak et al., 2016). For both the barotropic and baroclinic models, the RVM algorithm constructs a prognostic equation which is the advection of eddy kinetic energy (EKE), and captures 50-60% of the variance in the validation data (SI, S7). Changing the pruning threshold, the target equation, or the spatial-scale of the Gaussian filter for defining the eddy scale, did not modify the equation revealed by the algorithm.

4 Implementation into a Coarse-Resolution Ocean Model

Online performance, meaning when the parameterizations are coupled to a coarser-resolution model, is an important test for future implementation in global climate models. A key issue of any parameterizations is that diagnostic (offline) performance does not translate into prognostic (online) performance due to both the underlying model structure to be integrated forward (e.g., subgrid parameters, numerics) and the nonlinear nature of the equation of motions, in which the parameterizations continuously interact with the resolved scales. Here, the physics-driven parameterization from AZ17, $\hat{\mathbf{S}}^{AZ17}$, and the data-driven barotropic momentum expression (Eq. 5) revealed by the RVM, and the data-driven FCNN are implemented into a 30 km resolution version of a very idealized shallow-water model (Methods and SI). It is the first time that a CNN parameterization for ocean turbulence is implemented into an ocean model, therefore for easy implementation and testing we chose a model coded in Python. Choosing a model that is different than the model used for learning provides also a stronger (and more difficult) test for the success of the parameterizations. For all three parameterizations, conservation of global momentum and vorticity are satisfied. The goal of the parameterizations is to reduce model biases and in particular energize the flow, to replace the energy lost due to truncation of small-scales and large viscosity coefficients at coarse resolution.

We compare the 30 km-parameterized simulations, with the 30 km-simulation without parameterization and a 3.75 km high-resolution (the truth). We initially set the same parameter for both the RVM and AZ17 expressions to $\kappa = -4.87\times10^8$ m^2. However, this implementation led to issues of numerical stability for both the RVM and AZ17 parameterizations, while to the implementation of the FCNN led to over-energized flow, with an efficient inverse cascade and velocities reaching large values of O(10 m s^{-1}). To alleviate these issues, we attenuate the strength of each parameterization, i.e., at each time-step we simply multiply $\hat{\mathbf{S}}_{\mathbf{u}}$ by a coefficient τ between 0 and 1. Through trial and error, we use values of τ of 0.5, 0.5, and 0.7 for the RVM, AZ17, and FCNN parameterizations, respectively (SI, S8).

All three parameterizations increase the amount of kinetic energy in the model (Fig. 3A). Both the RVM and AZ17 expressions increase the kinetic energy to approximately halfway

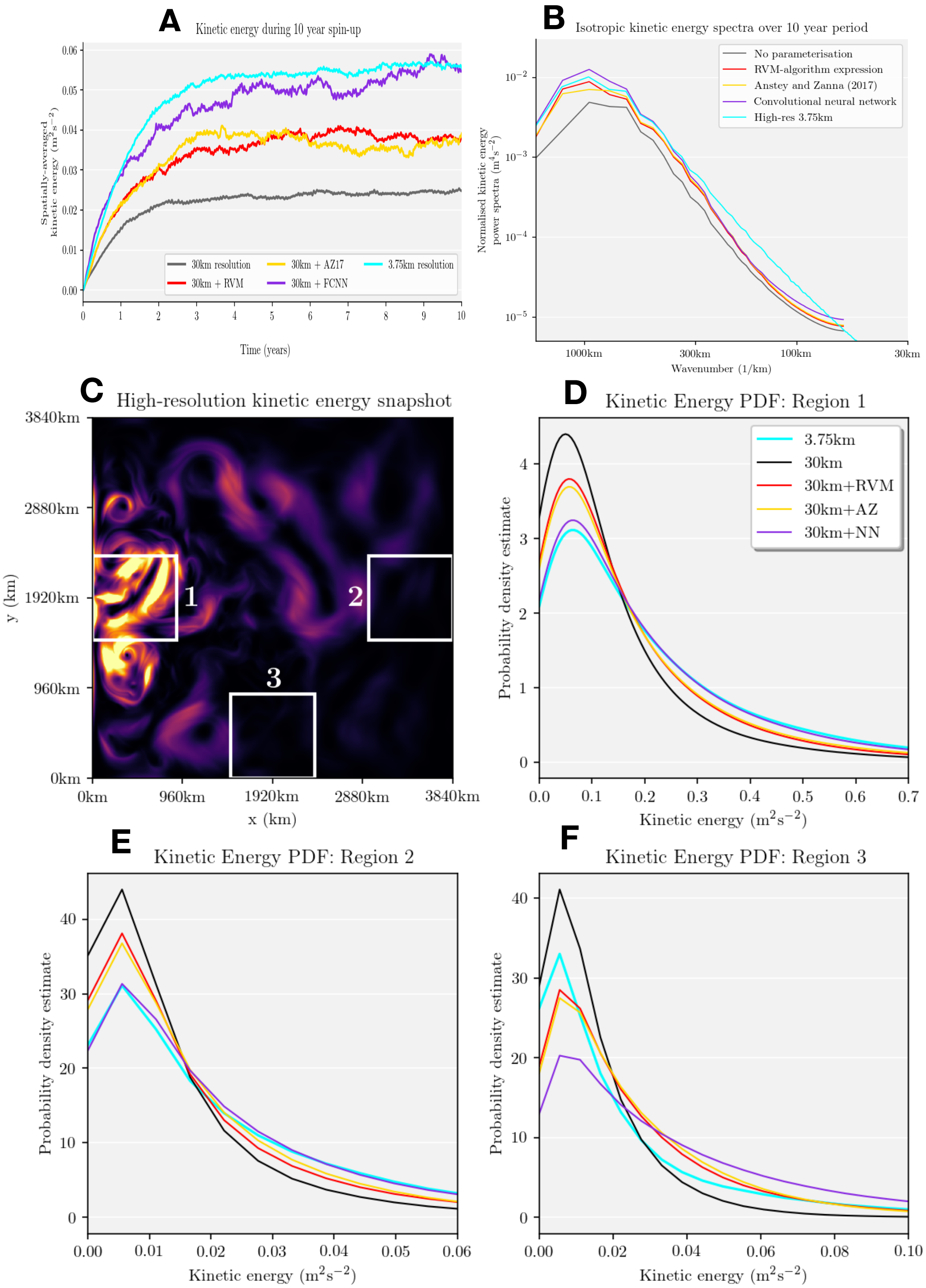


Figure 3. Kinetic energy diagnostics in the following idealized ocean simulation: high-resolution, 3.75 km (cyan), coarse-resolution 30 km without parameterizations (grey), coarse-resolution 30 km with FCNN (purple), coarse-resolution 30 km with RVM (red), and coarse-resolution 30 km with AZ17 (yellow). A) Time series of globally-averaged kinetic energy as a function of time; B) Kinetic Energy Spectrum as a function of wavenumber; C) Snapshot of kinetic energy in the high-resolution simulations, indicated three regions of interest (1-3) for extreme event diagnostics using probability distribution functions (PDF). E-D) PDF of kinetic energy for Regions 1-3.

between the 30 km and 3.75 km models, at a value of 0.038 m^2s^{-2}. It is not surprising that the RVM and AZ17 parameterizations lead to similar results in a shallow-water barotropic model, as their contributions to the vorticity budget are identical (SI, S4). The FCNN parameterizations increase the kinetic energy of the model to within approximately 5% of the high-resolution model at 0.056 m^2s^{-2}.

The kinetic energy power spectrum (Fig. 3B) shows evidence of increased kinetic energy for the parameterized simulations at all spatial scales, compared to the low-resolution unparameterized simulation. At spatial-scales larger than 300-400 km, all parameterizations increase the kinetic energy to approximately the same level as the high-resolution simulation, therefore implying a more efficient backscatter or inverse energy cascade. The FCNN parameterization increases the kinetic energy to above that of the high-resolution model. At length-scales smaller than 300 km, while all parameterizations increase the kinetic energy, it remains lower than that of the high-resolution simulation, likely due to viscosity.

In addition to the global mean kinetic energy, we consider the impact of the parameterizations on the statistics and extremes in kinetic energy. In the three representative regions selected (Fig. 3C), the high-resolution probability density function (PDF) has more probability in the tails compared to the 30 km model without parameterization (Fig. 3D-F). The effect of all parameterizations is to increase the probability in the tails, with little shift in the position of the peak. Therefore, the primary effect of the parameterizations is increasing the frequency of extreme kinetic energy values, as opposed to solely increasing the mean kinetic energy. In regions 1 and 2 (Fig. 3D-E), the FCNN is the best performing, with the kinetic energy PDF of the FCNN parameterization almost indistinguishable from the high-resolution model. AZ17 and the RVM expressions are almost indistinguishable from each other. However, in region 3 (Fig. 3F), all three parameterizations cause too much probability to be redistributed in the tails, as evident by the peaks of the RVM, AZ17, and FCNN kinetic energy PDFs all being below the high-resolution peak.

5 Summary

Machine learning algorithms can facilitate the discovery of physical processes, embedded within data from high-resolution simulations or observations. However, physical intuition remains critical to explain the physics discovered by these algorithms. We have introduced the data-driven equation discovery method of Zhang and Lin (2018), namely the RVM algorithm, for ocean eddy parameterizations, rather than for discovering fundamental equations of motions already known (Rudy et al., 2017). The mathematical expressions discovered by the RVM algorithm show that eddy momentum parameterizations should include up-gradient momentum fluxes and potentially a transfer between potential energy and kinetic energy. In addition, the RVM revealed that eddy temperature fluxes can act on vertical gradients of horizontal momentum with a magnitude that depends on the velocity gradient, and that eddy energy advection accounts for half of the time tendency of eddy kinetic energy. A CNN, constrained with physical conservation laws, appears to be an excellent representation of the eddy momentum forcing, leading to vastly improved coarser-resolution simulations which, under certain metrics, are indistinguishable from the high-resolution target, confirming results from Bolton and Zanna (2019). Yet, the reasons for the success of the CNN parameterization are difficult to extract. All parameterizations presented here have been shown to generalize well to other regimes (e.g., dynamical regions, Reynolds numbers or resolution; Pope, 1975; Mana & Zanna, 2014; Anstey & Zanna, 2017; Holm & Wingate, 2005; Bolton & Zanna, 2019). Unfortunately, the parameterizations, presented here are also subject to tuning when implemented in an ocean models, as all parameterizations in use in current climate models are. The parameterizations, when implemented in a very idealized model, did not vastly improved the mean state (SI, Figs. S10-11) , but tests in more complex models have showed that they have the ability to do so (Zanna et al., 2017). Here, the RVM (and the physics-based) expression, which performs well offline, does not show as good performance as the FCNN online due to numerical instabilities developing during the implementation. This result suggests that the complexity of a deep neural network may be more numerically stable compared to implementing a closed-form equation (Rasp et al., 2018), yet it is subject to heavy tuning (Brenowitz & Bretherton, 2019). However, we cannot rule out that improving the RVM expression by adding more functions, or by adding memory or stochasticity, which have been shown to drastically improve stability (Zanna et al., 2017) or finally by coupling the momentum parametrization to an eddy energy equation (Jansen & Held, 2014).

While the implementation of eddy forcing remains to be properly tested in more complex models, our results suggest that progress can be made using ML for physics discovery and interpretable parameterizations, which are more computationally efficient than running high-resolution simulations (Fig. S12). We hope that this manuscript provide a new road map for data-driven parameterizations to be developed, tested, interpreted, and implemented in ocean climate models in the future. A new strategy, which combines the interpretability of equation discovery with the predictive skill of complex neural networks, could be an effective approach to improving ocean models, and perhaps climate models in general.

Acknowledgments

We thank Alistair Adcroft for insightful discussions, and Malte Jansen, Redouane Lguensat, and an anonymous reviewer for their helpful comments which helped improve the manuscript. The work was partially supported by NERC, NSF-GEO 1912357 and NOAA CVP NA19OAR4310364. The code for the relevance vector machine algorithm can be found at `github.com/TomBolton/rvm-find`, `doi:10.5281/zenodo.3758659`.

References

Anstey, J. A., & Zanna, L. (2017). A deformation-based parametrization of ocean mesoscale eddy reynolds stresses. *Ocean Modelling*, *112*, 99–111.

Bachman, S. D. (2019). The gm+ e closure: A framework for coupling backscatter with the gent and mcwilliams parameterization. *Ocean Modelling*, *136*, 85–106.

Berloff, P. S. (2005). Random-forcing model of the mesoscale oceanic eddies. *Journal of Fluid Mechanics*, *529*, 71–95.

Beucler, T., Pritchard, M., Rasp, S., Gentine, P., Ott, J., & Baldi, P. (2019). Enforcing analytic constraints in neural-networks emulating physical systems. *arXiv preprint arXiv:1909.00912*.

Bishop, C. M. (2006). *Pattern recognition and machine learning.* springer.

Bolton, T., & Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, *11*(1), 376–399.

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. *Geophysical Research Letters*.

Brenowitz, N. D., & Bretherton, C. S. (2019). Spatially extended tests of a neural network parametrization trained by coarse-graining. *arXiv preprint arXiv:1904.03327*.

Cessi, P. (2008). An energy-constrained parameterization of eddy buoyancy flux. *Journal of Physical Oceanography*, *38*(8), 1807–1819.

Cooper, F. C., & Zanna, L. (2015). Optimisation of an idealised ocean model, stochastic parameterisation of sub-grid eddies. *Ocean Modelling*, *88*, 38–53.

Danabasoglu, G., McWilliams, J. C., & Gent, P. R. (1994). The role of mesoscale tracer transports in the global ocean circulation. *Science*, *264*(5162), 1123–

1126.

Eden, C., & Greatbatch, R. J. (2008). Towards a mesoscale eddy closure. *Ocean Modelling*, *20*(3), 223–239.

Ferrari, R., & Plumb, R. (2003). Residual circulation in the ocean. In *Near-boundary processes and their parameterization: Proc.aha hulikoa hawaiian winter workshop* (pp. 219–228).

Ferrari, R., & Wunsch, C. (2009). Ocean circulation kinetic energy: Reservoirs, sources, and sinks. *Annual Review of Fluid Mechanics*, *41*.

Gent, P. R., & Mcwilliams, J. C. (1990). Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography*, *20*(1), 150–155.

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could machine learning break the convection parameterization deadlock? *Geophysical Research Letters*.

Greatbatch, R. J., & Lamb, K. G. (1990). On parameterizing vertical mixing of momentum in non-eddy resolving ocean models. *Journal of Physical Oceanography*, *20*(10), 1634–1637.

Holm, D. D., & Wingate, B. A. (2005). Baroclinic instabilities of the two-layer quasigeostrophic alpha model. *Journal of physical oceanography*, *35*(7), 1287–1296.

IPCC. (2013). Summary for policymakers [Book Section]. In T. Stocker et al. (Eds.), *Climate change 2013: The physical science basis. contribution of working group i to the fifth assessment report of the intergovernmental panel on climate change* (p. 130). Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press. Retrieved from `www.climatechange2013.org` doi: 10.1017/CBO9781107415324.004

Jansen, M. F., & Held, I. M. (2014). Parameterizing subgrid-scale eddy effects using energetically consistent backscatter. *Ocean Modelling*, *80*, 36–48.

Jansen, M. F., Held, I. M., Adcroft, A., & Hallberg, R. (2015). Energy budget-based backscatter in an eddy permitting primitive equation model. *Ocean Modelling*, *94*, 15–26.

Kjellsson, J., & Zanna, L. (2017). The impact of horizontal resolution on energy transfers in global ocean models. *Fluids*, *2*(3), 45.

Kuhlbrodt, T., & Gregory, J. M. (2012). Ocean heat uptake and its consequences for the magnitude of sea level rise and climate change. *Geophysical Research Letters*, *39*(18). Retrieved from `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2012GL052952` doi: 10.1029/2012GL052952

Kutz, J. N. (2017). Deep learning in fluid dynamics. *Journal of Fluid Mechanics*, *814*, 1–4.

Ling, J., Kurzawski, A., & Templeton, J. (2016). Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, *807*, 155–166.

Mak, J., Maddison, J. R., & Marshall, D. P. (2016). A new gauge-invariant method for diagnosing eddy diffusivities. *Ocean Modelling*, *104*, 252–268.

Mana, P. P., & Zanna, L. (2014). Toward a stochastic parameterization of ocean mesoscale eddies. *Ocean Modelling*, *79*, 1–20.

Marshall, D. P., & Adcroft, A. (2010). Parameterization of ocean eddies: Potential vorticity mixing, energetics and arnolds first stability theorem. *Ocean Modelling*, *32*(3-4), 188–204.

Marshall, D. P., Maddison, J. R., & Berloff, P. S. (2012). A framework for parameterizing eddy potential vorticity fluxes. *Journal of Physical Oceanography*, *42*(4), 539–557.

Marshall, J., Adcroft, A., Hill, C., Perelman, L., & Heisey, C. (1997). A finite-volume, incompressible navier stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research: Oceans*, *102*(C3), 5753–5766.

Maulik, R., & San, O. (2017). A neural network approach for the blind deconvolution of turbulent flows. *Journal of Fluid Mechanics*, *831*, 151–181.

Meneveau, C., & Katz, J. (2000). Scale-invariance and turbulence models for large-eddy simulation. *Annual Review of Fluid Mechanics*, *32*(1), 1-32. doi: 10.1146/annurev.fluid.32.1.1

Nadiga, B. (2008). Orientation of eddy fluxes in geostrophic turbulence. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1875), 2489–2508.

O'Gorman, P. A., & Dwyer, J. G. (2018). Using machine learning to parameterize moist convection: potential for modeling of climate, climate change and extreme events. *Journal of Advances in Modelling Earth Systems*.

Pope, S. (1975). A more general effective-viscosity hypothesis. *Journal of Fluid Mechanics*, *72*(2), 331–340.

Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent sub-grid processes in climate models. *arXiv preprint arXiv:1806.04731*.

Rudy, S. H., Brunton, S. L., Proctor, J. L., & Kutz, J. N. (2017). Data-driven discovery of partial differential equations. *Science Advances*, *3*(4), e1602614.

Schneider, T., Lan, S., Stuart, A., & Teixeira, J. (2017). Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters*, *44*(24), 12–396.

Smagorinsky, J. (1963). General circulation experiments with the primitive equations: I. the basic experiment. *Monthly weather review*, *91*(3), 99–164.

Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of machine learning research*, *1*(Jun), 211–244.

Wang, R., Kashinath, K., Mustafa, M., Albert, A., & Yu, R. (2020). *Towards physics-informed deep learning for turbulent flow prediction.* Retrieved from `https://openreview.net/forum?id=Hkg5lAEtvS`

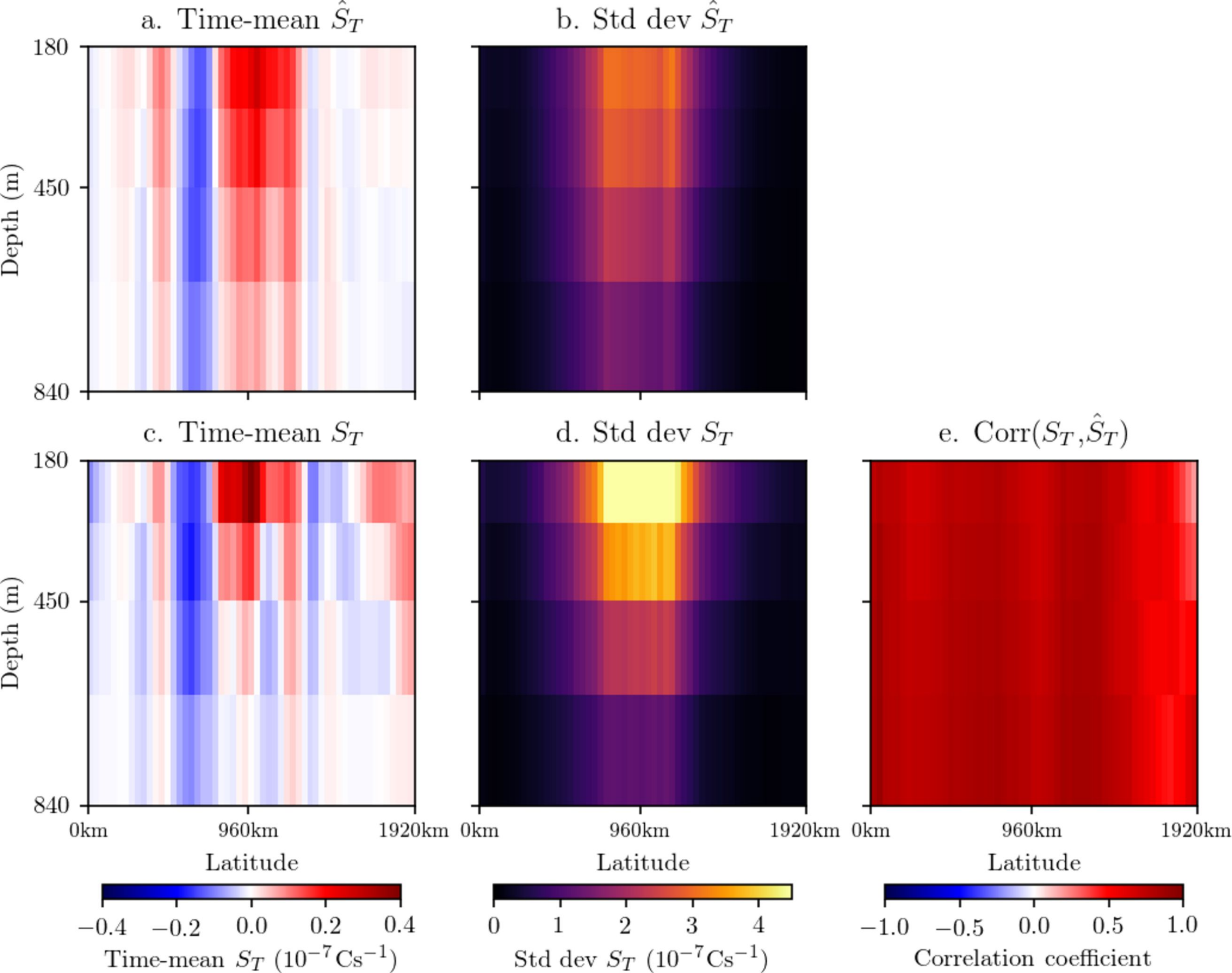
Warner, T. T. (2010). *Numerical weather and climate prediction.* Cambridge University Press. doi: 10.1017/CBO9780511763243

Zanna, L., Bachman, S., & Jansen, M. (2020, February). Energizing turbulence closures in ocean models. *CLIVAR Exchanges/US CLIVAR Variations*, *18*(1), 3–8. Retrieved from `https://doi.org/10.5065/g8w0-fy32` doi: 10.5065/g8w0-fy32

Zanna, L., Mana, P. P., Anstey, J., David, T., & Bolton, T. (2017). Scale-aware deterministic and stochastic parametrizations of eddy-mean flow interaction. *Ocean Modelling*, *111*, 66–80.

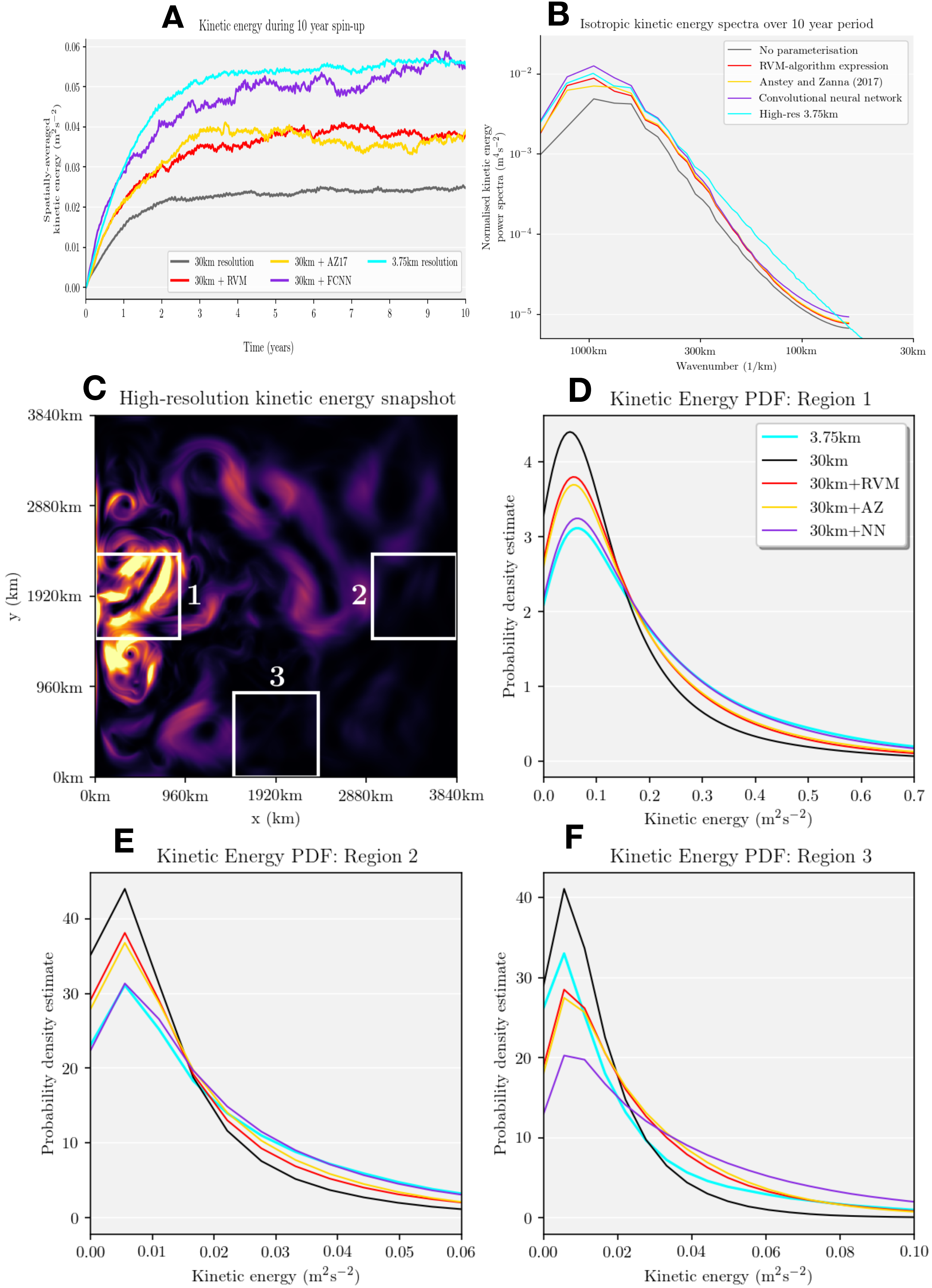
Zhang, S., & Lin, G. (2018). Robust data-driven discovery of governing physical laws with error bars. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *474*(2217), 20180305.

Figure2.



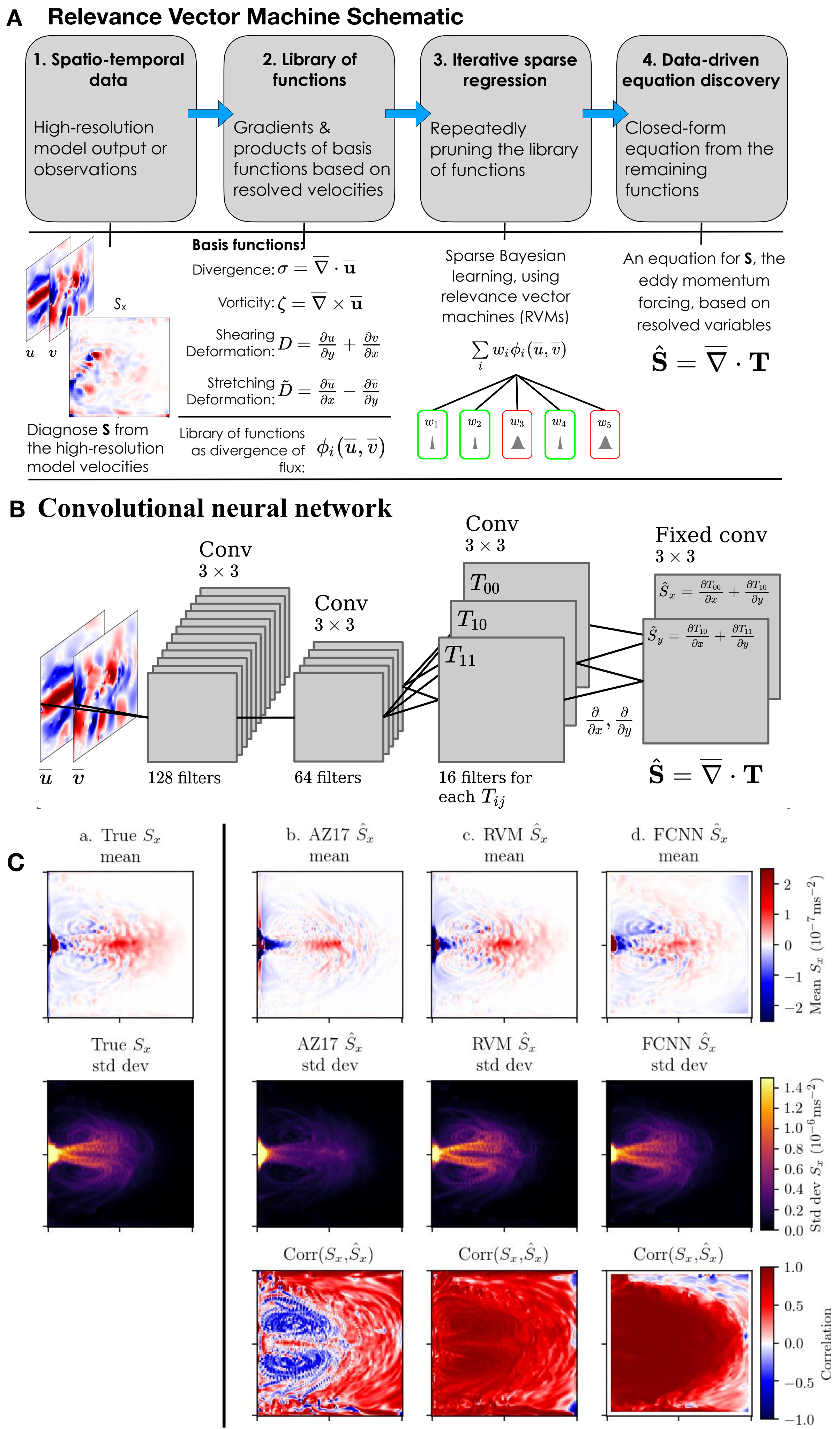
a. Time-mean $\hat{S}_T$
b. Std dev $\hat{S}_T$
c. Time-mean S_T
d. Std dev S_T
e. Corr(S_T,$\hat{S}_T$)
Depth (m)
180
450
840
0km
960km
1920km
Latitude
−0.4
−0.2
0.0
0.2
0.4
Time-mean S_T (10^{-7}Cs^{-1})
0
1
2
3
4
Std dev S_T (10^{-7}Cs^{-1})
−1.0
−0.5
0.0
0.5
1.0
Correlation coefficient

Figure3.



A
Kinetic energy during 10 year spin-up
Spatially-averaged kinetic energy (m^2s^{-2})
Time (years)
30km resolution
30km + RVM
30km + AZ17
30km + FCNN
3.75km resolution
B
Isotropic kinetic energy spectra over 10 year period
Normalised kinetic energy power spectra (m^4s^{-2})
Wavenumber (1/km)
No parameterisation
RVM-algorithm expression
Anstey and Zanna (2017)
Convolutional neural network
High-res 3.75km
C
High-resolution kinetic energy snapshot
y (km)
x (km)
D
Kinetic Energy PDF: Region 1
Probability density estimate
Kinetic energy (m^2s^{-2})
3.75km
30km
30km+RVM
30km+AZ
30km+NN
E
Kinetic Energy PDF: Region 2
F
Kinetic Energy PDF: Region 3

Figure 1.



A
Relevance Vector Machine Schematic
1. Spatio-temporal data
High-resolution model output or observations
2. Library of functions
Gradients & products of basis functions based on resolved velocities
3. Iterative sparse regression
Repeatedly pruning the library of functions
4. Data-driven equation discovery
Closed-form equation from the remaining functions
S_x
$\overline{u}$ $\overline{v}$
Diagnose S from the high-resolution model velocities
Basis functions:
Divergence: $\sigma = \overline{\nabla} \cdot \overline{\mathbf{u}}$
Vorticity: $\zeta = \overline{\nabla} \times \overline{\mathbf{u}}$
Shearing Deformation: $D = \frac{\partial \overline{u}}{\partial y} + \frac{\partial \overline{v}}{\partial x}$
Stretching Deformation: $\tilde{D} = \frac{\partial \overline{u}}{\partial x} - \frac{\partial \overline{v}}{\partial y}$
Library of functions as divergence of flux: $\phi_i(\overline{u}, \overline{v})$
Sparse Bayesian learning, using relevance vector machines (RVMs)
$\sum_i w_i \phi_i(\overline{u}, \overline{v})$
w_1 w_2 w_3 w_4 w_5
An equation for S, the eddy momentum forcing, based on resolved variables
$\hat{\mathbf{S}} = \overline{\nabla} \cdot \mathbf{T}$
B
Convolutional neural network
Conv 3×3
128 filters
Conv 3×3
64 filters
Conv 3×3
T_{00}
T_{10}
T_{11}
16 filters for each T_{ij}
Fixed conv 3×3
$\hat{S}_x = \frac{\partial T_{00}}{\partial x} + \frac{\partial T_{10}}{\partial y}$
$\hat{S}_y = \frac{\partial T_{10}}{\partial x} + \frac{\partial T_{11}}{\partial y}$
$\frac{\partial}{\partial x}, \frac{\partial}{\partial y}$
$\overline{u}$ $\overline{v}$
$\hat{\mathbf{S}} = \overline{\nabla} \cdot \mathbf{T}$
C
a. True S_x mean
b. AZ17 $\hat{S}_x$ mean
c. RVM $\hat{S}_x$ mean
d. FCNN $\hat{S}_x$ mean
Mean S_x $(10^{-7}\mathrm{ms}^{-2})$
2
1
0
−1
−2
True S_x std dev
AZ17 $\hat{S}_x$ std dev
RVM $\hat{S}_x$ std dev
FCNN $\hat{S}_x$ std dev
Std dev S_x $(10^{-6}\mathrm{ms}^{-2})$
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Corr(S_x,$\hat{S}_x$)
Corr(S_x,$\hat{S}_x$)
Corr(S_x,$\hat{S}_x$)
Correlation
1.0
0.5
0.0
−0.5
−1.0