Modelling biology in novel ways - an AI-first course in Structural Bioinformatics

Kieran Didi
University of Cambridge/Heidelberg
ked48@cam.ac.uk

Charles Harris
University of Cambridge
cch57@cam.ac.uk

Pietro Lio
University of Cambridge
pl219@cam.ac.uk

Rainer Beck
University of Heidelberg
rainer.beck@bzh.uni-heidelberg.de

Abstract

In recent years, there has been tremendous progress in applying data-driven methodologies to study biological questions. The rapidly evolving field of machine learning has gained a plethora of methods that can be applied to structural biology like protein structure prediction. However, the intricacies one faces when analyzing complex biological data are sometimes underappreciated in applications of machine learning methods. On the other hand, biologists often face a language- and method barrier when trying to understand and correctly apply machine learning tools. As a result, they might be using such methods without proper expertise, potentially resulting in incorrect predictions and questionable conclusions about the resulting data. To help remedy these issues, we have developed a holistic 11-unit course in AI-driven Structural Bioinformatics with the aim of (i) encouraging machine learning researchers to learn more about the biological complexity of the data they are analyzing and (ii) allowing biologists to better understand state-of-the-art machine learning algorithms for correct application to biological systems. The course includes video lectures, animated visualisations as well as in-depth exercises and further resources for each of the topics discussed. We hope that our course stimulates collaboration across research communities and lowers the entry barrier for newcomers to understand and investigate structural biology with data-driven tools. Our course is available at https://structural-bioinformatics.netlify.app.

1 Introduction

The application of machine learning (ML) to structural biology has causesd significant advancements in biological problem-solving, particularly in protein structure prediction [1] and *de novo* protein design [2, 3]. However, applying ML tools to biology is complicated, often neglecting the specific challenges and subtleties inherent to modelling biological data and leading to issues in real-world application scenarios [4, 5, 6, 7]. The barrier to effective understanding and application of machine learning in biology is intensified by a distinct disconnect between the two fields. On one hand, biologists grapple with the technical language and methodologies inherent to ML tools. On the other, ML researchers frequently lack the vital biological context that should inform their research. This divergence in expertise can culminate in well-intended but misdirected solutions.

Although bioinformatics courses are available [8], a gap remains in integrating the swiftly evolving domain of ML applied to bioinformatics, specifically in the feild of structural bioinformatics. To bridge this gap, we have developed an 11-unit AI-driven Structural Bioinformatics course. This course

NeurIPS 2023 AI for Science Workshop.

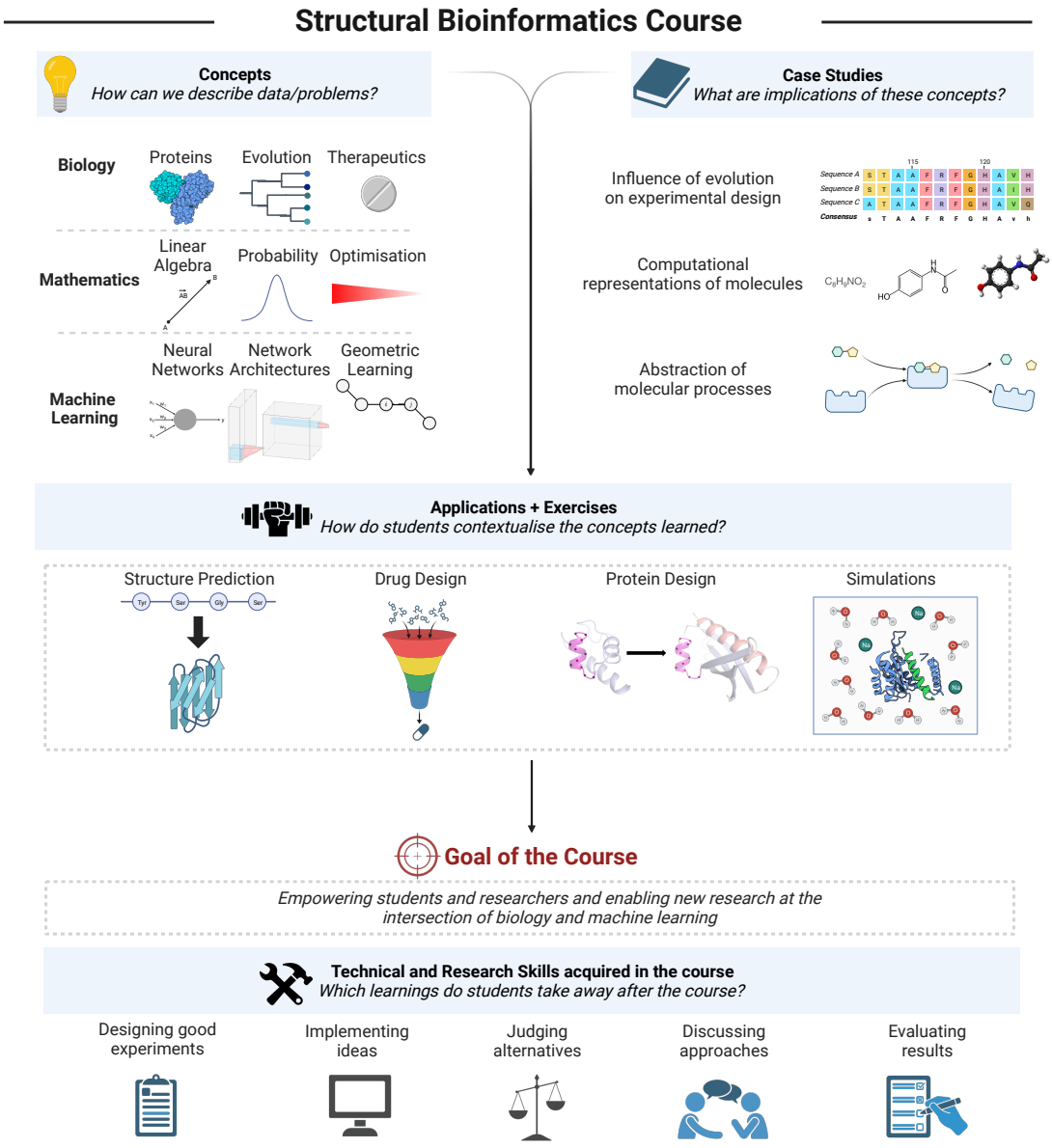

Figure 1: Course overview. Students get exposed to conceptual ideas in three different disciplines and learn about their interactions and implications via case studies. They use the taught concepts in exercises on practical applications and by this gain technical as well as research skills to investigate the intersection of biology and machine learning.

aims to equip ML researchers with biological understanding and enable biologists to understand and apply the latest ML research effectively, fostering interdisciplinary understanding and collaboration.

2 Background

The exact definition and curriculum of bioinformatics have been subjects of long-standing discussions [9, 10], which have gained renewed significance due to the advancements AI has brought to the field, enabling exploration of biological datasets in novel ways.

Bioinformatics, a multidisciplinary field, intertwines biology, mathematics, and computer science to interpret biological data [11]. Given its rapid evolution [12], creating up-to-date curricula is essential to equip students with both theoretical knowledge and practical skills, addressing the significant skill gap in life sciences [13, 14]. The integration of programming competencies like Python or Bash scripting is fundamental for applying theoretical knowledge in practical settings [15].

Curriculum Design and Interdisciplinary Approach Curricula need to facilitate knowledge consolidation through projects and case studies, enabling contextual application of acquired knowledge [16]. Considering the interconnectedness of courses within broader student curricula [17], this course adopts a model in which the core curriculum explores the interface of biology and machine learning in depth and primers are provided to accommodate diverse backgrounds and enables learners to fill in any existing knowledge gaps. Different courses set their focus differently (App. B); our course aims to integrate knowledge from machine learning and biology instead of teaching the two in isolation.

An interdisciplinary approach, involving educators from life sciences and computer science, is pivotal for a cohesive learning environment [18], preparing students for the future interdisciplinary demands and the integration of AI technologies in advancing structural bioinformatics.

3 Course Contents

The course aims to empower both students and researchers to dive into AI applications in structural biology and is structured into three main parts to achieve this goal (Fig. 1). Areas within structural bioinformatics such as protein structure prediction, evolutionary modelling, molecular dynamics simulations, early stage drug design and *de novo* protein design are introduced. In each area, both the traditional methods (e.g. docking software) and the latest machine learning methods (e.g. deep learning-based docking) are introduced, allowing for the comparison of similarities and differences as well as exploration of intricacies with are handled poorly by current ML methods [4, 6]

To provide the appropriate background knowledge to understand current applications, concepts from three areas are taught: biology, mathematics and machine learning. To offer students a curriculum specific to their needs, many of these fundamental concepts (e.g. linear algebra or probability) are presented as primers, meaning they are taught in a self-contained lesson at a suitable point in the curriculum and can be shortened for students who already possess the required knowledge. A detailed overview of the course content and concepts can be found in the Appendix in Fig. 3.

The concepts from these three domains often seem disconnected at first. That is why we integrate case studies into the curriculum; in our course, the students can see the connections between seemingly disparate concepts and how one concept can be applied to solve problems related to a different one. Examples include the choice of computational representation of molecules that is heavily influenced by the task at hand [19, 20] and also the influence of evolution on experimental design and methodologies such as the difficult of crafting robust train-test splits of biological data [4, 21]

Mixing passive with active learning elements promises to engage students more in the learning process and improve learning outcomes [22]. Therefore, we augment our concept lectures and case studies with exercises. In these hands-on problems, students learn to apply the learned concepts to an interesting problem of practical relevance such as drug design or molecular simulation and directly see the usefulness of concepts learned in the lectures.

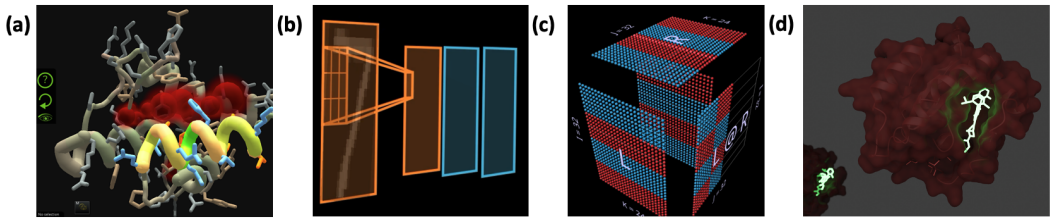

Figure 2: The curriculum includes both interactive exercises like (a) the FoldIt protein design game as well as interactive animations for concepts like (b) convolutions or (c) matrix multiplication. Students will also learn how to present their results with tools like (d) PyMol and Molecular Nodes [29].

4 Teaching Concept and Techniques

Curating and Developing Teaching Materials In alignment with guidelines for curriculum development in bioinformatics, the course strategically leverages existing teaching materials, curating and adapting them to serve the comprehensive educational objectives [23]. This didactically appropriate approach ensures the representation of varied contents and the development of new materials for underrepresented topics. Such a combination of curated and freshly developed resources, which are made publicly available, supports other educators in enhancing their instructional materials [23].

Specifically, the course incorporates adapted notebooks and packages from the TeachOpenCADD platform [24] and visualization tutorials [25], ensuring a rich and diversified learning experience.

Role of Conceptual Diagrams and Visualizations Conceptual diagrams are crucial to teaching complex concepts, but their creation poses considerable challenges and time investments [26]. The course addresses this by integrating a blend of distinguished visualizations available and custom-made diagrams and examples, aimed at providing clarity and enhancing comprehension (Fig. 2).

To visually illustrate the protein design process, the course employs PyMol [27], allowing students to interactively explore the intricacies of protein structures. This interaction is further enriched by additional tutorials based on the FoldIt tool [28], which gamifies the learning experience, fostering intuitive understanding and engagement.

Visualization of Machine Learning Concepts The course incorporates ManimML [30] to animate machine learning concepts such as convolutions or variational autoencoders, offering students intuitive insights into these advanced topics. This tool, in combination with the mm tool [31], facilitates the visualization of fundamental concepts like matrix multiplications, enabling students to grasp concepts related to efficient training and model interpretability effectively.

Additionally, Penrose [32] is deployed to generate clear diagrams for mathematical concepts like conditional independence and Bayes rule, ensuring that students gain a clear understanding of the mathematical underpinnings essential for bioinformatics.

5 Conclusion

In this paper, we detail an 11-unit course focused on AI-driven Structural Bioinformatics, constructed with the aim of addressing the notable divide between biology and machine learning disciplines. The curriculum is carefully developed, considering the complex nature of biological data and the intricacies of modern machine learning methods.

In summary, this course endeavours to provide a balanced perspective on the intersection of biology and machine learning, promoting interdisciplinary understanding and collaboration. The availability of course materials serves as a resource for those seeking to explore the possibilities and challenges in the evolving landscape of structural bioinformatics.

References

[1] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[2] Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, pages 1–3, 2023.

[3] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[4] Martin Buttenschoen, Garrett M Morris, and Charlotte M Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *arXiv preprint arXiv:2308.05777*, 2023.

[5] Yuejiang Yu, Shuqi Lu, Zhifeng Gao, Hang Zheng, and Guolin Ke. Do deep learning models really outperform traditional approaches in molecular docking? *arXiv preprint arXiv:2302.07134*, 2023.

[6] Charles Harris, Kieran Didi, Arian R Jamasb, Chaitanya K Joshi, Simon V Mathis, Pietro Lio, and Tom Blundell. Benchmarking generated poses: How rational is structure-based drug design with generative models? *arXiv preprint arXiv:2308.07413*, 2023.

[7] Gengmo Zhou, Zhifeng Gao, Zhewei Wei, Hang Zheng, and Guolin Ke. Do deep learning methods really perform better in molecular conformation generation? *arXiv preprint arXiv:2302.07061*, 2023.

[8] Lei Gao and Miao Guo. A course-based undergraduate research experience for bioinformatics education in undergraduate students. *Biochemistry and Molecular Biology Education*, 51(2): 189–199, 2023.

[9] N. M. Luscombe, D. Greenbaum, and M. Gerstein. What is bioinformatics? a proposed definition and overview of the field. 40(4):346–358. ISSN 0026-1270, 2511-705X. doi: 10.1055/s-0038-1634431. URL `http://www.thieme-connect.de/DOI/DOI?10.1055/s-0038-1634431`. Publisher: Schattauer GmbH.

[10] R B Altman. A curriculum for bioinformatics: the time is ripe. 14(7):549–550. ISSN 1367-4803. doi: 10.1093/bioinformatics/14.7.549. URL `https://doi.org/10.1093/bioinformatics/14.7.549`.

[11] Lonnie Welch, Fran Lewitter, Russell Schwartz, Cath Brooksbank, Predrag Radivojac, Bruno Gaeta, and Maria Victoria Schneider. Bioinformatics curriculum guidelines: Toward a definition of core competencies. 10(3):e1003496. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003496. URL `https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003496`. Publisher: Public Library of Science.

[12] Qanita Bani Baker and Maryam S Nuser. Design bioinformatics curriculum guidelines: Perspectives. *Your Passport to a Career in Bioinformatics*, pages 91–102, 2021.

[13] Smriti Sharma and Vinayak Bhatia. An appraisal of skill gaps in bioinformatics education. *Current Bioinformatics*, 16(9):1117–1125, 2021.

[14] Sandra G Porter and Todd M Smith. Bioinformatics for the masses: The need for practical data science in undergraduate biology. *OMICS: A Journal of Integrative Biology*, 23(6):297–299, 2019.

[15] Nicola Mulder, Russell Schwartz, Michelle D Brazas, Cath Brooksbank, Bruno Gaeta, Sarah L Morgan, Mark A Pauley, Anne Rosenwald, Gabriella Rustici, Michael Sierk, et al. The development and application of bioinformatics core competencies to improve bioinformatics training and education. *PLoS computational biology*, 14(2):e1005772, 2018.

[16] Michael J Wolyniak. Bioinformatics education for undergraduates: the need for project-based and experiential approaches. *International Journal of Smart Technology and Learning*, 3(2): 107–117, 2023.

[17] Derek Gatherer. Reflections on integrating bioinformatics into the undergraduate curriculum: The lancaster experience. *Biochemistry and Molecular Biology Education*, 48(2):118–127, 2020.

[18] Mohd Shahir Shamsir and Zeti Azura Mohamed Hussein. Across and beyond the divide: the role of inter-departmental teaching in bioinformatics. *Asean Journal Teaching and Learning in Higher Education*, 2(1):30–40, 2010.

[19] Daniel S Wigh, Jonathan M Goodman, and Alexei A Lapkin. A review of molecular representation in the age of machine learning. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1603, 2022.

[20] Arian Jamasb, Ramon Viñas Torné, Eric Ma, Yuanqi Du, Charles Harris, Kexin Huang, Dominic Hall, Pietro Lió, and Tom Blundell. Graphein-a python library for geometric deep learning and network analysis on biomolecular structures and interaction networks. *Advances in Neural Information Processing Systems*, 35:27153–27167, 2022.

[21] Jack Scantlebury, Lucy Vost, Anna Carbery, Thomas E Hadfield, Oliver M Turnbull, Nathan Brown, Vijil Chenthamarakshan, Payel Das, Harold Grosjean, Frank von Delft, et al. A small step toward generalizability: Training a machine learning scoring function for structure-based virtual screening. *Journal of Chemical Information and Modeling*, 2023.

[22] Paras Singh Minhas, Arundhati Ghosh, and Leah Swanzy. The effects of passive and active learning on student preference and performance in an undergraduate basic science course. *Anatomical sciences education*, 5(4):200–207, 2012.

[23] Susan McClatchy, Kristin M Bass, Daniel M Gatti, Adam Moylan, and Gary Churchill. Nine quick tips for efficient bioinformatics curriculum development and training. *PLoS Computational Biology*, 16(7):e1008007, 2020.

[24] Dominique Sydow, Andrea Morger, Maximilian Driller, and Andrea Volkamer. Teachopencadd: a teaching platform for computer-aided drug design using open source packages and data. *Journal of cheminformatics*, 11(1):1–7, 2019.

[25] Magnus Kjaergaard, Laura Skak Rasmussen, Johan Nygaard Vinther, Kasper Røjkjær Andersen, Ebbe Sloth Andersen, Esben Lorentzen, Søren S Thirup, Daniel E Otzen, and Ditlev Egeskov Brodersen. A semester-long learning path teaching computational skills via molecular graphics in pymol. *The Biophysicist*, 3(2):106–114, 2022.

[26] Dor Ma'ayan, Wode Ni, Katherine Ye, Chinmay Kulkarni, and Joshua Sunshine. How domain experts create conceptual diagrams and implications for tool design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14. ACM. ISBN 978-1-4503-6708-0. doi: 10.1145/3313831.3376253. URL `https://dl.acm.org/doi/10.1145/3313831.3376253`.

[27] Warren L DeLano and Sarina Bromberg. Pymol user's guide. *DeLano Scientific LLC*, 629, 2004.

[28] Robert Kleffner, Jeff Flatten, Andrew Leaver-Fay, David Baker, Justin B Siegel, Firas Khatib, and Seth Cooper. Foldit standalone: a video game-derived protein structure manipulation interface using rosetta. *Bioinformatics*, 33(17):2765–2767, 2017.

[29] Brady Johnston, Yuxuan Zhuang, Yinying Yao, William McCorkindale, Johannes Elferich, Patrick Kunzmann, Rich, Olivier Laprevote, Thibault Tubiana, Domenico Marson, James Hooker, Jessica A. Nash, and Joyce Kim. BradyAJohnston/MolecularNodes: v2.10.0 for Blender 3.5+, September 2023. URL `https://doi.org/10.5281/zenodo.8374654`.

[30] Alec Helbling and Duen Horng Chau. ManimML: Communicating machine learning architectures with animation. URL `http://arxiv.org/abs/2306.17108`.

[31] Basil Hosmer. mm - 3d matmul visualizer, 2023. URL https://bhosmer.github.io/mm/.

[32] Katherine Ye, Wode Ni, Max Krieger, Dor Ma'ayan, Jenna Wise, Jonathan Aldrich, Joshua Sunshine, and Keenan Crane. Penrose: from mathematical notation to beautiful diagrams. 39 (4). ISSN 0730-0301, 1557-7368. doi: 10.1145/3386569.3392375. URL https://dl.acm.org/doi/10.1145/3386569.3392375.

[33] Ashley Vater, Jaime Mayoral, Janelle Nunez-Castilla, Jason W Labonte, Laura A Briggs, Jeffrey J Gray, Irina Makarevitch, Sharif M Rumjahn, and Justin B Siegel. Development of a broadly accessible, computationally guided biochemistry course-based undergraduate research experience. *Journal of Chemical Education*, 98(2):400–409, 2020.

[34] Erin C Yang, Robby Divine, Christine S Kang, Sidney Chan, Elijah Arenas, Zoe Subol, Peter Tinker, Hayden Manninen, Alicia Feichtenbiner, Talal Mustafa, et al. Increasing computational protein design literacy through cohort-based learning for undergraduate students. *Journal of Chemical Education*, 99(9):3177–3186, 2022.

[35] Kathy H Le, Jared Adolf-Bryfogle, Jason C Klima, Sergey Lyskov, Jason W Labonte, Steven Bertolani, Shourya S Roy Burman, Andrew Leaver-Fay, Brian D Weitzner, Jack Maguire, et al. Pyrosetta jupyter notebooks teach biomolecular structure prediction and design. *The Biophysicist*, 2(1):108–122, 2021.

[36] Nuria B Centeno, Jordi Villà-Freixa, and Baldomero Oliva. Teaching structural bioinformatics at the undergraduate level. *Biochemistry and molecular biology education*, 31(6):386–391, 2003.

[37] Felipe Engelberger, Pablo Galaz-Davison, Graciela Bravo, Maira Rivera, and César A Ramírez-Sarmiento. Developing and implementing cloud-based tutorials that combine bioinformatics software, interactive coding, and visualization exercises for distance learning on structural bioinformatics, 2021.

[38] Janosch Menke, Samuel Homberg, and Oliver Koch. Introduction to artificial intelligence and deep learning using interactive electronic programming notebooks. *Archiv der Pharmazie*, page e2200628, 2023.

[39] Rachel Clune, Avishek Das, Dipti Jasrasaria, Elliot Rossomme, Orion Cohen, and Anne M Baranger. Development of a week-long mathematics intervention for incoming chemistry graduate students. *Journal of chemical education*, 2023.

A The curriculum in detail

In Fig. 3, we break the individual lessons of the course down into the tree main subject disciplines and which parts of the course teach what concepts. In addition, the case studies we look at are described. The actual course content can be found at the course website `https://structural-bioinformatics.netlify.app`.

Structural Bioinformatics, detailed curriculum

Lecture	Biology	Mathematics	CS/Machine Learning	Case Studies
L1: Introduction	Protein structure, history of the field	Intro to linear algebra + probability	Biological file formats + handling	PDB files
L2: ML Basics	-	Optimisation, gradient descent	Neural networks, basic notions	PyTorch
L3: ML Architectures	Computational representation of proteins	Matrix Algebra	CNNs, RNNs, transformers	AlexNet, transformers
L4: Language, Evolution and Bioinformatics	Homology, phylogeny	Distance metrics, clustering	Language models, data leakage	ESM
L5: Geometric Deep Learning	Computational representation of generic molecules	Invariance, equivariance, group theory	Graph Neural Networks (GNNs), geometric graph learning	GCN, GAT, EGNN
L6: Protein Structure Prediction	Structure-Function relationship, coevolution, protein dynamics/ interactions	End-to-end differentiability, quaternions	Inductive biases in model building, self-supervised learning	AlphaFold2, ESMFold
L7: Generative Modelling	-	distribution learning, score functions	Function modelling vs generative modelling, VAEs, diffusion models	Autoregressive VAEs, DDPMs
L8: Protein Design	Sequence- vs structure-based methods, catalysis, functional motifs	SO(3) group equivariance	Equivariant diffusion models	Rosetta, RFDiffusion, ProteinMPNN
L9: Simulations	Protein dynamics, conformational flexibility, structure ensembles	Numerical vs analytical integration, Newton's equations of motion	Performance/accuracy trade-off, coarse-graining, multiprocessing	GROMACS, Allegro
L10: Drug Design	Protein-ligand interactions, virtual screening	-	Rephrasing a problem as a generative one, data-driven vs rule-based methods	AutoDock, DIffDock, DiffSBDD
L11: Further Topics and Conclusion	Summary and Conclusion	Summary and Conclusion	Summary and Conclusion	-

Figure 3: The course curriculum in more detail. The lesson contents and learning outcomes are separated into the three main subject categories (biology, mathematics, CS/Machine Learning); in addition, the case studies discussed that illustrate the different concepts during the lesson are mentioned.

B Related courses

The course by Vater et al. integrates a design-to-data workflow in a biochemistry Course-based Undergraduate Research Experience (CURE), connecting students to a global community of protein researchers, thus enriching the undergraduate research landscape [33]. Yang et al. explored a cohort-based learning approach to improve computational protein design literacy among undergraduates, providing valuable research opportunities during the COVID-19 pandemic [34]. Le et al. developed a hands-on education strategy with sixteen modules using PyRosetta Jupyter notebooks to teach biomolecular structure and design, covering topics from conformational sampling to protein docking, and RNA structure prediction [35].

Centeno et al. described a course that employs hands-on computer approaches to teach structural biology, preparing students to build protein models based on sequence and structure information [36]. Engelberger et al. crafted cloud-based tutorials for distance learning on structural bioinformatics, combining bioinformatics software, interactive coding, and visualization exercises, which proved beneficial during the remote learning necessitated by the COVID-19 pandemic [37]. Menke et al. introduced a course on artificial intelligence and deep learning using interactive electronic programming notebooks, aiming to provide a hands-on learning experience in AI and deep learning fields [38]. Clune et al developed a 1-week mathematics boot camp for incoming chemistry graduate students and provided exercises and lecture materials for foundational mathematical topics like probability or differential equations [39].