Observational estimates of ocean energy from Argo floats

Katy Christensen

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington
2024

Reading Committee:
Alison R. Gray, Co-Chair
Stephen C. Riser, Co-Chair
Eric D'Asaro

Program Authorized to Offer Degree:
School of Oceanography

University of Washington

Abstract

Observational estimates of ocean energy from Argo floats

Katy Christensen

Co-chairs of the Supervisory Committee:

Alison R. Gray
School of Oceanography

Stephen C. Riser
School of Oceanography

This dissertation investigates energy in the ocean interior, focusing on the baroclinic conversion pathway between Eddy Available Potential Energy (EAPE) and Eddy Kinetic Energy (EKE), as well as the reservoir of EKE itself. This topic is explored using observational data from the global Argo float array, compared to using the classic approach in ocean energy analysis of examining model output. A major component of this dissertation is the development of methodology for estimating energy from observational data, which requires the consideration of numerous steps of quality control, spatial averaging techniques, and merging multiple sources of data.

Because the baroclinic conversion pathway is driven by vertical motion, I show the novel method for estimating mesoscale vertical velocities from Argo float data at 1,000 m. We find that the distribution of vertical velocities in the ocean is highly non-Gaussian, with a large number of small values near zero and a small number of large values up to the order of 10^{-4} m s^{-1}. The spatial variability of vertical velocity is linked to the ocean depth, where the largest values of vertical velocity appear to occur as horizontal flow interacts with topographic features. Due to the small magnitudes and high spatial variability, these estimates are amongst the first for vertical velocity across wide swaths of the interior ocean.

Using these estimates of vertical velocity, I include here the result of computing the baroclinic conversion

in the Southern Ocean. Using a local least-squares spatial regression for mapping, we find the anomaly components for vertical velocity and density that drive the conversion of energy by converting the EAPE in sloping isopycnals into EKE. We find that the largest values of baroclinic conversion occur in the ACC as it flows over large topographic features. The magnitude ranges from -5 to 5×10^{-5} kg m^{-1} s^{-3} but is positively skewed meaning that potential energy is being converted into eddy kinetic energy. This is found to agree well with model estimates of baroclinic conversion.

To further understand ocean energy, I also explore estimates of EKE throughout the water column using the horizontal velocity field from the surface to 2,000 m estimated by Argo float profiles. To examine the vertical structure in the EKE, we use the square of the Rossby wave vertical modes and apply two different boundary conditions: a flat bottom and a rough bottom. Comparing the proportion of variance in the EKE explained by the different regimes, we find that the flat-bottom boundary condition accounts for a greater proportion of the variance in fewer modes; however, this result is spatially variable where the rough-bottom modes are better suited for estimating the vertical structure of EKE where there are topographic features present.

The final part of this dissertation branches away from scientific research on ocean energy and into science education, where I discuss an introductory Python programming course for oceanographic data methods. Working collaboratively with another graduate student, we used evidence-based teaching practices and a constructivist approach to redesign a course to include a flipped structure, activities infused with active learning, an individualized final research project, and a focus on creating an accessible learning environment. By analyzing quantitative and qualitative data from surveys, online learning platforms, student work, assessments, and a focus group, we conclude that the instructional design facilitated learning and supported self-guided scientific inquiry. Students with less or no prior exposure to coding achieved similar success as peers with more experience, an outcome likely mediated by higher engagement with course resources.

The result of this dissertation is an examination of one branch of energy in the ocean from an observational standpoint and an in-depth description of effective science communication in the classroom. Capturing the structure of EKE and the pathway that contributes to it is fundamental for understanding the effects of a changing system on climate and carbon.

DEDICATION

To the Argo floats on the bottom of the ocean. Thanks for your data.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

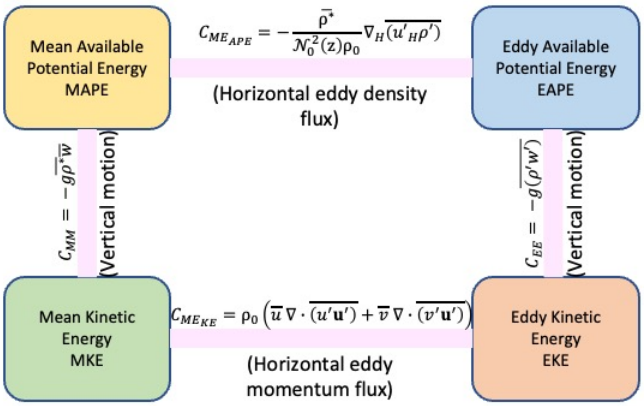

Figure 1.1: Reservoirs of Mean/Eddy, Kinetic/Available Potential energy in the Lorenz energy cycle. Equations for the conversion of energy between the different regimes are placed next to the relevant pathway, and labeled with the physical phenomena that drive that conversion. Equations adapted from Chen et al. [2014]; Youngs et al. [2017]; image adapted from Chen et al. [2014]; Aiki et al. [2016].

Energy drives motion, which influences the state of the ocean through circulation, mixing, and boundary interactions (e.g. air-sea, sea-ice, coastal processes). From the largest-scale currents to the smallest-scale turbulence, energy is being transferred and transformed throughout the ocean. And, though the avenue of energy to motion is an interesting physical process on its own, as the ocean moves it impacts global heat and carbon reservoirs that influence the climate as well as nutrient availability necessary for biological growth.

In the atmosphere, a great deal of research has been done to analyze all aspects of energy dynamics and their corresponding interactions within the field; this has somewhat been facilitated by ease of access where full-scale observations in the atmosphere are less challenging to obtain than in the ocean [Huang,

2004]. There has particularly been ample work on the development, growth, and persistence of atmospheric instabilities [Lorenz, 1955; Pierrehumbert and Swanson, 1995]. Similar instabilities do exist in the ocean. However, they are considerably less well-studied and, though there are fundamental similarities between the two fluids, all atmospheric energetics research is not necessarily directly applicable to the ocean. This is partially due to the fact that the ocean is "not a heat engine" in that it is primarily driven by mechanical processes rather than internal conversions like the atmosphere [Huang, 2004]. Yet, insights from the atmosphere can be useful in determining how the ocean system functions.

One useful tool that has come out of the energy analysis in the atmosphere is the deconstruction of an energy budget into a Lorenz Energy Cycle (LEC), first conceptualized in the seminal work done by Lorenz [1955]. This framework takes the total energy and separates it into Kinetic Energy (KE) and Potential Energy (PE) – including the internal energy of the system – components. Total PE is not truly a comparable measure of how much PE can be converted to KE since there is a certain amount of PE that simply cannot be converted. As such, a LEC further defines the metric of Available PE (APE), where the sum of APE and KE are conserved under adiabatic flow [Lorenz, 1955]. With this implemented, the only source of KE becomes APE; however, it is not the only sink of KE and dissipation must be accounted for. Both the APE and the KE can be further differentiated by separating out eddy and mean components of each (MAPE/EAPE, MKE/EKE) where the eddy component is the anomaly from the time-mean state. The governing equations for each of these four reservoirs of energy can be seen in the following set of equations [Chen et al., 2014; Youngs et al., 2017]:

$$\text{MAPE}(x,y,z) = \frac{0.5}{\mathcal{N}^2(z)\rho_0}\overline{\rho^*(x,y,z)^2} \tag{1.1}$$

$$\text{EAPE}(x,y,z) = \frac{0.5}{\mathcal{N}^2(z)\rho_0}\overline{\rho^{'}(x,y,z)^2} \tag{1.2}$$

$$\text{MKE}(x,y,z) = 0.5(\overline{u}^2 + \overline{v}^2) \tag{1.3}$$

$$\text{EKE}(x,y,z) = 0.5\overline{(u'^2 + v'^2)} \tag{1.4}$$

In these equations, $\mathcal{N}^2(z)$ refers to the squared buoyancy frequency from a time-mean density profile, ρ_0 refers to a constant reference density, overbars denote a time average, and u and v refer to the zonal and meridional velocities, respectively. The MAPE is the difference in potential energy of a time-mean state compared to a reference state (e.g., $\rho^*(x,y,z,t) = \rho(x,y,z,t) - \overline{\langle\rho(x,y,z,t)\rangle}$, where $\langle\rangle$ signifies an area mean). Whereas the EAPE is the difference in potential energy of the in-situ state compared to the time-mean state (e.g., $\rho^{'}(x,y,z,t) = \rho(x,y,z,t) - \overline{\rho(x,y,z,t)}$) [Chen et al., 2014]. In the ocean, most of the potential energy resides in the time-mean circulation, while most of the kinetic energy resides in time-varying flow.

In the context of an energy budget, the generation, conversion, and dissipation within each reservoir are balanced [Von Storch et al., 2012]. The generation terms are known to be the inputs of energy into the system from stress exerted by the atmosphere, evaporation and precipitation, heating and cooling, tides, or geothermal heating [Huang, 2004; Wunsch and Ferrari, 2004; Von Storch et al., 2012]. The dissipation terms are somewhat more elusive in their form and are often calculated as the remainder of the generation and conversion terms in an energy budget.

Of particular interest for this dissertation are the transformation terms, which are the pathways between the different reservoirs of energy (Figure 1.1). By gaining an understanding of these pathways, we can gain insight into how energy moves throughout the ocean. The transfer of energy between the different regimes in the LEC are caused by horizontal eddy momentum fluxes, horizontal eddy density fluxes, and vertical motions [Chen et al., 2014; Youngs et al., 2017]. Though the mathematical framework for these conversions is robust, the pathways themselves are not well sampled [Ferrari and Wunsch, 2009]. As such, much of the work presented here characterizes from observations one portion of the total energy budget by focusing on the baroclinic conversion pathway between EAPE and EKE.

In addition to the baroclinic conversion of energy, this research examines the reservoir of EKE itself. Specifically, we look at the vertical structure of EKE in the water column. At every depth except the surface where satellites can approximate energetics from sea surface height (SSH), observations of EKE in the ocean are sparse. In the past, they have been limited to short-term (<6 months) mooring deployments at select locations [Scott et al., 2010]. However, with the advent of Argo floats in the modern oceanographic era there is an unprecedented amount of data from the surface to 2,000 m that can be utilized in making estimates of EKE.

1.2 Approach and overview

The goal of the work described in this dissertation is to better understand one small portion of the overall energy in the ocean using observations. This research focuses on the mesoscale, as it contains a large proportion of the total EKE present in the ocean. Additionally, it is centered on the ocean interior as that is the least studied area of ocean energetics.

A significant connecting component factor between each of the different projects presented here is the use of data from the global Argo float array. Though the specific data used in analysis are described in each subsequent chapter, a brief overview of the Argo array is necessary here.

The global Argo float array was first conceptualized in the late 1990s when the need for consistent sampling of oceanographic variables (e.g. temperature, salinity) throughout the entirety of the world's oceans at regular intervals was recognized [Wong et al., 2020]. Though float design has evolved over time, the initial premise of matching the density of the float to the surrounding ocean to "park" at depth and then returning to the surface while taking a profile has remained consistent. Over the course of the past few decades there have been thousands of floats deployed and millions of profiles collected. There are now over 4,000 floats currently deployed still collecting data, with more floats still to be deployed in the future. New oceanographic variables have also been added to the sampling regime through the application of additional sensors [Claustre et al., 2020] and projects focused on specific regions, such as the Southern Ocean [Sarmiento et al., 2023], have been developed.

My approach is to use both the parking and profile data from Argo floats in calculating baroclinic conversion and EKE. Additionally, I use satellite data and model output for further analysis with the Argo data. The first two chapters are solely focused on obtaining estimates of baroclinic conversion, where the first chapter denotes a novel methodology for finding mesoscale vertical velocities, which drive energy transformation.

There is an additional chapter included within these works which addresses science education, where I worked collaboratively with a colleague to redesign an introductory Python programming course for oceanographic data analysis. We used evidence-based teaching practices to create a successful course design. Though this is not entirely related to the energy research presented elsewhere in this dissertation, effectively teaching geoscience programming is a key part of the field of Oceanography as a whole.

Chapter 2

Estimates of mesoscale vertical velocity near 1000 m

This chapter has been published as: Katy M. Christensen, Alison R. Gray, and Stephen C. Riser. 2024. Global estimates of mesoscale vertical velocity near 1,000 m from argo observations. *Journal of Geophysical Research: Oceans*, 129(1):e2023JC020003

2.1 Introduction

An essential component of many physical, chemical, and biological processes, vertical motion in the ocean occurs at all spatial and temporal scales, with a high degree of variability in both. As the pathway from the ocean interior to the surface, vertical flow directly impacts thermocline structure, nutrient transport, and global circulation [Sverdrup, 1947; Stommel and Arons, 1959; Munk, 1966; Martin and Richards, 2001; Pilo et al., 2018; Liang et al., 2017]. At the largest scales, the highly idealized model created by Stommel and Arons [1959] first estimated vertical velocities ranging between 0.5 and 3.0×10^{-7} m s^{-1}, averaged over large areas of the deep ocean, to compensate for dense water formation at high latitudes. As the spatio-temporal scale of the motions decrease, the magnitude of the associated vertical velocities tends to increase, as well as the variability across regions. The time-averaged vertical velocity at mid-depths associated with mesoscale motions has been estimated to be on the order of 10^{-6} m s^{-1} in a model [Liang et al., 2017], while values on the order of 10^{-4} m s^{-1} have been estimated within coherent vortices [Martin and Richards, 2001; Pilo et al., 2018]. Small-scale vertical velocities observed at mid-depths within internal waves and lee waves have been estimated to be on the order of 10^{-2} m s^{-1} [Merckelbach et al., 2010] and 10^{-1} m s^{-1} [Cusack et al., 2017], respectively. While these smaller-scale estimates are often inferred from observations at the locations and depths where phenomena at these scales occur, estimates at meso and larger scales are commonly determined from numerical models which do not resolve the small-scale processes, horizontally or vertically. Directly observing vertical velocities in the ocean at these scales has been an enduring challenge, due to the small magnitudes and high spatio-temporal variability [Stommel and Arons, 1959; Martin and Richards, 2001; Liang et al., 2017].

Despite the inherent difficulties, estimates of vertical velocities have previously been computed using observations across different scales, depths, and geographic locations, from many different sources including hydrographic profiles [Stommel and Arons, 1959; Munk, 1966; Martin and Richards, 2001], moorings [Sévellec et al., 2015], floats [Freeland, 2013; Cusack et al., 2017], and gliders [Merckelbach et al., 2010; Frajka-Williams et al., 2011]. However, these estimates are often only able to capture either large- or small-

scale vertical velocities due to limited sampling area or necessary assumptions of uniformity in depth and across smaller scales [Stommel and Arons, 1959; Freeland, 2013; Liang et al., 2017]. The vertical velocities of mesoscale phenomena, which have horizontal scales of $\sim$100 km, are not particularly well observed, even though such motions are in one of the most dynamic energy bands in the ocean [Ferrari and Wunsch, 2009]. Because mid-depth mesoscale velocities - both horizontal and vertical - are historically poorly characterized from observations, constraints on these flows in numerical models are completely lacking [Zilberman et al., 2023]. Oceanic models such as those used for coupled climate projections may therefore be misrepresenting these vertical motions and the associated fluxes of tracers. To advance our understanding of this critical component of the oceanic velocity field, a novel approach is needed to greatly expand our ability to estimate vertical velocities in the global ocean from observations.

The observations collected by the Argo array of profiling floats constitute the largest global, open-ocean, subsurface oceanic dataset. Since the late 1990s, the Argo program has vastly increased the coverage and quality of oceanographic observational data [Roemmich et al., 2019]. The classic Argo float cycle consists of a descent to 1000 dbar, a drift with the flow at that level in a quasi-isobaric fashion for approximately 10 days, another descent to 2000 dbar, an ascent to the surface while taking profile measurements of conductivity, temperature, and depth (CTD), and finally the transmission of data collected during the cycle to data acquisition centers using Iridium communications. The profile data from 0 to 2000 dbar are well-organized by the Argo program, comprehensively quality controlled, and have been extensively used in oceanographic research [Riser et al., 2016; Roemmich et al., 2019; Wong et al., 2020]. Argo float profiles have previously been used to estimate vertical velocities in the open northeast Pacific [Freeland, 2013]; however, that large-scale bulk estimate relied on computing the divergence within a well-defined control volume. Expanding this method to retrieve global vertical velocities at a finer scale, across a wide range of dynamic regimes, would require consideration of further processes and flows with the resulting uncertainties quickly dominating the estimates.

In addition to the profiles of temperature and salinity collected at the end of each cycle, a large number of Argo floats also record temperature and pressure measurements during the drift at 1000 dbar. The vast majority of floats do not measure salinity during the park phase. The temperature and pressure park phase data have been less frequently utilized than the profile measurements, and the associated quality control has been less consistent across platforms. Drift data at hourly resolution have previously been used to compute high-frequency isotherm displacement relative to the quasi-isobaric float, in order to examine internal wave characteristics near 1000 dbar [Hennon et al., 2014]. The observations recorded during the drift phase of Argo floats remain, however, relatively untapped as a source of information about the subsurface ocean. The work presented here uses these data to estimate mesoscale ($\sim$ 5-day mean) vertical velocities at mid-ocean depths. The resulting near-global picture of the vertical flow at 1000 dbar, quantified directly from observations for the first time at such scales, not only reveals important patterns of spatial variability but also provides a critical benchmark for the subsurface flow in numerical models.

Section 2.2 introduces the analysis framework underlying the method we have developed to estimate vertical velocity using data recorded during an Argo float's drift phase. In section 2.3 we apply the theory to the global Argo database and detail the extensive quality control steps that we have taken to ensure consistency in our estimates. The resulting vertical velocity estimates are presented in section 2.4, and several key geographic areas are highlighted. Additionally, we investigate how our estimates of vertical velocity relate to geographic location, bathymetry, horizontal flow, and sea level anomaly. Finally, section 2.5 compares the Argo-based estimates to model-based vertical velocities and discusses the implications of our results.

2.2 Theoretical Framework

To develop a method for estimating vertical velocity from high-frequency data recorded during the drift of Argo floats, we start with the evolution of temperature T following a water parcel

$$\frac{DT}{Dt} = k_z \frac{\partial^2 T}{\partial z^2} + k_h(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}) + Q \tag{2.1}$$

where $k_z \frac{\partial^2 T}{\partial z^2}$ represents the rate of change in temperature due to vertical mixing including the vertical diffusivity k_z, $k_h(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2})$ represents the rate of change in temperature due to horizontal mixing including the horizontal diffusivity k_h, and Q corresponds to the rate of change in temperature from external sources and sinks of heat. Consider now the temperature evolution of a water parcel at mid-depths in the open ocean, over approximately 5 days. For the associated space and time scales, we can assume that the influence of horizontal mixing on temperature (following a water parcel) is negligible. In addition, if the vertical temperature gradient $\frac{dT}{dz}$ is constant, the contribution of vertical mixing will vanish. Thus, given these conditions are upheld, we can neglect the temperature change due to mixing altogether.

In the open ocean there are negligible sources and sinks of heat at the float parking depth of 1000 dbar, so we can also approximate Q as zero. Thus, we are left with $\frac{DT}{Dt} \approx 0$, which can be expanded to show the contribution due to advection by both horizontal and vertical velocities (u_H and w, respectively),

$$\frac{\partial T}{\partial t} + u_H \cdot \nabla_H T + w \frac{\partial T}{\partial z} \approx 0. \tag{2.2}$$

Applying this framework now to an ideal, completely isobaric float that perfectly follows the horizontal projection of the flow causes the horizontal velocity relative to the float, u_H, to vanish. In other words, the temporal evolution of temperature observed by the float as it is drifting with the horizontal flow will solely reflect the vertical advection of the background temperature gradient. Further decomposing the temperature into a time-averaged mean $\bar{T}$ and a time-varying anomaly T' and assuming that the temperature gradient at depth is constant at these spatial and temporal scales allows Equation (2.2) to be solved for w as

$$w = -\frac{\partial(T' + \bar{T})}{\partial t} / \frac{\partial T}{\partial z} \approx -\frac{d}{dt}\left(T' / \frac{\partial T}{\partial z}\right). \tag{2.3}$$

Thus, vertical velocity at the park depth can be computed from the observed rate of change of isotherm displacement relative to the float.

Moving from a theoretical particle to a real world float, we must consider that the float is not perfectly isobaric. In fact, the floats move on equilibrium surfaces defined by their mass, which deviate from pressure surfaces depending on surrounding conditions and float compressibility [Swift and Riser, 1994]. Following Hennon et al. [2014], we account for the vertical motion of an Argo float away from the 1000-dbar isobar by subtracting the float displacement, given by the pressure anomaly during the drift. Doing this ensures that the isotherm displacement we estimate is a product of fluid motion alone. We then incorporate the pressure correction term into Eq.2.3. The pressure measured by a float must also be converted into units of depth to give velocity in m s^{-1}, which can be done by assuming a hydrostatic balance ($z = \frac{P}{\rho g}$). Following this we are left with the following equation for w:

$$w = -\frac{1}{\rho g}\frac{d}{dt}\left(\frac{T'}{(dT/dP)_{1000}} - P'\right) \tag{2.4}$$

where $T^{'}$ is the park phase temperature anomaly defined by $T^{'} = T - \bar{T}$ with $\bar{T}$ being the time average temperature, $P^{'}$ is the park phase float displacement defined by $P^{'} = P - \bar{P}$ with $\bar{P}$ being the time average pressure, $(dT/dP)_{1000}$ is the constant temperature gradient at 1000 dbar, ρ is the local density at 1000 dbar, and g is the gravitational constant. This final equation allows us to estimate vertical velocity solely using data from Argo floats, with $T^{'}$ and $P^{'}$ calculated from the float trajectory data and with $(dT/dP)_{1000}$ and ρ calculated from the float profile data.

2.3 Estimation of Argo-float based vertical velocity

2.3.1 Global Argo Data

Data used in this work include the drift trajectory, profile, technical (e.g. float systems reports during cycles), and metadata (e.g. float cycle configuration information) for the global array of Argo floats. Though there are over 16,000 floats with trajectory information available in the Global Data Acquisition Center (GDAC), many of the cycles from these floats cannot be used in estimating vertical velocity. Data that are usable for these estimates start in August 2005 and continue through April 2022. Data prior to 2005 are not usable because early float versions did not report an adequate number of samples during the drift cycle. Additionally, some later data are not usable because floats deployed by different programs or from different manufacturers report the drift data inconsistently with one another causing the necessary trajectory information to be less-uniformly available compared to that of the profile data. The biggest discrepancy is in the sampling regime during the park cycle, which results in floats reporting data ranging from sparse drift averages up to comprehensive hourly measurements of temperature and pressure. Note that the technical data and metadata for floats are similar to the trajectory data in they are inconsistently implemented between float programs. This can lead to missing file components causing the data from a float to be unusable in estimating vertical velocity. In particular, we are unable to use cycles that report temperature and pressure less frequently than every 6 hours during the park phase as well as cycles that have inadequate geographic coordinates, cycle numbering, and time information. To ensure that our results are as robust as possible, we have designated several quality control parameters (Section 2.3.3) in addition to the Argo program quality control system [Wong et al., 2020]. In consequence, our vertical velocity estimates are gleaned from 998 floats, encompassing 107,144 estimates.

2.3.2 Methods

From Eq.2.4, both the temperature gradient, $(dT/dP)_{1000}$, and the density, ρ, at 1000 dbar are found using float profile measurements. To get representative values for the entire park phase, the profiles directly before and after each individual park phase are averaged. The density at 1000 dbar is computed using the Gibbs-SeaWater Oceanographic Toolbox routines created by McDougall and Barker [2011] and is a function of absolute salinity and conservative temperature at a constant pressure of 1000 dbar. The temperature gradient is determined by finding the least squares best-fit line to the averaged temperature profile within 100 dbar from the parking depth [Hennon et al., 2014] (Figure 2.1b).

Starting with the raw temperature and pressure data reported during the entire float park, each drift cycle is split into a half-cycle of at least 4 days in length. The temperature and pressure anomalies during the park, $T^{'}$ and $P^{'}$, are found by subtracting the half-cycle mean values from the data for each half-cycle (Figure 2.1c). These anomalies are then applied to Eq.(4) to find the corrected isotherm displacement (Figure 2.1d). To examine phenomena on longer than inertial time scales, we use a fourth order lowpass Butterworth filter with a cutoff frequency of 0.3 cycles per day on the corrected isotherm displacement with odd reflected

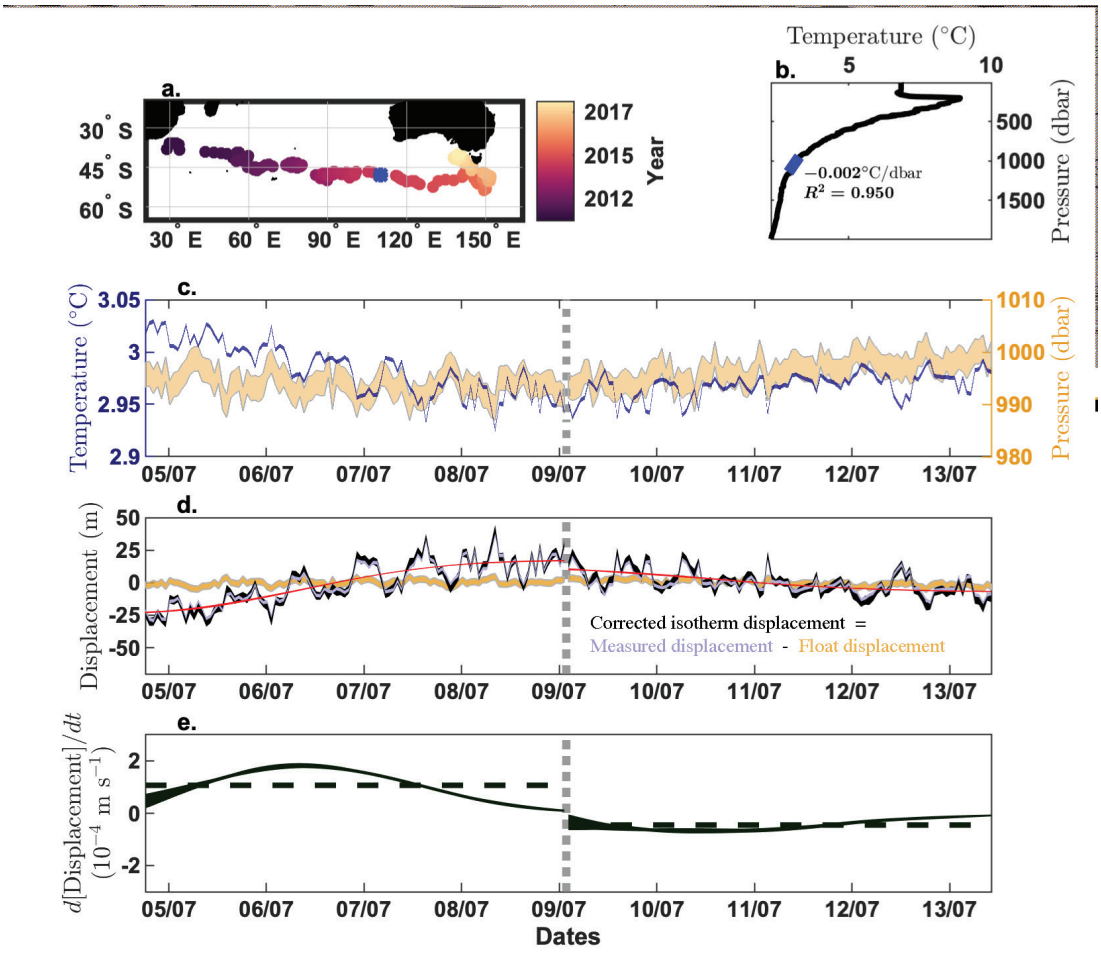

Figure 2.1: Example process for estimating vertical velocity using cycle number 134 from float 1901150 (WMO ID). No float buoyancy adjustments occurred during this cycle, so we are able to estimate two vertical velocities, one on each side of the gray dashed line in each panel. **a.** Map of all float cycle locations colored by date taken, except cycle number 134 which is noted in blue. **b.** Profile of temperature and pressure from cycle 134 with data surrounding 1000 dbar used to calculate $(dT/dP)_{1000}$, with at least 90% linear fit, highlighted in blue. **c.** Hourly sampled trajectory data where the temperature (blue) is shaded as the envelope of 100 iterations of added random error of ± 0.002°C and the pressure (orange) is shaded as the envelope of 100 iterations with added random error of ± 2.4dbar, as used in the Monte Carlo simulation described in section 2.3.3 **d.** Corrected isotherm displacement (black), calculated by subtracting float displacement (orange) from total measured displacement (blue), $[\frac{T'}{(dT/dP)}_{1000} - P']$. The corrected isotherm displacement is then filtered using a 4th Order Lowpass Butterworth filter and fit using a pchip function (red). Note that all of these values are converted to meters using an assumption of hydrostatic balance. **e.** Derivative of the filtered corrected isotherm displacement (solid black) and average value of the derivative (dashed black), which results in vertical velocity estimates of $(7.6 \pm 0.1) \times 10^{-5}$ m s^{-1} and $(-3.0 \pm 0.1) \times 10^{-5}$ m s^{-1} on the left and right sides of the cycle, respectively. Note that the error of these estimates is taken from the standard deviation of the 100 estimates that resulted from the Monte Carlo iterations.

endpoints. This effectively removes high frequency variability caused by internal tides and other short-lived events [Hennon et al., 2014]. After filtering, we fit the data using a piecewise cubic Hermite interpolating polynomial (pchip) [Fritsch and Carlson, 1980] and take the derivative with respect to time to get the slopes of the data during the drift half-cycle. Taking the average of this derivative and converting to depth units using float profile density in the equation for hydrostatic balance (the same assumption made to reach Eq. 2.4), gives us approximate values for the half-cycle-average vertical velocity (Figure 2.1e).

2.3.3 Quality Control

After a measurement cycle occurs, floats transmit their data to the data acquisition centers via the Argos-2 (<1% of usable floats) or the Iridium (>99% of usable floats) satellite systems. Then the data go through an automated quality control process that checks for reasonable dates, locations, velocities, pressures, temperatures, and salinities [Wong et al., 2020; data management, 2022]. Values are flagged with identifiers ranging from 1 to 4, where 1 corresponds to good data and 4 corresponds to bad data. This project uses quality control flags equal to 1, 2, or data that have been adjusted to reach the same level of qualification [Wong et al., 2020]. Within 24 hours after a float surfaces, the data have gone through the process described above and are publicly available as *Realtime-mode* data. In the following 5 months, more rigorous quality control should be done updating the cycle label to *Delayed-mode* [Wong et al., 2020]. However, this step is often only applied to the profile data and not to the trajectory, technical, or meta data. For estimating vertical velocity, we use the *Delayed-mode* data for the selection of floats where it is available (23.5% of usable cycles) and the *Realtime-mode* data in all other cases.

As stated in section 2.2, the vertical temperature gradient must be linear to maintain the assumption that there are limited vertical mixing effects on temperature. To enforce this condition, only cycles where the least squares best-fit line accounts for at least 90% of the temperature variance are considered [Hennon et al., 2014]. In some cases, the slope estimated from the profile data was near zero resulting in the magnitude of our vertical velocity estimates being unreasonably large. To counteract these cases, any vertical velocity with a magnitude larger than 10 standard deviations away from the mean of all the estimates are neglected (<0.05% of estimates).

Due to differences in float manufacturing and measurement goals, not all floats are programmed in the same manner. In particular, floats do not all have the same parking pressure and do not report data at the same interval during the drift. To keep the vertical velocity estimates comparable to one another, only floats that are assigned a park pressure of 1000 dbar in their metadata have been used. For differing drift data intervals, only floats that report both temperature and pressure measurements at a minimum of 6-hourly time increments during the park phase have been included. Previous studies that have analyzed isotherm displacement during the drift cycle could only use hourly-reporting floats because they were trying to capture the signal from internal gravity waves and tides [Hennon et al., 2014]. The mesoscale vertical velocities we are estimating persist over somewhat larger time scales, so we can use floats with a longer time between samples. As confirmation that the 6-hourly floats supply sufficient resolution, we conducted a test where we sub-sampled the hourly floats to every 6 hours, with randomly selected initial points, and re-estimated the vertical velocity using our method. When comparing the initial velocity estimates from the hourly samples with those from the sub-sampled regime, there is a 91% correspondence in the linear fit; the slope of this linear fit is 0.87. Though the 6-hour sub-sampling does slightly cause our method to underestimate the vertical velocity magnitude, particularly at large values, we feel that this error is sufficiently small with the average difference between the 6-hour sub-sampled values and the hourly true values being 2.2×10^{-7} m s^{-1}. Floats that measure once per day or transmit a cycle average value do not capture the necessary temperature variability to estimate isotherm motion around the float and are therefore discarded.

To maintain the programmed park pressure, floats will internally adjust their buoyancy under certain conditions. If a float is pushed outside of a pre-determined threshold (most often ± 10 dbar, located in the float metadata) from the programmed park pressure for three hours consecutively, the float increases or decreases its buoyancy to move back into the target zone. This adjustment causes additional variability in temperature measurements that is difficult to distinguish from temperature variability caused by isotherm displacement alone. Buoyancy adjustments can happen multiple times in a single drift cycle and the floats then report the number of times that they adjust in their technical data; however, they do not report when the adjustments actually occur. To manually determine when an adjustment happens, hourly data are required. Thus, for floats with a lower sampling frequency than hourly any cycles with reported adjustments are discarded. For floats that do measure on an hourly basis, we use the pressure data to compute the time of adjustment and keep only those cycles where the number of computed adjustments equals that reported in the technical data. Only half-cycles with no adjustments (omitting data in a 6 hour window after any prior adjustments) were used to compute vertical velocity. The floats do adjust preferentially in certain areas, which can likely be attributed to large magnitude vertical motions influencing the floats. Because we are neglecting cycle halves that include these adjustments, the estimates for vertical velocities shown in this chapter can be considered conservative.

To estimate the error involved in calculating the vertical velocity, we have conducted a Monte Carlo simulation that incorporates random error based on instrument accuracy. Floats are equipped with CTD sensors that have published accuracy of ± 0.002°C (0.0002°C drift per year) and ± 2.4 dbar (0.8 dbar drift per year) for temperature and pressure, respectively [Wong et al., 2020]. Because the average values are removed during the calculations, the precision of each measurement are more likely to impact the observation quality, but these values are smaller than those of the accuracy. We feel confident that using the larger values allows for robust testing in the Monte Carlo simulation. For each calculation of vertical velocity, 100 iterations of random error within the accuracy ranges were incorporated into the temperature and pressure data. These data were then used to compute 100 estimates of vertical velocity for each half cycle. The final value is the average of all the different iterations and the error is assumed to be represented by the standard deviation (Figure 2.1c). To compute aggregate statistics (mean, median, variance, etc.) on subsets of the data, we use all 100 of the Monte Carlo iterations for each half cycle and report the average of the desired statistic along with the standard deviation as our error estimate.

2.4 Results

2.4.1 Global estimates of near 1000-meter vertical velocity

The locations of vertical velocity estimates from the global Argo array have a spatial distribution that varies across geographic regions (Figure 2.2 a-c). Though the Argo array is well-distributed globally [Wong et al., 2020], these estimates of vertical velocity do not have uniform coverage and are particularly sparse in the Atlantic and North Pacific Oceans (Figure 2.2c). This is due to a lack of adequate 1000 dbar drift data in those areas in addition to a rigorous application of quality control measures (section 2.3.3).

The estimated vertical velocities have a particularly heavy-tailed distribution with a high peak (Figure 3). These data have the smallest sum square error (defined as $\sum_i \left(y_i - f(x_i)\right)^2$, where y_i is the given histogram value and $f(x_i)$ is the value predicted by the probability density function) when fit by the Cauchy function as Eq. 2.5,

$$f(x) = \frac{1}{\pi(1 + x^2)} \tag{2.5}$$

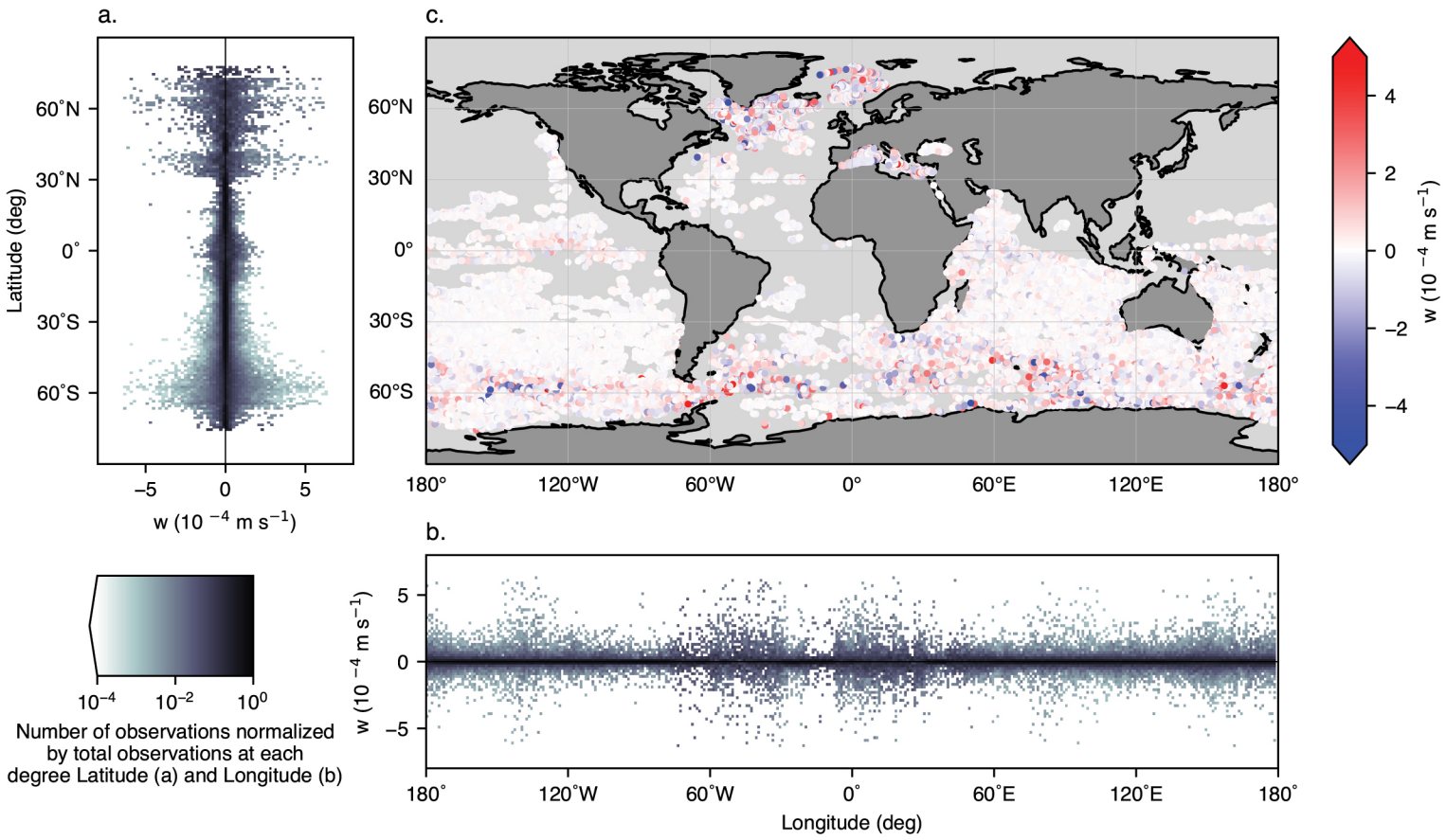

Figure 2.2: Argo-based, 5-day-averaged vertical velocities near 1000 dbar. Probability density functions of velocities as a function of (a) latitude and (b) longitude, normalized at each degree latitude or longitude, respectively. (c) Global map of all individual velocity estimates.

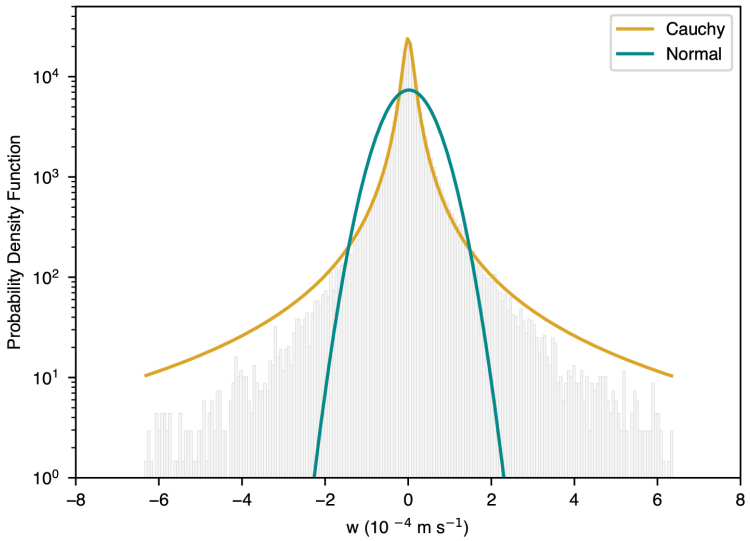

Figure 2.3: Probability density function (PDF) of global vertical velocities with Cauchy (gold) and Normal (teal) distributions plotted atop. Note that the y-axis is on a logarithmic scale.

which is the equivalent to a Student's t continuous fit with one degree of freedom [Forbes et al., 2010]. In comparison, a normal distribution is equivalent to a Student's t continuous fit with infinite degrees of freedom, meaning that these data are decidedly non-Gaussian [Forbes et al., 2010]. One characteristic of a Cauchy probability distribution is that the higher order statistical moments are undefined and do not converge with increased sampling [Forbes et al., 2010; Sugiyama, 2016]. Therefore, the mean and standard deviation that we compute for our finite dataset are highly dependent on the selection of sampling parameters and do not converge for large sample size. To characterize our dataset, we compute and report the median, which is less influenced by heavy tails. We also examine the mean and standard deviation for our data with the caveat that these values are dependent on the particular characteristics of sampling. With this in mind, this type of statistical analysis can still help provide insight into the variability of our estimates.

The median vertical velocity from the entirety of our estimates, is $(1.3 \pm 0.2) \times 10^{-7}$ m s^{-1} and the average vertical velocity is $(1.9 \pm 0.02) \times 10^{-6}$ m s^{-1}. If these data were collected everywhere globally, over a sufficiently long time period, the true average would be equivalent to zero due to the conservation of mass in the ocean [Stommel and Arons, 1959; Freeland, 2013]. However, there is no evidence to suggest that the vertical motion occurring at just the mesoscale should be balanced within itself, so we cannot assume that the average of our irregularly-sampled, mesoscale estimates should be equal to zero. The variance of our vertical velocity estimates is $(3.0 \pm 0.01) \times 10$[m s^{-1}]^2, which is impacted by both the high peak and heavy-tails of our distribution. There is a slight positive skew of 0.6 ± 0.03; however, due to the shape of the distribution discussed above, we cannot disentangle if this is a product of the uneven spatial and temporal distribution of our samples or a product of actual physical phenomena. To gain different insights into these estimates, we examine areas that are well sampled in context with sea level anomaly.

2.4.2 Sea level anomaly

To examine how our mid-depth vertical velocity estimates correspond with other physical phenomena in the ocean, we use the Data Unification and Altimeter Combination System (DUACS) gridded (L4) altimeter

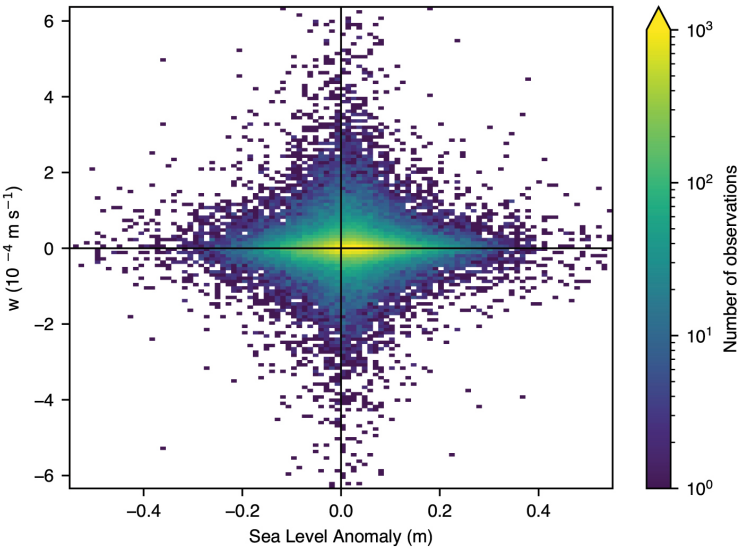

Figure 2.4: Global Sea Level Anomaly (SLA) vs. Vertical velocity estimates colored by the number of observations for each 0.01m $\times 10^{-5}$m s$^{-}1$ box

product with a 1/4° × 1/4 ° resolution with daily data ranging from 1993 to June 2020 [CMEMS, 2021]. From this gridded product, we use both the Sea Level Anomaly (SLA) and the horizontal (zonal [U] and meridional [V]) absolute geostrophic velocities at the surface.

The SLA in this dataset is given as the sea surface height above mean sea surface using a 20 year mean from 1993 through 2012; however, to better compare this data with that given by the Argo array, we have changed the reference period to be a 15 year mean from 2005 through 2020 following the methods in Pujol et al. [2016]. To directly link float estimates with satellite SLA, we located the nearest grid cell in the DUACS L4 product to each vertical velocity estimate location and averaged the SLA at that point for the actual dates of the float cycle. We repeat the process for the horizontal velocity magnitude, computed from surface U and V, by finding the nearest grid cells in the satellite product to our float estimates. However, we then average across all times(1993-2012) rather than just the float period. Taking the average over the entire time-series results in a mean state horizontal velocity, to which we can compare our vertical velocity.

Investigating the SLA from all locations where we have half cycles, we see that there is no distinct correlation between the SLA and the vertical velocity estimates (Figure 2.4). The highest values of vertical velocity would visually appear to be associated with smaller magnitude SLA, but there are considerably more data points present at smaller magnitudes than at higher magnitudes for both the vertical velocity and SLA (note Figure 2.4 has an exponential colorbar). Statistically this pattern is simply a product of the high peaks in both the vertical velocity and SLA distributions rather than showing a particular physical phenomena. Additionally, the SLA is measured at the surface via satellites, so comparing these measurements to our mid-depth estimates of vertical velocity is not direct. Certainly, not every eddy has a subsurface component that will have an effect at the Argo parking depth. There are ways to locate sub-surface phenomena solely using satellite data [Assassi et al., 2016]; however, these methods have difficulty in complex areas with strong currents, of which a great deal of our study area is composed. Instead, separating our vertical velocity estimates and their associated SLA into subsets of areas with high coverage will allow us to examine the degree of variability between different locations.

Table 2.1: Aggregate statistics –median, mean (μ), variance (σ^2), and ratio of mean to standard deviation (μ/σ) –of vertical velocity estimates globally and for selected areas shown in Figure 2.5.

Area	Median (m s^{-1})	μ (m s^{-1})	σ^2 ([m s^{-1}]^2)	$\lvert\mu\rvert/\sigma$
Global	$(1.3 \pm 0.2) \times 10^{-7}$	$(1.9 \pm 0.02) \times 10^{-6}$	$(3.0 \pm 0.01) \times 10^{-9}$	0.03
KP	$(1.2 \pm 0.2) \times 10^{-6}$	$(5.4 \pm 0.2) \times 10^{-6}$	$(7.8 \pm 0.09) \times 10^{-9}$	0.06
IOG	$(7.3 \pm 1.4) \times 10^{-7}$	$(8.5 \pm 0.7) \times 10^{-7}$	$(4.5 \pm 0.05) \times 10^{-10}$	0.04
PAR	$(7.4 \pm 0.2) \times 10^{-6}$	$(1.4 \pm 0.02) \times 10^{-5}$	$(1.0 \pm 0.01) \times 10^{-8}$	0.14
POG	$(-6.2 \pm 0.9) \times 10^{-7}$	$(-4.7 \pm 0.4) \times 10^{-7}$	$(1.2 \pm 0.01) \times 10^{-10}$	0.04
DP	$(1.1 \pm 0.1) \times 10^{-5}$	$(2.5 \pm 0.07) \times 10^{-5}$	$(1.7 \pm 0.03) \times 10^{-8}$	0.19
NA	$(-1.2 \pm 0.3) \times 10^{-6}$	$(-6.1 \pm 2.3) \times 10^{-7}$	$(1.5 \pm 0.01) \times 10^{-8}$	0.005

2.4.3 Spatial variability and areas of interest

Because of the varying data density across the globe, spatial analysis of the entire data set is not wholly informative. Looking closer at the variation in longitude (Figure 2.2b), we can see that the higher magnitude velocities do not occur uniformly, but instead have a unique distribution with certain areas being hot spots of higher magnitude vertical velocity. We have selected a few key areas for comparison. Namely, we chose two major topographic features in the Antarctic Circumpolar Current (ACC) in the Southern Ocean (Kerguelen Plateau (KP) and Pacific-Antarctic Ridge (PAR), Figure 2.5d-e) and selected juxtaposed examples in the gyres north of each (Indian Ocean Gyre (IOG) and Pacific Ocean Gyre (POG), Figure 2.5b-c). Each of these subsets of data has a comparable number of observations rather than equal study areas. The ACC generates considerable mesoscale and submesoscale variability that is expected to be associated with higher magnitude vertical velocities [Rosso et al., 2014; Cusack et al., 2017].

We find that the distributions of vertical velocity estimates over topographic features are similar to the global distribution in that they have high peaks and heavy tails that are best fit by the Cauchy function (Figure 2.5d-e). Our estimates in quiescent areas have much lighter tails and, though still not entirely normally distributed, are better fit by a normal distribution across most of the estimates with the only misfits being infrequent, high-magnitude events appearing in the tails (Figure 2.5b-c).

The median, mean, and variance of our vertical velocity estimates for each of the selected areas from Figure 2.5 are shown in Table 2.1. The mean values are similar in magnitude for the more energetic areas (KP, PAR) on the order of 10^{-6} to 10^{-5} m s^{-1}, as well as for the more quiescent areas (IOG,POG) on the order of 10^{-7} m s^{-1}. However, due to the heavy-tailed distributions, all of the mean values depend on the particular sampling scheme selected, even for large sample sizes, so these average results must be interpreted with caution. The differences in variance are more illuminating. In the ocean gyres, the variance of our estimates are on the order of 10^{-10} [m s^{-1}]^2 while the topographic features have a variance nearer to 10^{-8} and 10^{-9} [m s^{-1}]^2 for PAR and KP, respectively. This emphasizes the impact that the high-magnitude, heavy tails have in areas with significant bathymetry. As another large topographic feature in the ACC, we expect to see a similar pattern for Drake Passage (DP) (Figure 2.5g). Though the estimates in this area are sparser than in the others, the variance is once again two orders of magnitude larger than those of the ocean gyres and comparable to the other active regions. The scale of the global variance is approximately 10^{-9} [m s^{-1}]^2, showing the impact of the large vertical velocities that are localized in space.

Another area of interest is the North Atlantic (NA, Figure 2.5f), where the vertical velocity distribution is similar to the high energy areas of the Southern Ocean. As a region with mixed layers exceeding 1000 m, deep water formation, and significant eddy activity [Stommel and Arons, 1959; Ferrari and Wunsch, 2009],

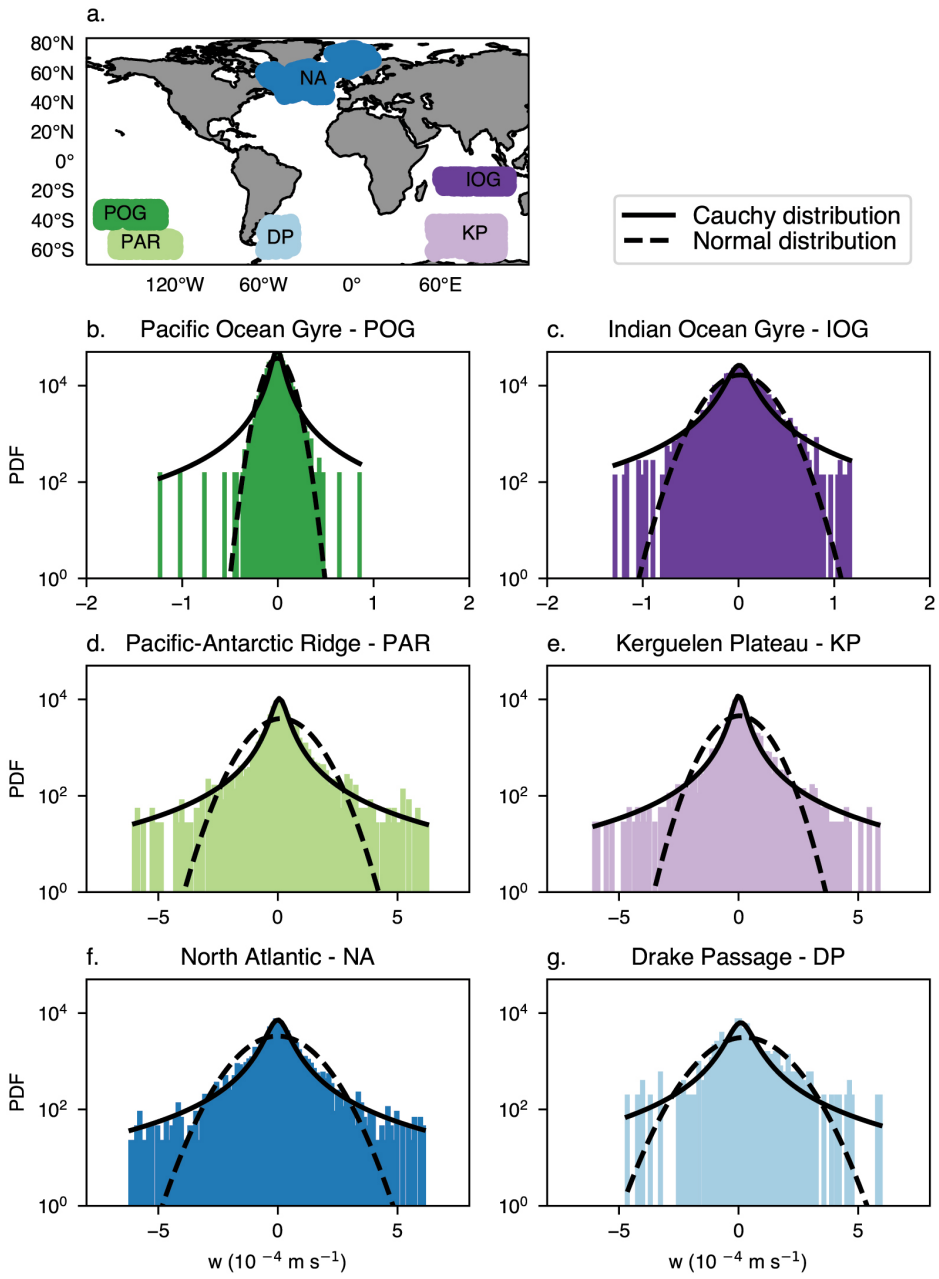

Figure 2.5: Probability density function distributions with Normal and Cauchy distribution functions plotted on top for select areas highlighted by corresponding color in a global map (a). b) Pacific Ocean Gyre [POG] with 3,143 observations (dark green), c) Indian Ocean Gyre [IOG] with 3,001 observations (dark purple), d) Pacific-Antarctic Ridge [PAR] with 3,053 observations (light green), e) Kerguelen Plateau [KP] with 3,057 observations (light purple), f) North Atlantic [NA] with 3,664 observations (dark blue), g) Drake Passage [DP] with 487 observations (light blue). Note that the x axis limits on panels (b) and (c) are decreased by a factor of 4 from the other PDFs.

it is expected that the North Atlantic mimics the other active regions. Indeed, the variance is once again two orders of magnitude larger than that of the quiescent areas.

The median values of our vertical velocity estimates range from 10^{-7} to 10^{-5} m s^{-1} (Table 1). The larger values occur in high-energy areas while the smaller values occur in the gyres. The global median is on the same order of magnitude as the gyre median values. This shows that, though the effects of large-magnitude events can be seen in the global variance, the small-magnitude phenomena that form the high peak in the distribution dominate the median (and mean).

The substantial impact of variability in mesoscale vertical velocities is further emphasized by the ratio of the absolute value of the mean and the standard deviation ($|\mu|/\sigma$, Table 2.1, Column 5), which varies between 0.005 and 0.19. In the gyre regions, this ratio is close to the global average value of 0.03, while the largest values are found in the energetic Southern Ocean areas.

Interestingly, the NA and POG both have a negative average vertical velocity, suggesting mean downwelling across 1000 m in these regions. Although the mean velocities are quite comparable, the observed variance is two orders of magnitude larger in the NA than in the POG. Given the resulting small ratio between the absolute value of the mean and the standard deviation (0.005 in the NA compared to 0.04 in the POG), we abstain from a physical interpretation of the negative mean vertical velocity in the NA. However, the larger ratio in the POG and the fact that the velocities there are more normally distributed (see discussion in Section 2.4.3) imply that the estimate of the mean in that region is more robust.

Considering a steady-state, one-dimensional balance between advection and diffusion in the temperature equation [Tsujino et al., 2000],

$$w\frac{\partial T}{\partial z} = \frac{\partial}{\partial z}\left(K_v \frac{\partial T}{\partial z}\right), \tag{2.6}$$

we can examine the implications of an average downward vertical velocity, which is contrary to the canonical model put forth by Munk [1966]. Near 1000 m, the first- and second-order vertical derivatives of temperature are positive (generally and in the POG specifically), and assuming down-gradient diffusion of heat gives a strictly positive value for vertical diffusivity (Kv). Therefore, in this one-dimensional model, a mean downward vertical velocity could only arise due to a negative gradient in diffusivity, implying an increase in mixing in the deeper waters relative to shallower depths. Such a vertical gradient in diffusivity at 1000 m has been shown to be necessary for models to more accurately replicate deep Pacific circulation [Tsujino et al., 2000].

The SLA associated with our select areas also highlight the variability between different locations. The SLA in the IOG and POG (Figure 2.6a-b) are similar in range going from -0.33 to 0.35 m with small vertical velocities. However, the SLA in the PAR (Figure 2.6c) is more limited ranging only between -0.11 to 0.15 m with larger vertical velocities. The SLAs near the KP (Figure 2.6d) are again different to those at IOG/POG and PAR, showing a wide range for both SLA (-0.49 to 0.41m) and vertical velocity. This reinforces that there is high variability in the spatial distribution of vertical velocities that are interacting with a variety of mechanisms, even in similarly energetic regimes.

2.4.4 Effects of bathymetry and horizontal flow

To examine other factors that may impact the observed vertical velocity variability, we utilize the satellite derived horizontal velocities (as described in section 2.4.2) and the ETOPO1 1 Arc Minute Global Relief Model [Amante and Eakins, 2009; NGDC, 2009]. We use the version of bathymetry that includes the bases of the Antarctic and Greenland ice sheets. To better compare between the gridded relief model and the Argo array, we located the nearest grid cell to each float cycle location that contains an ocean depth. Because

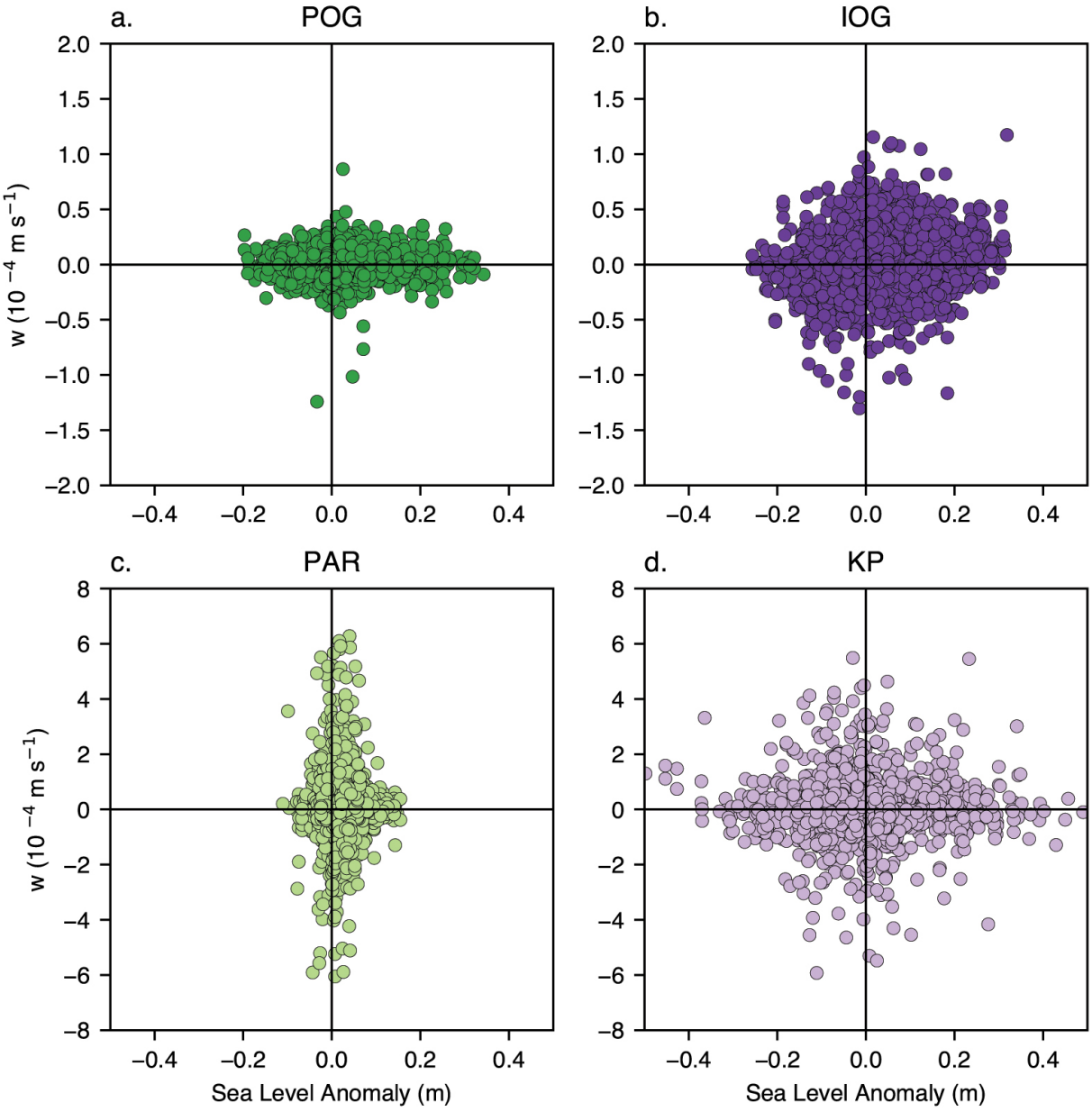

Figure 2.6: DUACS Sea Level Anomaly at the nearest grid point averaged over time of corresponding half cycles (as described in section 2.4.2 versus vertical velocity estimates for select areas highlighted by corresponding color in a global map in Figure 2.5a. a) POG (dark green), b) IOG (dark purple), c) PAR (light green), d) KP (light purple). Note that the y axis limits on panels (a) and (b) are decreased by a factor of 4 from panels (c) and (d).

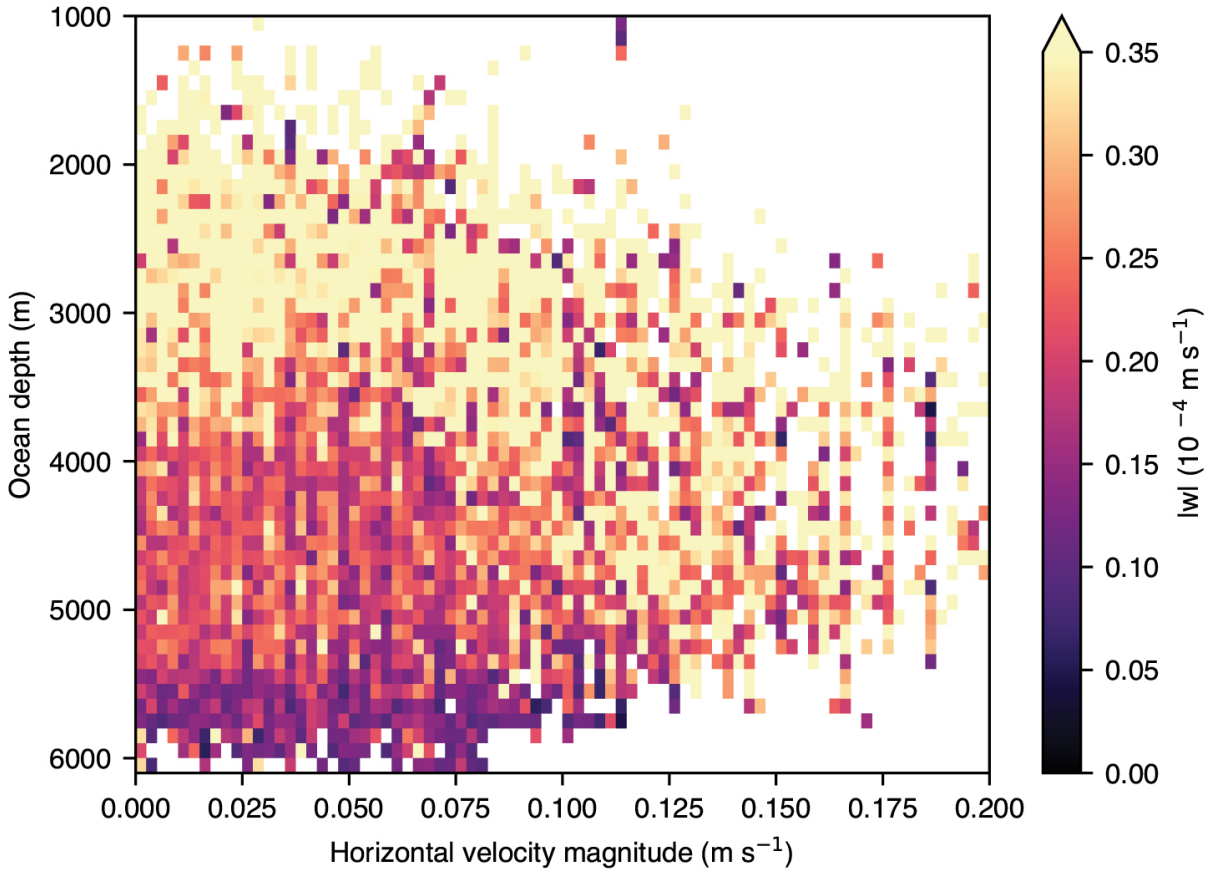

Figure 2.7: Average absolute value vertical velocity within grid cells of horizontal surface velocity averaged over 1993-2020 (bin size of 0.0025 m s^{-1}) and ocean depth (bin size of 75 m). Only grid cells with at least 5 estimates of vertical velocity are shown here. Note that floats are typically not deployed in waters shallower than 2000 m (due to a desire to prevent float grounding) so sparseness in samples above this depth is expected.

bathymetry changes on time scales much longer than the mesoscale phenomena we are focused on, we assume that the bathymetry remains constant.

Examining both the global surface velocities and the ocean bathymetry with their corresponding float w estimates, we see that deep areas with small horizontal surface velocities tend to have smaller values of w. However, areas with larger average horizontal surface flow (> 0.08m s^{-1}) have more high magnitude vertical velocities, even where the ocean is deep (between 4000-5000 m) (Figure 2.7). Prior research has shown elevated mesoscale eddy activity frequently occurs downstream of strong horizontal flows cross over topographic features [Rosso et al., 2014; Cusack et al., 2017; Liang et al., 2017]. This lag between the peak in bathymetry and the peak in vertical velocity may also contribute to high magnitude vertical velocities occurring where the ocean is deeper.

To analyze how the vertical velocity signal relates to bathymetry as the ocean flows across topographic features, we will highlight the ACC as a case study. This area is characterized by a strong zonal flow across several notable topographic features [Tamsitt et al., 2017; Yung et al., 2022] (Figure 2.8b) and it includes adequate float coverage across most longitudes except in the Atlantic Ocean. Consistent with the results discussed in section 2.4.3, hot spots of vertical velocity magnitude correspond roughly to known bathymetric features in the path of the ACC, including KP, PAR, and DP as well as Macquarie Ridge (MR) and the Southwest Indian Ridge (SWIR) [Tamsitt et al., 2017; Yung et al., 2022].

Here we define the ACC area as the polygon bounded by the zonally continuous streamfunction contours -0.7 to 3.4 dynamic m, as computed from Argo floats by Zilberman et al. [2023] (Figure 2.8a). We have taken all of the vertical velocity data within the ACC, and computed a running average with 4° longitude boxes at a 2° longitude spacing. We then did the same for all of the bathymetry data included in the ACC polygon. Finally, we compute a normalized lagged correlation between average ocean bottom depth and average absolute vertical velocity for areas with greater than 15 samples and get that they are somewhat correlated (r=0.55) (Figure 2.8b). The vertical velocity peaks lag the bathymetry peaks by approximately 9° longitude, meaning that higher magnitude values occur downstream in the ACC, as expected. However, this correlation certainly does not explain all of the variability in vertical velocity. Other factors are at play, including variable horizontal flow and multiple bathymetric features interacting with the current, particularly if features are near to one another or cover large areas.

Taken altogether, these results from the ACC indicate that at mid-depths, higher vertical velocities occur downstream of shallow bathymetry (Figure 2.8), which is supported by our findings of a relationship between bathymetry, horizontal flow, and vertical velocity magnitude in the global estimates(Figure 2.7). This provides evidence that the 5-day average vertical velocities determined from Argo observations primarily reflect mesoscale phenomena.

2.5 Discussion and summary

The results presented here are the first estimates of subsurface vertical velocity determined directly from observations, across wide swaths of the global ocean. Our approach of using measurements from floats in the globally distributed Argo array provides insight into three-dimensional flows occurring near 1000 dbar. The sampling distribution is uneven, with some regions well covered and others lacking data entirely, due to differences in how floats report data from the drift phase at the parking depth. As the global Argo array evolves, drift data suited for this method will continue to be collected from many floats, thereby increasing the coverage of these vertical velocity estimates [Riser et al., 2016; Roemmich et al., 2019; Wong et al., 2020; data management, 2022]. Furthermore, we encourage all float manufacturers and users to measure and report hourly temperature and pressure data during the drift phase when possible, to expand the spatial

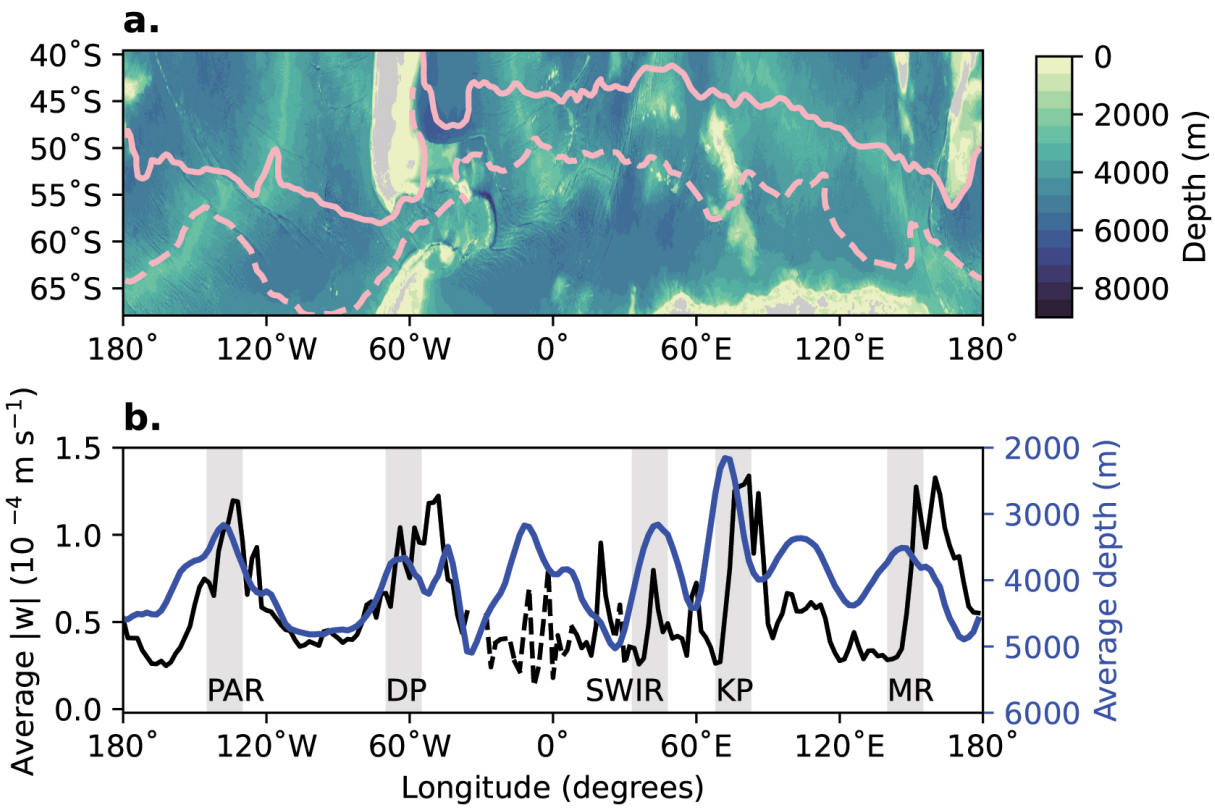

Figure 2.8: a) ETOPO1 bathymetry of the ACC and surrounding area. The pink lines show the stream-function contours (-0.7 to 3.4 dynamic m, as computed by Zilberman et al. [2023]) that define the edges of the ACC. b) (Black - left axis) Running average of absolute value vertical velocities within the ACC using 4° longitude boxes spaced every 2°, constrained in latitude by the streamfunction contours in panel (a). Note that the dashed line here refers to areas where there were fewer than 15 samples included in the average. (Blue - right axis) Running average of ETOPO1 depth within the ACC using 4° longitude boxes spaced every 2°, constrained in latitude by the streamfunction contours in panel (a) Significant topographic features highlighted with gray boxes [PAR, DP, KP, Southwest Indian Ridge (SWIR), and Macquarie Ridge (MR)].

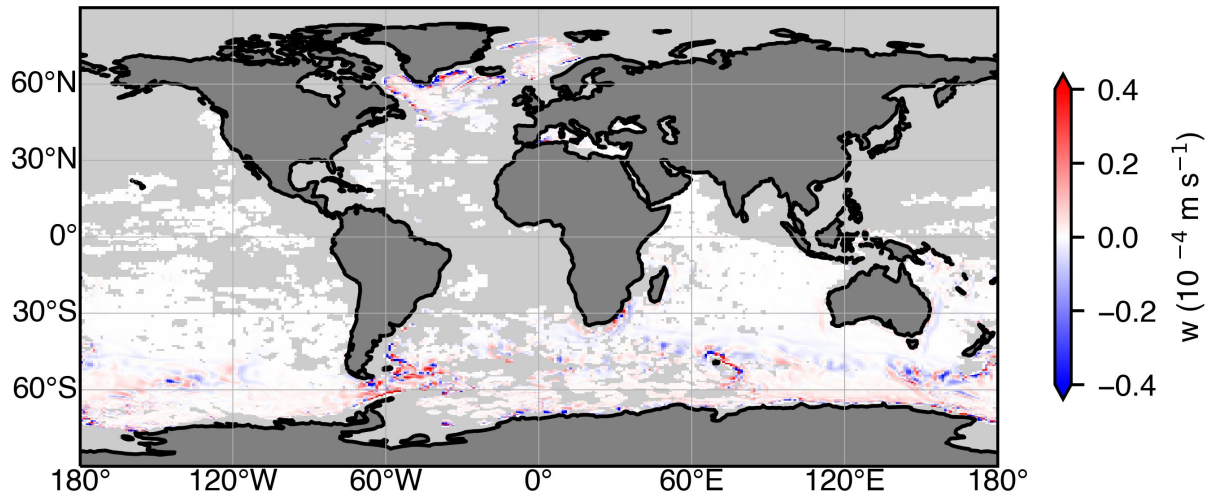

Figure 2.9: Global map of the vertical velocity field from the ECCOv4.4 model in grid cells that contain observational estimates from floats.

coverage of these vertical velocity estimates.

In areas with adequate sampling, our results highlight distinctive spatial variability in vertical velocities at 1000 dbar. In particular, there is clear evidence of a relationship between vertical motion and topographic features, with further influence in areas with strong horizontal flows. This topographic dominance is consistent with prior global estimates of mid-depth vertical velocity using model fields [Liang et al., 2017]. To better assess our estimates in relation to velocity output from a widely-used ocean state estimate, we retrieved data from the latest Estimating the Circulation and Climate of the Ocean (ECCO) project (version 4, release 4) [Forget et al., 2015]. We used the monthly-averaged vertical component of velocity from 2004 to 2017, which best overlaps with the time period available for the float estimates. We took the 1/2° resolution LLC90 grid, chose the depth level closest to 1000 dbar, and then selected only locations that have at least one float estimate of vertical velocity within 50km of center of the model grid-cell. Subsampling the model output in this way eliminates data from the coastal areas and shallow areas where floats do not sample.

Averaging the resulting model vertical velocity fields over the full 13-year period reveals spatial variability that matches the spatial pattern in the Argo-based estimates remarkably well. This similarity again emphasizes the topographic influence on vertical motions near 1000 dbar (Figure 2.2, Figure 2.9). The ECCO vertical velocities in Figure 2.9 are an order of magnitude smaller than the float-based velocities in Figure 2.2, reflecting the difference in the temporal scales of the two estimates and highlighting the influence from the spatial resolution of the model.

Examining instead the monthly-averaged velocities from ECCOv4.4, again restricted to only the locations of the float estimates, the normalized distribution of vertical velocity exhibits a high peak and heavy tails that is best fit by the Cauchy distribution, like the float-based estimates.

Although the 5-day averaged float estimates and the monthly model w fields still have different temporal scales, the similarity in the shape of the distributions provides support for the efficacy of the vertical velocity estimation method described here. Interestingly, the distribution of the observational w estimates is more symmetric than the distribution of the model vertical velocities, which is skewed towards negative values. Possible reasons for this include the uneven temporal spacing in the float estimates when compared to the

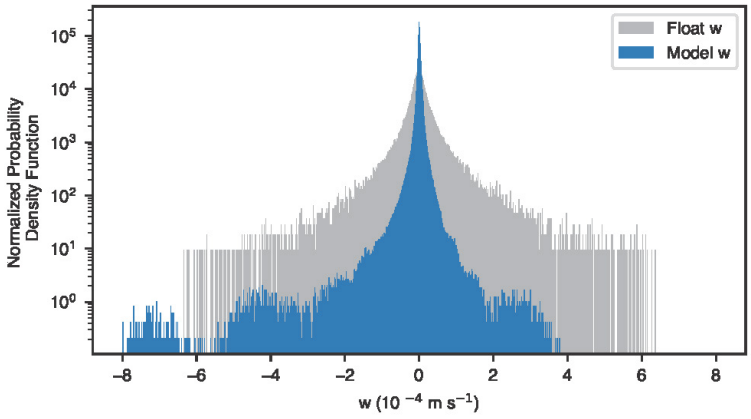

Figure 2.10: Distributions of model vertical velocity (blue) and observational estimates (gray), with each bin showing the count in the bin divided by the total number of observations and by the bin size (1×10^{-6} m s^{-1}).

model field or inadequate representation of mesoscale motions in the model. In addition, the tails of the float-based distribution decrease monotonically, while the model-based distribution exhibits multiple local maxima in both tails, likely due to limitations of the model's vertical resolution [Wong et al., 2020; Forget et al., 2015; Liang et al., 2017].

Advecting a synthetic float using model velocity fields and then recreating our vertical velocity estimation procedure might appear to be an ideal way to validate this technique. Unfortunately, the spatial and temporal scales required to accurately represent all the relevant dynamics, from the float interactions with the surrounding waters to oceanic mesoscale features, are immensely challenging to achieve [Swift and Riser, 1994; Liang et al., 2017; Wang et al., 2020]. Thus a direct model-based validation of the method developed here remains outside the scope of this work.

The direct estimates of 1000-dbar vertical velocity given here would not be possible without high-frequency data from the parking phase of Argo profiling floats. Though measuring vertical velocities was not an original intention of the Argo program [Riser et al., 2016; Roemmich et al., 2019], being able to adapt this globally-distributed in situ observational data set for novel methods such as the one presented here has immensely increased its scientific value. Continued improvements in float technology will allow for refinement and expansion of this method, as well as the development of others. These capabilities are made possible by the dedicated efforts of the international Argo program to ensure that all float data (i.e. trajectory, technical, metadata, and profile data) are provided in a consistent, quality-controlled format.

The estimates of w computed here directly from Argo float drift observations fit well within the range of values given by prior studies [Stommel and Arons, 1959; Liang et al., 2017; Martin and Richards, 2001; Pilo et al., 2018; Freeland, 2013; Cusack et al., 2017]. The distribution of these velocities, characterized by a narrow peak and heavy tails, emphasizes the distinctly non-Gaussian nature of vertical flows in the interior ocean. The variability in 5-day average w at 1000 dbar revealed here provides widespread direct observation-based evidence of the importance of topographic interactions in generating strong vertical motions at the oceanic mesoscale. This novel application of the Argo data set greatly expands our knowledge of the subsurface circulation of the global ocean and provides a unique observational constraint for model validation.

Chapter 3

Baroclinic conversion from Argo float observations

3.1 Introduction

Wind, buoyancy, and tidal forcing are major external contributors of energy to the ocean, driving motion across all spatial and temporal scales [Huang, 2004; Wunsch and Ferrari, 2004; Von Storch et al., 2012]. The complex processes of energy transformation, exchange, and dissipation influence carbon storage and air-sea interactions, thereby affecting the global climate system [Huang, 2004; Gruber et al., 2009]. The Southern Ocean is a high-energy region that is saturated with eddies associated with instabilities [Ferrari and Wunsch, 2009; Youngs et al., 2017] and has a disproportionate impact on Earth's climate, accounting for approximately 40% of the global uptake of anthropogenic carbon dioxide [Gruber et al., 2009], making it a key area for understanding underlying dynamical processes. However, the full scope of how energy moves and changes in the Southern Ocean is not fully understood.

Prior research has divided ocean energy into reservoirs of Kinetic Energy (KE) related to the direct displacement of fluid parcels, and Available Potential Energy (APE) influenced by gravitational, internal, and pressure work elements [Huang, 2004; Von Storch et al., 2012] (See Section 1.1). In general, large increases in APE should be accompanied by increases in KE, though exceptions do occur [Lorenz, 1955]. These two reservoirs are further divided into mean (MKE and MAPE) and eddy (EKE and EAPE) components, where the eddy state is defined as the deviation from the time-mean [Lorenz, 1955; Von Storch et al., 2012; Chen et al., 2014; Youngs et al., 2017]. Of particular interest for understanding dynamical processes in the Southern Ocean are the pathways between each of the four reservoirs of energy. This transfer of energy between different regimes can be caused by eddy momentum fluxes, horizontal eddy density fluxes, or vertical motion [Chen et al., 2014; Youngs et al., 2017]. Here, we focus on the vertical motion that is a fundamental component in the exchange between EAPE and EKE, known as baroclinic conversion. This transformation of energy can be mathematically represented as

$$\text{EAPE} \rightarrow \text{EKE} = -g\overline{(\rho' w')} \tag{3.1}$$

where g is the gravitational constant, ρ' is the density anomaly, and w' is the vertical velocity anomaly, while the notation $\overline{(\cdot)}$ represents the mean value [Von Storch et al., 2012].

The baroclinic energy conversion pathway is most often associated with areas of strong horizontal density gradients that reflect high EAPE. These density gradients present as sloping isopycnals maintained by

the constant addition of energy into the system from wind and other sources [Ferrari and Wunsch, 2009]. While the winds driving the Southern Ocean can vary, the isopycnal slopes have been shown to be robust in time meaning that the mesoscale eddies that are generated to compensate are consistently "saturating" the area [Hogg et al., 2008; Marshall et al., 2017; Youngs et al., 2017]. Baroclinic conversion occurs around these sloping isopycnals due to instabilities forming from perturbations in the horizontal density gradient [Lorenz, 1955; Wunsch and Ferrari, 2004; Chen et al., 2014; Youngs et al., 2017]. These perturbations can occur as a large-scale flow moves over topographic features, such as in the eastward-flowing current that is a major component the Southern Ocean, the Antarctic Circumpolar Current (ACC) [Youngs et al., 2017; He et al., 2024]. They are then resolved by the vertical motions in eddies on either side of the flow. The relaxation of the isopycnal slopes causes the gravitational center of the system to decrease, thereby diminishing the overall APE [Ferrari and Wunsch, 2009]. Baroclinic conversion is an important pathway from APE to KE, and ultimately leads to irreversible mixing processes [Wunsch and Ferrari, 2004].

The instabilities driving baroclinic conversion most often come in the form of turbulent eddies across many scales in the ocean [Lorenz, 1955; Pierrehumbert and Swanson, 1995; Chen et al., 2014; Youngs et al., 2017]. While large-scale circulation cannot be fully analyzed in isolation from the smaller scales, the mesoscale is a key component for understanding the mechanisms of energy conversion as it contains nearly 80% of all kinetic energy in the ocean [Ferrari and Wunsch, 2009]. Occurring on temporal scales of weeks to years and spatial scales of tens to hundreds of kilometers, mesoscale eddies were historically unresolved in climate models [Gent et al., 1995]. Because mesoscale eddies were poorly characterized in these early climate models, they do not accurately represent the associated vertical motions, and thereby baroclinic conversion. Recent climate models are now able to resolve eddies and their contribution to energy conversion, though there is some variability between different models caused by varying resolutions and ocean energy consistencies [Wunsch and Ferrari, 2004; Mazloff et al., 2010; Liang et al., 2017].

These inconsistencies present a need for widespread observational estimates of baroclinic conversion for comparison. However, difficulties in sampling at adequate resolution, depth, and frequency in the global oceans have previously made this challenging [Ferrari and Wunsch, 2009]. Many variables of ocean observations have been limited to either the entire water column at confined study sites or only the surface values measure by satellites across the globe [Huang, 2004]. Specifically, subsurface observations of mesoscale vertical velocity necessary for calculating baroclinic conversion are not commonplace due to their small magnitudes and high spatial variability compared to the horizontal flow [Liang et al., 2017]. Many observational estimates of vertical velocity are limited in spatial coverage or assumptions of homogeneity [Stommel and Arons, 1959; Freeland, 2013; Sévellec et al., 2015]. However, a novel method has been developed to estimate mesoscale vertical velocities that can be used in calculating baroclinic conversion from the global Argo array [Christensen et al., 2024].

3.2 Data and Methods

Argo floats have become a major source for observational oceanographic data in the world's oceans since their initial deployment as a global array [Riser et al., 2016; Roemmich et al., 2019; Wong et al., 2020; Claustre et al., 2020]. Each float continuously repeats a profiling cycle, where the float descends to 1,000 dbar, drifts with the flow at that level for approximately 10 days, descends again to 2,000 dbar, and finally ascends to the surface. Once on the surface, the float transmits all the data collected during the ascending profile and the drift (if applicable) using Iridium communications. These data include measurements of conductivity, temperature, and depth (CTD) as well as location and time information. While there is an assortment of other parameters that may be available from certain floats [Claustre et al., 2020], CTD data

are most applicable for computing baroclinic energy conversion. To examine just the Southern Ocean, we use all the CTD data from floats poleward of 25°S with quality control flags equal to 1 or 2, or data that have been adjusted to reach an equivalent level [Wong et al., 2020].

Based on the framework provided by Equation 3.1, we require observations of vertical velocity and density to complete our baroclinic conversion calculations. We estimate 5-day mesoscale vertical velocities at 1,000 m in the Southern Ocean from August 2005 to April 2022 following the same methodology and quality control metrics outlined in Christensen et al. [2024]. We first find the isotherm displacement around the float as it drifts at 1,000 m, followed by filtering out small-scale motions and fitting the data. Then, we find the time rate of change of the isotherm displacement function and take the average to be the equivalent to a vertical velocity for each half cycle. For each estimate of vertical velocity, we then calculate a corresponding density value. Using CTD profile data taken directly before and after the drift component of the Argo float cycle used to estimate the vertical velocity, we find the average value for density at 1,000 m $\pm$ 100 m, computed using the equation of state referenced to 1,000 m [McDougall and Barker, 2011]. With these values of vertical velocity and density, we now have all the inputs required for finding baroclinic conversion at 1,000 m. However, we must first separate out the anomaly components, which requires estimates of the mean field for each variable.

To get the mean field of both parameters, we perform local least-squares spatial regression in a 500 km circular window for each point of a $0.5^\circ \times 0.5^\circ$ grid in the Southern Ocean [Kuusela and Stein, 2018; Park et al., 2023]. The mean function applied takes the form

$$m(x,t) = \beta_0 + \beta_x x_c + \beta_y y_c + \beta_{xy} x_c y_c + \beta_{x^2} {x_c}^2 + \beta_{y^2} {y_c}^2 \tag{3.2}$$

where each β is a coefficient of fit and x_c and y_c are the longitude and latitude coordinates centered to location x_* and y_* at the center of the regression window. We do not include a temporally varying component to this regression due to the lack of seasonal influence on the vertical velocity data at 1,000 m [Christensen et al., 2024]. The stringent requirements for data availability and quality used in finding vertical velocity estimates cause there to be some sparse areas of observation that are not conducive to calculating a mean value, particularly in the Atlantic sector of the Southern Ocean. However, there is adequate data coverage elsewhere to allow us to compute both the mean and anomaly values for vertical velocity and density (Figure 3.1). To account for the lack of data in some areas, we impose that a regression window must contain at least 200 estimates of vertical velocity and density. Additionally, the geographic centroid of those samples must be located within 50% of the total size of the window radius, or 250 km, from the grid point.

These gridded mean fields are then linearly interpolated back to the original locations of the float data (Figure 3.1, a and c) where we find the anomaly values for vertical velocity ($w' = w - \overline{w}$) and density ($\rho' = \rho - \overline{\rho}$) from every half cycle (Figure 3.1, b and d). To calculate the baroclinic conversion, we multiply these two anomalies and use the same least-squares regression method detailed by Equation 3.2 to find the mean value of $\rho' w'$. To limit the effects of outliers in the data, we remove grid cells with magnitudes larger than 8 standard deviations from the mean. This allows us to create a gridded product of baroclinic conversion that can easily be compared to other gridded fields, such as models.

We use fields from the Southern Ocean State Estimate (SOSE) to compute a baroclinic conversion for model comparison. SOSE is a $\frac{1}{6}^\circ \times \frac{1}{6}^\circ$, eddy-permitting general circulation model that is constrained by observations [Mazloff et al., 2010]. We use the 5-day averaged output from the Version 2 solution from January 1, 2005, to December 31, 2010. Taking the salinity and potential temperature from the state estimate, we calculate the density using the equation of state referenced to 1,000 m [McDougall and Barker, 2011]. We then find the time-mean fields of the density and vertical velocity for all grid points. From this point, we follow the same procedure for finding baroclinic conversion as prescribed by Equation 3.1: subtract the

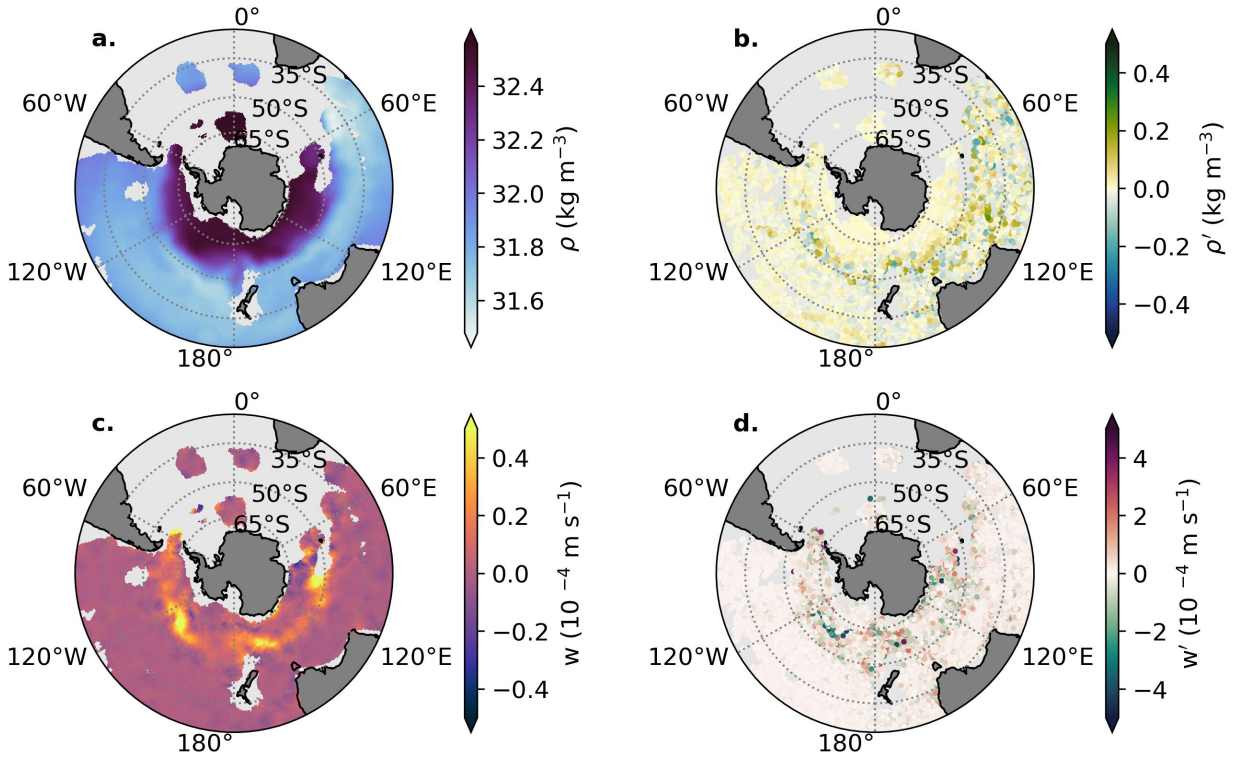

Figure 3.1: Results of the local least-squares spatial regression at 1,000 m for density and vertical velocity. a) Mean and b) anomaly density from Argo float profiles referenced to 1,000 dbar; c) mean and d) anomaly 5-day vertical velocity estimates from Argo float drift data. The anomalies shown here are obtained by subtracting the mean field from the original float data after linearly interpolating it to the profile location.

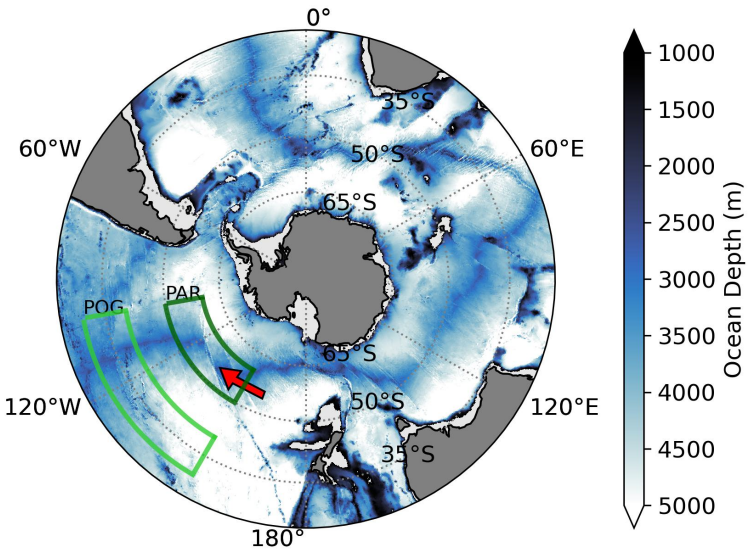

Figure 3.2: Ocean depth below 1,000 m from 1 Arc Minute Global Relief Model [Amante and Eakins, 2009; NGDC, 2009]. Boxes highlight areas of interest: the Pacific Ocean Gyre (POG - light green) and the Pacific-Antarctic Ridge (PAR – dark green). The red arrow denotes the direction of flow in the ACC.

mean values to find the anomalies of w and ρ, multiply the anomalies together, and take the final time-mean multiplied by the gravitational constant. We then apply the same local least-squares spatial regression to implement a spatial average that is comparable to the methods performed on the observational data. This has the benefit of having the SOSE baroclinic conversion on the same grid as the gridded observations from floats ($0.5^\circ \times 0.5^\circ$). Applying this same method to add a spatial average to the model field allows for better comparison between the floats and model field.

3.3 Results

Focusing on the areas of high data density clockwise from roughly 60°E to 60°W, we see that baroclinic energy conversion occurs non-uniformly in space (Figure 3.3a). Intuitively, this distribution is consistent with areas of high magnitude vertical velocity and density anomalies (Figures 3.2 and 3.1). Baroclinic conversion at 1,000 m in the Southern Ocean ranges from -5 to 5×10^{-5} kg m^{-1} s^{-3} (Figure 3.3b). The average value is 1.8 $\times 10^{-6}$ kg m^{-1} s^{-3}, with a standard deviation of 4.8 $\times 10^{-6}$ kg m^{-1} s^{-3}. The distribution is asymmetrical, with a large positive skew of 2.6, which means there is a net conversion from EAPE to EKE. This is an expected result for the area due to eddies actively forming in the ACC [Youngs et al., 2017]. There are some negative values throughout the dataset, which implies that there are other mechanisms at play than the baroclinic instabilities that balance sloping isopycnals, but most of the data are positive.

The ACC contributes greatly to this skewness, with the largest positive values occurring along the track of the current (Figures 3.2 and 3.3a). Areas downstream of topographic features are especially prominent; Macquarie Ridge ($\sim 150^\circ$E, 50-60°S), the Kerguelen Plateau ($\sim 65^\circ$E, 40-50°S), and the Pacific-Antarctic Ridge (PAR) ($\sim 150^\circ$W, 50-60°S) all show large and positive baroclinic conversion occurring downstream of where they intersect with the ACC [Christensen et al., 2024].

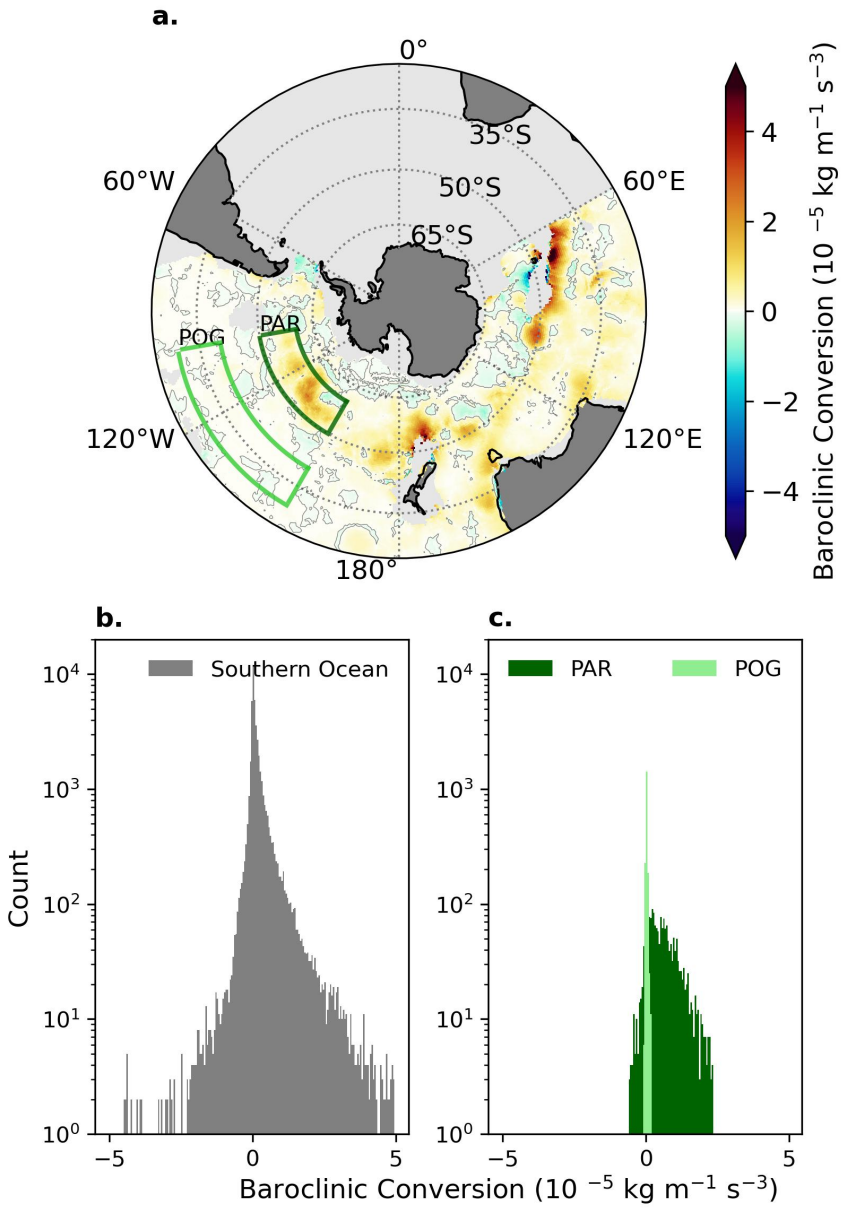

Figure 3.3: Estimates of the baroclinic conversion of energy at 1,000 m from Argo float data. Locations with magnitudes larger than 8 standard deviations from the mean have been removed. Positive values denote an exchange from EAPE to EKE. a) Map of gridded estimates ($0.5° \times 0.5°$); boxes highlight areas of interest: the Pacific Ocean Gyre (POG - light green) and the Pacific-Antarctic Ridge (PAR – dark green).The thin contour line denotes a shift from negative to positive while areas without adequate data coverage are masked with light gray. b) Histogram of all the data mapped in panel (a), with bin sizes of 5×10^{-7} kg m^{-1}s^{-3}. c) Histogram of data within the boxes (POG, PAR) from panel (a), with bin sizes of 5×10^{-7} kg m^{-1}s^{-3}. Note that the y-axes for panels (b) and (c) are shown on a logarithmic scale.

Here, we specifically highlight the PAR as it can easily be compared to the Pacific Ocean Gyre (POG) directly to the north, which is not in the ACC core. It also has the highest data density available from floats (Figure 3.2). In the PAR area (Figure 3.3, dark green), we see a range of baroclinic conversions from -1 to 3 $\times 10^{-5}$ kg m^{-1} s^{-3} (Figure 3.3c), with an average value of 6.2$\times 10^{-6}$ kg m^{-1} s^{-3} and standard deviation of 5.7$\times 10^{-6}$ kg m^{-1} s^{-3}, both of which are somewhat larger than the same metrics of the entire Southern Ocean. Comparing this to the POG area (Figure 3.3, light green), with an average value of 2.5$\times 10^{-7}$ kg m^{-1} s^{-3} and standard deviation of 2.7$\times 10^{-7}$ kg m^{-1} s^{-3}, we see that the PAR area has an order of magnitude larger conversion of energy from EAPE to EKE (Figure 3.3c). This comparatively larger magnitude is similar to what we see at other topographic features, while the smaller magnitude POG area is more consistent with other quiescent areas. These two different regimes shape the profile of baroclinic conversion in the entire Southern Ocean (Figure 3.3b).

Because observational estimates of baroclinic conversion are rare [Huang, 2004; Ferrari and Wunsch, 2009], we seek to put these results into context by comparing to model values of baroclinic conversion. As detailed in Section 3.2, we computed the baroclinic conversion term using output from SOSE. Though the model is data-constrained, it does not incorporate the Argo float vertical velocities, so it is independent of the float estimates in this case. The spatial pattern of baroclinic conversion at 1,000 m from the model agrees well with that of the float estimates (Figure 3.4a). Because we have included the local least-squares regression average for finding a mean-state baroclinic conversion in SOSE, the spatial distribution for baroclinic conversion from float estimates and from the model are quite similar. The model does appear somewhat more homogeneous and slightly reduced in magnitude due to the larger number of data points that are included in each window for averaging. However, we see the same spatial patterns of areas with high magnitudes.

These high-magnitude baroclinic conversions occur at the same topographic features in the path of the ACC (PAC and PAR) that the float observations highlighted (Figures 3.1, 3.2, and 3.3). The area surrounding Drake Passage did not have adequate observational data density to characterize the same high magnitude that is present in the model, but it is highly probable we would have the same result from our observational methods if there were more floats in the area.

The scale of baroclinic conversion from SOSE is similar to the observational estimates and has a range of values from -2 to 2 $\times 10^{-5}$ kg m^{-1} s^{-3} (Figure 3.4b). Though these values tend to be somewhat smaller than those from the floats, they are on the same order of magnitude. The average baroclinic conversion in SOSE is 5.6 $\times 10^{-7}$ kg m^{-1} s^{-3} and is considerably less positively weighted than the float estimates. There are far fewer large, positive values in the model than there are from observations. This is likely due to how the vertical velocity is calculated in the model as the divergence in horizontal velocities.

3.4 Summary

These observational estimates of baroclinic conversion at 1,000 m in the Southern Ocean clearly show that the transformation from EAPE to EKE is heavily impacted by strong horizontal flow interacting with topographic features. This reinforces our understanding of eddy formation and EKE distribution in the ACC, which has previously been characterized by through models [Youngs et al., 2017]. In general, large EKE anomalies are driven by baroclinic conversion [He et al., 2024]. As such, the topographically influenced areas of high baroclinic conversion are accompanied with high EKE values.

The agreement between our observational estimates and the SOSE model fields of baroclinic conversion demonstrates that this method is robust. This corroborates that mesoscale energetics are a dominant part of EKE generation, both in observations and models [Ferrari and Wunsch, 2009].

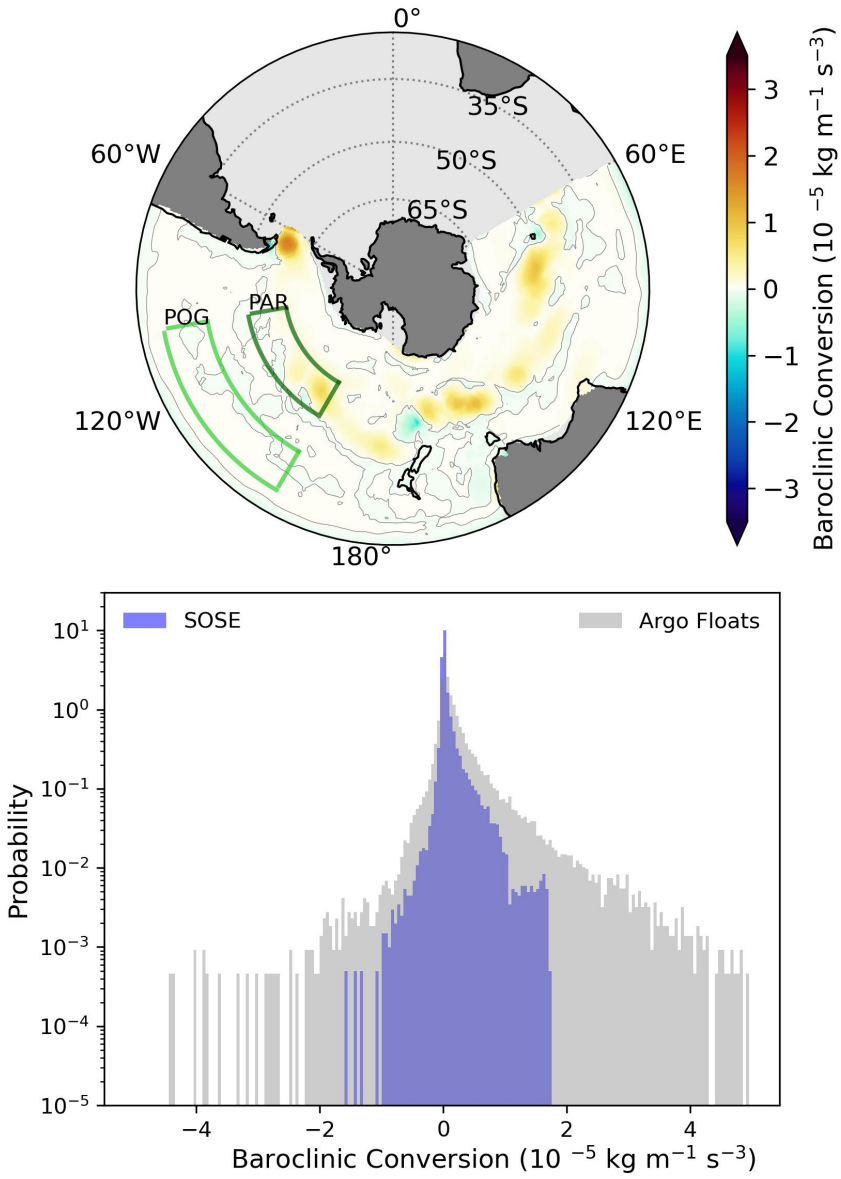

Figure 3.4: Gridded ($0.5° \times 0.5°$) estimates of the baroclinic conversion of energy at 1,000 m from SOSE model data. Positive values denote an exchange from EAPE to EKE. Boxes highlight areas of interest: the Pacific Ocean Gyre (POG - light green) and the Pacific-Antarctic Ridge (PAR – dark green).The contour line denotes a shift from negative to positive while areas without adequate float sampling are masked with light gray.

Our estimates are somewhat limited by only being available at a single depth level, by nature of where the floats are programmed to drift. Though EKE persists throughout the water column, the baroclinic conversion of energy is not constant at all depths [He et al., 2024]. Depth-integrated values of baroclinic conversion in strong horizontal flows interacting with bathymetry range from 0.1 to 0.3 kg m^{-2} s^{-3} [Chen et al., 2014; Youngs et al., 2017], making these observations just one component to understanding the full system.

These estimates from floats at 1,000 m are some of the first observationally derived values for sub-surface baroclinic conversion of energy across a large area of the world's oceans. As the global Argo array continues sampling, more data will become available to help constrain the areas with currently sparse float profiles.

Chapter 4

Eddy Kinetic Energy and its vertical structure

4.1 Introduction

Circulation in the ocean is driven by the input, transformation, and dissipation of energy across all scales. This energy budget is a key component to understanding the Earth's climate through the storage and flux of carbon in the ocean [Gruber et al., 2009]. Though energy flows through all scales, the energy partitioned to the mesoscale ocean is fundamental to this system as it contains nearly 80% of all kinetic energy in the ocean [Ferrari and Wunsch, 2009]. However, the vertical structure of mesoscale energy below the surface of the ocean is not well studied. Therefore, it is not fully understood how energy is attenuated from the surface to the ocean interior.

In particular, Eddy Kinetic Energy (EKE) has only been calculated globally at the surface using satellite measurements of sea surface height (SSH) to estimate horizontal velocities. These satellite derived fields show a lot of spatial variability within EKE, with the highest magnitudes occurring in fast-moving currents, such Western Boundary Currents and the Antarctic Circumpolar Current (ACC) [Martínez-Moreno et al., 2021].

Previous observations of EKE within the water column are from observational campaigns involving moorings and gliders that are spatially and temporally constrained [de La Lama et al., 2016]. Often, these data are collected only in areas of specific interest with little knowledge of phenomena occurring outside of the selected study location. Thus, they cannot readily capture global variability of EKE below the surface.

In modern oceanography, the use of data from the global array of Argo floats is ubiquitous in global analyses of ocean data. Due to their sampling from the surface to 2000 m, approximately every 10 days, they are an extremely useful tool for capturing the mesoscale vertical structure of the ocean interior across the globe [Wong et al., 2020].

Here, we use gridded estimates of EKE calculated from the horizontal velocity derived at the surface from satellites and at multiple depth levels from Argo floats. From these profiles of EKE, we use linear dynamical modes to examine the vertical structure. We do this using two different boundary conditions for formulating the modes [Early et al., 2020], a flat-bottomed ocean [Wunsch, 1997] as well as a rough-bottomed ocean [de La Lama et al., 2016; LaCasce and Groeskamp, 2020].

4.1.1 Framework for applying vertical modes to EKE

Our estimates of EKE are reliant on knowledge of the horizontal flow, following the mathematical representation shown here (the same as Equation 1.4):

$$\text{EKE}(x, y, z) = 0.5\overline{(u'^2 + v'^2)} \tag{4.1}$$

where u and v are the zonal and meridional components of horizontal velocity, the overbar ($^-$) denotes the mean field, and the prime ($'$) denotes an anomaly from the mean field (e.g. $u^{'} = u - \overline{u}$). To better understand the vertical structure of EKE, we can use the Rossby wave vertical modes for horizontal velocities that satisfy

$$\frac{d}{dz}\left(\frac{f_0^2}{\mathcal{N}^2(z)}\frac{dF(z)}{dz}\right) + \gamma^2 F(z) = 0 \tag{4.2}$$

which denotes an eigenvalue problem, where solving for the eigenvalues γ gives us the vertical dynamic modes [Gill, 1982; Wortham and Wunsch, 2014; Fuhr, 2015; LaCasce and Groeskamp, 2020]. With this, we can apply different boundary conditions to determine how topography affects the vertical structure of EKE.

Assuming a constant $\mathcal{N}^2(z)$, we can apply either a flat-bottom solution, where $\frac{du}{dz} = 0$ at the bottom and surface, or a rough-bottom solution, where $u = 0$ at the bottom (Figure 4.1). These two boundary conditions are most different from each other at the bottom, but do have some variability in the top 2,000 meters where data from Argo floats are readily available. Most notably the depths of where the profiles cross zero changes between the different regimes.

Using the solutions $F(z)$ to the eigenvalue problem, the horizontal velocities can then be represented in some number, N, of modes as

$$\mathbf{u} = \sum_{n=0}^{N} \alpha_{(n,\mathbf{u})}(t) F_n(z) + \varepsilon_{\mathbf{u}}(z, t) \tag{4.3}$$

where $\mathbf{u}$ consists of both the zonal (u) and meridional (v) components of horizontal velocity and $\alpha_{(n,\mathbf{u})}(t)$ is the nth coefficient. The residual is represented as $\varepsilon_{\mathbf{u}}(z, t)$, which denotes the structure that is not explained fully in N modes. It has previously been found that a large amount of the velocity structure in the ocean is indeed explained in the first few modes [Ni et al., 2023], making $\varepsilon_{\mathbf{u}}(z, t)$ quite small. In oceanographic literature, when $n = 0$ for the flat-bottom condition it is commonly referred to as the barotropic mode and it has been shown to account for up to 80% of the variance in structure for some locations [de La Lama et al., 2016]. Though this barotropic mode does not exist when using the rough-bottom boundary condition, an equivalent topographic mode has previously been implemented (Figure 4.1 F_0), which represents topographic waves being produced by interaction with the topography due to the conservation of potential vorticity [Fuhr, 2015]. Combining Equations 4.2 and 4.3, we can estimate the vertical structure of the EKE from

$$\text{EKE} = \frac{1}{2}\sum_{n=0}^{N}\left(\alpha_{(n,u)}^2 + \alpha_{(n,v)}^2\right) F_n^2(z) \tag{4.4}$$

provided that the coefficients are not correlated in time. If this is not the case, cross-terms play a role in the variance of the EKE structure, which creates a far more complex system [Wortham and Wunsch, 2014].

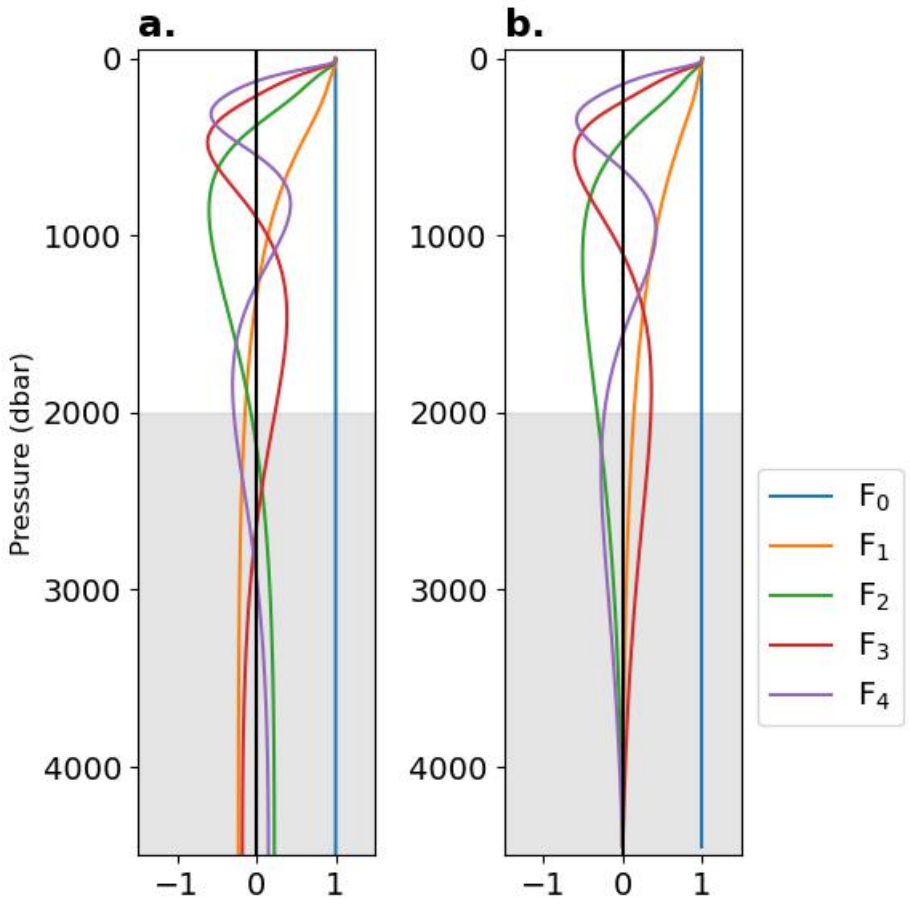

Figure 4.1: Normalized profiles of the first 5 modes at 36°N and 160°E for a) flat-bottom and b) rough-bottom boundary conditions. Depths below the maximum depth for our Argo estimate EKE profiles (2,000 m) are shaded in gray.

4.2 Data and methods

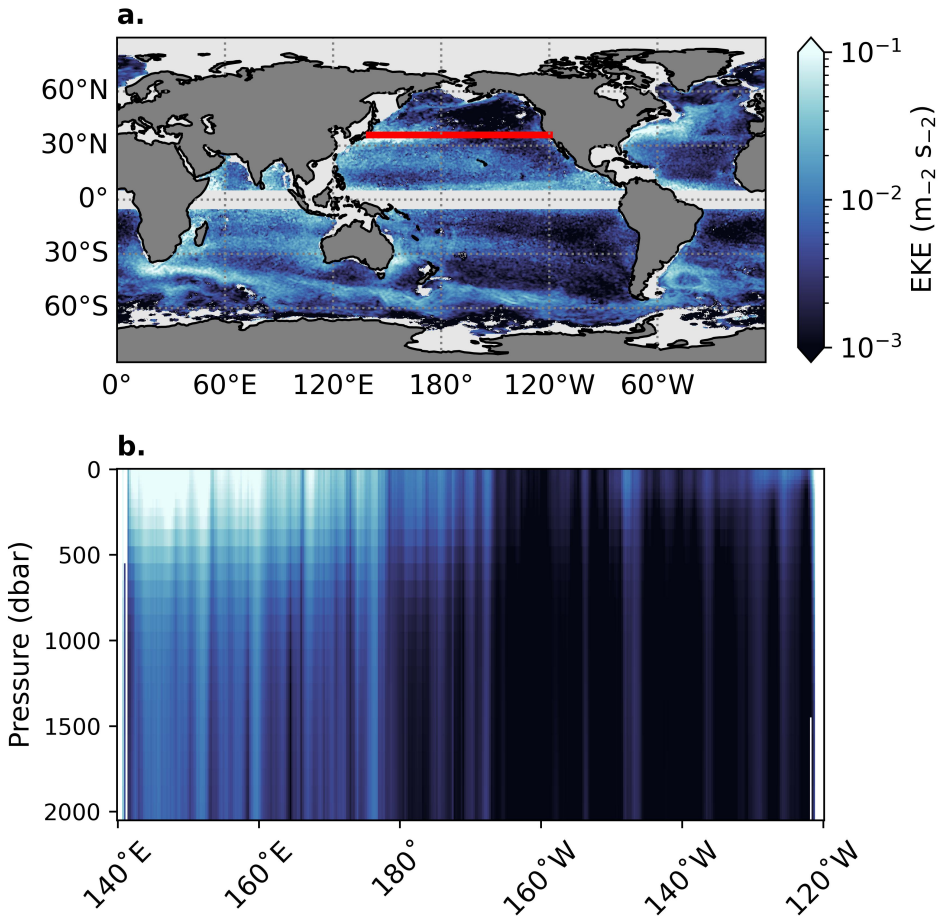

Figure 4.2: a) Map of gridded global EKE estimates at the surface and b) a slice of EKE of depth vs. longitude along the red line in panel (a) at 36°N. Note both panels use the same shading, consistent with the colorbar present in panel (a).

To calculate EKE, global horizontal velocities are derived using geostrophic velocities measured from satellites (surface) and from floats (ocean interior). Specifically, the velocities from the Argo array are calculated following the methods developed by Gray and Riser [2014]. Following their framework, we find profiles of geostrophic shear at 29 depth levels estimated from the float profile measurements of temperature and salinity. Any level that does not have adequate measurements to calculate the shear is discarded. Additionally, we implement a 5° buffer around the equator as assumptions of geostrophy deteriorate with a growing Rossby number [Gray and Riser, 2014]. We then reference each level of geostrophic shear to the surrounding levels emanating from the velocity calculated at the parking depth from the horizontal drift of the float. This give us horizontal velocity from the surface to 2,000 meters at each level (Figure 4.2). We then use these velocities in equation 4.1 to calculate EKE, with any values that are found to be larger than 10 times the standard deviation being neglected.

Because the float measurements are non-uniform in space and time, these velocities are gridded using a

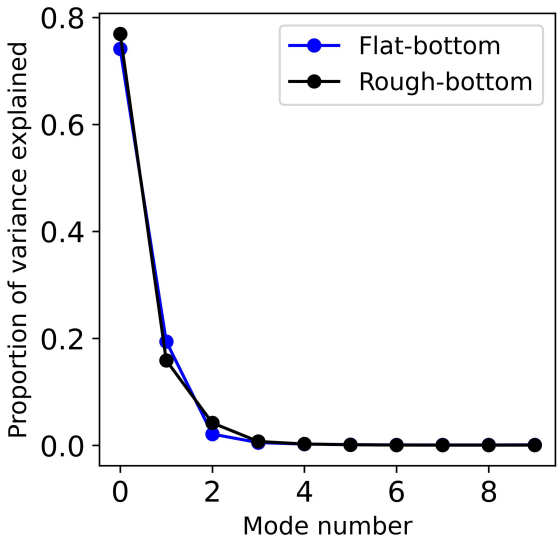

Figure 4.3: Proportion of variance explained vs. mode number for flat-bottom (blue) and rough-bottom (black) modes

locally parametric model, where the large-scale is determined by a local polynomial regression (similar to 3.2) and the small-scale is determined by zero-mean locally stationary Gaussian process [Kuusela and Stein, 2018; Park et al., 2023]. The float and satellite data are combined using least squares fit based on the Rossby wave vertical modes [Early et al., 2020].

To explore the vertical structure of EKE, we use an atlas of vertical modes that computes normalized global vertical modes that vary based on geography and stratification [Early et al., 2020], which allows us to better examine spatial variability in EKE. The mode profiles are interpolated to the same depth levels as our measurements of velocity. Then, we use a least squares fit to solve for the coefficients in Equation 4.4 for both the flat- and rough-bottom boundary condition. Our reconstruction of the EKE is based only on the first 10 modes ($N = 10$), with additional variance becoming a part of the residual. We believe this small number of modes used to reconstruct the EKE structure is adequate due to the vanishingly small residuals past the first three modes (<1% of the variance is explained with the addition of each further mode) (Figure 4.3).

Because find the proportion of variance that is explained in the first three modes is so high, these will be the focus of our analysis. Specifically, we will examine the barotropic (or equivalent topographic) mode (α_0), and the first two baroclinic modes (α_1 and α_2) for both bottom boundary conditions, as well as the sum of these different modes.

4.3 Results

On average, the proportion of variance explained is largest in the barotropic (topographic) mode for both boundary conditions, with 74% for the flat-bottom condition and 77% for the rough-bottom condition (Figure 4.3). Together, the first three modes explain approximately 96% and 97% of the vertical EKE structure, for the flat- and rough-bottom conditions, respectively. This implies that the two different solutions are almost equivalent in capturing the vertical structure of EKE in the ocean. However, this is only part of the

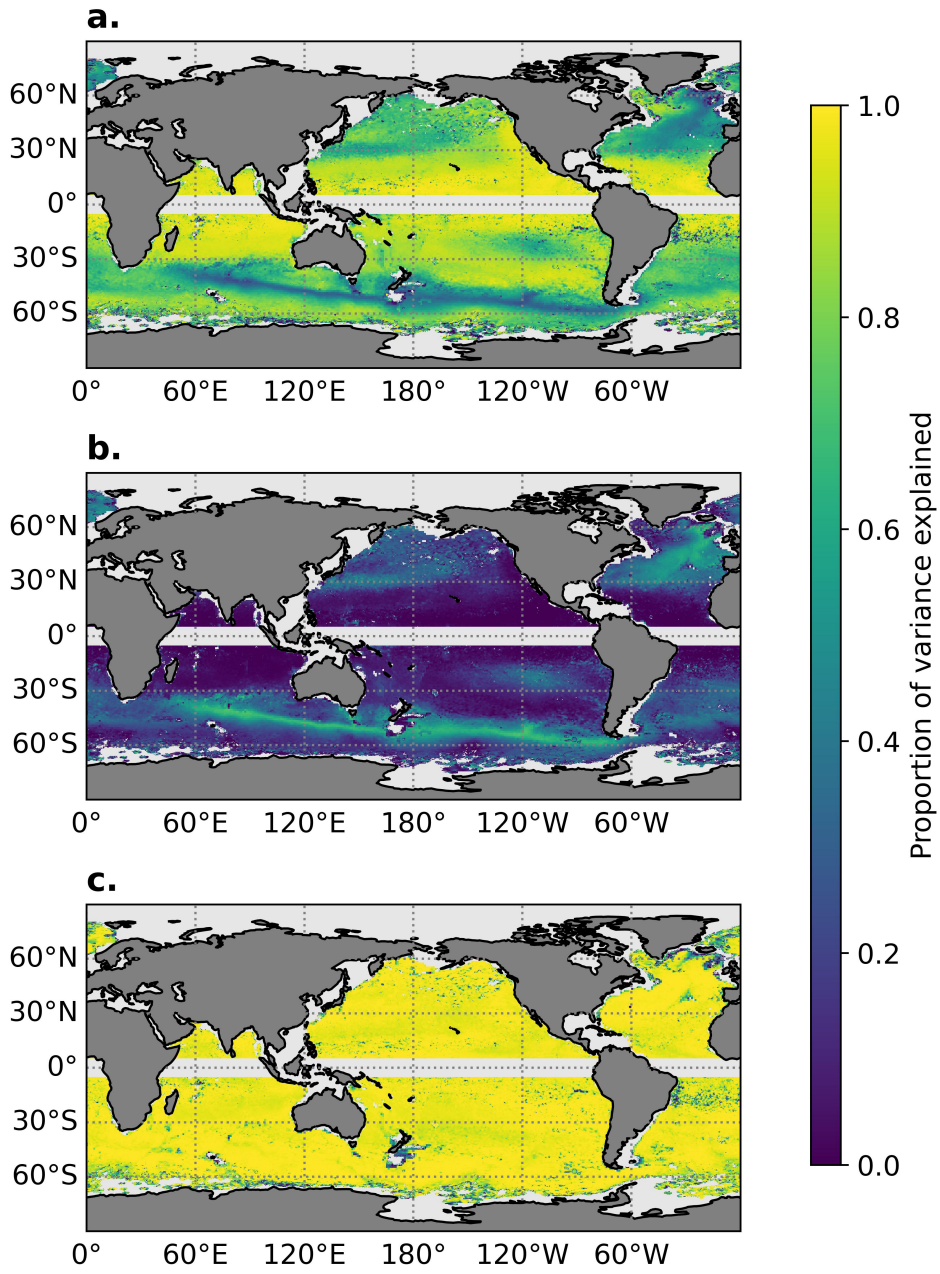

Figure 4.4: Global maps of the proportion of variance explained from the flat-bottom boundary condition by a) the barotropic mode, b) the first baroclinic mode, and c) the sum of these first two modes, where 0 is no variance explained and 1 is 100% of the variance is explained.

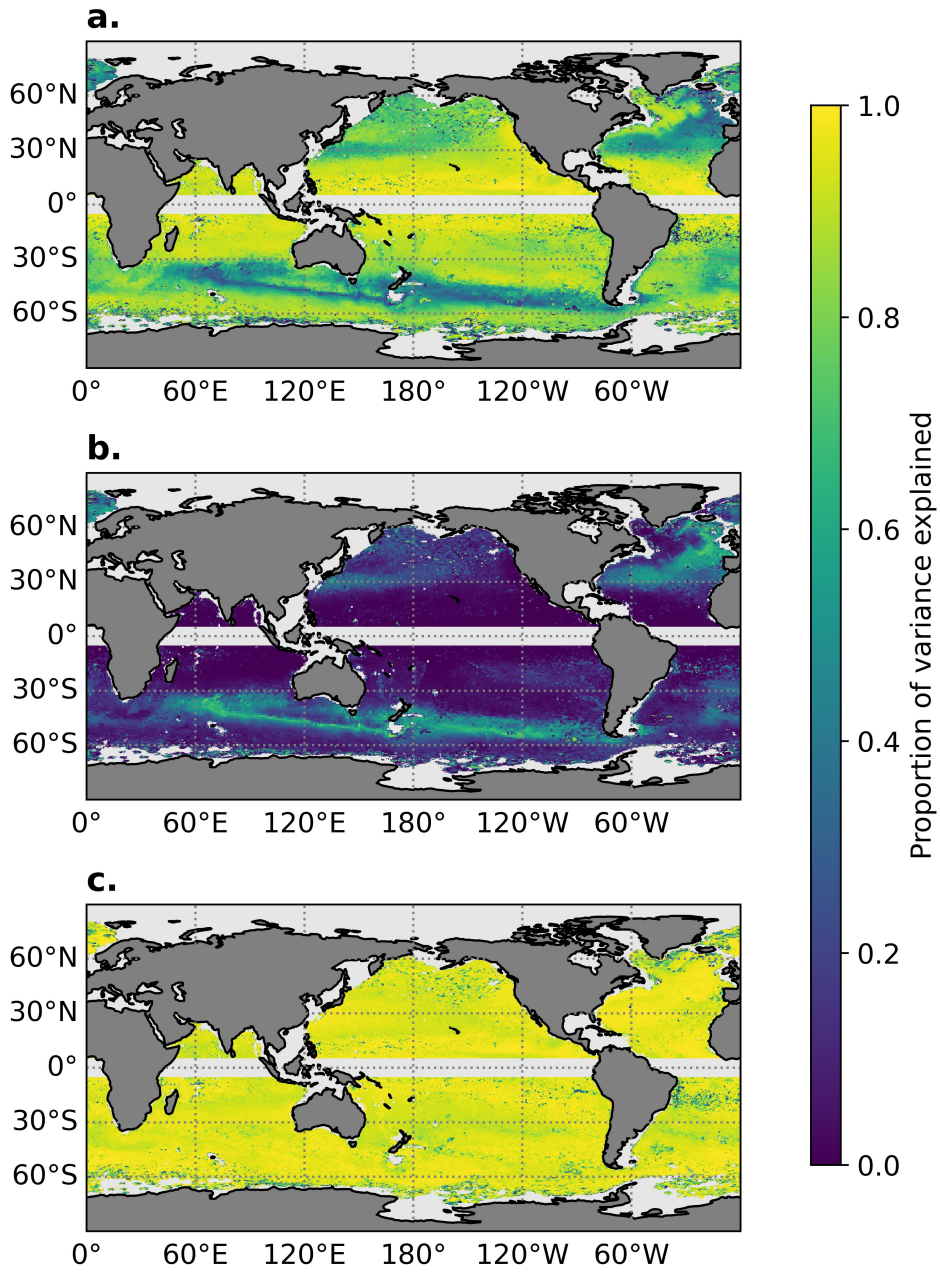

Figure 4.5: Global maps of the proportion of variance explained from the rough-bottom boundary condition by a) the topographic mode, b) the first baroclinic mode, and c) the sum of these first two modes, where 0 is no variance explained and 1 is 100% of the variance is explained.

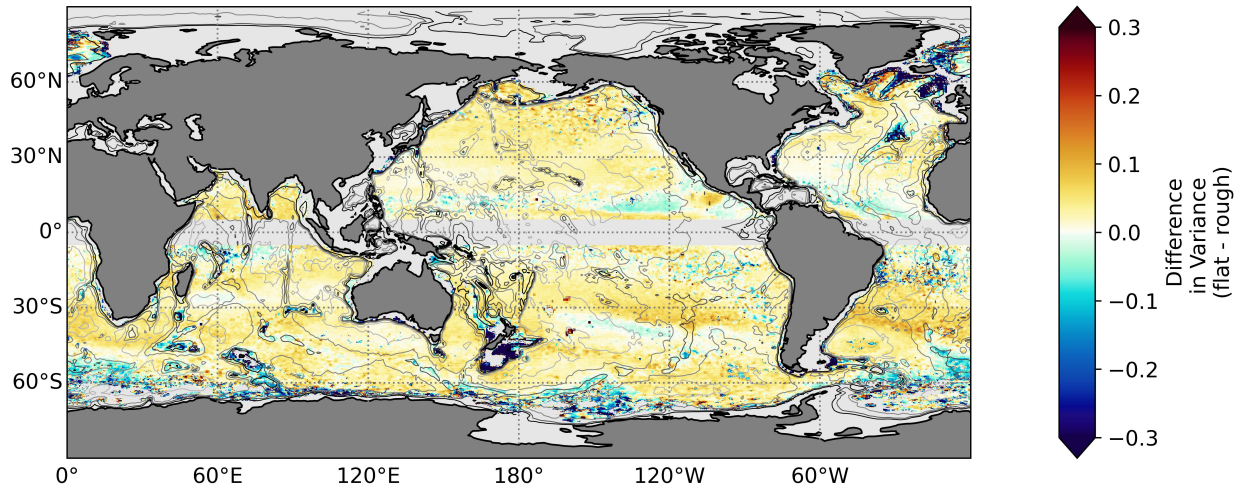

Figure 4.6: Global map of the difference in proportion of the variance explained by the sum of the first two modes of the flat-bottom minus the rough-bottom regime, where red denotes a larger proportion explained by the flat-bottom conditions and blue denotes a larger proportion explained by the rough-bottom conditions. Contours every 1,000 m (below 2,000 m) in ocean depth are included in gray with darker contours equating to shallower bathymetry.

picture as it does not address the spatial variability of EKE in the global ocean.

Looking at the amount of variance that is explained mapped in space we see a distinct pattern that is related to the flow of the ocean and topography (Figures 4.4 and 4.5). For both of the boundary conditions, we see that the barotropic and topographic modes explain the most variance in quiescent areas such as in the subtropics, while the first baroclinic mode explains most of the variance in high energy currents, such as the Kuroshio and Antractic Circumpolar Current (ACC). This result is consistent with previous analysis on horizontal velocities using vertical modes [de La Lama et al., 2016; LaCasce and Groeskamp, 2020; Ni et al., 2023]. In these strong currents, we not only expect larger values of EKE in general, but also more variability in depth that requires additional information that is provided by incorporating additional modes.

Though the first two modes are quite comparable between the different regimes, the topographic mode (rough-bottom) tends to characterize slightly more of the variance throughout the globe than the barotropic mode (flat-bottom). However, the inverse is true for the first baroclinic mode for each, where the flat-bottom first baroclinic mode captures more of the variance than the same mode in the rough-bottom scenario (Figures 4.3, 4.4a-b, and 4.5a-b).

The first two modes account for much of the variance in the vertical structure of EKE for both of the regimes throughout the globe (Figures 4.4c and 4.5c). However, when directly compared, the flat-bottom modes do not capture as much of the variance over prominent topographic features (Figure 4.6). In particular, near the Mid-Atlantic Ridge and in the ACC there are clear areas that follow the bathymetry where the rough-bottom condition better capture the variance in these small number of modes.

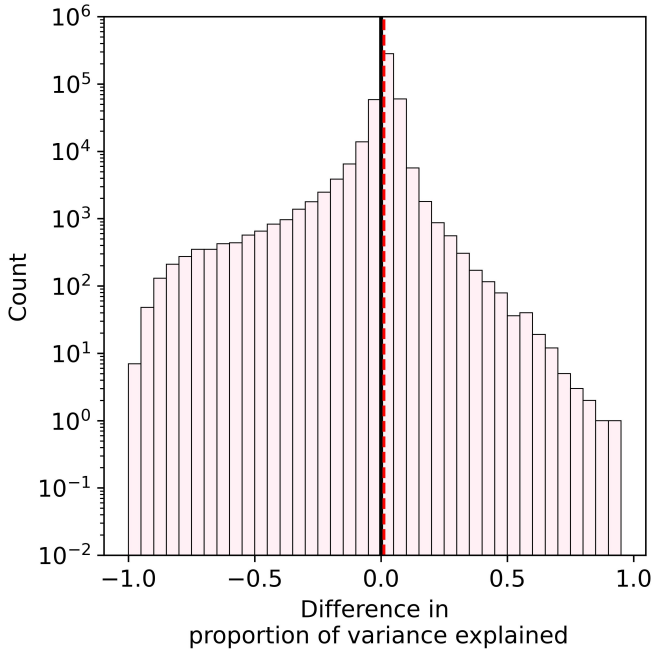

Figure 4.7: Histogram of the difference in proportion of the variance explained by the sum of the first two modes of the flat-bottom minus the rough-bottom regime with bin sizes of 0.05. The solid black line is centered on zero, while the dashed red line denotes the average difference of 0.01.

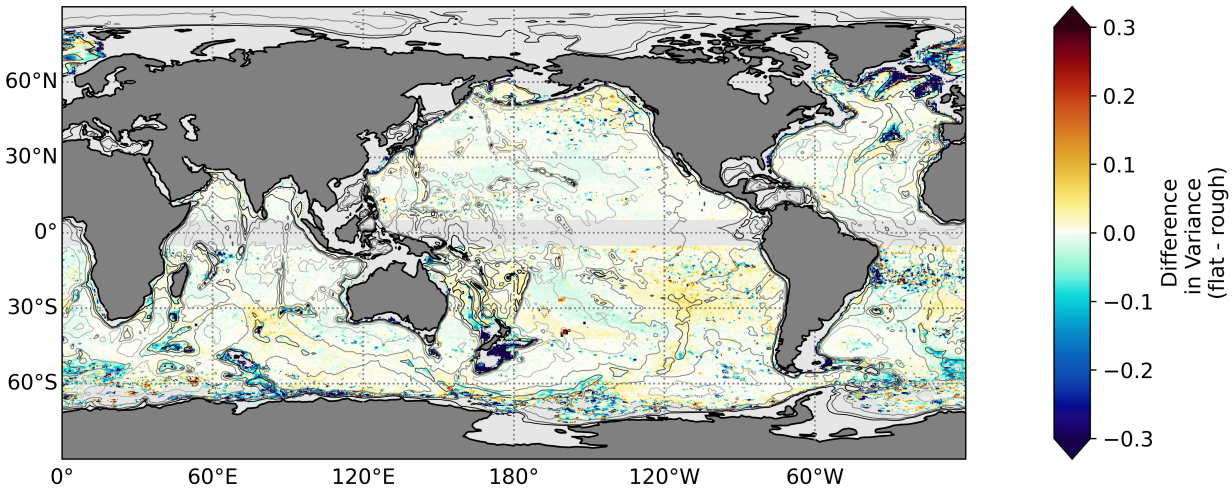

Figure 4.8: Global map of the difference in proportion of the variance explained by the sum of the first three modes of the flat-bottom minus the rough-bottom regime, where red denotes a larger proportion explained by the flat-bottom conditions and blue denotes a larger proportion explained by the rough-bottom conditions. Contours every 1,000 m (below 2,000 m) in ocean depth are included in gray with darker contours equating to shallower bathymetry.

The average difference between the flat- and rough-bottom using the first two modes is positive 0.01, meaning the flat-bottom boundary condition appears to better explain the variability in the EKE structure for large parts of the global ocean (where the difference between the variance explained by the mode types is largely positive in Figure 4.6). However, the distribution has a strong negative skew at -4.4 (Figure 4.7). This means that, in areas where the rough-bottom boundary condition is a better representation, it accounts for a considerably larger amount of the variance. As further modes are incorporated, the rough-bottom begins to do slightly better across the global oceans (where the difference becomes negative in Figure 4.8). Yet, the topographic features become somewhat less prominent, implying that bathymetry matters most in determining the best mode fit within the first two modes.

4.4 Summary

Comparing the two different boundary conditions, we can see the effects of topography on the vertical structure of EKE. Though the calculated field of EKE from the floats is from the surface to 2,000 m, rather than containing the entire water column (Figure 4.1), there is a clear signal that ocean depth matters in specific areas (Figure 4.6). Most notably, there are features in the Mid-Atlantic Ridge and in the ACC that are always best fit by the rough-bottom modes.

While additional deep samples are necessary to truly capture the best mode fit throughout the entire water column [de La Lama et al., 2016; Ni et al., 2023], these results are promising for understanding the variability in how EKE is partitioned in the ocean across wide areas. Previous estimates of EKE from moorings are not necessarily reflective of the global ocean due to mooring deployments being preferentially focused on areas with interesting physical phenomena rather than representatively sampling the entire ocean [de La Lama et al., 2016].

One limitation of this work is that the vertical structure of EKE can only be truly represented by the horizontal velocity modes if the modes are uncorrelated in time, otherwise cross-terms of the different modes play a role in the structure of EKE (Equation 4.4). Research has shown that the modes are likely correlated [Wortham and Wunsch, 2014], so there may be additional variance that is explained within the first few modes that is not attributed here.

These are some of the first global estimates of subsurface EKE. This gridded product incorporates multiple sources of data (satellites and floats) to capture an in-depth view of ocean energy. Understanding the vertical structure of EKE helps to quantify the attenuation of energy in depth and better capture the effects of mesoscale circulation on the global system.

Chapter 5

Evidence-Based Teaching for Python in an Undergraduate Earth Science Setting

This is a collaborative chapter with E.C. Campbell and has been published as: Ethan C. Campbell, Katy M. Christensen, Mikelle Nuwer, Amrita Ahuja, Owen Boram, Junzhe Liu, Reese Miller, Isabelle Osuna, and Stephen C. Riser. 2024. Cracking the code: An evidence-based approach to teaching python in an undergraduate earth science setting. *Journal of Geoscience Education*, pages 1–20

5.1 Introduction

Data programming has become a foundation of research in today's geoscientific disciplines. As the volume and size of data sets have steadily increased, so have the complexity and ubiquity of the computational techniques used for analysis and visualization. Some argue that innovation in earth science research will increasingly be driven by one's competency in translating ideas into computer code [Jacobs et al., 2016].

The field of oceanography is no exception to this "data tsunami," with more hydrographic casts collected in the past two decades than over the previous 100 years [Brett et al., 2020]. Unprecedented collaborative initiatives such as the Argo profiling float array [Wong et al., 2020], the National Science Foundation's Ocean Observatories Initiative (OOI; Greengrove et al. [2020]), and remote sensing platforms such as satellite altimeters [Scheick et al., 2023] are continuously adding to expansive, publicly available data sets. In addition to these observational programs, hard drives at institutions across the world are being filled with terabytes of data generated by numerical simulations. From highly resolved ocean general circulation models to the lower-resolution global climate models assessed in the Intergovernmental Panel on Climate Change (IPCC) reports, the natural ocean is being reproduced with ever-increasing fidelity [Haine et al., 2021]. The resulting challenges in accessing and analyzing these data require new computational tools that enable truly open science, further motivated by the notion that "research conducted openly and transparently leads to better science" [of Sciences Engineering and Medicine, 2018]. At the same time, modeling and observation-focused oceanographers use highly unstandardized computational methods that may deviate from best practices in software engineering, as highlighted in an ethnography of oceanographers' programming practices [Kuksenok et al., 2017].

Discipline-specific computational coursework and data literacy are thus a critical part of a modern oceanographic undergraduate curriculum, and we infer the same applies across many geoscience disciplines. While students can collect and analyze small-scale data sets through hands-on fieldwork and labs that are common elements of undergraduate earth science curricula, working with larger, professionally collected

data sets may require familiarity with a programming language [Kastens et al., 2015]. Historically, introductory programming education has been the responsibility of computer science departments, with a focus on data structures and algorithms. Geoscience-specific programming instruction will necessarily reflect distinct goals and tools compared to computer science [Grapenthin, 2011] or data science [Anderson et al., 2015; Lasser et al., 2021], namely, the use of coding to derive insight into natural systems through mathematical manipulation, visualization, and interpretation of idiosyncratic data, often in the time and space domains. Yet formal scientific computing instruction is often absent in earth science curricula, including oceanography [Old, 2019], except for highly scaffolded modules that employ coding in courses where programming is not the primary focus [Rowe et al., 2021]. Even in courses that more extensively utilize programming within activity modules, such as those distributed by Project EDDIE (Environmental Data-Driven Inquiry and Exploration), pre-written code is usually provided to students [O'Reilly et al., 2022]. In this void, brief but intensive hands-on workshops like those offered by Software Carpentry (https://software-carpentry.org; Wilson [2016]), Data Carpentry (https://datacarpentry.org/; Irving [2019]), scientific societies (e.g., Arms et al. [2020]), and academic institutions (e.g., Timms and Guyon [2023]) have provided crucial training to young scientists. These short workshops, however, give learners limited opportunities to apply new coding skills to their own research in a supervised setting. In lieu of formalized instruction, many earth science students teach themselves programming during research experiences or in graduate programs, which can lead to the propagation of ad hoc, inefficient, and outdated practices.

Incorporating programming into an earth science curriculum additionally opens the door to a constructivist approach to teaching scientific concepts—one that encourages students to use experimentation and self-guided inquiry to build on previous learning, construct new knowledge, and engage in critical reflection [Bada, 2015; Hadjerrouit, 2008]. The iterative, reflective process of writing and refining scientific code makes it naturally suited to this individualized model of learning. In practice, a constructivist pedagogy often involves active techniques such as project-based investigation, cooperative learning, and inquiry-based activities. These have been shown to improve student competencies in information recall, analysis, and quantitative reasoning in a large-enrollment introductory oceanography course [Yuretich et al., 2001].

Throughout higher education, there is an increasing recognition that effective teaching requires a focus on active learning, which can be described broadly as students engaged in their learning due to the use of intentional teaching practices [Prince, 2004]. Active modalities include those designated as "high-impact educational practices" [Kuh et al., 2017]. Yet traditional lecturing still represents about three-quarters of class time across STEM undergraduate and graduate courses today [Stains et al., 2018]. In a survey of almost 200 undergraduate oceanography professors, for example, three-quarters indicated that they use data in their instruction but are most likely to teach using lectures, rather than creating opportunities for active inquiry [McDonnell et al., 2015]. There is strong evidence that using active learning techniques increases students' understanding and retention of material in STEM courses, with disproportionate benefits for underrepresented students and students who learn in different ways [Freeman et al., 2014; Haak et al., 2011; Theobald et al., 2020]. One reason these strategies appear to be effective is that they often require an instructor to implement more structure in their course through, for example, regular and intensive practice using scaffolded activities [Haak et al., 2011]. Evidence supports the efficacy of active learning strategies in geoscience classrooms – particularly peer instruction, case studies, and problem-based activities [McConnell et al., 2017].

Embedding computing skills into a geoscience curriculum faces the challenge of introducing students to unfamiliar skills such as algorithmic thinking and overcoming a steep learning curve, similar to teaching a foreign language [Jacobs et al., 2016]. Perhaps for these reasons – as well as a lack of accessible software tools and insufficient computational power in previous decades [Hays et al., 2000] – existing examples of

courses using geoscience data have often focused on interactive online modules, portals, or widgets that are constrained in their data sets and capabilities [Ellwein et al., 2014; Greengrove et al., 2020; Klug et al., 2017]. Software such as Microsoft Excel or specialized tools like Ocean Data View face similar limitations. In comparison, programming skills are more versatile, enabling the analysis of virtually any data set from any domain and empowering students to conduct independent or mentored research projects.

This chapter reports on an evidence-based redesign of an undergraduate oceanography course that teaches introductory Python programming and data analysis techniques. In subsequent sections, we highlight key course elements (summarized schematically in Figure 5.1) and assess the efficacy of the redesign from the standpoint of student engagement and learning.

5.2 Implementation

5.2.1 Course history and development

"Methods of oceanographic data analysis" (OCEAN 215) has been taught annually as a degree requirement in the School of Oceanography at the University of Washington since its establishment in 2015. It was the first introductory Python course offered by the department. Over a ten-week quarter, classes met in person two times each week in two-hour sessions that featured a mix of traditional lecturing and dedicated homework time. The course was well-received by students, who rated it as "very good" (4 on a scale from 1-5) across a variety of metrics in final course evaluations from 2015, 2016, 2017, and 2019 (Figure 5.2), and has been perceived as demanding relative to other courses in students' curricula (see Figure A.1).

However, instructors of other oceanography courses reported that many students who completed OCEAN 215 later had difficulty with core Python programming tasks. A review of past senior theses – projects in which students execute original research – revealed that students often reverted to less versatile, non-coding solutions like Microsoft Excel and Ocean Data View. Given that former students rated the course content as highly useful in evaluations (see Figure A.1), we speculate that their subsequent hesitancy in applying Python skills was due to a lack of recurrent exposure in the curriculum (see also Section 5.7.2) as well as weaknesses in the course design. Possible shortcomings include an overreliance on non-interactive lectures, a lack of student-driven inquiry, assignments' use of unrealistically clean scientific data, and course elements that were not reflective of current scientific Python practices.

The course was restructured (Figure 5.1, Table 5.1) and subsequently co-taught during a 10-week quarter in 2020 by two graduate students, both of whom had served as TAs in past years. Twenty-five undergraduate students completed the course, a typical class size (Figure 5.2). All except two were oceanography majors or later declared an oceanography major. The majority were third-year students, with other years represented, reflecting the flexibility of the oceanography curriculum. To ensure the broadest possible audience for the course, previous coding experience was not required, and a prerequisite of one quarter of calculus from previous iterations of the course was removed. Elements retained from previous iterations included the basic format of four structured programming assignments as well as twice-weekly classes and office hours; however, the latter were conducted virtually rather than in a physical classroom.

In 2020, the COVID-19 pandemic forced a swift transition to virtual instruction. The timing of this course in Autumn 2020, however, allowed for careful planning of an online learning framework, rather than the forced adoption of emergency remote instruction necessary in the first half of 2020 [Donham et al., 2022; Hodges et al., 2020]. Nonetheless, disruptions outside of the classroom were still present: students were isolated on campus or sequestered at home with family, mental health declined, and some became sick or had loved ones fall ill [Furman and Moldwin, 2021]. With these realities in mind, the course redesign paid

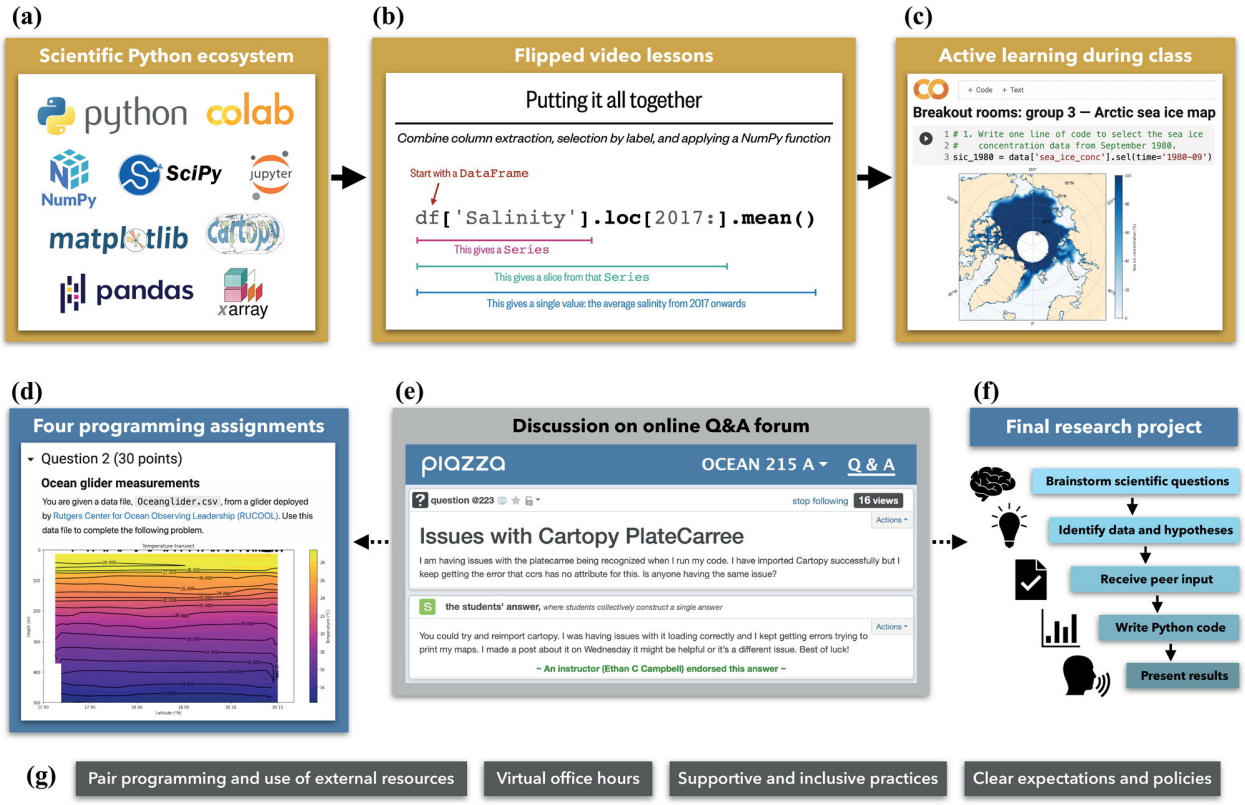

Figure 5.1: Key course elements: (a) Python platforms and software libraries that were taught (see Table A.1 for specific functions, operators, and methods); (b) flipped video lessons, with a slide demonstrating how colors, fonts, design elements, and a minimal working example help to explain Python syntax; (c) class sessions focused on active learning, showing a completed portion of a group activity; (d) programming assignments, with an illustrative plot; (e) discussion on the Piazza Q&A forum, showing a student question and a peer answer endorsed by an instructor; (f) the final research project, represented as the sequence of assigned components; (g) underlying course elements that fostered an effective learning environment. Solid arrows indicate the progression from foundational material (a) to content delivery (b) and application (c); dashed arrows indicate the contributions of discussion forum engagement (e) to students' work on assignments (d) and the final project (f).

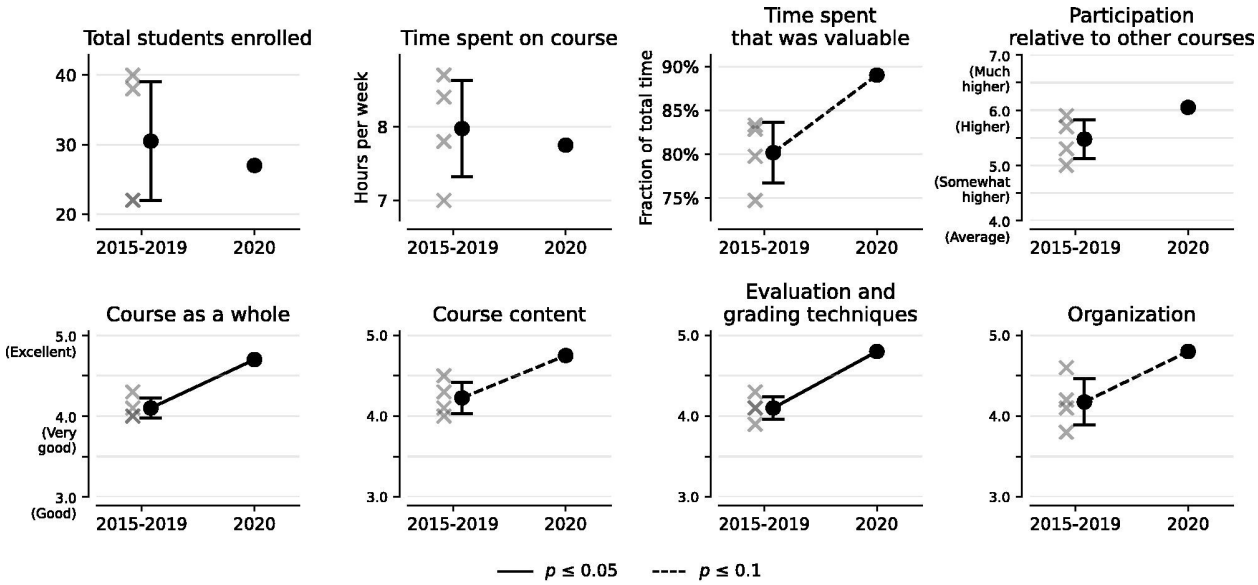

Figure 5.2: Selected metrics from anonymous final student evaluations in 2015, 2016, 2017, 2019, and 2020. Metrics shown are class medians for 2015, 2016, 2017, and 2019 (gray crosses), except for "Total students enrolled"; 2015-2019 mean or 2020 class median (black points); and 2015-2019 standard deviation (bars). Changes from 2015-2019 to 2020 were tested for a significant increase at the 95% (solid line) or 90% (dashed line) confidence level using a one-tailed t-test for all metrics except for "Total students enrolled" and "Time spent on course," which were tested for a significant change using a two-tailed t-test (no significant change was identified for either). For more details, see Section 5.3.2. An absence of a line connecting the 2015-2019 and 2020 data indicates no statistically significant improvement or difference. Note that y-axes have been truncated from the full 1-5 scale ("Very poor" to "Excellent") or 1-7 scale ("Much lower" to "Much higher"). For the full set of evaluation metrics, see Figure A.1.

Table 5.1: Core topics and concepts taught in Ocean 215. Topics listed here are not necessarily in chronological order as taught in the course, and class time was not necessarily allocated in equal proportions to each topic.

Topic	Key Concepts and skills
Why code in Python?	The power of programming is its versatility. Python is open source, stable, popular, free, and ideal for scientific data analysis. Google Colab offers advantages in a classroom setting compared to other programming environments.
Variables and object types	Variables store Python objects, which include numbers, booleans, strings, lists, tuples, dictionaries, and module-specific objects. Objects can be altered, indexed, sliced, iterated over, or used in mathematical operations. Assigning meaningful variable names makes for clearer code.
Logical operations and control flow	Objects can be compared using logical operations (and, or, is/equals, greater/less than, in, not). Loops and if-statements facilitate repetitive and conditional actions.
Packages and functions	Installing and using packages extends the capabilities of Python. Built-in, imported, and user-created functions accomplish common tasks and make for more compact, efficient code. Online documentation can be used to understand functions' arguments and outputs.
Data files	Oceanographic data are often stored in CSV and netCDF files, which can be read into Python, displayed, indexed, sliced, and manipulated using functions in the NumPy, Pandas, and Xarray packages. Real-world data sets can be obtained from public repositories and frequently contain messy or missing data.
Working with data	Data can be stored in multi-dimensional NumPy arrays and labeled structures specific to the Pandas and Xarray packages. These packages, as well as others like SciPy, have functions that average, sort, group, correlate, resample, smooth, regress, interpolate, and perform other computations on the data. Understanding common error types and tracing errors from their line of origin allow for methodical debugging of code.
Plotting	Line, scatter, bar, contour, pseudocolor, and other types of plots available from the Matplotlib package can be used to visualize data. Geospatial data can be projected onto maps using Cartopy. Appropriately customizing and labeling a plot is essential for interpretability.
Scientific skills	The modern scientific method is driven by data exploration, but also relies on traditional research skills like formulating hypotheses, interpreting the scientific significance of visualizations, effectively communicating results, and giving and receiving feedback from peers and mentors.

special attention to the need for a supportive and accommodating learning environment [Shay and Pohan, 2021].

The updates to the course were guided by past experience as TAs, consultation with previous teaching teams and department faculty, the need for fully virtual instruction during the pandemic, and a desire to infuse the course with evidence-based pedagogical elements. Changes included content that reflected the current scientific Python ecosystem (Table 5.1), cloud-based coding notebooks, flipped video lessons, discussions on an online question-and-answer (Q&A) forum, use of data from a wider range of earth science domains, a student-designed final research project, encouragement of pair collaboration and use of external resources, efforts to center accessibility, and a syllabus with explicit policies, expectations, and the following end-of-quarter student learning objectives (SLOs):

1. Understand why the Python programming language is ideal for data analysis.
2. Write, execute, and debug Python code.
3. Access, read, transform, visualize, and interpret oceanographic data with confidence using Python.
4. Explore the ever-expanding universe of packages and tools available for creating and sharing code.
5. Formulate and investigate scientific research questions using programming and data analysis skills.
6. Adopt best practices in programming and data visualization that facilitate collaboration and information-sharing, both within the classroom and the broader scientific community environment.

All course materials were original, created by the graduate instructors, and are available for free reuse and adaptation under a CC-BY-4.0 license at `https://ethan-campbell.github.io/OCEAN_215/`.

5.2.2 Course content

In an introductory classroom setting, the choice of programming language matters. Python is an ideal candidate, as it is easy to learn, versatile, and free to use. First released three decades ago, Python is increasingly ubiquitous within earth science [Lin, 2012] and is widely used outside the scientific community, particularly in industry, making it valuable for students seeking a career outside of academia [Srinath, 2017]. The language features concise, easily read, higher-level syntax that allows one to focus on data exploration, enabling more efficient science, while streamlining workflows starting from remote data access through to analysis and visualization [Ayer et al., 2014; Jacobs et al., 2016; Lin, 2012]. For those learning programming for the first time, a primary challenge is thinking algorithmically, that is, developing structured code to solve a problem. Compared to Python, lower-level programming languages commonly taught in introductory computer science courses (such as Java and C++) require substantial syntactical overhead that can distract from achieving that pedagogical goal [Pears et al., 2007; Srinath, 2017].

Python offers other advantages. Its open-source nature has fostered a large active developer community, which has contributed to its stability and the dissemination of numerous multipurpose packages that extend its functionality. The fact that Python is free prevents a reliance on expensive commercial solutions that can render analysis code inaccessible to scientists outside of well-resourced university environments [Gentemann et al., 2021]. These qualities stand in contrast to MATLAB, a scientific programming language also popular in geoscientific research. Despite the clear benefits of teaching Python in an earth science

context, we find only one documented example of an instructional approach for a conventional (quarter- or semester-long) course in the existing literature [Jacobs et al., 2016].

The updated OCEAN 215 covered scientific Python skills needed for oceanographic data analysis, starting with fundamental Python syntax, as well as data management and research practices (Table 5.1). Students learned core functions (see Table A.1) from versatile, interoperable, and open-source software libraries widely used in climate-related disciplines: NumPy, a fundamental library for multidimensional array computing [Harris et al., 2020]; Matplotlib, a visualization library [Hunter, 2007]; Cartopy, a mapping toolbox [Met Office, 2022]; SciPy, a scientific and statistical analysis library [Virtanen et al., 2020]; Pandas, a toolkit for working with 1-D and 2-D data [McKinney, 2010]; and Xarray, a toolkit for label-based, coordinate-aligned manipulation of multidimensional netCDF files [Hoyer and Hamman, 2017]. No textbook was required in order to allow flexibility in the topics addressed and avoid high textbook costs that have a disproportionately negative impact on historically underserved students [Jenkins et al., 2020]. Students were encouraged to reference online documentation and use their knowledge of general function syntax to expand their Python capabilities beyond the course content. Lessons also addressed programming best practices, such as modularizing code, adhering to variable naming conventions, writing comments, and applying consistent style and formatting [Wilson et al., 2014], as well as effective visualization principles, including legibility and labeling [Hepworth et al., 2020] and considerations of accuracy and accessibility when choosing colormaps for visualizations [Thyng et al., 2016]. These concepts were introduced with examples and data from oceanographic disciplines (physics, chemistry, biology, and marine geology) and other domains (e.g., cryosphere, atmosphere, and climate) using scaffolding to familiarize students with new topics.

5.2.3 Course elements

5.2.3.1 Programming platform

Google Colaboratory (Colab), a cloud-based, in-browser Python development environment modeled after Jupyter notebooks, was chosen as the coding platform for the course. Jupyter notebooks are widely used and can include a mix of interactive code blocks and narrative text, allowing for easy exploration of data and documentation of scientific workflows [Granger and Pérez, 2021; Perkel, 2021]. In general, cloud-based computing has democratized the ability to conduct complex analyses of earth science data sets, creating new opportunities for innovation, transparency, and reproducibility [Gentemann et al., 2021].

Google Colab was chosen as the teaching platform over alternatives like an integrated development environment (IDE) and Jupyter notebooks. Unlike IDEs, Colab requires no local installation of Python or additional software, so students can start coding immediately with minimal device-specific troubleshooting. Notebooks also avoid the cognitive overhead associated with learning command-line syntax or a professional-level IDE [Jacobs et al., 2016; Pears et al., 2007]. Unlike Jupyter notebooks, Colab does not require server configuration and integrates with Google Drive, facilitating file sharing and submission of assignments. Comments can be added to notebooks for grading purposes, similar to Google Docs, and built-in edit history can confirm students' compliance with deadlines. While constraints exist, such as a lack of transparent package management, computational limitations, and the need for an internet connection, the advantages of Google Colab outweigh its disadvantages in a classroom setting.

5.2.3.2 Flipped structure

For novice programmers, blended learning models have been shown in a systematic review to improve the learning experience, as they allow class time to be reserved for active learning and afford students more flexi-

bility to plan and customize their study [Alammary, 2019]. A flipped classroom approach was implemented by assigning 14 recorded lessons of approximately 30 minutes each to be watched before synchronous (Zoom) classes. The lessons were divided into 41 tightly scripted segments of about 10 minutes each (see Figure A.2c). This was done with the goal of helping students maintain focus, as some evidence suggests the average student has an attention span of 15–20 minutes during traditional lecturing [Middendorf and Kalish, 1996]. Prior study in a flipped classroom setting has also found condensing material into shorter (under 5 minute or 10-20 minute) videos is associated with higher student proficiency, engagement, and satisfaction than use of longer (over 30 minute) videos [Yu and Gao, 2022]. In addition to segmenting videos, students were reminded to take breaks between segments. Flipped video watching and in-class participation were not graded, partially in recognition of pandemic stressors but also to accommodate individual circumstances without requiring students to disclose possibly sensitive information. The expectation was that assignment grades would be sufficiently impacted if students were not engaged in these activities.

Most lessons consisted of lectures that illustrated Python concepts using multiple representations, which has been suggested as a core pedagogical strategy for teaching programming [Hadjerrouit, 2008]. For example, slides introducing a new concept would often include three distinct representations: a simplified overview of syntax and function arguments, a minimal example of the function or concept being used (e.g., Figure 5.1b), and a schematic or illustrative plot. Consistent fonts, color schemes, and other design elements were used to reliably indicate relationships between concepts and distinguish examples from core syntax. Some lessons used live-coding demonstrations rather than slides. Accompanying Colab notebooks were provided with each lesson to allow students to run code while watching.

5.2.3.3 Synchronous class sessions

In-class sessions were conducted using the Zoom platform. Students and instructors co-created community guidelines on the first day of the course. Each following synchronous class started with simple icebreakers asking students about their well-being and anonymous polls to gather feedback about previous video lessons. Warm-up activities like these allay anxiety about classroom engagement, connect students with each other, and create a safer environment more conducive to active learning [Bledsoe and Baskin, 2014; Chlup and Collins, 2010]. Concepts from the relevant flipped videos were then briefly reviewed, leaving ample time for students to ask lingering questions. In some class sessions, short activities were used to introduce topics not covered in lesson videos.

The majority of synchronous class time was spent conducting live coding demonstrations and facilitating tutorials that integrated concepts taught in the videos. Compared to using slides or copying and pasting blocks of existing code, live coding forces slower, more digestible instruction, allows instructors to be responsive to student questions in real-time, and inevitably allows students to see instructors' mistakes and how they are diagnosed and fixed [Wilson, 2016]. Tutorials were designed with multiple goals in mind, in alignment with core considerations for programming activities laid out by Hadjerrouit [2008]: (1) to encourage students to analyze the problem at hand and develop stepwise solutions; (2) to build on concepts that students previously learned, encouraging reuse and modification of previous code; and (3) to compare and contrast different ways of achieving the same analytical or graphical result. Based on positive mid-quarter feedback, the instructors emphasized these tutorials and live coding in the second half of the course.

A Google Colab notebook was prepared for each class, presenting a tutorial with four or five related but distinct problems that applied different concepts or functions to a real-world data set from oceanographic and related disciplines (e.g., Figure 5.1c). Data were curated by the instructors for their instructional potential. These exercises created opportunities to divide the classroom into 4-5 person groups that worked cooperatively within Zoom breakout rooms. A "think-pair-share" model [McConnell et al., 2017; Yuretich

et al., 2001] was adopted: students first individually attempted a problem for a few minutes, then teamed up in their breakout room to discuss challenges encountered and optimal solutions, and lastly returned to the main Zoom room, at which point a designated reporter from each group reviewed their results with the full class. Instructors monitored student discussions by moving between breakout rooms and provided guidance when needed. Groups' progress was tracked by watching a shared Google Doc configured ahead of time with templates in which each group filled in their final coding solutions. Occasionally randomizing group members allowed students to gain exposure to a variety of coding styles, social dynamics, and levels of confidence with the material.

5.2.3.4 Q&A forum

An online Q&A board, Piazza, was offered as an outlet for students to connect asynchronously with peers and instructors outside of class and office hours (Figure 5.1e). Piazza enables students to seek help on logistical or clarifying questions as well as their problem-solving processes, thereby reducing individual emails to instructors. The platform allows students to select the audience for their questions (instructors and/or classmates), post anonymously, respond to peers in threaded discussions, and collaboratively construct answers. Instructors may endorse and comment on student answers. Four brief check-ins (including Assignment #0) required Piazza submissions, and an additional quota of five substantive posts per student (i.e., those that contribute further insight to the discussion, rather than simply "Good work" or "I agree") was prescribed in the syllabus. We note that alternative platforms exist that provide similar functionality to Piazza and may be similarly effective.

5.2.3.5 Assignments and final project

Similar to the previous version of the course, students completed four programming assignments at two-week intervals. In the updated version, assignments consisted of approachable, multi-part problems in a Google Colab notebook that utilized real scientific data (Figure 5.1d). For example, one assignment tasked students with importing data collected by an ocean observing platform (a seaglider), identifying key summary statistics, creating a visualization of the glider's location and temperature measurements, and calculating trends in the data. Appropriate geoscience problems were selected to exercise specific skills: atmospheric reanalysis fields depicting historical local weather conditions lent themselves to practicing dimensionality reduction operations using Xarray (e.g., calculating area-averaged snowfall rates over time), while a global bathymetry data set was ideal for applying knowledge of how to create and format contour plots using Cartopy.

Assignments incorporated elements of both "structured inquiry" and "guided inquiry," the second and third levels in the hierarchy of [Banchi and Bell, 2008]. Questions were somewhat less structured compared to class activities, allowing students more flexibility to design their own solutions. This created opportunities to practice both programming skills and data literacy, creating a foundation for more sophisticated independent analysis of data sets. Without a midterm exam, assignments were instructors' main window into student progress. Instructors offered a one-time, two-week extension to allow flexibility while still requiring students to learn foundational material.

Students also completed an individually driven or collaborative final project (see Text A.3 for the project description handout). The goal was for students to write code to answer a scientific question by exploring a data set of their choice, supported by ample guidance from the instructors and peer review from classmates. Similar to the structure of an introductory data programming course described by Anderson et al. [2015], low-stakes checkpoints throughout the quarter required students to share their topic, data set, scientific

questions, and hypotheses on the Piazza Q&A board, as well as offer feedback on at least three classmates' choice of data or questions. The project culminated in the delivery of a short final presentation. A rubric was provided to clearly communicate expectations and evaluation techniques for code, figures, and presentation content and delivery (see Table A.2). Rubrics may lead to increased student performance, and in any case, rubrics are recognized as a user-friendly tool for setting guidelines and enabling self-assessment [Brookhart and Chen, 2015].

Students were offered the option to collaborate in pairs on both the assignments and final project. When programming as a pair, one student serves as the "driver," writing code, while the other observes, monitoring the code for defects and helping to problem-solve. Students were also allowed to reference external resources such as online documentation sites and Stack Overflow. Citations and acknowledgment of collaboration were expected in assignments and the final project, and students confirmed their agreement with the integrity policy in the initial survey (Assignment #0).

5.3 Evaluation

We adopted a two-pronged approach by first evaluating student achievement of SLOs using final project assessments and then exploring instructional approaches that helped students learn by using a variety of other data. Quantitative data include standardized course evaluations, an end-of-quarter student survey, engagement and usage metrics provided by the video and Q&A platforms, and graded assessments. Student focus groups and open-ended evaluation questions were qualitatively analyzed. Prior to analysis, all student-specific metrics were de-identified and coded by a coauthor who was not directly involved in quantitative analyses; identified versions were not used thereafter. This study was approved as qualifying for exempt status for institutional review by the Human Subjects Division at the University of Washington.

5.3.1 Student achievement

We use students' final projects as a barometer of their level of scientific reasoning, their final coding competency, and their achievement of course SLOs. First, questions and hypotheses posed by students in their projects were assessed based on the seven levels of the cognitive process dimension of the revised Bloom's taxonomy [Bloom et al., 1956; Krathwohl, 2002] (see rubric in Table 5.2 for examples referenced in our classification), similar to the methodology of Kastens et al. [2020]. Second, students' breadth of programming skills was evaluated computationally as the fraction of Python syntax elements taught in the course – namely, functions, operators, and methods – that were employed at least once in each student's submitted project code notebook (see Table A.1 for search terms used in the analysis). This metric varies widely between students (see Section 5.4.1) and thus offers significant discriminatory power, albeit limited by our exclusion of miscellaneous functions that were not taught in the course but were used by some students at higher skill levels. Third, the submitted projects were graded using a rubric that was provided to students ahead of time to delineate expectations and evaluation techniques (Table A.2). By mapping rubric subcategories onto four of the six corresponding SLOs and combining the graded scores within each category for each student, we create aggregate metrics of each student's final achievement of those key objectives.

5.3.2 Surveys and evaluations

Here we detail our use of four distinct assessments: an introductory survey, anonymous mid-quarter and final evaluations, and an end-of-quarter survey.

Table 5.2: Rubric used to classify students' final project questions and hypotheses based on the cognitive process dimension of the revised Bloom's taxonomy [Krathwohl, 2002]. For the analyses in Figures 5.3b-c, multiple hypotheses and/or questions offered by students (up to three each) were assessed separately and weighted such that a student's three hypotheses, for example, would each contribute 1/3 of a point to their respective cognitive level's total count.

Cognitive level	Questions	Hypotheses
Level 1: Remember	*"Can the data be visualized using my skills?"* Intention to **recall** coding techniques taught in the course and **recognize** their proper use	**Recall of course material** (e.g., the data can be depicted using a scatter/line/pseudocolor plot)
Level 2: Understand	*"What stands out in the data?"* Intention to **summarize** the data; or *"Do the data resemble what we expect the ocean to look like?"* Intention to **interpret** the data and classify what is present by **comparison** to known examples	**Factual interpretation** (e.g., the data will have X, Y features; the data will resemble X, Y other ocean data)
Level 3: Apply	*"What [happens if...]"* Intention to **execute** or **implement** a specific procedure, such as **calculating** a correlation; or *"Does [...]"* Intention to **answer** a binary (yes/no) question	**Specific results and relationships** (e.g., the answer will be yes/no; X will show an increase over time; X and Y will show a positive correlation)
Level 4: Analyze	*"How [does/do/is/are...]"* Intention to **characterize** or **test** a **straightforward** or **single-dimensional** relationship, phenomenon, or difference	**Contextual results and relationships** (e.g., X and Y will show a positive correlation, but only under Z conditions; X and Y will vary with Z; X is characterized by Y patterns)
Level 5: Evaluate	*"How [does/do...] affect... "* *"What [is/are...] the relationship between... "* Intention to **characterize** or **attribute** in an **open-ended** or **multidimensional way**; or *"Why [does/do/is/are...]"* Intention to **establish** causality by **integrating** external ideas or models and/or **connecting**, **contrasting**, or **weighing** multiple sources of information	**Explanations** (e.g., X and Y will show a positive correlation because of mechanism Z; X and Y exhibit different features because of Z)
Level 6: Create	*"What [does/do...] mean... "* *"How [does/do...] fit into... "* Intention to **evaluate** the **implications** of findings, **place** findings within old or new paradigms, **construct** or **produce** new frameworks, or **investigate the consequences** of phenomena using an open-ended approach	**Discovery** (e.g., X is important because Y; X will differ from a past model Y, where a model is composed of two or more mechanisms; X can be explained using Y model; or a hypothesis cannot be established due to lack of prior information)

To gauge initial exposure to the Python programming language and coding in general, students were asked to share their prior experiences in an introductory survey distributed in the first week of class (Assignment #0). The instructors translated students' short-answer responses into a numeric rating (1-5) using a subjective analysis of their word choice (see rubric in Table A.3 and Figure A.3). The factors considered were any previous coding languages learned, the reported efficacy of past learning experiences, and time since last exposure to coding. The introductory survey also encouraged students to introduce themselves to the teaching team by sharing their pronouns and any anticipated accessibility, technology, or learning needs, which led to instructors making an effort to accurately caption all lesson videos.

We also obtained summary reports from final Instructional Assessment System (IAS) evaluations completed by OCEAN 215 students in Autumn 2015, 2016, 2017, 2019, and 2020 (results from Spring 2015 and Autumn 2018 were unavailable), which were administered and anonymized by the University of Washington. Standardized questions asked students to evaluate aspects of the course quality and their engagement with the course. While most questions were consistent across years, others evolved in their wording and thus required mapping or aggregation to enable comparison between years (as shown in Table A.4). Questions that could not be tracked across years were excluded. Students completed evaluations either in paper or online format, with the class response rate of around 70% in 2020 being somewhat higher than in past years (Figure A.1). As IAS summary reports correspond to specific instructors, we averaged the class median responses between the two graduate instructors for each question in 2020. Changes between 2015-2019 and 2020 were tested for a statistically significant increase using a one-sided t-test for questions where increases could objectively be viewed as a desired improvement: metrics on a 1-5 ("Very poor" to "Excellent") scale and the metrics "Time spent that was valuable" and "Participation relative to other courses." Remaining metrics were tested for a statistically significant change in either direction using a two-sided t-test.

Furthermore, we apply a standard qualitative approach [Creswell, 1998] to extract meaning from students' anonymous responses to open-ended questions in two IAS evaluations in 2020: a mid-quarter evaluation administered during weeks 4-5 of the course and the final evaluation. The prompts are listed in Table A.5. We identified common or unique themes mentioned by students, grouped similar themes, coded responses by noting whether a theme was mentioned in either a subjectively positive context (e.g., an appreciative or affirming comment; assigned a value of +1) or subjectively negative context (e.g., an unenthusiastic or critical comment; assigned a value of –1), and tabulated the frequency of each context for all themes. We also excerpt illustrative quotes from students' responses throughout the text.

In addition to the university-managed IAS evaluations, a voluntary end-of-quarter Google Form survey was administered during the week after the final class to measure students' perceived success relative to the course SLOs. The response rate was 92%. Submissions were not anonymous, but instructors guaranteed to students that their responses would not impact their final course grades. As a final self-assessment of students' Python skills, we use responses to the question, "How proficient do you feel in writing, executing, and debugging Python code?", which were on a 6-point scale from "Least proficient" to "Most proficient."

5.3.3 Flipped video viewership

Panopto, the video hosting and delivery platform used in the course, provides instructors with usage statistics, including view counts, minutes delivered, percent completed, and last view time. Those metrics – associated with individual students, individual videos (both aggregated and disaggregated by student), and distinct video viewing sessions, where applicable – were downloaded, and student identities were anonymized as described above. Student-specific Panopto metrics computed include total minutes watched, minutes watched before the class for which a video was assigned, and minutes watched after class for the first time (i.e., late views).

5.3.4 Q&A forum engagement

Piazza, the online Q&A platform, also makes usage statistics available to instructors. The following student-specific metrics were downloaded, then anonymized as described above: days online, answers, and total contributions (which include questions, notes, answers, and comments). Additionally, a time series of engagement was constructed based on unique users per day, as provided by Piazza. The time series was supplemented by a manual tabulation of daily Piazza activity within the following categories: student questions and notes related to programming; student scheduling, extension, or logistical requests; student answers and comments; student posts that were required for assignments; and instructor posts, answers, or comments. Where relevant, those categories were further divided by chosen audience into total posts that were public and signed, public and anonymous, or private (i.e., visible to instructors only).

5.3.5 Final grades

Students' final grades are used to represent overall student achievement. Grades were recalculated to ignore two students' incomplete assignments (0% grades) that occurred due to personal circumstances, and the following weights were re-applied: 60% for assignments #0-#4 (weighted equally), 15% for Piazza posts, and 25% for final projects. Recalculating final grades raised their average from 95.0% to 95.9% and decreased their standard deviation from 5.7% to 3.8%.

5.3.6 Student focus group

Undergraduate students who completed OCEAN 215 in Autumn 2020 were considered for a focus group based on responses to the voluntary Google Form survey asking students to rate their interest in the project and provide a short paragraph about course elements that affected their learning positively or negatively. Five students were chosen based on the thoughtfulness of their written responses and the diversity of their academic backgrounds and experiences within the course. Selection was not dependent on students' grades in the course, and it was made clear that survey responses would not impact course grades. Three focus group sessions were held in the quarter following Autumn 2020, each lasting 1-2 hours. In the sessions, the instructors asked questions designed to provoke open and candid discussion about students' perception of course elements and took notes by paraphrasing comments. Focus group participants did not have access to the anonymized raw student data.

The five students were additionally invited to share short testimonials detailing their unique experiences in the course and were offered coauthorship on the study. The four testimonials that were submitted are presented in Text A.3 and excerpted throughout the text. The final testimonials were assembled from students' responses to their selection of a subset of the guiding questions included as Table A.6 and were edited for style and grammar and to limit redundancy of themes mentioned. Insights gleaned from the focus group or testimonials are clearly denoted in the text. We use them as supporting evidence to depict students' perspectives about the course more holistically and accurately and to indicate areas where students felt the course could be modified to improve their experience.

5.4 Results

5.4.1 Student learning outcomes

Overall final project grades were all above 80%, with most students scoring high marks (80% or above) on four project rubric categories representing the quality of their code, visualizations, use of data, and scientific

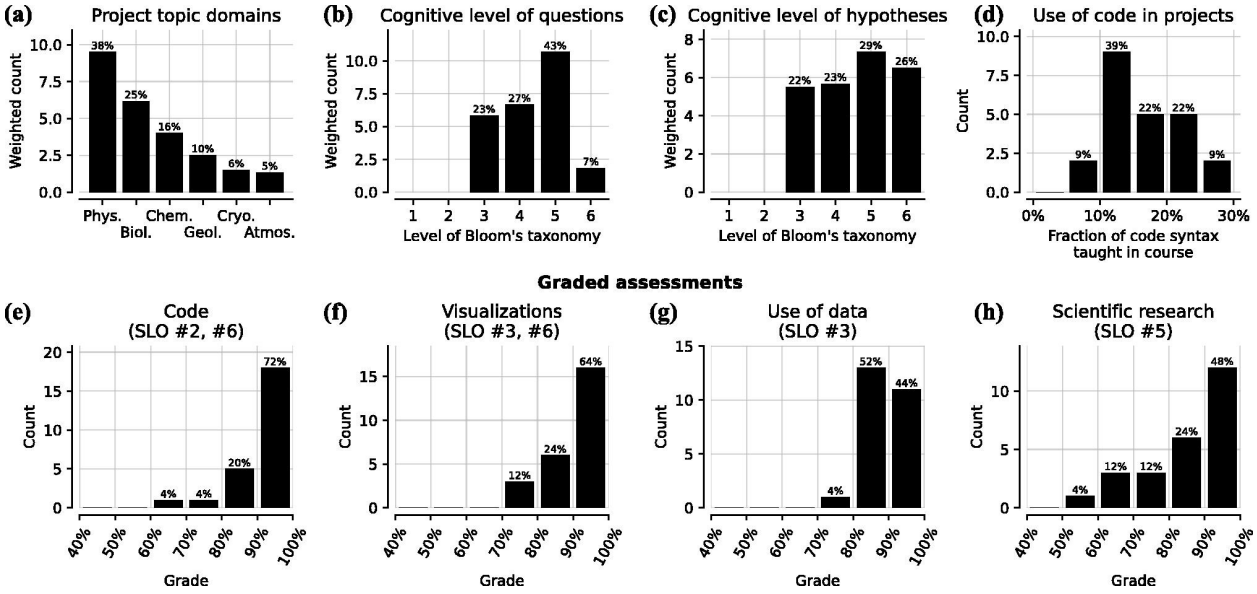

Figure 5.3: Assessment of students' final projects. (a) Distribution of domains of students' final projects. If a project topic touched multiple domains, each domain was weighted such that, for example, a project spanning three domains would contribute 1/3 of a point to each of the domains' total count. (b-c) Distribution of cognitive level of students' questions and hypotheses. Each student's questions and hypotheses (up to three each per student) were assessed based on the cognitive process dimension of the revised Bloom's taxonomy [Krathwohl, 2002] using the rubric and weighting described in Table 5.2, with higher levels of Bloom's taxonomy representing higher-order questioning and prediction. (d) The fraction of code syntax taught in the course that students used in their projects (see Table A.1 for assessment methodology and search terms). (e-h) Project grades within four named categories that correspond to student learning objectives (SLOs) outlined in the text (see Table A.2 for grading rubric and mapping to SLO categories). These categories (with rubric subcategories in parentheses) are code (correctness, functionality, tidiness, perseverance), visualizations (clarity, colormaps, labels, creativity), use of data (data collection, processing, results/interpretation), and scientific research (background, questions/hypotheses, explanations). Note the x-axes are truncated to 40%-100% for readability.

research (Figures 5.3e-h). These categories correspond to SLOs #2, #3, #5, and #6, with some overlap (see Table A.2). The widest spread in grades was in the category of scientific research (Figure 5.3h), in which 28% of students scored below 80

Students' final project topics spanned the oceanographic, cryosphere, and atmospheric domains (Figure 5.3a). Examples include applying time series analysis techniques to ocean monitoring data to assess the impact of climate change on coral bleaching, cleaning a global database to characterize the traits of hydrothermal vent fauna, and mapping blue whale tracks and satellite sea surface temperatures to infer relationships between environmental conditions and whale behavior. Scientific questions and hypotheses posed by students largely map onto higher levels of Bloom's taxonomy, exemplifying higher-order questioning and prediction (Figures 5.3b-c). The percentage of code syntax taught in the class that was used in each final project ranged widely from 6% to 29% (Figure 5.3d) and exhibits no significant correlation with the assessed cognitive level of students' questions or hypotheses (not shown). In other words, students' level of scientific reasoning was not predictive of the analytical complexity of their finished projects.

5.4.2 Connecting course elements to student learning

By calculating correlations between a variety of anonymized data sources (see Section 5.3), presented in Figure 5.4, we explore the impact of students' varying backgrounds and learning strategies on their course experiences and outcomes. Significantly, neither students' final grades nor their code usage in final projects is correlated with prior coding experience, indicating that previous exposure to Python was not predictive of success in the course. On the other hand, prior coding experience is positively correlated with self-assessments of Python skills in end-of-quarter surveys, the only metric showing a relationship with the self-assessments. Dichotomizing the class by prior coding experience (none/little versus some/moderate/lots) also reveals no statistically significant difference in final grades (Figure S3). That said, less prior experience was associated with higher engagement with lesson videos and the Q&A forum (Figure 5.4). The breadth of Python skills used in final projects positively correlates with three key metrics: total lesson minutes watched, number of Q&A forum answers, and forum days online. This indicates that students who demonstrated strong coding competency had likely acquired more content knowledge, frequently shared that knowledge with peers, and were more engaged with the course. Variations in students' demonstrated Python skills cannot fully explain differences in their final grades, but the two show a positive nonlinear correlation. Students who earned higher grades tended to monitor the Q&A forum more frequently, collaborate more often with classmates using pair programming, and watch lesson videos before class.

5.4.3 Role of course elements in student learning

5.4.3.1 Course content

Overall, students perceived the course positively in final evaluations, rating its content, evaluation techniques, organization, and the course as a whole markedly higher than in past years (Figure 5.2). Students' view of the course content evolved from a critical stance expressed in mid-quarter evaluations, with comments citing its abstract or challenging nature, to an appreciative view in final evaluations of the data skills they had acquired and their utility for conducting scientific research (Figure 5.5). One focus group participant who was a first-time coder wrote in their testimonial (Text A.3):

> *"I have always viewed research as something that is extraordinarily complicated. This class demonstrated that knowing a few basic Python functions and packages can provide a solid foundation to start conducting research."*

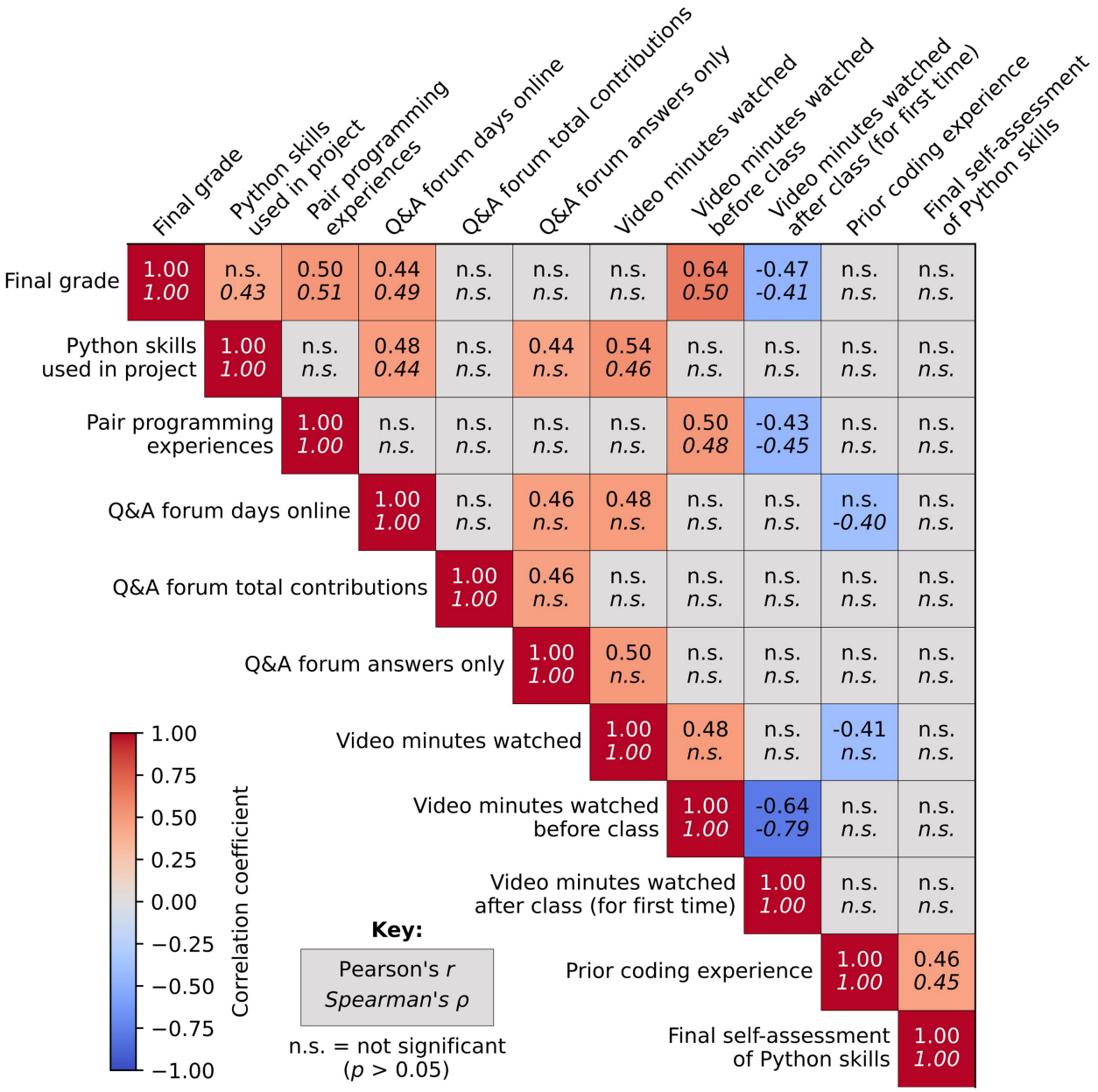

Figure 5.4: Correlations between student-specific anonymized metrics. Two tests were applied: Pearson's r (top values) and Spearman's ρ (lower values, italicized). Higher Pearson correlations indicate stronger positive linear relationships, while higher Spearman values indicate stronger monotonic relationships, which may not necessarily be linear. Correlations without statistical significance ($p > 0.05$) are indicated by "n.s." Colors correspond to the larger of the two correlation coefficients by absolute value. For detailed information about each metric presented, see Section 5.3.5 (for "Final grade"; column 1), Table A.1 (for "Python skills used in project"; column 2), Section 5.4.3.5 (for "Pair programming experiences; column 3), Section 5.3.4 (for Q&A forum-related metrics; columns 4-6), 5.3.3 (for video-related metrics; columns 7-9), Table A.3 (for "Prior coding experience"; column 10), and Section 5.3.2 (for "Final self-assessment of Python skills; column 11).

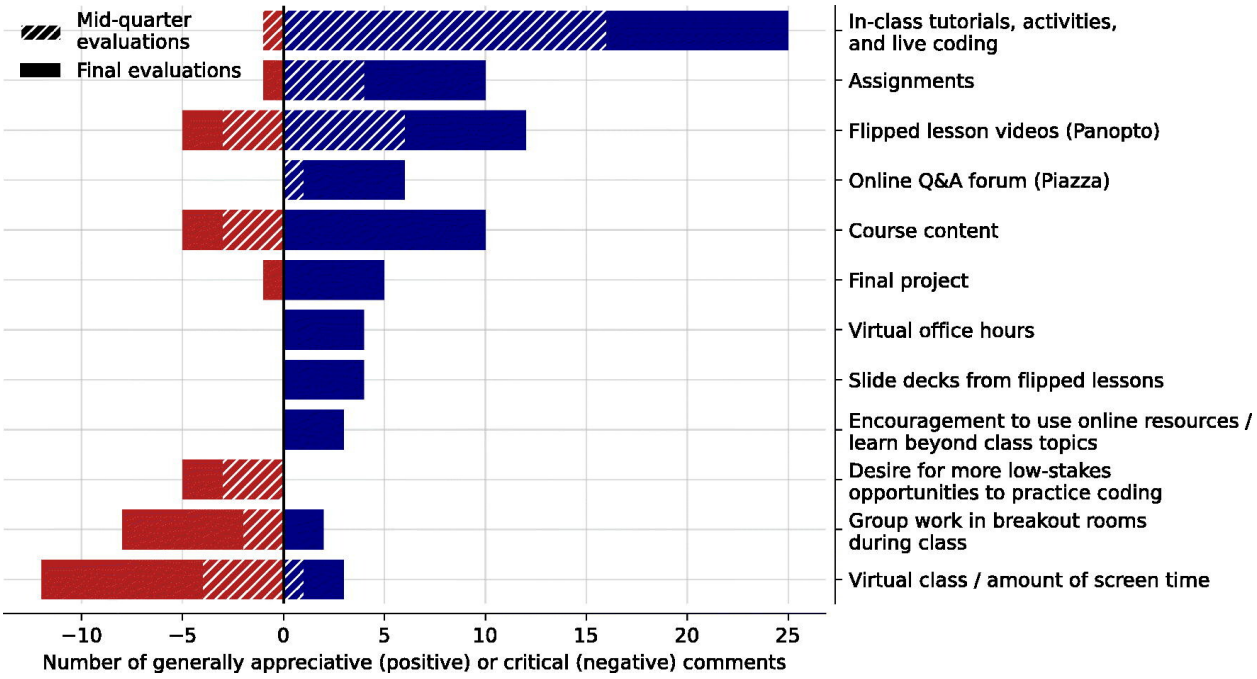

Figure 5.5: Themes identified in anonymous, open-ended student responses to mid-quarter (hatched bars; n = 15 students) and final (solid bars; n = 20 students) evaluations in 2020, ranked according to the net positivity (blue) or negativity (red) of comments regarding those themes. Totals for mid-quarter and final evaluations are stacked, not overlapping, within each bar. Original question prompts are listed in Table A.5

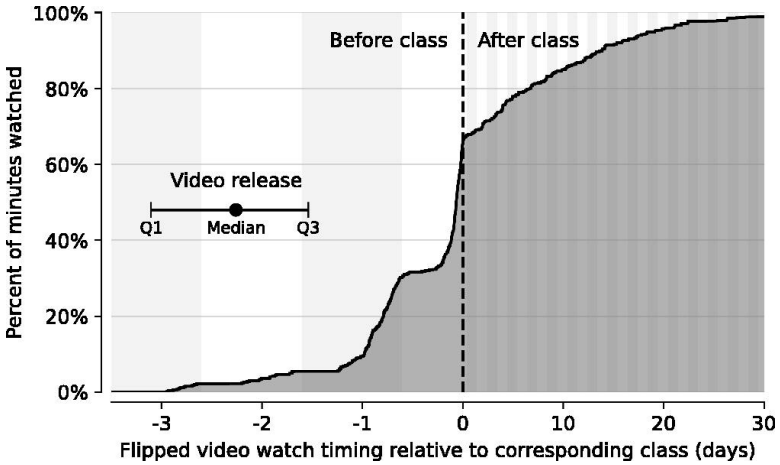

Figure 5.6: Timing of flipped Panopto video viewing sessions relative to the class for which each video was assigned. Viewing sessions were binned along the x-axis according to their timing before or after their corresponding class (with each viewing session weighted by its duration), then total minutes for each timing bin were summed from left to right to produce the cumulative distribution of watch timing shown here. The y-axis is the cumulative fraction of total video time delivered during the course (166.3 hours over 41 videos), with video rewatches included. The median and interquartile range (25%-75%) of video releases by instructors, relative to the corresponding class, is included for reference, indicating that videos were generally released 1.5 to 3 days before they were due. Vertical shading corresponds to days; note the compressed positive x-axis scale.

5.4.3.2 Flipped structure

In total, students spent 166 hours watching lesson videos on the Panopto platform. According to viewership statistics, two-thirds of the watch time occurred before the class for which the video was assigned (Figure 5.6). Most lessons were released 1.5-3 days before the Zoom class meeting, and students generally watched lessons during the 24 hours prior to class. The remaining one-third of total watch time occurred throughout the month following the relevant class, of which three-quarters were first-time views. While the total video lesson minutes watched by a student were correlated with the breadth of Python skills used in their final project (see Section 5.3.1), the timing of their video lesson views was not (Figure 5.4).

Students in the focus group expressed that they appreciated the opportunity to watch videos at a convenient time and the ability to take breaks. Some shared that they would have viewed videos immediately before class regardless of release timing, while others said they would have taken advantage of a longer period of availability. While one student reported in their final course evaluation that "occasionally the length of the recorded lectures prevented [them] from finishing them entirely," we find no significant correlation in Panopto statistics between video or lesson duration and fraction watched (see Figures A.2f and h). These statistics also show that half of students watched nearly every video, with class-wide average video completion between 80-90% in most weeks (Figure 5.7a). Completion rates dropped near the end of the course, which student focus group participants suggested was due to high end-of-quarter demands in other courses and because the material covered didn't appear in assignments. Some students in the focus group reported re-watching videos to review material or using corresponding slide decks for the same purpose; one student took notes on the videos and later referenced those notes. In final course evaluations, students noted that having slide decks available benefitted their learning (Figure 5.5), with one student sharing, "I was able to

surprise myself with how much I could figure out through review when feeling helpless at first." Despite the addition of watching flipped videos (as well as a final project) to the overall course workload, students estimated in final evaluations that the amount of time they spent each week was similar to past years. Yet out of students' total time spent on the course, nearly 90% was perceived as valuable in advancing their education – a significant increase from past years ($p \leq 0.1$; Figure 5.2).

5.4.3.3 Synchronous class sessions

Interactive tutorials involving live coding demonstrations and individual activities were the most positively reviewed course element in students' mid-quarter and final evaluations (Figure 5.5). On the other hand, the large amount of screen time was the most frequently mentioned criticism in course evaluations (Figure 5.5). Students also offered criticism on the use of breakout groups in their evaluations, with one noting, "I didn't find the small group coding breakout rooms very helpful for coding, but they were nice for getting to know my classmates." Several students wished for more time and instructor guidance in breakout rooms, which contributed to their overall negative rating (Figure 5.5). Nonetheless, one focus group participant noted in their testimonial (Text A.3) that breakout rooms "forced us to come well-prepared for class" and in final course evaluations, students rated their overall participation as higher relative to other courses (6.0 on a 7-point scale, where 4.0 is "average"; Figure 5.2).

5.4.3.4 Q&A Forum

Students visited Piazza once every 1-5 days on average, and engagement in the form of questions, answers, and comments closely tracked assignment deadlines and peaked while students worked on the final project (Figure 5.7a). Many questions from students were simple – for example, diagnosing a coding bug or clarifying the goal of an assignment – while others were more complex – such as seeking strategies to efficiently work with large data sets for one's final project. The forum saw 530 total student contributions, out of which two-thirds were voluntary, i.e., not required by a check-in or Assignment #0 (Figure 5.7b). The average response time to student questions by instructors or peers was 17 minutes.

Students selected the three audience options (public, either signed or anonymous, and private posts) with approximately equal frequency, depending on their needs (Figure 5.7b). Student focus group participants shared that the anonymous and private posting options were useful when they were worried that a question would be perceived as obvious or simple, or when they were less sure of their answer. Final course evaluations show that students overall felt positively about having access to Piazza (Figure 5.5). One student shared their appreciation for the ability to post anonymously, stating that it "alleviated some anxiety about asking questions."

5.4.3.5 Assignments and final project

In course evaluations, most students viewed the assignments and final project as beneficial (Figure 5.5). We find that nearly half of the class – 48% of students – took advantage of the pair programming option at some point, with 34% of students collaborating on any given assignment or the project on average. Students generally chose the same classmate as their partner throughout the course. The number of times that a student worked collaboratively is presented as the metric "Pair programming experiences" in Figure 4. One focus group participant shared their experience in their testimonial (Text A.3):

> *"... we coded in completely different ways, and it was fascinating to see those differences. We were more effective together because we learned to compromise and collaborate to find the*

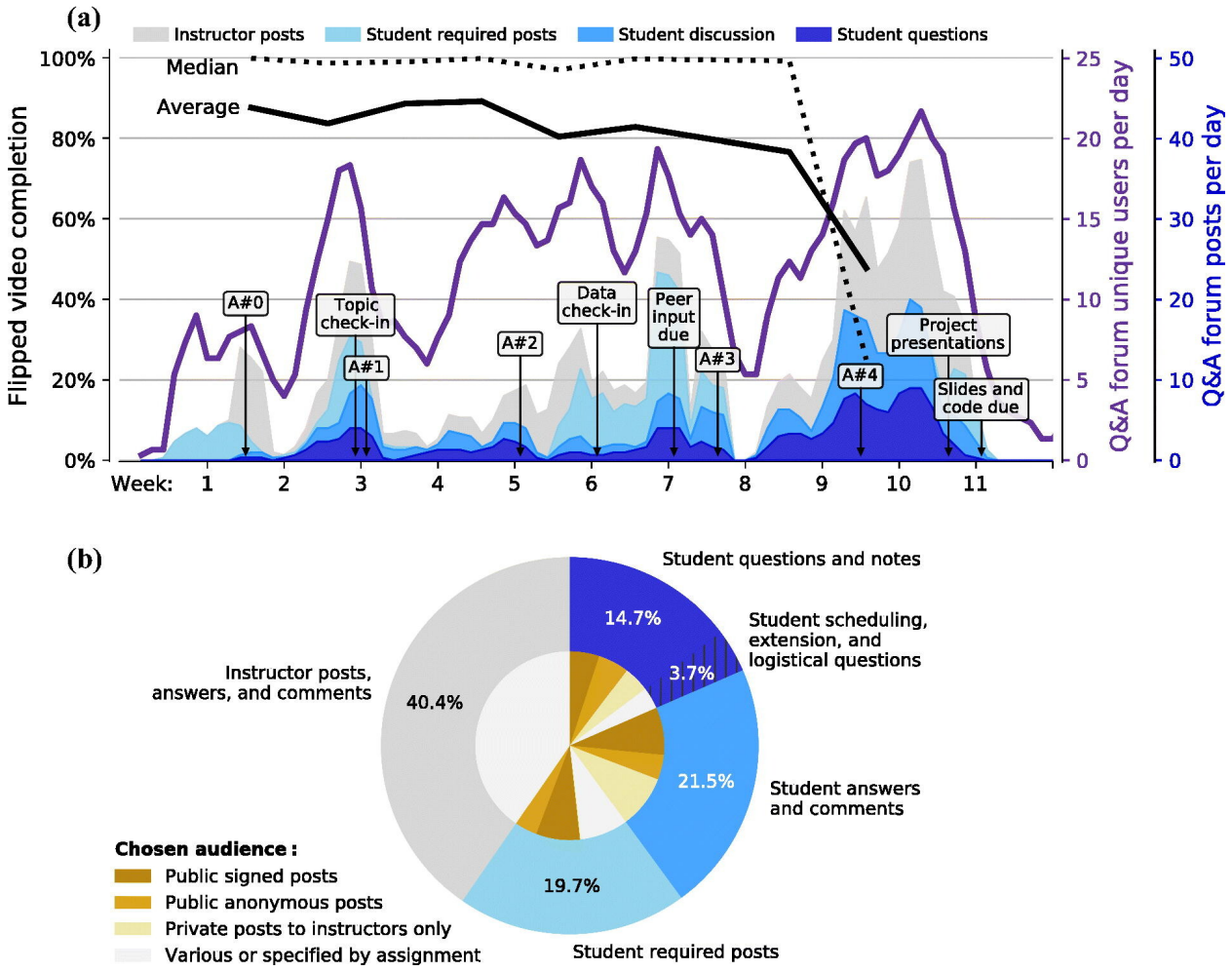

Figure 5.7: Student engagement with online platforms. (a) Flipped video completion rates over time from Panopto are presented as both the class-wide median (dotted black line) and average (solid black line). Note that video completion by student was allowed to exceed 100% due to repeat views. Piazza Q&A forum engagement is shown as unique users per day (purple) and posts per day, segmented by the type of post (shaded curves; see colors in legend). The timing of coursework deadlines (assignments ["A#..."] and final project checkpoints) are indicated with arrows. (b) Usage of the Piazza Q&A online forum by students and instructors, segmented by type of post (outer) and further divided by chosen audience (inner). "Required posts" were those requested from every student for Assignment #0 and final project check-ins. "Public posts" were viewable by all users, while "private posts" were visible to instructors only. "Anonymous posts" refer to those in which the author was hidden from other students, but not from instructors.

cleanest and fastest method between the two of us."

The opportunity to synthesize course knowledge and the option to collaborate with classmates on final projects were specifically cited in students' final evaluations as positive elements of the course. The ability to use external materials and learn beyond class topics was similarly welcomed in final evaluations (Figure 5.5), and another student expressed in their testimonial:

> *"[Accessing online resources like StackOverflow] developed essential skills and gave me the confidence to apply new concepts in my final project. This meant my research could be dictated by my curiosity and questions, as it should be, and not by the limitations of what concepts we had covered in class."*

That said, one critical final evaluation comment related to ambiguity about the rigor of science expected and the open-ended nature of project checkpoints.

5.5 Discussion

5.5.1 Student learning outcomes

We measured students' achievement of key SLOs (#2, #3, #5, and #6) by assessing their final projects, with the assumption that the projects represent a holistic demonstration of students' capabilities. Those assessments indicated clear success in achieving learning objectives. Students produced impressive and original work that reflected earnest attempts to investigate questions within the geosciences using effective coding and visualization techniques. The strength of the correlation between project code usage and video minutes watched suggests that students' content acquisition was particularly impactful for their coding competency (SLOs #2 and #3).

Consistent with research that found a weak correlation between tutor grades and self-assessments by over 3,000 undergraduate students [Lew et al., 2010], we saw no link between students' self-assessment of programming skills in the end-of-quarter survey and their final grades. A caveat is that students were asked to rate their Python competence, rather than their final grade, and the two metrics may not be entirely comparable. That said, this result could still reflect the Dunning-Kruger effect, a cognitive bias in which those with the least knowledge tend to overestimate their performance or ability because they lack the competencies required for self-assessment [Kruger and Dunning, 1999]. The lack of a relationship between students' final self-assessments and any metrics other than prior coding experience points to a persistent confidence from previous Python exposure that contributed to a perception of competence not necessarily reflected in higher grades or course-acquired skills. In contrast, our results suggest a leveling of the playing field in which those who came in with less previous knowledge of programming took full advantage of class resources, like lesson videos and Piazza, to ultimately reach the same level of proficiency as their peers, as shown in final grades and project code usage.

5.5.2 Role of course elements in student learning

5.5.2.1 Flipped Structure

Consistent with a review supporting the efficacy of blended learning models [Alammary, 2019], our analysis of video watch timing, student focus group feedback, and course evaluations shows that our flipped structure enabled a diversity of strategies for content acquisition. Exposure to video content before working on related

in-class activities may have helped students prepare for assignments, which comprised the majority of final grades. Nonetheless, our correlation metrics suggest that the total amount of time spent viewing lessons, not whether those lessons were watched before or after a class, was most influential in students' application of course content, as measured by Python skills used in their final projects.

In line with prior research on students' perspective of the flipped model [McCallum et al., 2015], our course structure generally received student approval in course evaluations (Figure 5.5). Students' overall positive evaluations of the course are notable given hardships related to the COVID-19 pandemic, as well as findings that show students often prefer passive lecturing over active learning due to the additional cognitive effort required to engage actively with material [Deslauriers et al., 2019].

Virtual teaching offers inherent accessibility benefits for students facing long commutes, disability-related challenges, and other barriers to in-person participation [Pichette et al., 2020]. We believe that virtual office hours – regarded positively by students in course evaluations – indeed offered benefits for students who may have perceived office hours as an unfamiliar or unsafe space. Similarly, breakout rooms created privacy for students with questions on assignments or personal matters.

5.5.2.2 Synchronous class sessions

Course evaluations indicated that in-class activities and demonstrations were well-liked and engaging. However, the facilitation of breakout groups and large amount of screen time presented challenges for students and instructors and were met with critical reviews. "Zoom fatigue" is a particular form of exhaustion that may result from the intensity of continuous, close-up eye contact and seeing oneself, reduced mobility when having to stay in a video frame, and increased cognitive load from having to exaggerate nonverbal cues [Bailenson, 2021]. To mitigate these effects, regular breaks were taken during class, students were encouraged to take breaks during recorded videos, a video-optional policy was instituted on Zoom, and students were allowed to use the chat function to participate, though students' criticisms about screen time show Zoom fatigue remained a challenge. These solutions are also imperfect—breaks take class time, teaching to students with cameras off can be disorienting, and chat messages can be difficult to monitor during instruction.

Though breakout rooms can allow for more individualized attention, the instructors had difficulty with distributing their finite time across groups and eliciting participation. Both can be linked to group size, and student focus group participants indeed shared mixed views on the number of students per group. Smaller groups could have encouraged more individual accountability at the expense of increasing demands on instructors' time as they cycle between breakout rooms. Larger groups would have enabled instructors to provide more efficient guidance and increased opportunities for peer instruction but often suffer from uneven participation. The optimal configuration may depend on individual classroom circumstances.

5.5.2.3 Q&A Forum

The wide range of question types that we observe on Piazza are in line with previous research in an undergraduate computer science setting, which similarly showed high participation rates when students are encouraged to use the platform by teaching staff [Vellukunnel et al., 2017]. Our correlation analysis of student metrics also matches the positive relationship between question-asking on a Q&A forum and final grades found in that prior study. The apparent efficacy of Piazza may reside in the fact that voluntarily asking a question on a discussion forum, by definition, constitutes a form of active learning, though posts may vary in their level of reasoning and connectedness [Vellukunnel et al., 2017]. Active learning would presumably be maximized if students use Piazza to seek help after they have invested time into trying differ-

ent solutions and consulting other resources, which is encouraged by the asynchronous nature of the forum. However, at times, the rapid responses to questions on Piazza observed in our course from both instructors and peers may have short-circuited this problem-solving process. While prompt instructor engagement is vital for establishing a strong teaching presence in a remotely taught course [Prince et al., 2020], it could be valuable for instructors to delay their responses so that peers have an opportunity to provide input first.

5.5.2.4 Q&A Forum

In each assignment notebook, copious scaffolding around each problem (e.g., step-by-step instructions, expected intermediate results, and links to documentation websites) was provided, which created an environment of "structured inquiry" and "guided inquiry," the second and third levels of the continuum of inquiry proposed by Banchi and Bell [2008]. The assignments were designed to be challenging yet were viewed favorably by both the student focus group and the final evaluation respondents. Both, however, indicated a desire for more short, frequent, low-stakes practice opportunities to help reinforce concepts and check understanding. Such practice exercises centered on coding skills rather than scientific inquiry would exist outside of the hierarchy of Banchi and Bell [2008] and yet may be critical for providing a strong foundation when teaching a programming language. In contrast to instructor-generated assignments, the final project elicited student-designed questions and hypotheses indicating cognition on higher levels of Bloom's taxonomy [Bloom et al., 1956; Krathwohl, 2002]. The resulting research spanned a wide range of disciplines, topics, data sources, and analysis techniques. This reflects "open inquiry," an experience that is exceedingly rare in undergraduate oceanography teaching [McDonnell et al., 2015]. Consistent with a constructivist approach to learning [Bada, 2015], students in OCEAN 215 answered complex or potentially ill-structured questions using messy and incomplete real-world dataset [Ellwein et al., 2014; Klug et al., 2017] with instructor guidance mostly related to feasibility. As demonstrated by graded assessments of the projects, students successfully conducted research despite it having unknown outcomes. In similar courses that incorporate open-ended research, students have reported learning gains equivalent to those of dedicated summer research programs [Lopatto, 2010].

Pair programming has been known to improve student learning, performance, and satisfaction in the computer science classroom, without loss of competency on exams [McDowell et al., 2002; Williams and Upchurch, 2001]. While not a causal analysis, we find a positive correlation between pair programming experiences and final grades. In a survey of undergraduates who conducted collaborative research, almost 80% reported that working in teams or pairs enhanced their research experience [Lopatto, 2010], consistent with feedback from our focus group and final evaluations. We found pair programming to be readily adaptable to the virtual classroom using Zoom screen-sharing, with the caveat that Colab notebooks must be refreshed to show updates and thus edits must be made by one user at a time rather than synchronously. One lesson learned was that some pairs will gravitate towards asynchronous collaboration (i.e., a division of labor, rather than true pair programming) unless it is specified that the coding must be done synchronously. Additionally, collaborations appeared to prove more successful when coding partners had a pre-existing working relationship; naturally, this is less likely to occur in a remotely taught introductory class setting. Nonetheless, previous work has found equal benefits to student performance and confidence for students who pair program remotely using screen-sharing and audio connectivity compared to physically collocated student pairs [Hanks, 2005].

5.5.3 Accessibility and inclusivity

Instructional approaches focused on active learning and student engagement can help to combat inequities in the classroom [Theobald et al., 2020], but equally important are strategies that promote a culture of respect and foster a sense of belonging for students [Dewsbury and Brame, 2019]. A classroom community built on mutual understanding and respect promotes engagement, especially among students with marginalized identities, by creating a supportive space to share ideas and ask questions [Barrett, 2021]. With that in mind, efforts were made to ensure that the course was accessible for all students despite pandemic-related disruptions so that those with varying backgrounds and needs felt welcome. These supportive strategies, some of which are discussed elsewhere in this chapter, included accommodating disabilities, captioning lesson videos, co-creating classroom norms, affirming students' gender identities, starting class with icebreakers and anonymous feedback polls, endeavoring to mitigate 'Zoom fatigue,' setting flexible attendance and extension policies, offering virtual office hours, facilitating the participation of students located in remote time zones, and offering one-on-one mentoring as needed. While our study design did not specifically assess their impact, the intentional incorporation of these elements likely influenced student outcomes and perceptions in the course.

Our overall approach of providing multiple modalities for student learning was consistent with a universal design for learning (UDL) framework that prioritizes equitable and inclusive teaching [Capp, 2017; Meyer et al., 2014]. UDL outlines three core principles: (1) multiple means of representation, which our course accomplished through recorded lessons with text, auditory, and visual components, live coding demonstrations, and permissive use of external resources; (2) multiple means of action and expression, facilitated through practice opportunities and assignments with varying degrees of structure; and (3) multiple means of engagement, enabled by our use of individual as well as group work, verbal as well as chat-based participation, peer instruction, office hours, and the online forum.

5.6 Limitations

The robustness of our conclusions is limited by the relatively small sample size (25 students) and the study's focus on a single academic quarter. Additionally, missing IAS evaluations from Spring 2015 and Autumn 2018 are not included in our longitudinal comparison to previous years' course evaluations. In this comparison, we also cannot disentangle the various influences of the COVID-19 pandemic on learning from the impact of the curriculum changes that we made. Furthermore, we cannot quantify the impact of the new teaching team's positionality as graduate students on students' impression of course quality. A previous study, for example, found that professors who were perceived as younger received higher evaluations than professors teaching identical content who were perceived as older [Arbuckle and Williams, 2003].

While a pre-quarter assessment of student coding competency and attitudes would have been an ideal way to assess student growth, such an assessment was not conducted as the study design was conceived after the course had concluded. Data on students' age, race, and ethnicity were not collected for similar reasons, so we were unable to explore relationships between demographic profiles and students' experiences or success in the course. Likewise, student achievement for two of the six course SLOs (#1 and #3) could not be explicitly measured using available data, although an assessment of final projects found that students successfully met the remaining four SLOs.

While a systematic approach is used to identify and tabulate themes in the evaluation responses (see Section 5.3), we do not apply the same technique to qualitative data from the student focus group or their testimonials. The small sample size (five students) and the non-representative nature of the group selected by instructors would limit the appropriateness and utility of such an approach. Furthermore, the focus group

conversations were not open-ended, but rather guided by questions formulated by instructors after initial analyses of other data (e.g., evaluation results, student learning metrics, etc.). Focus group discussions were documented through paraphrased notes rather than an exact transcription, so direct quotes are not presented. Testimonials were edited by instructors (as described in the Section 5.3.6), further restricting the possibility of a quantitative thematic analysis approach.

5.7 Summary

5.7.1 Recommendations for future teaching

We recommend without reservations adopting the key elements that we describe in this chapter, particularly flipped instruction, an online coding platform and discussion board, and strong attention to accessibility. That said, we encourage others to improve on our framework and regularly seek feedback from students, preferably in a format that allows for anonymity. For example, in course evaluations, students encouraged the addition of more frequent, low-stakes practice of basic skills to reinforce fundamental concepts (see Section 5.5.2.4). New practice opportunities would ideally be coupled with immediate feedback that guides further practice, which promotes efficient learning and refinement of conceptual understanding [Ambrose et al., 2010]. While we did not implement graded comprehension checks for videos, these could be useful in a situation of lower engagement [Jacobs et al., 2016]. Additionally, data literacy skills could be taught through higher-level exercises asking students to scrutinize the limitations, biases, and provenance of scientific data sets and make predictions and recommendations grounded in their analysis of data (see, e.g., Kastens and Krumhansl [2017]). Instructors may consider expanding our offering into a multi-course sequence to incorporate these elements.

We acknowledge the ongoing paradigm shift in many scientific fields towards "open science," a broadly defined set of ethics that encapsulates practices like code reproducibility, curation of data for reuse, and open journal access [Brett et al., 2020; Ramachandran et al., 2021]. While these practices were not explicitly taught in this course, its emphasis on collaborative programming, well-documented code, and the scientific method as an open, transparent endeavor speak to fundamental open science principles. Explicit instruction on advanced topics like reproducibility, data archival, version control using Git and GitHub [Blischak et al., 2016], manipulation of large data sets stored on the cloud [Gentemann et al., 2021], and command-line interfaces may be more appropriate for a separate, higher-level course. While this course relied on original content, instructors could consider offering excerpts from existing earth science-oriented Python textbooks as a resource [Alyuruk, 2019; DeCaria and Petty, 2021; Esmaili, 2021; Palomino et al., 2021; Wei-Bing Lin et al., 2022].

The pandemic likely accelerated existing trends in higher education towards multi-modal instruction and more engaging teaching practices [Lockee, 2021]. Though universities have transitioned back to in-person teaching, an interested and highly-engaged instructor team could still offer a fully remote version of this course, potentially with minimal penalty in student performance and satisfaction compared to in-person instruction [Ghosh et al., 2022; Ramirez et al., 2022]. We believe that the framework developed for this course is also well-suited to a hybrid approach that incorporates in-person tutorial and work sessions but retains the pedagogical and accessibility benefits of recorded lesson videos, virtual office hours, and platforms that enable regular online engagement. Since 2020, this course has been offered annually in-person at the University of Washington by other graduate instructor teams with a flipped structure and most of the key curriculum elements introduced in this study.

5.7.2 Impact

The impact of this course extends beyond the students who enrolled in Autumn 2020. The flipped lesson videos were uploaded to a dedicated YouTube channel (`https://www.youtube.com/@ocean215python`), where they have been collectively viewed more than 27,000 times as of June 2024, reaching 40 different countries. Furthermore, the graduate student instructors have benefited from the professional experience of developing a curriculum and managing a classroom. Opportunities such as this have been linked with the success of doctoral students attaining future employment in higher education [Bettinger et al., 2016]. Our department plans for a rotating cast of two graduate students to continue serving as the primary teaching team, with the guidance and support of a dedicated teaching mentor to develop their pedagogical skills.

For many undergraduate oceanography majors without a deeper interest in data science, multiple quarters may pass after completing OCEAN 215 before their next opportunity to use programming. For most, this comes in the form of their senior thesis. Students' demonstrated loss of coding skills during past intervening years (see Section 5.2.1) is consistent with previous findings from online programming courses [Teusner et al., 2018]. An exploratory analysis of programming usage in the senior theses of students who completed OCEAN 215 in Autumn 2020 compared to past senior theses nonetheless shows a notable increase in the use of coding languages, including Python, for computation and figure generation. While potential confounding factors related to differences between years prevent definite conclusions, this improvement suggests the value of our instructional design in fostering content acquisition and retention of skills.

Yet even an effective programming course's impacts will wane over time without regular, scaffolded opportunities for students to practice and apply programming skills. Barriers to incorporating programming into geoscience undergraduate curricula include coalescing around a primary language of instruction while realizing the benefits of exposing students to other languages – many instructors, for example, use MATLAB for research – and implementing a curriculum map that communicates a standard set of programming skills for students. Mastery and retention of a programming language requires repeated exposure and practice opportunities, and so programming should be integrated across all levels of an earth science curriculum. Effort should also be invested in creating supervised research opportunities for students that involve the use of programming and data analysis. More broadly, we see the need for earth science undergraduate curricula to adopt active, student-centered pedagogical practices that more frequently allow students to construct knowledge through hands-on exploration of real-world data. Infusing earth science curricula with current data programming practices will naturally facilitate the achievement of these goals.

Chapter 6

Conclusion

6.1 Summary of EKE and baroclinic conversion

Examining EKE and the pathway to it from observations assists us in understanding the impacts of energy on global circulation. We have built these projects on a vast collection of previous work to understand ocean energy (e.g. Ferrari and Wunsch [2009]; Aiki et al. [2016]; Youngs et al. [2017]). Each of the research chapters in this dissertation addresses a new component of the energy system that has not previously been studied using wide-scale observations. The use of the global Argo array has allowed for unprecedented sampling across large swaths of the ocean that can observe spatial variability in the ocean interior.

In Chapters 2 through 4, we find a common theme that the bathymetry affects the spatial variability of vertical velocity, baroclinic conversion, and EKE. In particular, the bathymetry matters most for vertical exchange when interacting with the horizontal flow. For this reason, strong currents, like the ACC, have complex energy dynamics with large fluxes in energy. These areas have been shown to be a key component in the circulation of the ocean [Gille et al., 2004; Tamsitt et al., 2017; Youngs and Flierl, 2023]. Our results support the importance of accounting for topographic features in the study of the ocean.

In general the reservoirs of energy are considerably larger than the fluxes between them [Ferrari and Wunsch, 2009; Chen et al., 2014]. However, the fluxes are a fundamental component for the effects of energy on ocean circulation. In Chapter 3, we find that the baroclinic conversion of energy in the Southern Ocean occurs largely in the ACC. This is, in part, due to the wind-driven sloping isopycnals that contain a large amount of EAPE. As the current interacts with topography the potential energy is converted into kinetic through vertical motion, estimated in Chapter 2.

These fluxes then establish the EKE field that we observed in Chapter 4. In general, we see that profiles of EKE show decreasing energy with depth. The shape of the profile is influenced by topographic features, which change how the energy is attenuated at depth. This is also accentuated by the presence of strong horizontal flow.

These are some of the first observational estimates of vertical velocity, baroclinic conversion, and EKE in the ocean interior across wide areas of the ocean. As such, there are a number of applications for these results that can build our understanding of energy in ocean circulation.

6.2 Applications and future work

The results of this work are one small corner of all ocean energetics (Figure 1.1). While trying to create a full energy budget from observations would be a monumental undertaking, estimating the other reservoirs

and fluxes of energy would help inform us on the full evolution of energy as it enters the ocean. Having these observations would also help to address how energy dissipates in the ocean, which is not currently well understood.

Additionally, the observations presented in this dissertation can help inform model dynamics. Because vertical velocity in the ocean is not well-measured, model fields have not been compared to observations. For models that aim to predict future ocean conditions and their global impact, like climate models [Mazloff et al., 2010; Liang et al., 2017], these observations can help validate model fields and determine the effects of the mesoscale that may not be included in the resolution of the model.

The novel methods we have developed are the groundwork for further understanding of ocean energy from observations. As the global Argo array continues its sampling mission, sparse areas of float coverage will be better sampled which will allow for an even more expansive view into ocean energy using the same methodologies outlined here. With the addition of new biogeochemical sensor packages being deployed on floats [Claustre et al., 2020], we can also start to examine how ocean energy affects a wide variety of variables in the ocean. All of this work would not have been possible without the global Argo array. The data from floats are indispensable to this research and to the oceanographic community at large [Wong et al., 2020]. The continued growth of the Argo array is paramount to successfully understanding the current and future state of the ocean through observations.

The vertical velocity and baroclinic conversion of energy that are observed here are some of the first observations of wide-scale fluxes in the ocean interior. This is unique in that we can estimate rates of change and movement in ocean conditions. Specifically, these fluxes facilitate the transport of carbon, nutrients, and heat, all of which affect the ocean's impact on the climate [Gruber et al., 2009].

One limitation to this work is that we are constrained to only the depths at which Argo floats can sample. For the baroclinic conversion and the vertical velocity, this means that we only have measurements at a single depth level of 1,000 m. For the EKE, this means that we have profiles that go from the surface to 2,000 m, rather than the full ocean depth. Therefore, one future pathway for this work is to bring in other data sources to create a complete view of the ocean. Combining our estimates of EKE with those from moored current meters would allow us to have the widespread measurements of floats with the full-depth measurements of the moorings. Additionally, the baroclinic conversion and EKE fields that we observe can be incorporated into model fields that would result in profiles of EKE and it conversion. This would allow us to more readily understand how energy moves in the ocean interior.

While further efforts need to be made to entirely comprehend energy in the ocean, the range of possible applications and future directions of this research makes the work presented here a key part in understanding ocean energetics. We have found that the reservoirs of energy and their pathways are fundamental to the circulation of the ocean, though they are also spatially variable and highly complex. Their effects are difficult to fully discern, but doing so is a key part to understanding the changing system of Earth and its climate.

Bibliography

Hidenori Aiki, Xiaoming Zhai, and Richard J Greatbatch. 2016. Energetics of the global ocean: The role of mesoscale eddies. *Indo-Pacific climate variability and predictability*, pages 109–134.

Ali Alammary. 2019. Blended learning models for introductory programming courses: A systematic review. *PLoS ONE*, 14(9):e0221765.

Hakan Alyuruk. 2019. *R and Python for oceanographers: A practical guide with applications*. Elsevier, Amsterdam.

Christopher Amante and Barry W Eakins. 2009. Etopo1 arc-minute global relief model: procedures, data sources and analysis. *NOAA Technical Memorandum NESDIS NGDC-24*.

Susan A. Ambrose, Michael W. Bridges, Michele DiPietro, Marsha C. Lovett, and Marie K. Norman. 2010. What kinds of practice and feedback enhance learning? In *How learning works: Seven research-based principles for smart teaching*, chapter 5, pages 121–152. John Wiley & Sons, Inc.

Ruth E. Anderson, Michael D. Ernst, Robert Ordóñez, Paul Pham, and Ben Tribelhorn. 2015. A data programming CS1 course. In *SIGCSE 2015 - Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 150–155.

Julianne Arbuckle and Benne D. Williams. 2003. Students' perceptions of expressiveness: age and gender effects on teacher evaluations. *Sex Roles*, 49(9-10):507–516.

Sean Arms, Julien Chastang, Maxwell Grover, Jon Thielen, Matthew Wilson, and Douglas Dirks. 2020. Introducing students to scientific Python for atmospheric science. *Bulletin of the American Meteorological Society*, 101(9):E1492–E1496.

C. Assassi, Y. Morel, F. Vandermeirsch, A. Chaigneau, C. Pegliasco, R. Morrow, F. Colas, S. Fleury, X. Carton, P. Klein, and R. Cambra. 2016. An index to distinguish surface- and subsurface-intensified vortices from surface observations. *Journal of Physical Oceanography*, 46(8):2529 – 2552.

Vidya M. Ayer, Sheila Miguez, and Brian H. Toby. 2014. Why scientists should learn to program in Python. *Powder Diffraction*, 29(S2):S48–S64.

Steve Olusegun Bada. 2015. Constructivism learning theory: A paradigm for teaching and learning. *Journal of Research & Method in Education*, 5(6):66–70.

Jeremy N. Bailenson. 2021. Nonverbal overload: A theoretical argument for the causes of Zoom fatigue. *Technology, Mind, and Behavior*, 2(1).

Heather Banchi and Randy Bell. 2008. The many levels of inquiry. *Science and Children*, 46(2):26–29.

Sarah Elizabeth Barrett. 2021. Maintaining equitable and inclusive classroom communities online during the COVID-19 pandemic. *Journal of Teaching and Learning*, 15(2):102–116.

Eric P. Bettinger, Bridget Terry Long, and Eric S. Taylor. 2016. When inputs are outputs: The case of graduate student instructors. *Economics of Education Review*, 52:63–76.

T. Scott Bledsoe and Janice J. Baskin. 2014. Recognizing student fear: The elephant in the classroom. *College Teaching*, 62(1):32–41.

John D. Blischak, Emily R. Davenport, and Greg Wilson. 2016. A quick introduction to version control with Git and GitHub. *PLOS Computational Biology*, 12(1):e1004668.

Benjamin S. Bloom, Max D. Engelhart, Edward J. Furst, Walker H. Hill, and David R. Krathwohl. 1956. *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain*. Longmans, London.

Annie Brett, Jim Leape, Mark Abbott, Hide Sakaguchi, Ling Cao, Kevin Chand, Yimnang Golbuu, Tara J. Martin, Juan Mayorga, and Mari S. Myksvoll. 2020. Ocean data need a sea change to help navigate the warming world. *Nature*, 582(7811):181–183.

Susan M. Brookhart and Fei Chen. 2015. The quality and effectiveness of descriptive rubrics. *Educational Review*, 67(3):343–368.

Ethan C. Campbell, Katy M. Christensen, Mikelle Nuwer, Amrita Ahuja, Owen Boram, Junzhe Liu, Reese Miller, Isabelle Osuna, and Stephen C. Riser. 2024. Cracking the code: An evidence-based approach to teaching python in an undergraduate earth science setting. *Journal of Geoscience Education*, pages 1–20.

Matthew James Capp. 2017. The effectiveness of universal design for learning: a meta-analysis of literature between 2013 and 2016. *International Journal of Inclusive Education*, 21(8):791–807.

Ru Chen, Glenn R Flierl, and Carl Wunsch. 2014. A description of local and nonlocal eddy–mean flow interaction in a global eddy-permitting state estimate. *Journal of Physical Oceanography*, 44(9):2336–2352.

Dominique T. Chlup and Tracy E. Collins. 2010. Breaking the ice: Using ice-breakers and re-energizers with adult learners. *Adult Learning*, 21(3-4):34–39.

Katy M. Christensen, Alison R. Gray, and Stephen C. Riser. 2024. Global estimates of mesoscale vertical velocity near 1,000 m from argo observations. *Journal of Geophysical Research: Oceans*, 129(1):e2023JC020003.

Hervé Claustre, Kenneth S Johnson, and Yuichiro Takeshita. 2020. Observing the global ocean with biogeochemical-argo. *Annual review of marine science*, 12(1):23–48.

CMEMS. 2021. Global Ocean Gridded L4 Sea Surface Heights and Derived Variables Reprocessed (1993-ongoing). `https://doi.org/10.48670/moi-00148`. Accessed Oct. 8, 2021.

John W. Creswell. 1998. Data analysis and representation. In *Qualitative Inquiry and Research Design: Choosing Among Five Traditions*, 1 edition, chapter 8, pages 139–165. SAGE Publications, Thousand Oaks, CA.

Jesse M. Cusack, Alberto C. Naveira Garabato, David A. Smeed, and James B. Girton. 2017. Observation of a large Lee wave in the Drake Passage. *Journal of Physical Oceanography*, 47(4):793–810.

Alex DeCaria and Grant W. Petty. 2021. *Python programming and visualization for scientists*, 2 edition. Sundog Publishing.

Louis Deslauriers, Logan S. McCarty, Kelly Miller, Kristina Callaghan, and Greg Kestin. 2019. Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom. *Proceedings of the National Academy of Sciences*, 116(39):19251–19257.

Bryan Dewsbury and Cynthia J. Brame. 2019. Inclusive teaching. *CBE—Life Sciences Education*, 18(2):fe2.

Cristine Donham, Cathy Pohan, Erik Menke, and Petra Kranzfelder. 2022. Increasing student engagement through course attributes, community, and classroom technology: Lessons from the pandemic. *Journal of Microbiology & Biology Education*, 23(1):1–6.

Jeffrey J Early, M Pascale Lelong, and K Shafer Smith. 2020. Fast and accurate computation of vertical modes. *Journal of Advances in Modeling Earth Systems*, 12(2):e2019MS001939.

Amy L. Ellwein, Laurel M. Hartley, Sam Donovan, and Ian Billick. 2014. Using rich context and data exploration to improve engagement with climate data and data literacy: Bringing a field station into the college classroom. *Journal of Geoscience Education*, 62(4):578–586.

Rebekah B. Esmaili. 2021. *Earth observation using Python: A practical programming guide*, volume 75. American Geophysical Union and Wiley, Hoboken, NJ.

Raffaele Ferrari and Carl Wunsch. 2009. Ocean circulation kinetic energy: Reservoirs, sources, and sinks. *Annual Review of Fluid Mechanics*, 41:253–282.

Catherine Forbes, Merran Evans, Nicholas Hastings, and Brian Peacock. 2010. *Cauchy Distribution*, chapter 10. John Wiley & Sons, Ltd.

G. Forget, D. Ferreira, and X. Liang. 2015. On the observability of turbulent transport rates by Argo: Supporting evidence from an inversion experiment. *Ocean Science*, 11(5):839–853.

Eleanor Frajka-Williams, Charles C. Eriksen, Peter B. Rhines, and Ramsey R. Harcourt. 2011. Determining vertical water velocities from Seaglider. *Journal of Atmospheric and Oceanic Technology*, 28(12):1641–1656.

Howard J Freeland. 2013. Vertical velocity estimates in the north pacific using argo floats. *Deep Sea Research Part II: Topical Studies in Oceanography*, 85:75–80.

Scott Freeman, Sarah L. Eddy, Miles McDonough, Michelle K. Smith, Nnadozie Okoroafor, Hannah Jordt, and Mary Pat Wenderoth. 2014. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415.

Frederick N Fritsch and Ralph E Carlson. 1980. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246.

Helle Kristine Fuhr. 2015. On the vertical partition of kinetic energy in the ocean. a comparison between flat and steep bottom modes. Master's thesis, University of Oslo.

Tanya Furman and Mark Moldwin. 2021. Higher education during the pandemic: Truths and takeaways. *Eos*, 102(9):17–19.

Peter R Gent, Jurgen Willebrand, Trevor J McDougall, and James C McWilliams. 1995. Parameterizing eddy-induced tracer transports in ocean circulation models. *Journal of physical oceanography*, 25(4):463–474.

Chelle L. Gentemann, Christopher Holdgraf, Ryan Abernathey, Daniel Crichton, James Colliander, Edward J. Kearns, Yuvi Panda, and Richard P. Signell. 2021. Science storms the cloud. *AGU Advances*, 2(2):e2020AV000354.

Sattik Ghosh, Stephanie Pulford, and Arnold J. Bloom. 2022. Remote learning slightly decreased student performance in an introductory undergraduate course on climate change. *Communications Earth & Environment*, 3:177.

Adrian E Gill. 1982. *Atmosphere—ocean dynamics*. Elsevier.

Sarah T Gille, E Joseph Metzger, and Robin Tokmakian. 2004. Seafloor topography and ocean circulation. *Oceanography*, 17(1):47–54.

Brian E. Granger and Fernando Pérez. 2021. Jupyter: Thinking and storytelling with code and data. *Computing in Science & Engineering*, 23(2):7–14.

Ronni Grapenthin. 2011. Computer programing for geosciences: Teach your students how to make tools. *Eos, Transactions American Geophysical Union*, 92(50):469–470.

Alison R Gray and Stephen C Riser. 2014. A global analysis of sverdrup balance using absolute geostrophic velocities from argo. *Journal of Physical Oceanography*, 44(4):1213–1229.

Cheryl Greengrove, Sage Lichtenwalner, Hilary Palevsky, Anna Pfeiffer-Herbert, Silke Severmann, Dax Soule, Stephanie Murphy, Leslie Smith, and Kristen Yarincik. 2020. Using authentic data from NSF's Ocean Observatories Initiative in undergraduate teaching. *Oceanography*, 33(1):62–73.

Nicolas Gruber, Manuel Gloor, Sara E Mikaloff Fletcher, Scott C Doney, Stephanie Dutkiewicz, Michael J Follows, Markus Gerber, Andrew R Jacobson, Fortunat Joos, Keith Lindsay, et al. 2009. Oceanic sources, sinks, and transport of atmospheric co2. *Global biogeochemical cycles*, 23(1).

David C. Haak, Janneke HilleRisLambers, Emile Pitre, and Scott Freeman. 2011. Increased structure and active learning reduce the achievement gap in introductory biology. *Science*, 332(6034):1213–1216.

Said Hadjerrouit. 2008. Towards a blended learning model for teaching and learning computer programming: A case study. *Informatics in Education*, 7(2):181–210.

Thomas W. N. Haine, Renske Gelderloos, Miguel A. Jimenez-Urias, Ali H. Siddiqui, Gerard Lemson, Dimitri Medvedev, Alex Szalay, Ryan P. Abernathey, Mattia Almansi, and Christopher N. Hill. 2021. Is computational oceanography coming of age? *Bulletin of the American Meteorological Society*, 102(8):E1481–E1493.

Brian Hanks. 2005. Student performance in CS1 with distributed pair programming. *ACM SIGCSE Bulletin*, 37(3):316–320.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. *Nature*, 585(7825):357–362.

James D. Hays, Stephanie Pfirman, Benno Blumenthal, Kim A. Kastens, and William Menke. 2000. Earth science instruction with digital data. *Computers & Geosciences*, 26(6):657–668.

Yunzhu He, Meng Zhou, and Dujuan Kang. 2024. Strong eddy kinetic energy anomalies induced by baroclinic instability in the southwest region of the kerguelen plateau, east antarctica. *Journal of Geophysical Research: Oceans*, 129(6):e2023JC020667.

Tyler D. Hennon, Stephen C. Riser, and Matthew H. Alford. 2014. Observations of Internal Gravity Waves by Argo Floats. *Journal of Physical Oceanography*, 44(9):2370–2386.

Katherine Hepworth, Cesunica E. Ivey, Chelsea Canon, and Heather A. Holmes. 2020. Embedding online, design-focused data visualization instruction in an upper-division undergraduate atmospheric science course. *Journal of Geoscience Education*, 68(2):168–183.

Charles B. Hodges, Stephanie Moore, Barbara B. Lockee, Torrey Trust, and Aaron A. Bond. 2020. The difference between emergency remote teaching and online learning. *Educause Review*.

Andrew Mc C Hogg, Michael P Meredith, Jeffrey R Blundell, and Chris Wilson. 2008. Eddy heat flux in the southern ocean: Response to variable wind forcing. *Journal of Climate*, 21(4):608–620.

Stephan Hoyer and Joe Hamman. 2017. xarray: N-D labeled Arrays and Datasets in Python. *Journal of Open Research Software*, 5(1):10.

Rui Xin Huang. 2004. Ocean, energy flows in. *Encyclopedia of Energy*, 4:497–509.

John D. Hunter. 2007. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95.

Damien Irving. 2019. Python for atmosphere and ocean scientists. *Journal of Open Source Education*, 2(11):37.

Christian T. Jacobs, Gerard J. Gorman, Huw E. Rees, and Lorraine E. Craig. 2016. Experiences with efficient methodologies for teaching computer programming to geoscientists. *Journal of Geoscience Education*, 64(3):183–198.

J. Jacob Jenkins, Luis A. Sánchez, Megan A. K. Schraedley, Jaime Hannans, Nitzan Navick, and Jade Young. 2020. Textbook broke: Textbook affordability as a social justice issue. *Journal of Interactive Media in Education*, 2020(1):3.

Kim A. Kastens and Ruth Krumhansl. 2017. Identifying curriculum design patterns as a strategy for focusing geoscience education research: A proof of concept based on teaching and learning with geoscience data. *Journal of Geoscience Education*, 65(4):373–392.

Kim A. Kastens, Ruth Krumhansl, and Irene Baker. 2015. Thinking big. *The Science Teacher*, 82(5):25–31.

Kim A. Kastens, Melissa Zrada, and Margie Turrin. 2020. What kinds of questions do students ask while exploring data visualizations? *Journal of Geoscience Education*, 68(3):199–219.

Jennifer L. Klug, Cayelan C. Carey, David C. Richardson, and Rebekka Darner Gougis. 2017. Analysis of high-frequency and long-term data in undergraduate ecology classes improves quantitative literacy. *Ecosphere*, 8(3):e01733.

David R. Krathwohl. 2002. A revision of Bloom's taxonomy: An overview. *Theory Into Practice*, 41(4):212–218.

Justin Kruger and David Dunning. 1999. Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77(6):1121–1134.

George Kuh, Ken O'Donnell, and Carol Geary Schneider. 2017. HIPs at ten. *Change: The Magazine of Higher Learning*, 49(5):8–16.

Kateryna Kuksenok, Cecilia Aragon, James Fogarty, Charlotte P. Lee, and Gina Neff. 2017. Deliberate individual change framework for understanding programming practices in four oceanography groups. *Computer Supported Cooperative Work*, 26(4-6):663–691.

Mikael Kuusela and Michael L Stein. 2018. Locally stationary spatio-temporal interpolation of argo profiling float data. *Proceedings of the Royal Society A*, 474(2220):20180400.

Marta Sanchez de La Lama, JH LaCasce, and Helle Kristine Fuhr. 2016. The vertical structure of ocean eddies. *Dynamics and Statistics of the Climate System*, page dzw001.

Joseph Henry LaCasce and Sjoerd Groeskamp. 2020. Baroclinic modes over rough bathymetry and the surface deformation radius. *Journal of Physical Oceanography*, 50(10):2835–2847.

Jana Lasser, Debsankha Manik, Alexander Silbersdorff, Benjamin Säfken, and Thomas Kneib. 2021. Introductory data science across disciplines, using Python, case studies, and industry consulting projects. *Teaching Statistics*, 43(S1):S190–S200.

Magdeleine D. N. Lew, W.A.M. Alwis, and Henk G. Schmidt. 2010. Accuracy of students' self-assessment and their beliefs about its utility. *Assessment & Evaluation in Higher Education*, 35(2):135–156.

Xinfeng Liang, Michael Spall, and Carl Wunsch. 2017. Global Ocean Vertical Velocity From a Dynamically Consistent Ocean State Estimate. *Journal of Geophysical Research: Oceans*, 122(10):8208–8224.

Johnny Wei-Bing Lin. 2012. Why Python is the next wave in earth sciences computing. *Bulletin of the American Meteorological Society*, 93(12):1823–1824.

Barbara B. Lockee. 2021. Online education in the post-COVID era. *Nature Electronics*, 4(1):5–6.

David Lopatto. 2010. Undergraduate research as a high-impact student experience. *Peer Review*, 12(2):27–30.

Edward N Lorenz. 1955. Available potential energy and the maintenance of the general circulation. *Tellus*, 7(2):157–167.

Aro data management. 2022. Argo user's manual. Report (normative document), Ifremer.

David P Marshall, Maarten HP Ambaum, James R Maddison, David R Munday, and Lenka Novak. 2017. Eddy saturation and frictional control of the antarctic circumpolar current. *Geophysical research letters*, 44(1):286–292.

Adrian P. Martin and Kelvin J. Richards. 2001. Mechanisms for vertical nutrient transport within a North Atlantic mesoscale eddy. *Deep-Sea Research Part II: Topical Studies in Oceanography*, 48(4-5):757–773.

Josué Martínez-Moreno, Andrew McC Hogg, Matthew H England, Navid C Constantinou, Andrew E Kiss, and Adele K Morrison. 2021. Global changes in oceanic mesoscale currents over the satellite altimetry record. *Nature Climate Change*, 11(5):397–403.

Matthew R Mazloff, Patrick Heimbach, and Carl Wunsch. 2010. An eddy-permitting southern ocean state estimate. *Journal of Physical Oceanography*, 40(5):880–899.

Shelly McCallum, Janel Schultz, Kristen Sellke, and Jason Spartz. 2015. An examination of the flipped classroom approach on college student academic involvement. *International Journal of Teaching and Learning in Higher Education*, 27(1):42–55.

David A. McConnell, LeeAnna Chapman, C. Douglas Czajka, Jason P. Jones, Katherine D. Ryker, and Jennifer Wiggen. 2017. Instructional utility and learning efficacy of common active learning strategies. *Journal of Geoscience Education*, 65(4):604–625.

Janice McDonnell, Sage Lichtenwalner, Scott Glenn, Carrie Ferraro, Kristin Hunter-Thomson, and Jim Hewlett. 2015. The challenges and opportunities of using data in teaching from the perspective of undergraduate oceanography professors. *Marine Technology Society Journal*, 49(4):76–85.

T.J. McDougall and P.M. Barker. 2011. Getting started with TEOS-10 and the Gibbs Seawater (GSW) Oceanographic Toolbox. *Scor/Iapso Wg127*, page 28pp.

Charlie McDowell, Linda Werner, Heather Bullock, and Julian Fernald. 2002. The effects of pair-programming on performance in an introductory programming course. *ACM SIGCSE Bulletin*, 34(1):38–42.

Wes McKinney. 2010. Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference*, pages 56–61.

Lucas Merckelbach, David Smeed, and Gwyn Griffiths. 2010. Vertical water velocities from underwater gliders. *Journal of Atmospheric and Oceanic Technology*, 27(3):547–563.

Met Office. 2022. Cartopy: a cartographic Python library with a Matplotlib interface.

Anne Meyer, David H. Rose, and David Gordon. 2014. *Universal design for learning: theory and practice*. CAST.

Joan Middendorf and Alan Kalish. 1996. The "change-up" in lectures. *The National Teaching and Learning Forum*, 5(2):1–5.

Walter H. Munk. 1966. Abyssal recipes. *Deep-Sea Research and Oceanographic Abstracts*, 13(4):707–730.

NGDC. 2009. ETOPO1 1 Arc-Minute Global Relief Model. Accessed Feb. 27, 2022.

Qinbiao Ni, Xiaoming Zhai, Joseph Henry Lacasce, Dake Chen, and David P Marshall. 2023. Full-depth eddy kinetic energy in the global ocean estimated from altimeter and argo observations. *Geophysical Research Letters*, 50(15):e2023GL103114.

Patrick L. Old. 2019. *Bridging the gap in oceanographic data science curriculum: prototyping experiential learning materials in Python*. Undergraduate senior thesis, University of Washington.

Catherine M. O'Reilly, Tanya Josek, Rebekka D. Darner, and Sarah K. Fortner. 2022. Pedagogy of teaching with large datasets: Designing and implementing effective data-based activities. *Biochemistry and Molecular Biology Education*, 50(5):466–472.

Jenny Palomino, Leah Wasser, and Max Joseph. 2021. Introduction to earth data science textbook.

Beomjo Park, Mikael Kuusela, Donata Giglio, and Alison Gray. 2023. Spatiotemporal local interpolation of global ocean heat transport using argo floats: A debiased latent gaussian process approach. *The Annals of Applied Statistics*, 17(2):1491–1520.

Arnold Pears, Stephen Seidman, Lauri Malmi, Linda Mannila, Elizabeth Adams, Jens Bennedsen, Marie Devlin, and James Paterson. 2007. A survey of literature on the teaching of introductory programming. *ACM SIGCSE Bulletin*, 39(4):204–223.

Jeffrey M. Perkel. 2021. Ten computer codes that transformed science. *Nature*, 589(7842):344–348.

Jackie Pichette, Sarah Brumwell, and Jessica Rizk. 2020. Improving the accessibility of remote higher education: Lessons from the pandemic and recommendations. Technical report, Higher Education Quality Council of Ontario, Toronto.

RT Pierrehumbert and KL Swanson. 1995. Baroclinic instability. *Annual review of fluid mechanics*, 27(1):419–467.

Gabriela S. Pilo, Peter R. Oke, Richard Coleman, Tatiana Rykova, and Ken Ridgway. 2018. Patterns of Vertical Velocity Induced by Eddy Distortion in an Ocean Model. *Journal of Geophysical Research: Oceans*, 123(3):2274–2292.

Michael Prince. 2004. Does active learning work? A review of the research. *Journal of Engineering Education*, 93(3):223–231.

Michael Prince, Richard Felder, and Rebecca Brent. 2020. Active student engagement in online STEM classes: Approaches and recommendations. *Advances in Engineering Education*, 8(4):1–25.

Marie-Isabelle Pujol, Yannice Faugère, Guillaume Taburet, Stéphanie Dupuy, Camille Pelloquin, Michael Ablain, and Nicolas Picot. 2016. Duacs dt2014: the new multi-mission altimeter data set reprocessed over 20 years. *Ocean Science*, 12(5):1067–1090.

Rahul Ramachandran, Kaylin Bugbee, and Kevin Murphy. 2021. From open data to open science. *Earth and Space Science*, 8(5):e2020EA001562.

Salvador Ramirez, Samantha Teten, Martha Mamo, Carol Speth, Timothy Kettler, and Meghan Sindelar. 2022. Student perceptions and performance in a traditional, flipped classroom, and online introductory soil science course. *Journal of Geoscience Education*, 70(1):130–141.

Stephen C Riser, Howard J Freeland, Dean Roemmich, Susan Wijffels, Ariel Troisi, Mathieu Belbéoch, Denis Gilbert, Jianping Xu, Sylvie Pouliquen, Ann Thresher, et al. 2016. Fifteen years of ocean observations with the global argo array. *Nature Climate Change*, 6(2):145–153.

Dean Roemmich, Matthew H. Alford, Hervé Claustre, Kenneth S. Johnson, Brian King, James Moum, Peter R. Oke, W. B. Owens, Sylvie Pouliquen, Sarah Purkey, Megan Scanderbeg, Toshio Suga, Susan E. Wijffels, Nathalie Zilberman, Dorothee Bakker, Molly O. Baringer, Mathieu Belbeoch, Henry C. Bittig, Emmanuel Boss, Paulo Calil, Fiona Carse, Thierry Carval, Fei Chai, Diarmuid O. Conchubhair, Fabrizio D'Ortenzio, Giorgio Dall'Olmo, Damien Desbruyères, Katja Fennel, Ilker Fer, Raffaele Ferrari, Gael Forget, Howard Freeland, Tetsuichi Fujiki, Marion Gehlen, Blair Greenan, Robert Hallberg, Toshiyuki Hibiya, Shigeki Hosoda, Steven Jayne, Markus Jochum, Gregory C. Johnson, Ki Ryong Kang, Nicolas Kolodziejczyk, Arne Koertzinger, Pierre Yves Le Traon, Yueng Djern Lenn, Guillaume Maze, Kjell A. Mork, Tamaryn Morris, Takeyoshi Nagai, Jonathan Nash, Alberto N. Garabato, Are Olsen, Rama R. Pattabhi, Satya Prakash, Stephen Riser, Catherin Schmechtig, Emily Shroyer, Andreas Sterl, Philip Sutton, Lynne Talley, Toste Tanhua, Virginie Thierry, Sandy Thomalla, John Toole, Ariel Troisi, Tom Trull, Jonathan D. Turton, Pedro J. Velez-Belchi, Waldemar Walczowski, Haili Wang, Rik Wanninkhof, Amy Waterhouse, Andrew Watson, Cara Wilson, Annie P. Wong, Jianping Xu, and Ichiro Yasuda. 2019. On the future of Argo: A global, full-depth, multi-disciplinary array. *Frontiers in Marine Science*, 6(JUL):1–28.

Isabella Rosso, Andrew McC. Hogg, Peter G. Strutton, Andrew E. Kiss, Richard Matear, Andreas Klocker, and Erik van Sebille. 2014. Vertical transport in the ocean due to sub-mesoscale structures: Impacts in the kerguelen region. *Ocean Modelling*, 80:10–23.

Penny M. Rowe, Lea Fortmann, Timothy L. Guasco, Aedin Wright, Amy Ryken, Emma Sevier, Grace Stokes, Amanda Mifflin, Rachel Wade, Haiyan Cheng, William Pfalzgraff, Justin Beaudoin, Isha Rajbhandari, Kena Fox-Dobbs, and Steven Neshyba. 2021. Integrating polar research into undergraduate curricula using computational guided inquiry. *Journal of Geoscience Education*, 69(2):178–191.

Jorge L Sarmiento, Kenneth S Johnson, Lionel A Arteaga, Seth M Bushinsky, Heidi M Cullen, Alison R Gray, Roberta M Hotinski, Tanya L Maurer, Matthew R Mazloff, Stephen C Riser, et al. 2023. The southern ocean carbon and climate observations and modeling (soccom) project: A review. *Progress in Oceanography*, page 103130.

Jessica Scheick, Wei Ji Leong, Kelsey Bisson, Anthony Arendt, Shashank Bhushan, Zachary Fair, Norland Raphael Hagen, Scott Henderson, Friedrich Knuth, Tian Li, Zheng Liu, Romina Piunno, Nitin Ravinder, Landung Don Setiawan, Tyler Sutterley, J. P. Swinski, and Anubhav. 2023. icepyx: querying, obtaining, analyzing, and manipulating ICESat-2 datasets. *Journal of Open Source Software*, 8(84):4912.

National Academies of Sciences Engineering and Medicine. 2018. *Open science by design: Realizing a vision for 21st century research*. The National Academies Press, Washington, D.C.

Robert B Scott, Brian K Arbic, Eric P Chassignet, Andrew C Coward, Mathew Maltrud, William J Merryfield, Ashwanth Srinivasan, and Anson Varghese. 2010. Total kinetic energy in four global eddying ocean circulation models and over 5000 current meter records. *Ocean Modelling*, 32(3-4):157–169.

F. Sévellec, A. C. Naveira Garabato, J. A. Brearley, and K. L. Sheen. 2015. Vertical flow in the Southern Ocean estimated from individual moorings. *Journal of Physical Oceanography*, 45(9):2209–2220.

Jackie E. Shay and Cathy Pohan. 2021. Resilient instructional strategies: Helping students cope and thrive in crisis. *Journal of Microbiology & Biology Education*, 22(1):1–8.

K.R. Srinath. 2017. Python–the fastest growing programming language. *International Research Journal of Engineering and Technology*, 4(12):354–357.

Marilyne Stains, Jordan Harshman, Megan K. Barker, Stephanie V. Chasteen, Renee Cole, Sue Ellen DeChenne-Peters, M. K. Eagan, Joan M. Esson, Jennifer K. Knight, F. A. Laski, Marc Levis-Fitzgerald, C. J. Lee, Stanley M. Lo, Lisa McDonnell, Timothy A. McKay, Nicole Michelotti, Amanda Musgrove, Michael S. Palmer, Kathryn M. Plank, Tamara M. Rodela, Erin R. Sanders, Natalie G. Schimpf, Patricia M. Schulte, Michelle K. Smith, MacKenzie R. Stetzer, B. Van Valkenburgh, Erin Vinson, Laura K. Weir, Paul J. Wendel, Lindsay B. Wheeler, and Anna M. Young. 2018. Anatomy of STEM teaching in North American universities. *Science*, 359(6383):1468–1470.

Henry Stommel and A. B. Arons. 1959. On the abyssal circulation of the world ocean - II. An idealized model of the circulation pattern and amplitude in oceanic basins. *Deep Sea Research (1953)*, 6(C).

Masashi Sugiyama. 2016. Chapter 4 - examples of continuous probability distributions. In Masashi Sugiyama, editor, *Introduction to Statistical Machine Learning*, pages 37–50. Morgan Kaufmann, Boston.

H. U. Sverdrup. 1947. Wind-driven currents in a baroclinic ocean; with application to the equatorial currents of the eastern pacific. *Proceedings of the National Academy of Sciences*, 33(11):318–326.

Dana Swift and Stephen Riser. 1994. Rafos floats: Defining and targeting surfaces of neutral buoyancy. *Journal of Atmospheric and Oceanic Technology - J ATMOS OCEAN TECHNOL*, 11:1079–1092.

Veronica Tamsitt, Henri F Drake, Adele K Morrison, Lynne D Talley, Carolina O Dufour, Alison R Gray, Stephen M Griffies, Matthew R Mazloff, Jorge L Sarmiento, Jinbo Wang, et al. 2017. Spiraling pathways of global deep waters to the surface of the southern ocean. *Nature communications*, 8(1):1–10.

Ralf Teusner, Christoph Matthies, and Thomas Staubitz. 2018. What stays in mind? - Retention rates in programming MOOCs. In *2018 IEEE Frontiers in Education Conference*.

Elli J. Theobald, Mariah J. Hill, Elisa Tran, Sweta Agrawal, E. Nicole Arroyo, Shawn Behling, Nyasha Chambwe, Dianne Laboy Cintrón, Jacob D. Cooper, Gideon Dunster, Jared A. Grummer, Kelly Hennessey, Jennifer Hsiao, Nicole Iranon, Leonard Jones, Hannah Jordt, Marlowe Keller, Melissa E. Lacey, Caitlin E. Littlefield, Alexander Lowe, Shannon Newman, Vera Okolo, Savannah Olroyd, Brandon R. Peecook, Sarah B. Pickett, David L. Slager, Itzue W. Caviedes-Solis, Kathryn E. Stanchak, Vasudha Sundaravardan, Camila Valdebenito, Claire R. Williams, Kaitlin Zinsli, and Scott Freeman. 2020. Active learning narrows achievement gaps for underrepresented students in undergraduate science, technology, engineering, and math. *Proceedings of the National Academy of Sciences*, 117(12):6476–6483.

Kristen M. Thyng, Chad A. Greene, Robert D. Hetland, Heather M. Zimmerle, and Steven F. DiMarco. 2016. True colors of oceanography: Guidelines for effective and accurate colormap selection. *Oceanography*, 29(3):9–13.

Geoffrey P. Timms and Jeffrey R. Guyon. 2023. From zero to Python in 10.5 hours: Building foundational programming skills with marine biology graduate students and researchers in an introductory workshop series. *Science & Technology Libraries*, 42(3):371–390.

Hiroyuki Tsujino, Hiroyasu Hasumi, and Nobuo Suginohara. 2000. Deep pacific circulation controlled by vertical diffusivity at the lower thermocline depths. *Journal of Physical Oceanography*, 30(11):2853–2865.

Mickey Vellukunnel, Philip Buffum, Kristy Elizabeth Boyer, Jeffrey Forbes, Sarah Heckman, and Ketan Mayer-Patel. 2017. Deconstructing the discussion forum: Student questions and computer science learning. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pages 603–608. ACM.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, Aditya Vijaykumar, Alessandro Pietro Bardelli, Alex Rothberg, Andreas Hilboll, Andreas Kloeckner, Anthony Scopatz, Antony Lee, Ariel Rokem, C. Nathan Woods, Chad Fulton, Charles Masson, Christian Häggström, Clark Fitzgerald, David A. Nicholson, David R. Hagen, Dmitrii V. Pasechnik, Emanuele Olivetti, Eric Martin, Eric Wieser, Fabrice Silva, Felix Lenders, Florian Wilhelm, G. Young, Gavin A. Price, Gert-Ludwig Ingold, Gregory E. Allen, Gregory R. Lee, Hervé Audren, Irvin Probst, Jörg P. Dietrich, Jacob Silterra, James T. Webber, Janko Slavič, Joel Nothman, Johannes Buchner, Johannes Kulick, Johannes L. Schönberger, José Vinícius de Miranda Cardoso, Joscha Reimer, Joseph Harrington, Juan Luis Cano Rodríguez, Juan Nunez-Iglesias, Justin Kuczynski, Kevin Tritz, Martin Thoma, Matthew Newville, Matthias Kümmerer, Maximilian Bolingbroke, Michael Tartre, Mikhail Pak, Nathaniel J. Smith, Nikolai Nowaczyk, Nikolay Shebanov, Oleksandr Pavlyk, Per A. Brodtkorb, Perry Lee, Robert T. McGibbon, Roman Feldbauer, Sam Lewis, Sam Tygier, Scott Sievert, Sebastiano Vigna, Stefan Peterson, Surhud More, Tadeusz Pudlik, Takuya Oshima, Thomas J. Pingel, Thomas P. Robitaille, Thomas Spura, Thouis R. Jones, Tim Cera, Tim Leslie, Tiziano Zito, Tom Krauss, Utkarsh Upadhyay, Yaroslav O. Halchenko, and Yoshiki Vázquez-Baeza. 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272.

Jin-Song Von Storch, Carsten Eden, Irina Fast, Helmuth Haak, Daniel Hernández-Deckers, Ernst Maier-Reimer, Jochem Marotzke, and Detlef Stammer. 2012. An estimate of the lorenz energy cycle for the world ocean based on the storm/ncep simulation. *Journal of physical oceanography*, 42(12):2185–2205.

Tianyu Wang, Sarah T Gille, Matthew R Mazloff, Nathalie V Zilberman, and Yan Du. 2020. Eddy-induced acceleration of argo floats. *Journal of Geophysical Research: Oceans*, 125(10):e2019JC016042.

Johnny Wei-Bing Lin, Hannah Aizenman, Erin Manette Cartas Espinel, Kim Gunnerson, and Joanne Liu. 2022. *An introduction to Python programming for scientists and engineers*. Cambridge University Press.

Laurie Williams and Richard L. Upchurch. 2001. In support of student pair-programming. *ACM SIGCSE Bulletin*, 33(1):327–331.

Greg Wilson. 2016. Software Carpentry: lessons learned. *F1000Research*, 3:62.

Greg Wilson, Dhavide A. Aruliah, C. Titus Brown, Neil P. Chue Hong, Matt Davis, Richard T. Guy, Steven H. D. Haddock, Kathryn D. Huff, Ian M. Mitchell, Mark D. Plumbley, Ben Waugh, Ethan P. White, and Paul Wilson. 2014. Best practices for scientific computing. *PLoS Biology*, 12(1):e1001745.

Annie P.S. Wong, Susan E. Wijffels, Stephen C. Riser, Sylvie Pouliquen, Shigeki Hosoda, Dean Roemmich, John Gilson, Gregory C. Johnson, Kim Martini, David J. Murphy, Megan Scanderbeg, T. V.S.Udaya Bhaskar, Justin J.H. Buck, Frederic Merceur, Thierry Carval, Guillaume Maze, Cécile Cabanes, Xavier

André, Noé Poffa, Igor Yashayaev, Paul M. Barker, Stéphanie Guinehut, Mathieu Belbéoch, Mark Ignaszewski, Molly O.Neil Baringer, Claudia Schmid, John M. Lyman, Kristene E. McTaggart, Sarah G. Purkey, Nathalie Zilberman, Matthew B. Alkire, Dana Swift, W. Brechner Owens, Steven R. Jayne, Cora Hersh, Pelle Robbins, Deb West-Mack, Frank Bahr, Sachiko Yoshida, Philip J.H. Sutton, Romain Cancouët, Christine Coatanoan, Delphine Dobbler, Andrea Garcia Juan, Jerôme Gourrion, Nicolas Kolodziejczyk, Vincent Bernard, Bernard Bourlès, Hervé Claustre, Fabrizio D'Ortenzio, Serge Le Reste, Pierre Yve Le Traon, Jean Philippe Rannou, Carole Saout-Grit, Sabrina Speich, Virginie Thierry, Nathalie Verbrugge, Ingrid M. Angel-Benavides, Birgit Klein, Giulio Notarstefano, Pierre Marie Poulain, Pedro Vélez-Belchí, Toshio Suga, Kentaro Ando, Naoto Iwasaska, Taiyo Kobayashi, Shuhei Masuda, Eitarou Oka, Kanako Sato, Tomoaki Nakamura, Katsunari Sato, Yasushi Takatsuki, Takashi Yoshida, Rebecca Cowley, Jenny L. Lovell, Peter R. Oke, Esmee M. van Wijk, Fiona Carse, Matthew Donnelly, W. John Gould, Katie Gowers, Brian A. King, Stephen G. Loch, Mary Mowat, Jon Turton, E. Pattabhi Rama Rao, M. Ravichandran, Howard J. Freeland, Isabelle Gaboury, Denis Gilbert, Blair J.W. Greenan, Mathieu Ouellet, Tetjana Ross, Anh Tran, Mingmei Dong, Zenghong Liu, Jianping Xu, Ki Ryong Kang, Hyeong Jun Jo, Sung Dae Kim, and Hyuk Min Park. 2020. Argo Data 1999–2019: Two Million Temperature-Salinity Profiles and Subsurface Velocity Observations From a Global Array of Profiling Floats. *Frontiers in Marine Science*, 7(September):1–23.

Cimarron Wortham and Carl Wunsch. 2014. A multidimensional spectral description of ocean variability. *Journal of physical oceanography*, 44(3):944–966.

Carl Wunsch. 1997. The vertical partition of oceanic horizontal kinetic energy. *Journal of Physical Oceanography*, 27(8):1770–1794.

Carl Wunsch and Raffaele Ferrari. 2004. Vertical mixing, energy, and the general circulartion of the oceans. *Annual Review of Fluid Mechanics*, 36(Volume 36, 2004):281–314.

Madeleine K Youngs and Glenn R Flierl. 2023. Extending residual-mean overturning theory to the topographically localized transport in the southern ocean. *Journal of Physical Oceanography*, 53(8):1901–1915.

Madeleine K Youngs, Andrew F Thompson, Ayah Lazar, and Kelvin J Richards. 2017. Acc meanders, energy transfer, and mixed barotropic–baroclinic instability. *Journal of Physical Oceanography*, 47(6):1291–1305.

Zhonggen Yu and Mingle Gao. 2022. Effects of video length on a flipped English classroom. *SAGE Open*, 12(1):215824402110684.

Claire K Yung, Adele K Morrison, and Andrew McC Hogg. 2022. Topographic hotspots of southern ocean eddy upwelling. *Frontiers in Marine Science*, 9:855785.

Richard F. Yuretich, Samia A. Khan, R. Mark Leckie, and John J. Clement. 2001. Active-learning methods to improve student performance and scientific interest in a large introductory oceanography course. *Journal of Geoscience Education*, 49(2):111–119.

NV Zilberman, M Scanderbeg, AR Gray, and PR Oke. 2023. Scripps argo trajectory-based velocity product: Global estimates of absolute velocity derived from core, biogeochemical, and deep argo float trajectories at parking depth. *Journal of Atmospheric and Oceanic Technology*.

Chapter A

Appendix One: Chapter 5 supplemental

The materials of this chapter are supplemental to Chapter 5 and are referenced in the text by using the letter A in the reference. It has been divided into Figures, Tables, and Text sections for convenience. These are included in the publication of Campbell et al. [2024] as supplemental materials.

A.1 Tables

Table A.1: Functions, operators, and methods taught in the course that were used as search terms to assess the complexity of students' final project code. A Python script was used to count instances of each search term in students' project code notebooks, and the number of search terms used at least once (expressed as a percent of all search terms below) is presented as the metric "Python skills used in project" in Figure 5.4.

Topic	Search terms
Basic functions	'len(', 'print(', 'display(', 'range(', 'enumerate(', 'zip(', 'int(', 'float(', 'complex(', 'bool(', 'tuple(', 'type(', 'readline('
Lists	'list(', '.append(', '.extend(', '.insert(', '.remove(', 'del', '.pop(', '.reverse(', '.copy(', '.join(', '.sort('
Strings	'str(', '.lstrip(', '.rstrip(', '.upper(', '.lower(', '.count(', '.replace(', '.split(', '.format('
NumPy	'np.array(', '.dtype', '.astype(', 'np.append(', 'np.insert(', 'np.flip(', 'np.tolist(', '.sum(', '.mean(', '.median(', '.max(', '.min(', 'np.std(', 'np.pi', 'np.e', 'np.inf', 'np.nan', 'np.absolute(', 'np.round(', 'np.sqrt(', 'np.exp(', 'np.sin(', 'np.cos(', 'np.zeros(', 'np.ones(', 'np.full(', 'np.arange(', 'np.linspace(', '.size', '.ndim', '.shape', '.reshape(', ',flatten(', '.transpose(', '.vstack(', '.hstack(', 'np.genfromtxt(', 'np.meshgrid('
Time	'datetime.now()', '.year', '.month', '.day', '.hour', '.minute', '.second', '.microsecond', 'datetime.strptime', 'datetime.strftime', '.total_seconds()', 'timedelta', 'mdates.date2num('
Pandas	'.Series(', '.index', '.values', '.loc[', '.iloc[', 'pd.concat(', 'pd.DataFrame(', '.describe(', '.to_csv(', '.read_csv(', '.read_excel('
Xarray	'.open_dataset(', '.open_mfdataset(', '.attrs', '.isel(', '.sel(', '.item'
SciPy	'stats.linregress(', 'interpolate.interp1d(', 'interpolate.griddata('
Plotting	'.figure(', '.subplots(', '.xlabel(', '.ylabel(', '.set_xlabel(', '.set_ylabel(', '.grid', '.colorbar', '.set_label', '.clabel', '.invert_yaxis', '.gca', '.axes', '.coastlines', '.add_feature(', '.set_extent('
Plot types	'.plot(', '.scatter(', '.hist(', '.contour(', '.contourf(', '.pcolormesh('
Logic	' if ', ' while ', ' for ', ' is ', ' in ', ' not ', ' else:', ' elif ', ' and ', ' ', '==', '!=', '>=', '<='

Table A.2: Grading rubric for students' final research projects. In the first column, corresponding main student learning objectives (SLOs) are appended to the rubric (see Implementation section "Course history and development" for the full numbered list of SLOs). Figure 3 depicts assessments of students' final projects using this rubric, grouped by theme and SLOs.

Corresponding student learning outcome(s)		Limited (0-50%)	Good (50-75%)	Exceptional (75-100%)	
	Presentation Content				
SLO #5 ("Formulate and investigate scientific research questions")	Background	Topic background is missing or severely lacking in detail.	Topic background is sufficient, but missing some details or lacks coherency.	Topic background is clear, complete, and relevant.	3 points
	Questions / hypotheses	Questions are not well-defined. Hypotheses are not substantiated.	Questions are well-defined. Hypotheses draw on prior knowledge.	Questions are well-defined and pertinent for the topic. Hypotheses draw on prior knowledge and have clear explanations for why they are expected.	2 points
SLO #3 ("Access, read, transform... and interpret oceanographic data with confidence using Python")	Data collection	Information about the data collection process is missing key details or is inaccurate. The limitations of the data are missing or not realistic.	Information about the data collection process is accurate, but missing some minor details. The limitations of the data are explained.	Information about the data collection process is complete and accurate. Underlying problems and limitations of the data are explained. Use of these data to answer the project questions is justified.	3 points
	Data processing	The student has made errors in processing their data. The student is missing steps.	The student has processed the data correctly. Steps for obtaining, loading, cleaning, and analyzing the data are well-defined.	The student has processed the data correctly and taken precautions to ensure that their results are appropriate. Steps for obtaining, loading, cleaning, and analyzing the data are well-defined.	3 points

	Results / interpretation	Results of the project do not attempt to answer the scientific questions. The data visualizations are not relevant.	Results of the project somewhat answer the scientific questions. Data visualizations are mostly appropriate for the data.	Results of the project answer, or earnestly attempt to answer, the scientific questions. Data visualizations are entirely appropriate for the data.	3 points
	Presentation Skills				
N/A	Organization	The presentation is not in a logical order and the student makes no effort to guide the audience.	The presentation is organized in a logical order and takes some care to guide the audience.	The presentation is organized in a logical order and shows exceptional attention to guiding the audience.	2 points
	Timing	The student far exceeds their allotted time and/or has not made an effort to practice.	The student completes the presentation in somewhat over 5 minutes.	The student completes the presentation within 5 minutes and it is clear that they have practiced.	1 points
SLO #5 ("Formulate and investigate scientific research questions")	Explanation of ideas/ information	The ideas and information explained in the presentation were not clear and were not relevant.	The ideas and information explained in the presentation were clear and relevant.	The ideas and information explained in the presentation were exceptionally clear, relevant, and coherent.	3 points
	Code				
SLO #2 ("Write, execute, and debug Python code"), SLO #6 ("Adopt best practices in programming")	Correctness	The student misuses code and does not produce reasonable results.	The student uses some coding techniques/tools learned throughout the quarter. The analysis produces reasonable answers that can be replicated with some effort.	The student properly and efficiently uses the coding techniques/tools learned throughout the quarter. The analysis produces reasonable answers that can be replicated easily.	8 points
	Functionality	The code does not run and has egregious errors.	The code is mostly able to run, but has some (small) errors.	The code runs efficiently with no errors.	5 points

	Tidiness	The code breaks proper etiquette and should not be shared with others.	The code mostly follows proper coding etiquette. The organization is somewhat lacking and would need review before sharing.	The code follows proper coding etiquette. It is organized and commented effectively so that it can easily be shared with another person.	6 points
	Perseverance	The student has made no effort to work through problems and hurdles.	The student has made some effort to work through problems.	The student has made a gallant effort to work through problems and documented in their code their best understanding of the problems they are facing.	5 points
	Visualizations				
SLO #3 ("... visualize... oceanographic data with confidence using Python"), SLO #6 ("Adopt best practices in... data visualization")	Plot clarity	The plots are unclear and do not make sense in the context of the project.	The plots are mostly clear and show some thought from the students about ways to present their data.	The plots are extremely clear and are effective tools to help the audience understand the results/analysis.	5 points
	Colormaps	The colormaps are not appropriate for the data being shown.	The colormaps are appropriate for the data being shown.	The colormaps are appropriate for the data being shown and take into account colorblindness, and perceptual accuracy.	3 points
	Proper labels	The plots are missing most/all labels or have improper labels.	The plots are labeled with general accuracy and completion.	The plots are labeled extremely accurately in a way that guides the audience through the figure.	5 points
	Creativity	The student made no effort to create original plots.	The student has made some effort to create original plots.	The student has created original plots that show the data/analysis in an extremely effective manner.	3 points

Table A.3: Rubric used to assess students' prior coding experience based on their written responses to the Assignment #0 survey during Week 1 of the course. Students were asked: "Do you have prior coding experience, and if so, with what language?" and "How comfortable do you feel using technology?" Responses to the first question were graded subjectively based on word choice on a scale from 1-5, using the keywords in quotes (e.g., "a little") when present. As noted below, additional points were awarded to weight responses in favor of prior exposure to Python or similar high-level and/or interpreted languages (MATLAB, Java, R). Points were subtracted to account for less relevant prior experience. If no level of coding proficiency was provided, the base number used was from the students' "comfort with technology" statement ("Very comfortable": 4; "Fairly comfortable": 2). Results are used in Figure A.3 and presented as the metric "Prior coding experience" in Figure 5.4

1	2	3	4	5
No experience	Minimal experience (e.g., "a little", "small", "tiny amount")	"Some" or "moderate" experience	Experience	Experience (with full additions)

Additions (maximum total: +1.0)	**Subtractions (maximum total: -0.5)**
+0.5 for one of MATLAB, Java, R	-0.5 if response mentions many years since their previous experience
+1.0 for Python or multiple languages	-0.5 if response mentions that their previous experience was not useful

Table A.4: Mapping of university-administered IAS final course evaluation questions from 2015-2019 to 2020. The mapping allows the slightly different evaluation questions from the two periods to be compared in Figure 5.2 and Figure A.1. Metrics listed are the median of responses collected for each class.

Paraphrased question	**Original evaluation question(s) (2015-2019)**	**Original evaluation question(s) (2020)**	**Metric and units**
Time spent on course	On average, how many hours per week have you spent on this course, including attending classes, doing readings, reviewing notes, writing papers and any other course related work?		Hours per week
Time spent that was valuable	From the total average hours above, how many do you consider were valuable in advancing your education?		Hours per week, expressed as percent relative to response to question above
Expected grade	What grade do you expect in this course?		GPA scale (0.0-4.0)
Expected grade relative to other courses	Do you expect your grade in this course to be:		1-7 scale ("Much lower" to "Much higher")
Effort invested relative to other courses	The amount of effort you put into this course was:		

Participation relative to other courses	Your involvement in course (doing assignments, attending classes, etc.) was:	Relative to similar courses taught in person, your participation in this course was:	
Intellectual challenge relative to other courses	The intellectual challenge presented was:		
Course as a whole	The course as a whole was:	The remote learning course as a whole was:	0-5 scale ("Very poor" to "Excellent"))
Course content	The course content was:		
Usefulness of course content	Relevance and usefulness of course content were:	Average of: "Usefulness of reading assignments in understanding course content was:", "Usefulness of written assignments in understanding course content was:", "Usefulness of online resources in understanding course content was:"	
Facilitation of learning	Amount you learned in the course was:	The effectiveness of this remote course in facilitating my learning was:	
Evaluation and grading techniques	Evaluative and grading techniques (tests, papers, projects, etc.) were:		
Reasonableness of assigned work	Reasonableness of assigned work was:		
Organization	Course organization was:	Organization of materials online was:	
Clarity of student responsibilities	Clarity of student responsibilities and requirements was:		
Instructor's contribution to the course	The instructor's contribution to the course was:		
Effectiveness of instructor's teaching	The instructor's effectiveness in teaching the subject matter was:		
Quality of instructor answers and feedback	Average of: "Explanations by instructor were:", "Instructor's ability to present alternative explanations when needed was:", "Instructor's interest in whether students learned was:", "Answers to student questions were:"	Quality/helpfulness of instructor feedback was:	

Table A.5: Open-ended questions asked in university-administered IAS mid-quarter and final course evaluations in 2020. Students' anonymous responses are tabulated in Figure 5.5 and are excerpted throughout this study.

Evaluation period	**Question**
Mid-quarter	What is helping you to learn in this course? What is hindering your learning in this course? What can your instructor do to improve your learning in this course?
Final	Was this class intellectually stimulating? Did it stretch your thinking? Why or why not? What aspects of this class contributed most to your learning? What aspects of this class detracted from your learning? What suggestions do you have for improving this class generally? If this course were offered remotely again, what suggestions do you have to improve the student experience?

Table A.6: List of guiding questions offered to undergraduate student focus group for structuring their testimonial submissions, which are presented in Text A.3 (also see Section 5.3.6). Students were encouraged to address one or more of the questions in their submissions.

1. How did your prior experience with coding (or lack of prior experience) impact your experience with the course? If you have prior coding experience and it was self-taught, what do you see as the benefits of learning scientific programming in a structured environment rather than teaching it to yourself? If your prior coding knowledge was learned from course(s), how did we teach programming that was different and more or less effective than those past course(s)?
2. How did the accessibility elements that we implemented (e.g., captioning, syllabus late policy, extensions, not grading on attendance, breaks during class, virtual office hours, making slide decks available, video optional on Zoom, ability to use chat during class, no course prerequisites, extra credit opportunities, etc.) affect your success in the course?
3. How did the expectations and norms established in the course impact your experience?
4. How did you navigate the course policies we created on collaboration and original work? If you worked with a partner on assignments and/or the final project, what was your experience like? Was it productive/challenging/surprising, and how did the technological tools we used (Colab, Zoom) facilitate it? What advice would you give to professors who are teaching a programming course and want to create opportunities for collaboration?
5. How did the key course elements (recorded videos, in-class activities, assignments, final project, etc.) and technological platforms (Google Colab, Piazza, Zoom, Google Drive/Docs, Canvas) help or hinder your learning?
6. Instead of a textbook, we allowed use of external resources (e.g., documentation websites, Stack Overflow, etc.). How did this compare to having a textbook for the course?
7. How did guidance from the instructors and classmates (via Piazza or in class) help you complete assignments and shape and execute your final project?
8. In what ways did the class help you learn about oceanography sub-disciplines (marine geology, chemistry, physics, biology) or other earth science subjects adjacent to oceanography (e.g., cryosphere, meteorology, climate)? What value do you see in teaching programming in an oceanography curriculum rather than a computer science department?
9. How do you feel this course fit into your overall undergraduate education? How did this course prepare you for future research, like your senior thesis? In what ways do you feel more capable now that you have Python in your arsenal?
10. How do you feel this course shaped your career/life goals or motivation to pursue oceanography or data science during and after college?
11. What was it like taking this class during the pandemic? How does this course compare to other classes you've taken remotely during the pandemic?

A.2 Figures

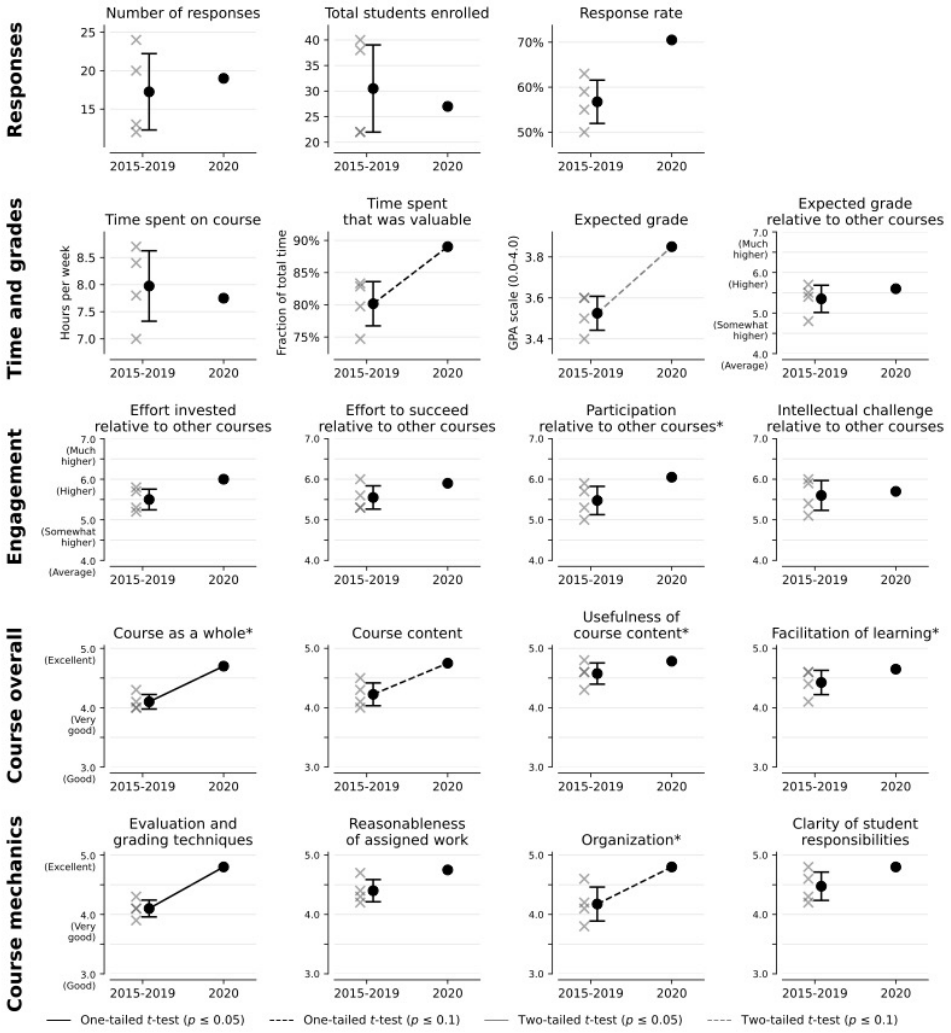

Figure A.1: All metrics from anonymous final student evaluations in 2015, 2016, 2017, 2019, and 2020 (see Evaluation section "Surveys and evaluations"). Differently worded questions, indicated with an asterisk (), were mapped between years as shown in Table A.4. Metrics shown are class medians for 2015, 2016, 2017, and 2019 (gray crosses, except for those in the first row ["Responses"]); 2015-2019 mean or 2020 class median (black points); and 2015-2019 standard deviation (bars). Changes from 2015-2019 to 2020 were tested for a significant increase using a one-tailed (black line) t-test or a significant change using a two-tailed (gray line) t-test at the 95% (solid line) or 90% (dashed line) confidence level according to the methodology detailed in Section 5.3.2. An absence of a line connecting the 2015-2019 and 2020 data indicates no statistically significant improvement or difference. Note that y-axes have been truncated from the full 1-5 scale ("Very poor" to "Excellent") or 1-7 scale ("Much lower" to "Much higher"). Evaluation questions for which a consistent mapping across years was not possible were excluded; instructor-specific questions are also not shown.

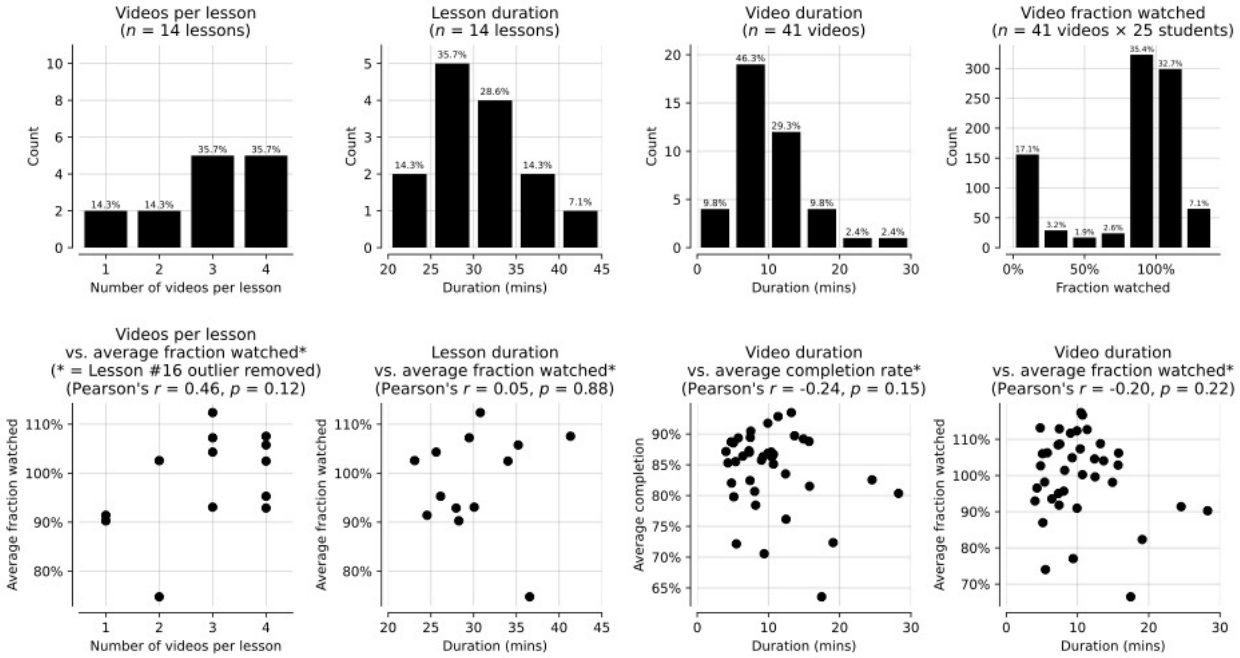

Figure A.2: Additional statistics on flipped lesson videos that were posted and viewed on the Panopto platform, based on video-specific metrics obtained from Panopto. Pearson's r represents the linear correlation between two variables. Note that none of the correlations tested in panels (e)-(h) were significant at the 95% ($p \leq 0.05$) or 90% ($p \leq 0.1$) confidence level. (a) Distribution of number of videos included per lesson (as the 14 topical lessons were usually split into multiple videos). (b) Distribution of the total duration of lessons. (c) Distribution of individual video duration. (d) Distribution of fraction of each video watched for each student. Fraction watched represents the total minutes that a specific video was viewed by a specific student divided by its duration, and thus can exceed 100% due to rewinds and repeat views. (e) Videos per lesson vs. video fraction watched, averaged across all students. Note that the final video lesson (Lesson #16) was excluded as an outlier due to its lower viewership where indicated using an asterisk (). (f) Lesson duration vs. fraction watched, averaged across all students. (g) Video duration vs. completion rate, averaged across all students. Completion rate represents the fraction of a video that was viewed at least once, and thus is capped at 100% for a specific student and video (unlike "fraction watched"). (h) Video duration vs. fraction watched, averaged across all students.

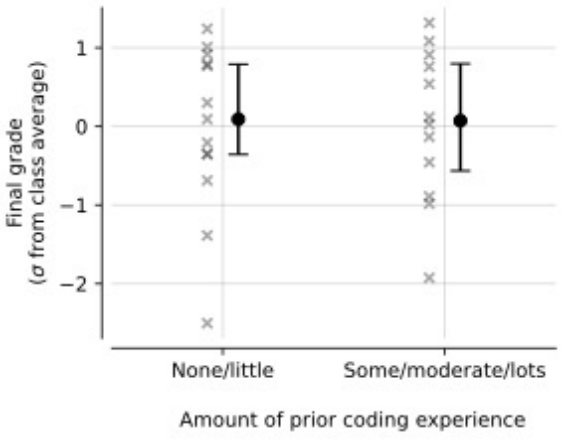

Figure A.3: Final course grades dichotomized by amount of prior coding experience. Coding experience was assessed using students' written responses to the Assignment #0 survey (see Table A.3 for rubric and methodology) and is divided here into two groups with none/little experience (score of 1 or 2) and some/-moderate/lots of experience (score of 3, 4, or 5) containing approximately equal numbers of students. Final grades were recalculated to ignore two students' incomplete assignments (see Section 5.3.5) and are expressed as standard deviations from the class average (gray crosses). Error bars represent the median and interquartile range (25%-75%) of final grades for each population. No significant difference in final grades was found between the two groups using a two-sided t-test ($p = 0.89$).

A.3 Text

Text A.1

Final project assignment handout. Note that the grading rubric presented as Table A.2 (modified to include corresponding SLOs) was also distributed to students. A list of due dates is not included here.

OCEAN 215 | Autumn 2020 | **Final project**

Project Description

During this course, you will conduct a small scientific research project from start to finish. You will choose a topic, produce a scientific question related to your topic, suggest a hypothesis, locate data that will help support or reject your hypothesis, analyze/visualize this data using Python, and present your findings to the class. Along the way, you will be responsible for giving your input on other students' projects and you will receive input from other students as well. To further reflect the collaborative nature of scientific research, we also encourage you to post any questions or challenges you encounter during this project to the class on Piazza. If you wish, you may work with a partner on this project. See below for important information if you choose this option. The majority of this project will involve writing Python code to analyze and visualize your chosen data. We will dedicate a substantial amount of class time for this work, during which instructors and peers will be present to help you work through coding challenges. Throughout the quarter, there are a number of due dates for different parts (see the table above) designed to guide you through completing your research. The expectations for the deliverables of this project are detailed below:

1. **Topic check-in:** Consider a topic of research that you would like to examine. If you are having trouble

identifying a topic, contact the instructors privately on Piazza so they can help you find something that interests you. Once you have identified your topic, create a private note to the instructors on Piazza in the 'final_project' folder that answers the questions below.

- What research topics or questions are you interested in?
- What type(s) of data would help you look into those topics/questions?

2. **Data check-in:** Locate data that will help you look into your selected topic. We will set aside some class time for students to work on this. You can start your exploration by using an internet search engine to look up background information on your topic and find possible data sources. You can also use the oceanography data repositories (e.g. PO.DAAC, NASA Giovanni, BCO-DMO, etc.) listed in the Class #1 slides. As always, the instructors are also available to help you locate a fitting data source. In the 'final_project' folder of Piazza, respond to the data check-in post with answers to the questions below. Make sure that your response is visible to the whole class.

   - What data set(s) do you plan to use?
   - What is one scientific question that you might be able to answer using these data?
   - What is your hypothesis? What do you anticipate the answer to your scientific question is, and why? (try to bring in scientific knowledge from previous courses, published literature, and/or reliable internet sources)

3. **Piazza responses:** Respond to at least 3 other data check-in posts written by your classmates on Piazza with an additional question that they might be able to investigate using their data or about their topic. To reflect the collaborative nature of research, where colleagues often help to dictate research priorities, you will choose one question suggested by a classmate and one question of your own to investigate.

4. **Project presentations:** Present the results of your project to the class in a 5 minute presentation. Presentation schedules will be posted to Canvas later this quarter. Your presentation should include the following:

   - Scientific background on your topic [$\sim$1 slide]
   - Two scientific questions (yours and a classmate's from Piazza) with your hypotheses [$\sim$1 slide]
   - Information about your data (How/when/where was it collected? What instruments were used? Are there any limitations to your data?) [$\sim$1 slide]
   - Your process for obtaining, loading, cleaning, visualizing and analyzing the data. Describe your data file format(s) and structure(s) as well as any challenges you encountered [$\sim$1-2 slides]
   - Answers to your scientific questions with associated plots and an explanation of your analysis results [$\sim$2-3 slides, $\sim$2-3 figures]

5. **Slides and code:** Submit the slides from your project presentation, your data files/folders, and the code you wrote to analyze your data and create your figures. Your code should follow proper coding etiquette and your figures should be formatted properly. To submit your code, data, and slides, save them and put them in your individual class Google Drive folder. There is no written essay required for this project.

NOTE: Piazza posts that are required for the final project do not count towards the required 5 Piazza posts detailed in the syllabus.

Pair programming option

If you wish, you may work with a partner on this project. This could be a valuable opportunity to experience a research collaboration, work through coding challenges together, and accomplish even more analysis! If you choose this option, the following expectations supersede (override) the requirements listed elsewhere in this document:

- Starting with the data check-in, you and your partner may choose a single data set together, and share this identical data set on Piazza. However, please each offer a different scientific question and hypothesis in your Piazza posts (i.e. the two of you will come up with two questions and two hypotheses in total).

- For the Piazza responses, you and your partner should each respond to three classmates' posts (not including your partner's post), for a total of six posts. You will jointly choose one question from a classmate and two questions of your own to investigate for your project, for a total of three questions to investigate.

- For the final project presentations, please prepare a single 8-10 minute slideshow, instead of a 5 minute slideshow. Include at least the number of slides specified above for each category. Trade off roles when presenting (i.e. each person should be presenting for about 4-5 minutes).

- You may submit separate Colab code notebooks, or a single joint Colab notebook. However, in all notebooks submitted, please indicate which student wrote each section of code using Python comments. We expect that both partners will contribute approximately equally to writing code for the project.

- You will be graded jointly and will receive the same grade for the project, except under extenuating circumstances to be determined on a case-by-case basis.

Grading breakdown

Project Part	**Grading**	**Points**
Topic check-in:	Complete/Incomplete	10 points
Data check-in:	Complete/Incomplete	15 points
Piazza responses:	Complete/Incomplete	15 points
Project presentation:	See rubric	20 points
Code	See rubric	40 points

Text A.2

Testimonials shared by undergraduate student coauthors (see Section 5.3.6 for more details). The students were encouraged to address one or more of the guiding questions listed in Table A.6 in their submissions.

Other coding classes that I have taken have generally failed to place skills in the context of applications. Without examples of methods being used, there is less of an incentive to understand them. In contrast, this course provided the opportunity to work with oceanographic data, allowing us to recognize the significance of the methods we were applying. For instance, ocean glider data was used to teach about interpolation. This was engaging because we first visualized the original, non-interpolated data and could see the gaps due to the physical motion of the device, then compared this with the data interpolated using the same axes and color scale. Additionally, the lack of a textbook in this course made it easier to approach methods beyond what we learned in class. Instead, we learned to answer questions by accessing online resources like Stack Overflow. Doing so developed essential skills and gave me the confidence to apply new concepts in my final project. This meant my research could be dictated by my curiosity and questions, as it should be, and not by the limitations of what concepts we had covered in class. In general, research can seem intimidating to many students because it relies on an individual's creativity. In other classes with exclusively rigid assignments and predetermined tasks, there is little opportunity for students to form original ideas, let alone develop them. In this class, we used creativity and critical thinking skills to develop a final project that answered an independently formed question. This experience has helped to prepare me for research. -O.B.

I previously took a Fortran class at the Ocean University of China, which had two traditional lectures and one lab each week. In that class, most students were not engaged during the lectures, which led them to be bewildered when doing real coding. I have also been teaching myself MATLAB for three years, basically learning by doing tasks with the help of the internet. This process has often been time-consuming, and it has been hard to organize my notes in a logical way. In comparison to those experiences, this course provided a logical pathway into Python, especially for oceanography applications. Without this class, it would have taken ten times longer to acquire the same knowledge, which would also have been less clear. In class, Zoom breakout rooms forced everyone to discuss and practice the coding, which in turn forced us to come well-prepared for class. Though Google Colab has limited storage (RAM) and is unable to process large data sets, it is great for starters. Most of my other classes have been about theory and previously derived conclusions in the field, but this class has provided a bridge between theory and practice. After taking this course, I would say that we can now start to connect math and data to discover the areas of science we are interested in. -J.L.

I have always viewed research as something that is extraordinarily complicated. This class demonstrated that knowing a few basic Python functions and packages can provide a solid foundation to start conducting research. Additionally, offering this class as part of an oceanography curriculum instead of relying on a computer science department allowed us to learn about programming skills in a way that directly applied to our interests and studies. I liked the way that the course was set up, in which we learned the material in an asynchronous video first and then practiced it in class. This helped me to discover where my gaps in understanding were and to learn from other people who may have understood a concept better than I did. Google Colab may not be the most powerful programming platform, but it is streamlined and easy to use, which made it great for first-time coders like me. Piazza was also an exceptionally useful resource. Many classes present an idealized version of how research works. This class didn't. It was an important learning experience when my final research project didn't yield the correlation I expected. This was frustrating since I put so much time and effort into the project, but it showed that a lack of correlation can be an important result and that one's research doesn't always have to produce a major scientific breakthrough. -R.M.

I came in with a little prior coding experience thanks to robotic projects that I completed with my father as a child. In taking this class, the love of coding that I had as a child was reignited. I hadn't realized how beneficial and necessary knowing a programming language would be for research. Having Python in my arsenal opened up research opportunities that I wouldn't have been qualified for before and can aid me in branching out beyond oceanography in the future. The great experience I had in this class – and my realization that research and coding are extremely integrated – inspired me to pursue a minor in Data Science. In this class, the coding assignments were based on real-world problem solving. I loved having the opportunity to work with a partner because we coded in completely different ways, and it was fascinating to see those differences. We were more effective together because we learned to compromise and collaborate to find the cleanest and fastest method between the two of us. Writing code on Zoom was a good alternative to in-person collaboration because we could share our screens and help pinpoint issues in each other's code. In addition, Piazza was helpful for me because it allowed anonymous or private questions, which avoids the uncomfortable feeling of asking a question that you think might be silly. I liked that we were able to get quick and helpful feedback on our code. It was a better way of communicating than those I have used in other classes, like email, which might get drowned out in a teacher's inbox, or Slack, which doesn't provide the anonymity that Piazza does. -I.O.
