

PYDA—A Python Dashboard for Easy and Customizable Visualization and Fitting of Biophysical Data

Alexander S. Hebbelstrup^{1,*}, Kaare Teilum¹, Birthe B. Kragelund^{1,2}

¹Structural Biology and NMR Laboratory, Linderstrøm-Lang Center for Protein Science, Department of Biology, University of Copenhagen, Copenhagen, Denmark

²REPIN, Department of Biology, University of Copenhagen, Copenhagen, Denmark

ABSTRACT Research-integrating teaching offers a collective approach to engage students and researchers in real research questions, introducing them to authentic problem-solving strategies. In experimental research-integrating teaching, various different data are acquired and often analyzed by using specialized software programs. However, learning to operate these programs can sometimes divert attention from the research itself. To keep the focus on the research, we have developed a Python-based dashboard (PYDA) and implemented it successfully in a research-integrating course on intrinsically disordered proteins. PYDA simplifies data analysis by handling experimental data from techniques such as liquid chromatography, far-ultraviolet circular dichroism, and fluorescence spectroscopy in a unified way. Developed in Python by using the Plotly Dash framework and hosted on Heroku, PYDA allows students to visualize, convert, and analyze data without extensive prior knowledge of Python. By providing a streamlined, flexible, and customizable solution, PYDA enables students to focus on research and hypothesis development, fostering digitally competent and inquiry-driven problem solvers. PYDA is supplied with a set of example files that allows the user to test the program. These files can be accessed as the last item in the dropdown menu.

KEY WORDS research-integrating teaching; digitization; plotting; protein science; intrinsically disordered proteins; protein-protein interactions; protein stability; fitting; circular dichroism; fluorescence; spectroscopy; biochemistry; Python; dashboard

I. INTRODUCTION

With increasing demands for inquiry-competent problem solvers in an ever more digital world, universities are rethinking teaching formats and styles to promote students as active contributors to exploring and solving scientific problems (1, 2). One way to do this is through research-integrating teaching (RIT) (3). RIT is characterized by a dynamic relation between the processes of realization of both the student and the researcher and is centered around a scientific area of interest to both parties (4, 5).

A critical ingredient of a recently developed RIT-based biophysics course is the formulation of a scientific question, which is then addressed by the students through access to various biophysical methods. By independently testing a self-defined hypothesis, students take ownership and ultimately generate a large amount of varied and complex scientific data. As an example, an emerging scientific

“*” corresponding author

Received: 6 January 2024

Accepted: 13 August 2024

Published: 7 October 2024

© 2024 Biophysical Society.

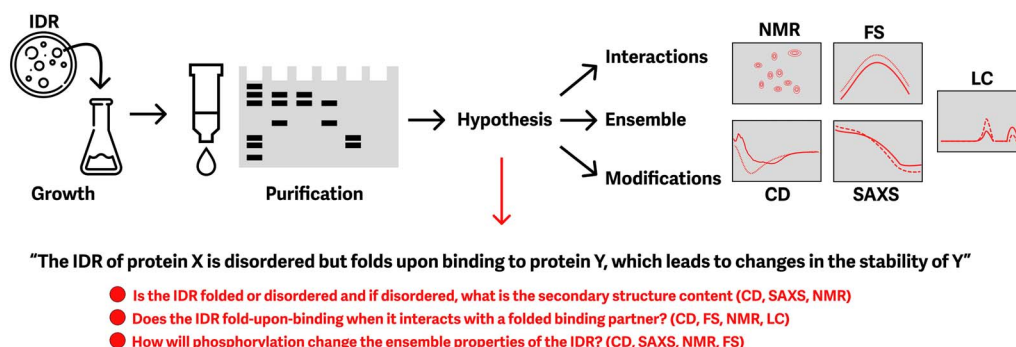


Fig 1. Workflow of the RIT course with use of, among other techniques, circular dichroism, fluorescence spectroscopy, and size-exclusion chromatography. (Top) Graphical illustration of the workflow of the research-integrating teaching course from selection of the intrinsically disordered region, purification and hypothesis formulation to data acquisition and analyses. (Bottom) The types of research questions typically asked are illustrated. NMR, nuclear magnetic resonance; SAXS, small-angle X-ray scattering.

field—the intrinsically disordered proteins—that encompasses dynamic and cutting-edge scientific advancements was chosen for this study (6, 7). Together, with a lack of detailed descriptions in most textbooks, the novelty spurs students' curiosity and empowers them to actively engage in investigation and problem-solving. This in turn encourages exploration of unanswered scientific questions.

In the RIT course, students express and purify an intrinsically disordered region (IDR) from a protein that was selected from a list of available proteins precloned into an *Escherichia coli* expression vector. The properties of IDRs differ from those of globular proteins, which are the primary focus in biochemistry and structural biology classes (8). Intrinsically disordered proteins and IDRs challenge the structure-function paradigm, which is still often introduced in textbooks (9).

The expression protocols, ensemble properties, and interactions of the IDRs selected for use in this course are not published. Therefore, students must design and execute expression and purification protocols for their selected IDR by using standard chromatographic systems. They then formulate a hypothesis to test. Typically, students investigate the ensemble properties of their IDR by using circular dichroism (CD) spectropolarimetry, nuclear magnetic resonance spectroscopy, small-angle x-ray scattering, and size-exclusion chromatography. They examine interactions with folded potential binding partners through the conformational

stability of the partner by using fluorescence spectroscopy (FS) or other techniques (Fig 1). In the course, reverse-phase high-pressure liquid chromatography, CD, and FS were central techniques. Thus, in a typical workflow, students perform various analyses, frequently using CD spectra and measurements of stability using CD and FS by increasing the temperature or adding a chemical denaturant.

Visualization and fitting data to known models are often complex and beyond the capabilities of freely available software or that accessible through a university. Additionally, alternative third-party software solutions are typically expensive and offer much more functionality than required for an RIT course. These programs can create barriers rather than supporting curiosity-driven solutions. As a result, students may ultimately use free plotting and fitting software or custom scripts, with which they rarely have experience. These solutions can be overly complicated and may never be used again, leading to time-consuming processes of learning a new software tool and diverting focus from the primary goal of the learning process. This lack of access to simple data digitalization has hindered the final phase of our RIT course, limiting research integration and learning outcomes.

To enhance the research outcomes for biochemistry students, we created an integrated Python-based Dashboard (PYDA) that seamlessly became a part of the course. PYDA allowed the students to visualize and fit their data to a set of

predefined models without any prior experience with Python. This freed time for learning and developing practical and hands-on skills and allowed students more time to focus on the science, which is central for the successful development of the students.

II. RESULTS

A. Development of PYDA

PYDA was developed in Python 3.9.6 with visualization and a dashboard build in Dash from Plotly (10), which allowed existing Python scripts to be integrated with little additional code. Dash is tightly integrated with Plotly's powerful visualization package, which allows the user to dynamically interact with the dashboard to zoom, scale, remove curves or points, and download figures as scalable vector graphics (SVG), all without having to write additional code. All background manipulation of data was achieved by using a combination of Pandas (11), NumPy (12), and SciPy (13). Finally, PYDA is hosted on the cloud platform Heroku (San Francisco, CA), which allows students to access PYDA through their browser of choice via a simple URL. This alleviated any requirements to install Python or other programs locally on the students' computers.

B. Implementation

Uploading of the exported files from the different research instruments can be achieved in PYDA via the web-based interface by using a web browser of choice. The process is divided into two steps. First, the type of instrument (CD, FS, or ÄKTA go) used is selected in a drop-down menu, and the "upload file(s)" button is clicked (Fig 2). Second, the model that the data should be fit to, if any, is selected. Here, the students gain access to features and models specific for the chosen type of data; for example, the ability to fit a protein-folding stability model would be available only for temperature or denaturant unfolding datasets. By presenting the students with only the options appropriate for their dataset, the time spent on data processing is reduced.

C. Examples of data visualization and fitting

1. Liquid chromatography (LC)

LC purification of proteins is conducted on ÄKTA go purification systems by using various columns and buffers. Although the ÄKTA go system provides reasonable visualization software, it can still be cumbersome to perform basic graphical manipulations, such as overlaying multiple chromatograms or filtering displayed elements. The LC plotter in PYDA offers a solution by allowing users to upload and visualize one or more chromatograms. This visualization includes key parameters such as absorption at 280 nm, the percentage of buffer B, conductivity (mS/cm), fraction numbers, and, if used, a secondary wavelength. Additionally, users can easily select which combinations of data to display (Fig 2). The integration of Plotly further enhances the user experience by enabling the ability to zoom and scale and the option to download results as SVG files. Overall, this streamlined approach simplifies the protein purification analysis for the students.

2. Far-ultraviolet (UV) CD data

Far-UV CD data are recorded on two different Jasco instruments (J810, J815). Either full CD spectra at a constant temperature or temperature unfolding followed at a single wavelength may be recorded. The instruments measure the ellipticity in millidegrees (other systems may report ΔA or $\Delta \epsilon$), which needs to be converted to mean residual ellipticity (MRE; Eq. 1) (14). This conversion was implemented by using two input boxes in which students insert the number of peptide bonds in their protein and the protein concentration in micromolar units. In this course, the light path length was fixed at 1 mm. If other path lengths are needed, they can be adjusted by using Python scripting. In PYDA, the CD data are then automatically converted to MRE by ticking the box "Convert to MRE" (Fig 3a). The second graph on the display is the dynode voltage of the photomultiplier, which gives information on saturation. If these data are not included in the uploaded file, PYDA

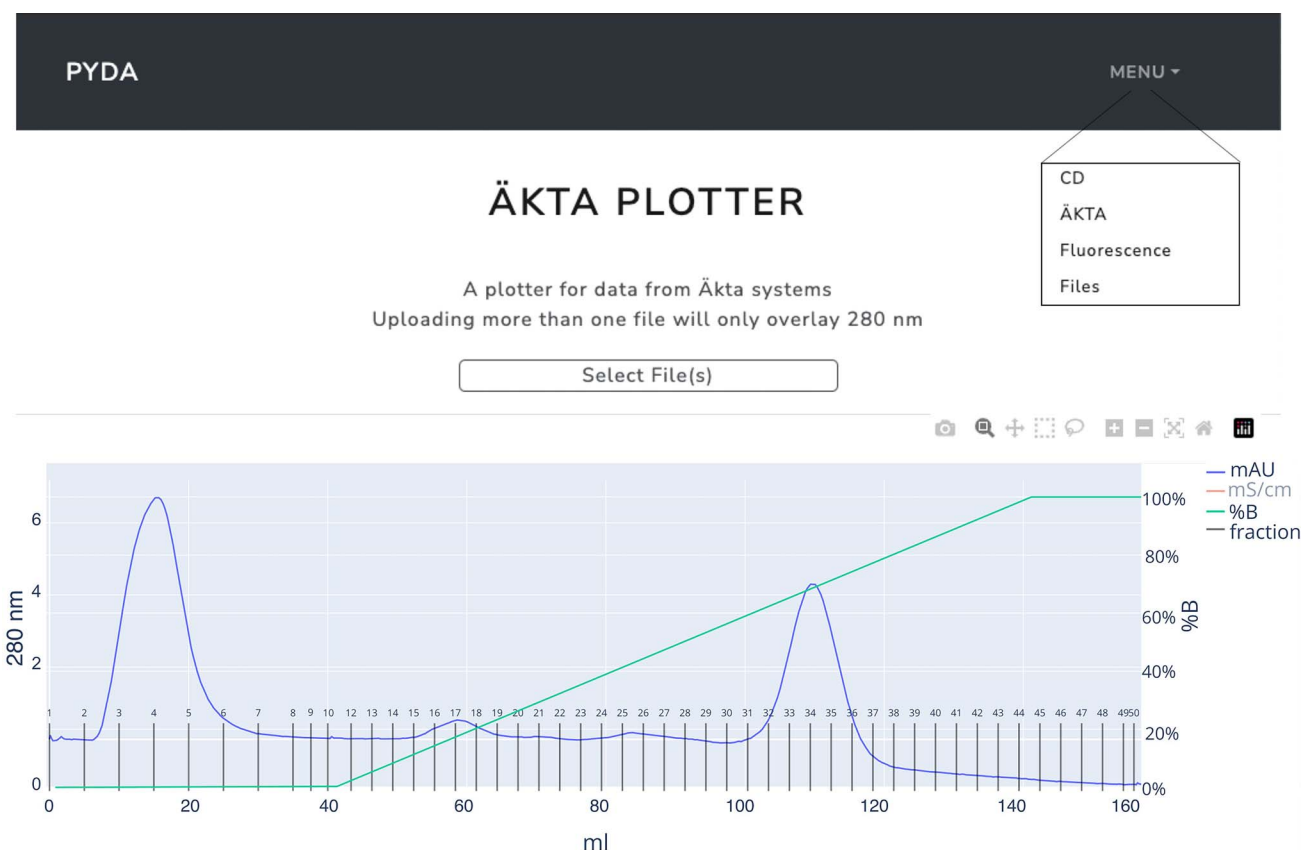


Fig 2. The ÅKTA plotting interface. Representation providing an overview of the ÅKTA plotting interface in a Python-based dashboard. Users can upload their data by clicking the “Select File(s)” button. Elution volumes are depicted with milliliters on the x axis and the absorbance at 280 nm on the y axis. The second y axis is dynamic and can display either the percentage of buffer B (%B) or absorbance at a second wavelength, depending on the data available. The legend located to the right of the chromatogram selectively displays data available in the dataset. The toolbar situated just above the legend offers a range of functions: (left to right) downloading the plot in .svg format, zooming in or out, panning, auto-scaling, and resetting the axis. The “Menu” option allows the user to choose which type of data they want to fit (circular dichroism, ÅKTA, or fluorescence spectroscopy).

will recognize this and display only the CD spectrum.

For far-UV CD data, PYDA allows users to upload multiple spectra at once and overlay them for easy visualization of spectral changes. These changes may be related to changes in protein secondary structure induced by an additive (Fig 3b). If only a single file is uploaded, PYDA checks whether the data contains a column with varying temperature. If so, PYDA changes to the temperature unfolding mode. In this mode, the change in ellipticity as a function of temperature can be fitted to Eq. 2 (Fig 3b). The initial guesses for the model must be provided by the student. These include guesses for the linear baseline effects of the changing temperature on the

native and denatured states, $A_n + B_n \cdot T$ and $A_d + B_d \cdot T$, respectively. Additionally, the changes in entropy, ΔS , and enthalpy, ΔH , for unfolding should be considered.

After fitting, the result is visualized as a dashed red line together with the experimental data as a blue line, and the parameters of the fit (including their standard deviations) are displayed in the legend (Fig 3b). The students can evaluate the fit quality both visually and by using the R^2 value and the standard deviations of the fitted parameters. On the basis of this evaluation, they can decide whether to adjust the initial guesses to obtain an improved fit.

3. Fluorescence spectroscopy

Fluorescence emission measurements are conducted on a Perkin Elmer LS50B instrument.

PYDA

MENU ▾

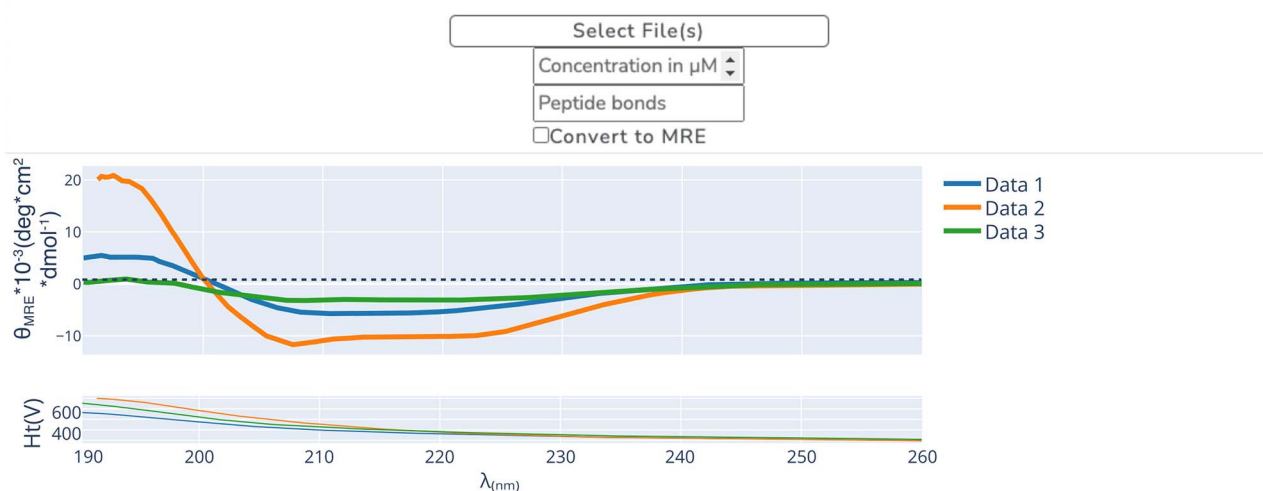
a

CD PLOTTER

A plotter for data from JASCO CD systems

Uploading more than one file will overlay them

Uploading file containing temperature data will allow for fitting of denaturation



b

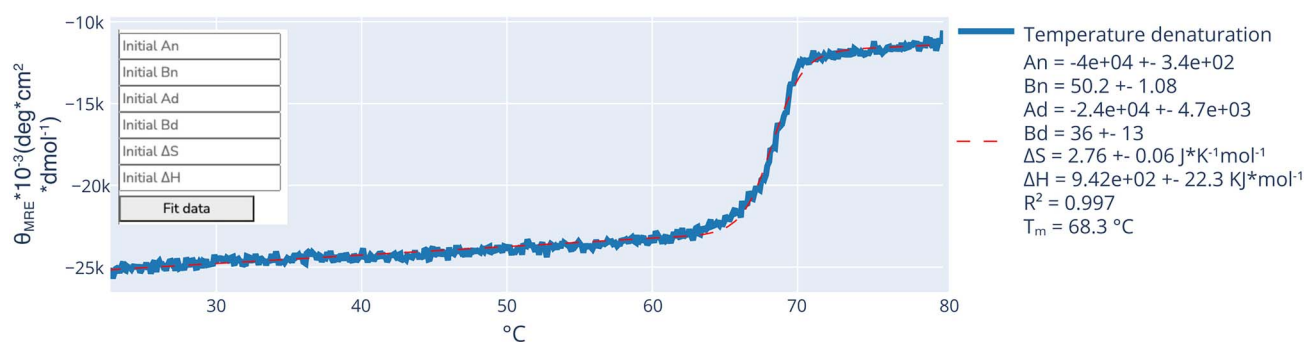


Fig 3. Illustration of the far-UV CD plotting interface in a PYDA. (a) The user can upload one or more files through the “Select file(s)” button. Uploading one or more files, without a temperature component in the dataset, will overlay the different spectra, displaying the wavelength, λ , in nanometers on the x axis and ellipticity in millidegrees on the y axis. The ellipticity can be converted by PYDA to mean residual ellipticity through inputting the protein concentration in micromolar units and the amount of peptide bonds in the appropriate input boxes. Photomultiplier voltage (reporting on saturation) displayed in the separate graph below the CD spectra is critical for assessing data quality (not shown). The displayed spectra can be chosen by toggling through options presented on the right of the interface. (b) Temperature unfolding by far-UV CD displayed with temperature in Celsius along the x axis and the measured fixed wavelength on the y axis. Input for the initial guesses for fitting the temperature unfolding model (Eq. 2) is displayed to the left. Clicking “Fit data” will optimize to find the best fit; the fit will be displayed in a dashed orange line on top of the data. Parameters of the fit are display in the legend.

Often a set of emission spectra will be recorded as a function of an additive (e.g., denaturant, ligand, lipids, or reducing agent) depending on the type of question asked. If a single file is uploaded to PYDA, it is visualized with the measured emission wavelength (x axis) and the corresponding intensities in arbitrary units (y axis; Fig 4a). If more than one file is uploaded, PYDA will overlay the spectra, and a button appears that allows the student to assign a concentration (or other feature) to each uploaded file.

Furthermore, it is possible in PYDA to choose additional ways of visualizing the data. Option 1 plots the emission intensity at a specific wavelength for all spectra uploaded (Fig 4b). The wavelength to visualize can be chosen by the student using a slider, making it easy to assess which wavelength to follow to maximize the explored effect. Option 2 plots λ_{max} , the wavelength at which the individual spectra exhibit maximum emission intensity (Fig 4c). The λ_{max} is the population average only if $I(\lambda_{\text{max}})$ is the same for the titrating species. This is valid only in special cases, which must be kept in mind. Different datasets will yield a better result by using option 1 over option 2 and vice versa; this accessible way of visually exploring both options has helped students understand their data more in depth and has enabled them to choose the better option.

Additionally, this visual approach can empower students to uncover unexpected patterns within their data. In the presented case, a three-state unfolding becomes apparent only when plotting the intensity at a specific wavelength (Fig 4c). This feature could easily have been missed had the students plotted only the intensities at λ_{max} (Fig 4b). Finally, in PYDA, the student has the option to fit the data to an appropriate model (ligand binding or protein unfolding). This is completed by selecting either Eq. 3 (unfolding) or Eq. 4 (binding) by clicking the “fitting” button and entering initial guesses. The fit will be visualized as a dotted line together with the data and the parameters of the fit. The standard deviation of all parameters will be displayed in the legend, as

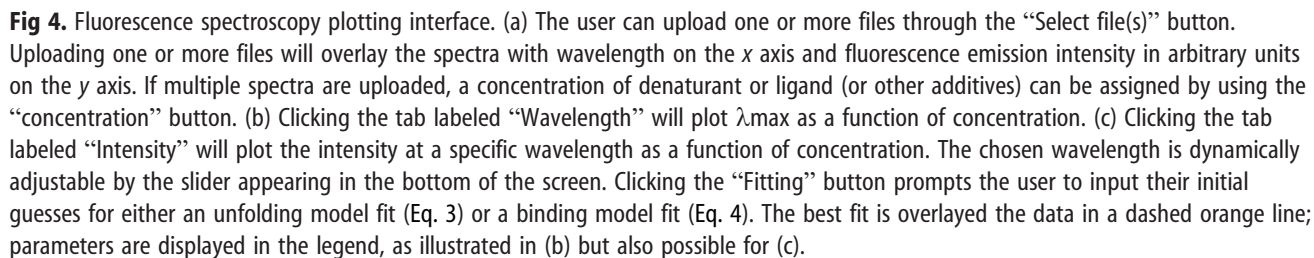
described for far-UV CD. Options to adjust the initial guesses and reperforming the fit are similarly available.

D. Student feedback

At the end of the course, students anonymously responded to a survey on the usage, strengths, and limitation of PYDA (Fig 5). The results showed a general satisfaction with the ease of use and the effectiveness in time reduction provided by PYDA. Despite the lower hands-on time with model fitting, students generally reported a good to thorough understanding of the used models. Less than 10% of the students found PYDA difficult and ineffective, although this seemed to have no apparent bearing on their self-reported understanding of the model. Only 50% of the students reported being competent in other software, which could have substituted PYDA. This highlights the considerable need for an easy-to-use and effective solution because more complicated alternatives are always available. PYDA provides the easy solution.

E. Implementation of PYDA into other RIT courses

PYDA, with the features described here, is freely available for use at the Structural Biology and Nuclear Magnetic Resonance Laboratory (University of Copenhagen, Denmark, <https://www1.bio.ku.dk/english/research/bms/sbinlab>) and can be accessed through any browser. The integration of a dashboard into other RIT-based courses is directly possible. With the functionalities described here, PYDA (code available at <https://github.com/Hebbelstrup/PYDA>) can also easily be modified to include additional or other types of data or other models for fitting than described here. The latter necessitates an initial time investment in developing the dashboard. Nevertheless, when compared to alternative solutions, the skill set required for creating a dashboard in Dash is relatively modest and can, in RIT courses with a different focus, even be a student task itself. This would require at least an intermediate level of proficiency in Python and experience in writing



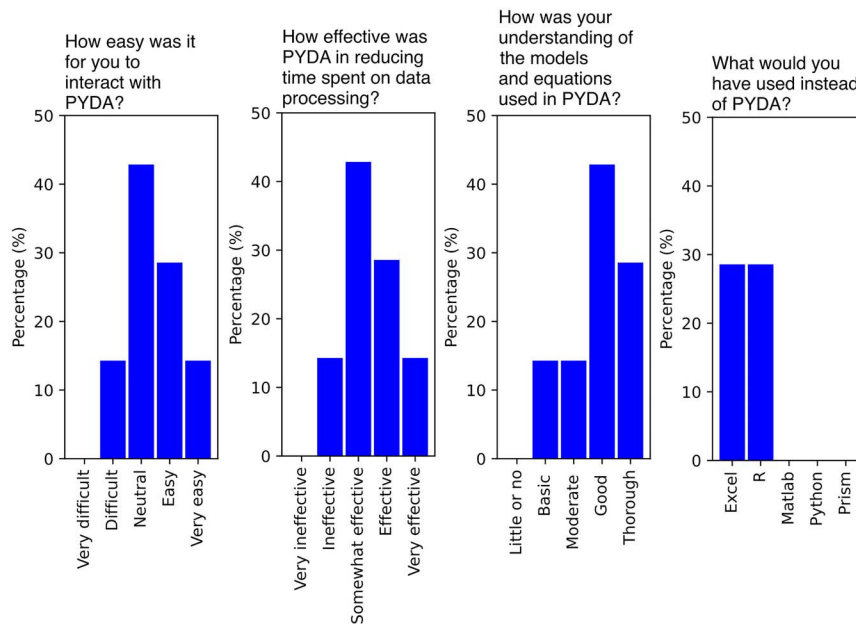


Fig 5. Survey on the PYDA, providing student feedback on PYDA use. Responses are reported as a percentage of total answers. In total, seven groups of three students answered the survey.

Python scripts. Newly developed scripts can then be integrated into the dashboard with a modest amount of additional code and tailored to the precise requirements of individual RIT courses.

III. DISCUSSION

The integration of a Python-based dashboard into an RIT-based laboratory course was successful. According to student evaluations, the incorporation of a web-based graphic interface made it easy for them to navigate and use the tool. PYDA was a significant improvement over previous years, when students who were unfamiliar with Python were supplied with Python scripts that required Python and relevant packages to be installed. PYDA significantly reduced the amount of time each student spent learning new software. This freed time and mental resources for students and mentors to focus on laboratory work and hypothesis development.

As an additional positive outcome, the easy visualization and modulation of the data provided the students with insight into the robustness of their data. In some instances, this insight made the students repeat their experiment because they were inspired to do better.

A potential downside of the easy access approach is that students do not develop a deeper understanding of the underlying data

analysis and models; however, according to the student survey, this seemed not to be a dominant effect. All students reported that they obtained a good to thorough understanding of the models, despite the obstacles felt by a few students in using PYDA. The RIT course is not evaluated quantitatively through an exam; therefore, insight into model understanding was not quantitatively assessed. At the final poster session, though, questions directly testing such insight were asked by the poster committee. Answers from the students generally supported the insight that they claimed in their self-reporting evaluation. To alleviate potential uncertainty in learning outcomes, a gated dashboard—in which students can access certain elements only when they have demonstrated an understanding of the used functions or the processes underlying them—can be developed. In this case, students were adequately involved in their projects so that they, by themselves, acquired the theoretical knowledge needed.

In an increasingly digital world where programming and scripting are becoming essential for learning, a need for activities that lower the barriers to data digitalization for untrained students is paramount. These students are typically following curricula in the biological sciences. With PYDA, the students gained access to fitting and data visualization of complex

formats with few data repetitions and more than usual experimental noise. This has transformed students' resistance toward data digitalization and complex data fitting and has empowered them to explore their data more thoroughly. This easy accessibility has directly lowered the barrier for digital learning.

IV. EQUATIONS

Eq. 1 shows

$$MRE = \frac{100 * \theta}{L * c * N} \quad (1)$$

where θ is the ellipticity, c is the molar concentration of the protein(s), N is the number of peptide bonds, and L is the light pathlength (in centimeters) (14).

Eq. 2 is as follows:

$$y = \frac{(\alpha_N + \beta_N * T) + (\alpha_D + \beta_D * T) * \exp \frac{\Delta S_{D-N(T_M)} * T - H_{D-N(T_M)}}{RT}}{1 + \exp \frac{\Delta S_{D-N(T_M)} * T - H_{D-N(T_M)}}{RT}} \quad (2)$$

where $(\alpha_N + \beta_N * T)$ is the influence on the native state (N), $(\alpha_D + \beta_D * T)$ is the influence on the denatured state (D), T is the temperature in Kelvin, T_M is the melting temperature in Kelvin, $\Delta S_{D-N(T_M)}$ is the change in entropy at T_M , $H_{D-N(T_M)}$ is the change in enthalpy at T_M , and R is the gas constant (15).

Eq. 3 shows

$$f_{obs} = \frac{(f_N + a_1[D]) + (f_D + a_2[D]) * \exp \frac{m * ([D] - [D]_{50\%})}{RT}}{1 + \exp \frac{m * ([D] - [D]_{50\%})}{RT}} \quad (3)$$

where $[D]$ is the denaturant concentration, $(f_N + a_1[D])$ is the influence of the denaturant on the native state, $(f_D + a_2[D])$ is the influence of the denaturant on the denatured state, $[D]_{50\%}$ is the concentration of denaturant at which N and D exist in equal concentrations and $\Delta G(D) = 0$, m is the slope in the equilibrium point, T is the

temperature in Kelvin, and R is the gas constant (15).

Finally, Eq. 4 shows

$$y = y_0 + \frac{y_f - y_0}{2 * [P_{total}]} * ([P_{total}] + x + K_d - \sqrt{([P_{total}] + x + K_d)^2 - 4 * [P_{total}] * x}) \quad (4)$$

where y_0 is the fluorescence before titration, y_f is estimated fluorescence at the end of titration, $[P_{total}]$ is the total protein concentration, x is the concentration of titrated ligand, and K_d is the dissociation constant.

AUTHOR CONTRIBUTIONS

All authors contributed to designing PYDA. ASH wrote the code and performed the student survey, BBK supervised the implementation of PYDA in the teaching, ASH wrote the initial draft of the manuscript, and BBK and KT edited and wrote the final version of the manuscript.

ACKNOWLEDGMENTS

The authors thank all the students of the Protein Research Lab and Protein Science A courses in the years 2021–2023 for inspiring us to develop PYDA. We are grateful to Nina L. Jacobsen and Lasse Staby for allowing us to use their published and unpublished data for illustrative purposes (16). The development of PYDA was supported by REPIN funded by the Novo Nordisk Foundation (NNF18OC0033926 to BBK) and Independent Research Foundation Denmark (0136-00164B to KT). The authors declare no competing interests.

REFERENCES

- Deslauriers, L., L. S. McCarty, K. Miller, K. Callaghan, and G. Kestin. 2019. Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom. *Proc Natl Acad Sci U S A* 116:19251–19257. <https://doi.org/10.1073/pnas.1821936116>.
- Waldrop, M. M. 2015. Why we are teaching science wrong, and how to make it right. *Nature* 523:272–274. <https://doi.org/10.1038/523272a>.
- Skriver, K., S. A. Sjørup, A. E. Langkilde, E. Balouka, C. S. Christensen, K. Carbel, J. N. V. Decker, D. N. Essenbæk, J. F. Gräf, C. H. Jessen, P. Kristensen, C. Merrild, T. S. Mortensen, I. F. Nalepa, B. W. Nordsteen, S. K. Svoren, M. van Hall, J. Weicher, M. L. Wind, D. Zhang, D. Saar, H. Blæsild, M. Stahlhut, K. V. Andersen, R. Dagil, B. Vestergaard, M. L. Ryberg, and B. B. Kragelund. 2023. Intrinsically disordered proteins as an instrument for research-integrating teaching. *Biophysicist* 4:82–88. <https://doi.org/10.35459/tbp.2022.000221>.
- Chang, H. 2005. Turning an undergraduate class into a professional research community. *Teach High Educ* 10:387–394. <https://doi.org/10.1080/13562510500122339>.

5. Jenkins, A., R. Breen, R. Lindsay, and A. Brew. 2002. Reshaping Teaching in Higher Education, A Guide to Linking Teaching with Research. Routledge, London. <https://doi.org/10.4324/9780203416884>.
6. Holehouse, A. S., and B. B. Kragelund. 2024. The molecular basis for cellular function of intrinsically disordered protein regions. *Nat Rev Mol Cell Biol* 25:187–211. <https://doi.org/10.1038/s41580-023-00673-0>.
7. Pedersen, C. P., P. Seiffert, I. Brakti, and K. Bugge. 2020. Production of intrinsically disordered proteins for biophysical studies: tips and tricks. *Methods Mol Biol* 2141:195–209. https://doi.org/10.1007/978-1-0716-0524-0_9
8. Dunker, A. K., J. D. Lawson, C. J. Brown, R. M. Williams, P. Romero, J. S. Oh, C. J. Oldfield, A. M. Campen, C. M. Ratliff, K. W. Hipps, J. Ausio, M. S. Nissen, R. Reeves, C.-H. Kang, C. R. Kissinger, R. W. Bailey, M. D. Griswold, W. Chiu, E. C. Garner, and Z. Obradovic. 2001. Intrinsically disordered protein. *J Mol Graph Model* 19(1):26–59. [https://doi.org/10.1016/s1093-3263\(00\)00138-8](https://doi.org/10.1016/s1093-3263(00)00138-8).
9. Buda, K., K. Cermakova, H. C. Hodges, E. F. Fornasiero, S. Sukenik, and A. S. Holehouse. 2023. Using graphs and charts in scientific figures. *Trends Biochem Sci*, 48:913–916. <https://doi.org/10.1016/j.tibs.2023.08.011>.
10. Hossain, S. 2019. Visualization of bioinformatics data with dash bio. In *Visualization of Bioinformatics Data with Dash Bio*. C. Calloway, D. Lippa, D. Niederhut, and D. Shupe, editors, pp. 126–133. <https://doi.org/10.25080/Majora-7ddc1dd1-012>.
11. Pandas Development Team. 2024. Zenodo: pandas-dev/pandas. <https://doi.org/10.5281/zenodo.10957263>.
12. Harris, C. R., K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. Fernández del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. 2020. Array programming with NumPy. *Nature* 585:357–362. <https://doi.org/10.1038/s41586-020-2649-2>.
13. Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A. P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C. N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D. A. Nicholson, D. R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G. A. Price, G.-L. Ingold, G. E. Allen, G. R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J. T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J. Vinícius de Miranda Cardoso, J. Reimer, J. Harrington, J. L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N. J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P. A. Brodtkorb, P. Lee, R. T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T. J. Pingel, T. P. Robitaille, T. Spura, T. R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y. O. Halchenko, Y. Vázquez-Baeza, and SciPy 1.0 Contributors. 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat Methods* 17:261–272. <https://doi.org/10.1038/s41592-019-0686-2>.
14. Woody, R. W. 1996. Theory of circular dichroism of proteins. In *Circular dichroism and the Conformational Analysis of Biomolecules*. G. D. Fasman, editor. Springer, New York, NY, USA, pp. 25–67.
15. Lindorff-Larsen, K., and K. Teilum. 2021. Linking thermodynamics and measurements of protein stability. *Protein Eng Des Sel* 34: gzab002. <https://doi.org/10.1093/protein/gzab002>.
16. Staby, L., K. R. Kemplen, A. Stein, M. Ploug, J. Clarke, K. Skriver, P. O. Heidarsson, and B. B. Kragelund. 2021. Disorder in a two-domain neuronal Ca²⁺-binding protein regulates domain stability and dynamics using ligand mimicry. *Cell Mol Life Sci* 78:2263–2278. <https://doi.org/10.1007/s00018-020-03639-z>.