Computational Biology (BIOSC 1540) Syllabus

University of Pittsburgh | Spring 2026[1]

Modern life scientists rely on tools to probe genomes, model proteins, and screen drug candidates. While many researchers can follow a tutorial to run these programs, a computational biologist knows why they work and, more importantly, when to trust them. We will not focus on running tools, but rather on understanding their foundations.

As an introductory course, our focus is on breadth rather than depth. We will explore the field's wide-ranging landscape to give you an overview of how computation intersects with biology. For those who wish to dive deeper into specific domains, this course serves as the essential precursor to advanced offerings like Computational Genomics (BIOSC 1542) or Simulation and Modeling (BIOSC 1544). Because this course has no programming prerequisite, we will not delve into software implementation; those looking for that level of technical depth will find it in Computational Biology Research (BIOSC 1640).

Teaching Team

The teaching team for this course is composed of individuals dedicated to supporting you in this course. Our collective role is to facilitate your learning and provide the guidance necessary to help you achieve the course objectives.

Position	Name	Pronouns	Email
Instructor	Alex Maldonado, PhD	*he/him*	alex.maldonado@pitt.edu
	UTAs were removed for FERPA compliance.		

Prerequisites

Students must have received a C or above in the following classes to enroll in this course:

- Foundations of Biology 1 (BIOSC 0150 | 0155 | 0715);
- Foundations of Biology 2 (BIOSC 0160 | 0165 | 0716).

While the courses above are required to enroll, there is always additional assumed knowledge when starting a course.

Biology

At the molecular level, you should have a working knowledge of gene structure and the Central Dogma, specifically how information flows from DNA sequences to functional proteins. Regarding protein chemistry, you should be familiar with the general properties of amino acids and the basics of protein structure; we do not require the rote

---

[1]Classes are held from January 12 to April 24 with final exams from April 27 to May 1.

memorization of every side chain, but you should understand the physical principles that allow these linear chains to fold into complex 3D shapes. Beyond individual molecules, you should be fluent in the principles of genetics and inheritance. This includes the distinction between haploid and diploid states and a clear grasp of how mutations manifest within a sequence.

Chemistry

We expect an understanding of molecular architecture, including the nature of single, double, and triple bonds. Because computational models frequently simulate molecular recognition and protein folding, you should be familiar with the noncovalent interactions that dictate these processes.

Mathematics

You should be comfortable with basic algebraic manipulation and conceptualizing slopes as a rate of change to understand the logic of a derivative. Proficiency with exponents and logarithms is critical, as we frequently move between these forms to handle probabilities and log-likelihood scores. We also expect you to have a strong intuition for what a probability represents and to be familiar with mathematical shorthand, such as sum and product notation. Finally, you must be able to organize data into tables and navigate them accurately using row and column indexing; while we use these structures constantly to represent biological data, linear algebra is not required for this course.

Programming

We will use Python as our primary vehicle for exploring biological data. Rather than building complex software from scratch, you will read pre-written code, manipulate parameters, and interpret the resulting outputs.

Meetings

We have in-person lectures on Tuesdays and Thursdays from 9:30 - 10:45 am every week in 1501 Posvar Hall.

Textbooks

There are no required textbooks for this course; however, you may peruse the list of supplemental resources if you are interested.

Assessments

In the era of generative AI, the ability to type code is becoming commoditized. Yet, the ability to understand, debug, optimize, and interpret computational logic is paramount for employability and growth. Consequently, this course de-emphasizes the mere production of scripts and prioritizes understanding the methods that drive modern biology.

Homework Assignments

Homework serves as the laboratory component of this course. Just as a wet-lab biologist uses a pipette to manipulate a sample physically, you will use Python to manipulate biological data digitally. We utilize Google Colab, a browser-based environment that requires no software installation, ensuring that every student has access to the same high-performance computing resources regardless of their personal laptop.

You will rarely be asked to write a complex program from a blank page. Instead, I will provide you with Jupyter notebooks containing pre-written data loaders and visualization code. Your assignments will involve modifying parameters, writing specific logic snippets, and interpreting the outputs. This approach allows us to focus on the biological logic of the algorithms rather than software engineering.

Assignment	Covers	Due by 11:59 pm
A1.1	L1.1 - L1.2	January 22
A1.2	L1.3 - L1.4	January 29
A2.1	L2.1 - L2.2	February 12
A2.2	L2.3 - L2.4	February 19
A3.1	L3.1 - L3.2	March 5
A3.2	L3.3 - L3.4	March 19
A3.3	L3.5 - L3.6	March 26
A4.1	L4.1 - L4.2	April 9
A4.2	L4.3 - L4.4	April 16

Exams

There will be five in-person exams throughout the semester: four module exams and one cumulative final. Each exam will take place during the regular class period the dates listed below. These dates are firm and will not change; students are expected to plan accordingly.

To support flexibility, the lowest exam score will be dropped at the end of the semester. Accordingly, there will be no make-up exams. If an exam is missed for any reason (including illness, religious observance, university obligations, or personal emergencies), the resulting zero will serve as the dropped score. Students who miss more than one exam must meet with the instructor to discuss their course standing. If you are satisfied with your exam performance, you may skip the final exam and have it counted as your drop.

Module Exams

Module exams will assess the material covered in lectures and assignments associated with each module. While exams are not formally cumulative, the course is designed in a

progressive manner: later material builds upon foundational concepts introduced earlier. Students should be prepared to apply prior knowledge throughout the term. The instructor may revisit earlier topics in exams if they were widely misunderstood.

	Module	Date
1	Search and Similarity	February 3, 2026
2	Signal Detection	February 24, 2026
3	Physical Modeling	March 31, 2026
4	Cheminformatics	April 23, 2026

Final Exam

The final exam is cumulative and will be administered at the university-designated time. No early or make-up finals will be permitted for any reason. Students with three or more finals on the same day must follow the university registrar's official exam conflict procedure failure to do so will result in the expectation to sit for the final as scheduled.

Our final exam is scheduled for **April 29, 2026** from **12:00 to 1:50 pm** in **1501 Posvar Hall**.[2]

Point Distribution

There are 264 possible points for this course. If not enough assessments are provided, students will be awarded full credit for the unaccounted points.

Assessment	Points Each	Quantity	Number of Drops	Cumulative Points
Homeworks	8	9	1	64
Exams	50	5	1	200

Late Policy

Assignments will never be accepted more than 24 hours after the due date. I will follow the late assignment and extension policy outlined below.

• Each assignment has a specified due date and time.
• Assignments submitted after the due date will incur a late penalty.
• The late penalty for that assignment is calculated using the function: % Penalty = $(25/144) \times$ (hours late)^2 rounded to the nearest tenth. This results in approximately:

Hours Late	1	3	6	9	12	18	21	24
Penalty (%)	0.2	1.6	6.3	14.1	25.0	56.3	76.6	100.0

---

[2]Last verified on January 4, 2026.

Extra Credit

Submitting all assignments on time earns you a 1% bump to your final grade.

For each assignment you submit on time, you will receive a proportion of this bonus. If you have an 89.3% (B+) in the course and submitted 8 out of 10 assignments on time, you would receive a 0.8% boost to your final grade to 90.1% (A-). Even if a low-scoring assignment is dropped from your grade calculation, its submission timestamp still counts toward this participation bonus. However, unsubmitted assignments do not count as on-time. Since this bonus is automatically available to everyone, I will not provide an additional adjustment (i.e., bump) to your final grade.

Expectations

What you can expect from me

My role is not just to lecture to you, but to train you. I approach this course with the mindset of a physicist and engineer: I value logic, systems, and first principles. My goal is to teach you to view biological complexity through that same structured lens, turning messy problems into solvable puzzles.

I set very high expectations because I know what is required to excel in this field. You can meet them, provided you are willing to engage deeply with the material. In return, I promise to be reliable and transparent. I will provide the scaffolding, the resources, and the honest feedback you need to grow. I view errors as data points, not failures; we will debug your process together.

I deliberately chose to teach. I pivoted from a career in the tech industry to academia because I found mentorship more rewarding than product development. My goal is to help you figure out where you want to go and give you the tools to get there. I invest deeply in students who take the time to come to office hours—not just to debug code, but to discuss their ambitions. I want to know you as a person, not just a roster entry, so I can be a resource for your career long after you leave this classroom.

I take our subject matter seriously, but I don't take myself too seriously. You can expect dry humor and a candid, direct lecture style. True confidence comes from mastering difficult things, not from having the bar lowered. However, learning requires vulnerability: I will be honest about what I know and admit when I make a mistake (or a typo).

To be the best coach I can be, I practice sustainable working habits. You can expect me to be fully present and dedicated to you during class and scheduled office hours. In exchange, I expect you to respect my time boundaries.

- I respond to emails during business hours (M-F, 9:00 am – 5:00 pm). Emails sent on Friday evenings will generally receive a response on Monday/Tuesday. I encourage you to use that time to work on other problems or rest.
- Assessments will be graded within seven business days so that you can iterate on feedback quickly.

What I expect from you

I treat this classroom as a professional environment and view you as a junior colleague. I expect you to approach this course with professional curiosity, ownership, and resilience. In a professional setting, you aren't paid to sit passively; you are paid to solve problems. The same applies here.

Authentic learning happens when you push past the edge of what you already know. If you are confused, that is not a sign of failure; it is a sign that you are building a new mental model. Do not shy away from the friction of learning complex systems. I expect you to persist through that initial difficulty. I will help, but you must put in meaningful effort.

You cannot master this material simply by listening to me talk. My lectures provide some mental framework and logic, but they will not teach you everything. You must build the structure yourself through independent study and practice.

I respect your time, and I do not assign busy work. Every reading and assignment has been designed to help you achieve a specific learning objective. In return, I expect you to be fully present in the classroom. Please restrict phone use and distractions so we can make the most of our contact time. Plan to engage with the course material for at least 5 hours each week outside of class. This is the standard required to succeed in a course of this rigor.

We are all individuals with different backgrounds. We will likely disagree on ideas, but our discourse must always remain civil, logical, and evidence-based.

Because I value mentorship, I want to help you before a problem becomes a crisis. Do not wait until you are drowning to ask for a life raft. Come to the office hours when you first smell smoke, not when the building is on fire.

While I strive for precision, I am human. If you catch an error on a slide or a mistake in the materials, please let me know! I am committed to getting it right, and I appreciate the peer review.

Attendance Policy

I do not take attendance for a grade because you are capable of determining how you spend your time. If you believe you can learn the material by reading the documentation

and working independently, you are free to do so. Your grade is based on your output (i.e., deliverables and exams), not your input (i.e., seat time).

However, I strongly encourage you to attend. My lecture slides will be uploaded to Canvas, but they are not designed to be passive information dumps. Historically, students who attempt to learn the course from the slide decks alone struggle to perform on exams.

While attendance is optional, my time is a finite resource. To be fair to the class as a whole, I must prioritize my capacity for support. Office hours are intended to refine your understanding, not to replace the lecture. If you choose to miss a class, you accept the responsibility of catching up independently (e.g., getting notes from a peer). I cannot use office hours to privately reteach a lecture to a student who did not attend.

Errors in Grading

We strive for precision, but errors can occur. If you identify a discrepancy in a grade, you have a one-week window from the time it is posted to flag it for review.

Grading Scale

Letter grades for this course will be assigned based on Pitt's recommended scale[3].

Letter	Percentage	GPA	Attainment
A +	97.0 - 100.0%	4.00	
A	93.0 - 96.9%	4.00	Superior
A −	90.0 - 92.9%	3.75	
B +	87.0 - 89.9%	3.25	
B	83.0 - 86.9%	3.00	Meritorious
B −	80.0 - 82.9%	2.75	
C +	77.0 - 79.9%	2.25	
C	73.0 - 76.9%	2.00	Adequate
C −	70.0 - 72.9%	1.75	
D +	67.0 - 69.9%	1.25	
D	63.0 - 66.9%	1.00	Minimal
D −	60.0 - 62.9%	0.75	
F	0.0 - 59.9%	0.00	Failure

---

[3]University of Pittsburgh's grading system.

Artificial Intelligence

Summary

You may use AI (e.g., ChatGPT, Claude, Gemini) as a private tutor to explain concepts or debug your understanding. However, you strictly cannot use AI to generate text, code, or solutions that you submit as your own work. To ensure you are learning the necessary skills, exams will be in-person and will require you to demonstrate knowledge without digital assistance.

We are in an era of generative artificial intelligence (AI) development, marked by the release of tools such as Gemini, ChatGPT, Claude, Grok, and Copilot, among many others. While these tools can be powerful aids for learning, their use in this course is subject to strict guidelines to ensure academic integrity and meaningful engagement with the material.

My Philosophy

Generative AI is a powerful tool with immense potential when used ethically and critically. It enables individuals to produce outcomes that exceed their traditional skill sets, which can be particularly beneficial for completing specific tasks. If the primary value of an activity is to generate an output as fast as possible, then AI can be beneficial.

However, the purpose of education, especially in a university setting, is fundamentally different. Our goal is not just to produce a final product; it is to equip you with the critical thinking skills and foundational knowledge necessary for you to achieve your ambitions. Your degree is not merely a golden ticket. Your degree is a rigorous endorsement by the university and your professors that you possess the capabilities and understanding described by your major. Learning here is an active process of acquiring tools and developing your intellect, not passively attending classes to receive a credential.

It's also important to consider the real-world implications of generative AI. Many companies currently prohibit or heavily restrict their use. This means there's a significant chance you won't be able to rely on these technologies in your professional life. Furthermore, while generative AI can make output accessible to many, employers seek individuals who add unique value and solve problems. Why would an employer hire someone whose primary skill is operating a large language model when they could outsource that task for a fraction of the cost? The value you bring to an organization lies in your ability to evaluate information critically, adapt to new challenges, and innovate beyond what AI can generate.

From my perspective, relying on generative AI in your role as a student actively harms your prospects and diminishes your education. While it might seem like an easier route to complete assignments, that convenience comes at the cost of genuine learning and

skill development. It undermines the very purpose of this course, which is to build your capabilities.

Given the current environment and the unknown long-term effects of generative AI on learning, in-person exams remain the most reliable way for me to assess your proper understanding of the material. This approach ensures that your knowledge and skills are genuinely your own, providing a more accurate measure of your capabilities. We will continue to evaluate and adapt our assessment methods as we gain a clearer understanding of how generative AI impacts education.

Possible Uses

It is your decision whether to use generative AI. I strongly advise you against it, but I cannot effectively monitor your usage of generative AI outside the classroom. Thus, if you plan to use it, here are some ways you can do so responsibly.

- **Clarifying Complex Topics and Concepts.** If you're struggling to grasp a difficult concept from a lecture or reading, you can ask a generative AI to explain it in simpler terms or provide different analogies. For example, if we're discussing a complex algorithm, you might ask, "Explain *[algorithm name]* in a way a beginner could understand," or "What are some real-world examples of *[concept]*?" This can help solidify your understanding before you tackle problems independently.
- **Exploring Alternative Explanations for Difficult Material.** Sometimes, a different perspective can make all the difference. If you're stuck on a particular problem or explanation, you can prompt the AI for alternative viewpoints or methods. For instance, you could ask, "Are there other ways to approach this type of problem?" or "Can you provide a different explanation for [specific theory]?" This can broaden your understanding and provide new insights.

The key here is that AI should supplement, not supplant, your own thinking. Your goal is to engage with the material and develop your problem-solving skills. You must always complete assignments and assessments based on your own understanding and work, not by relying on AI-generated solutions.

> **Warning**: Remember, generative AI is a massive word probability model. After observing millions of biased examples of materials, generative AI will predict what word it often sees after the current one. There are computational techniques to try to maintain the accuracy of these probability models, but they are far from achieving this reliably.

Prohibited Uses

To ensure academic integrity and genuine learning, the following uses of generative AI are strictly prohibited and will be treated as academic integrity violations:

- **Generating or Modifying Assignments.** You cannot use generative AI to produce or alter any part of your homework assignments, projects, papers, or any other graded work. This includes generating entire responses, crafting paragraphs, or even rephrasing significant portions of your work that you didn't write. For example, using an AI to write an answer for you or to generate code that you then submit as your own falls under this prohibition.
- **Using AI-Generated Answers as Direct Solutions.** Simply put, you cannot copy and paste or transcribe answers directly from a generative AI tool as solutions to problems or questions. The purpose of this course is for you to develop your problem-solving skills, not to have an AI solve them for you. If a problem asks you to derive a formula, you must show your own derivation, not just paste an AI-generated result.
- **Submitting AI-Generated Work.** Any work you submit must be entirely your own. Submitting content created by generative AI, regardless of how minor the contribution, is considered a form of academic dishonesty. This means you cannot submit AI-generated text, images, code, or any other output as if it were your original creation.
- **Circumventing Course Policies.** You cannot use generative AI to bypass any course policies. For example, if a policy requires you to show your work or explain your reasoning, using AI to generate only the final answer without demonstrating your thought process is a violation.
- **Uploading Course Materials to Generative AI Models.** You are strictly prohibited from uploading any course-specific materials (e.g., lecture notes, assignment prompts, readings, exam questions, solutions, or discussions from the learning management system) to any generative AI model or platform. This includes, but is not limited to, pasting text into chat interfaces or uploading documents. This prohibition is crucial for several reasons:

  ▸ Course materials are often copyrighted and proprietary.

  Uploading them to a third-party AI model may violate intellectual property rights and the university's licensing agreements with content providers, potentially leading to legal issues for both you and the university.

  ▸ The information you upload to AI models may become part of their training data, making it potentially accessible to others or used for purposes beyond your control.

  This risks compromising the privacy of course content and potentially exposing sensitive or confidential academic information.

  ▸ Uploading course materials could inadvertently "train" the AI model on course-specific content, potentially making it easier for future users (including other students) to generate answers or solutions to assignments, undermining the integrity of the course and its assessments.

- ▸ To ensure fair and valid assessments, it is essential that the content of assignments and exams remains contained within the learning environment.

Uploading these materials to external AI platforms compromises the integrity of current and future assessments.

In essence, if you're using AI to think or write for you or share course materials in a way that could compromise academic integrity or intellectual property, it's prohibited. Your work should reflect your understanding, effort, and critical engagement with the course material.

Academic Integrity

Students in this course will be expected to comply with the University of Pittsburgh's Policy on Academic Integrity. Any student suspected of violating this obligation during the semester will be required to participate in the procedural process initiated at the instructor level, as outlined in the University Guidelines on Academic Integrity. This may include, but is not limited to, the confiscation of the examination of any individual suspected of violating University Policy. Furthermore, no student may bring unauthorized materials to an exam, including dictionaries and programmable calculators.

To learn more about Academic Integrity, visit the Academic Integrity Guide for an overview. For hands-on practice, complete the Understanding and Avoiding Plagiarism tutorial.

Disability Services

If you have a disability for which you are or may be requesting an accommodation, you are encouraged to contact both your instructor and Disability Resources and Services (DRS), 140 William Pitt Union, (412) 648-7890, drsrecep@pitt.edu, (412) 228-5347 for P3 ASL users, as early as possible in the term. DRS will verify your disability and determine reasonable accommodations for this course.

Email Communication

Upon admittance, each student is issued a University email address (username@pitt.edu). The University may use this email address for official communication with students. Students are expected to read emails sent to this account regularly. Failure to read and react to University communications promptly does not absolve the student from knowing and complying with the content of the communications. The University provides an email forwarding service that allows students to read their email via other service providers (e.g., Gmail, AOL, Yahoo). Students who forward their email from their pitt.edu address to another address do so at their own risk. If email is lost due to forwarding, it does not absolve the student from responding to official communications sent to their University email address.

Equity, Diversity, and Inclusion

The University of Pittsburgh does not tolerate any form of discrimination, harassment, or retaliation based on disability, race, color, religion, national origin, ancestry, genetic information, marital status, familial status, sex, age, sexual orientation, veteran status or gender identity or other factors as stated in the University's Title IX policy. The University is committed to taking prompt action to end a hostile environment that interferes with the University's mission. For more information about policies, procedures, and practices, visit the Civil Rights & Title IX Compliance web page.

I ask that everyone in the class strive to help ensure that other members of this class can learn in a supportive and respectful environment. If there are instances of the aforementioned issues, please contact the Title IX Coordinator, by calling 412-648-7860 or emailing `titleixcoordinator@pitt.edu`. Reports can also be filed online. You may also choose to report this to a faculty/staff member; they are required to communicate this to the University's Office of Diversity and Inclusion. If you wish to maintain complete confidentiality, you may also contact the University Counseling Center (412-648-7930).

Religious Observance

The observance of religious holidays (activities observed by a religious group of which a student is a member) and cultural practices are an important reflection of diversity. As your instructor, I am committed to providing equivalent educational opportunities to students of all belief systems. At the beginning of the semester, you should review the course requirements to identify foreseeable conflicts with assignments, exams, or other required attendance. If possible, please contact me (your course coordinator/s) within the first two weeks of the first class meeting to allow time for us to discuss and make fair and reasonable adjustments to the schedule and/or tasks.

Sexual Misconduct, Required Reporting, and Title IX

If you are experiencing sexual assault, sexual harassment, domestic violence, and stalking, please report it to me and I will connect you to University resources to support you.

University faculty and staff members are required to report all instances of sexual misconduct, including harassment and sexual violence to the Office of Civil Rights and Title IX. When a report is made, individuals can expect to be contacted by the Title IX Office with information about support resources and options related to safety, accommodations, process, and policy. I encourage you to use the services and resources that may be most helpful to you.

As your instructor, I am required to report any incidents of sexual misconduct that are directly reported to me. You can also report directly to Office of Civil Rights and Title

IX: 412-648-7860 (M-F; 8:30am-5:00pm) or via the Pitt Concern Connection at: Make A Report.

An important exception to the reporting requirement exists for academic work. Disclosures about sexual misconduct that are shared as a relevant part of an academic project, classroom discussion, or course assignment, are not required to be disclosed to the University's Title IX office.

If you wish to make a confidential report, Pitt encourages you to reach out to these resources:

- The University Counseling Center: 412-648-7930 (8:30 A.M. TO 5 P.M. M-F) and 412-648-7856 (AFTER BUSINESS HOURS)
- Pittsburgh Action Against Rape (community resource): 1-866-363-7273 (24/7)

If you have an immediate safety concern, please contact the University of Pittsburgh Police, 412-624-2121

Any form of sexual harassment or violence will not be excused or tolerated at the University of Pittsburgh.

For additional information, please visit the full syllabus statement on the Office of Diversity, Equity, and Inclusion webpage.

Statement on Classroom Recording

To ensure the free and open discussion of ideas, students may not record classroom lectures, discussions and/or activities without the advance written permission of the instructor, and any such recording properly approved in advance can be used solely for the student's private use.

Schedule

The following tentative schedule outlines the topics we will cover. We will do our best to adhere to this schedule; however, changes may be necessary.

Module 1 - Search and similarity

L1.1 - Course Tour *(Jan 13)*

We establish the computational environment that serves as our digital laboratory. While this is not an introductory programming course, we define the specific subset of Python needed to manipulate biological data efficiently.

L1.2 - The Digital Genome *(Jan 15)*

We confront the reality that genomic data is inherently noisy, focusing on the FASTQ format as the field's standard currency. You will learn that a DNA sequence is a probabilistic assertion rather than a fixed string. By deriving Phred quality scores, we establish how to filter statistical noise before it propagates into downstream analysis.

L1.3 - Pairwise Sequence Alignment *(Jan 20)*

We address the fundamental challenge of comparing biological sequences to quantify evolutionary distance. Using Dynamic Programming, we demonstrate how to optimally align DNA and protein sequences by translating biological intuition into mathematical scoring matrices and gap penalties. You will learn to view alignment not just as text matching, but as a rigorous optimization problem that seeks to maximize a specific objective function.

L1.4 - Heuristic Searches *(Jan 22)*

Since dynamic programming is computationally expensive for large databases, we pivot to heuristic search algorithms that trade sensitivity for speed. We deconstruct the seed-and-extend paradigm and the use of hash tables to identify matches rapidly. We introduce statistical significance, using E-values to distinguish biologically relevant hits from random matches.

L1.5 - Read Mapping *(Jan ~~27~~ 29)*

Due to Pitt cancelling in-person classes on Jan 27, this lecture was moved to Jan 29. Instead, we did a virtual exam review session and dropped pseudo-alignment from this module.

We move from pairwise comparison to reconstructing samples against a reference genome. This lecture covers the high-throughput application of alignment and decoding standard formats such as SAM/BAM and CIGAR strings. We also examine how probabilistic errors and alignment scores are synthesized to report mapping confidence.

~~L1.6 - Pseudo-Alignment *(Jan 29)*~~

~~Challenging the need for base-to-base alignment, we introduce alignment-free methods that drive modern transcriptomics. We examine K-mers to quantify expression without the cost of traditional alignment. This shifts the conceptual focus from read location to read abundance, bridging the gap to statistical counting models.~~

Module 2 - Signal Detection

L2.1 - Variant Calling *(Feb 5)*

We address the challenge of distinguishing sequencing errors from biological mutations. We demonstrate why simple pileups fail near insertions and deletions and how local re-assembly resolves these artifacts. We introduce variant formats and genotype likelihoods, applying Bayesian reasoning to distinguish true heterozygotes from noisy homozygotes.

L2.2 - Genome-Wide Association Studies *(Feb 10)*

We treat the genome as a numerical matrix of dosages to perform Genome-Wide Association Studies (GWAS). We cover regression and the statistical burden of testing millions of variants simultaneously. You will interpret diagnostic plots to distinguish true biological signals from the genomic inflation found in poorly designed studies.

L2.3 - Genome Correlation *(Feb 12)*

We address the two main confounders in genetics: Population Stratification and Linkage Disequilibrium (LD). We introduce dimensionality reduction techniques to correct for stratification and explore how LD complicates the "fine-mapping" of causal variants. We discuss methods for aggregating genetic effects into risk scores for precision medicine.

L2.4 - RNA Quantification *(Feb 17)*

Quantifying gene expression requires transforming raw alignment data into a structured count matrix. We dissect the logic of feature counting, addressing the complexities of alternative splicing, multi-mapping reads, and gene overlaps. This establishes the rigorous counting standards necessary for downstream differential expression analysis.

L2.5 - Differential Gene Expression *(Feb 19)*

We demonstrate why raw read counts are insufficient and derive normalized metrics to correct for composition bias. We introduce probabilistic distributions to model the overdispersion inherent in RNA-seq data. We apply statistical hypothesis testing frameworks and correct for multiple testing using the False Discovery Rate (FDR).

Module 3 - Physical Modeling

L3.1 - Atomistic Structure *(Feb 26)*

We move from 1D strings to the 3D reality of biological macromolecules. This lecture introduces the coordinate systems used to define molecular topology and examines how experimental data are discretized into atomic coordinates and quality metrics. We establish that static structures are simplified models of dynamic objects, setting the geometric primitives for physics simulations.

L3.2 - Force Fields *(Mar 3)*

To simulate motion, we define the rules of engagement between atoms using Force Fields. We introduce the mathematical functions that approximate quantum-mechanical reality using Newtonian mechanics. By exploring specific energy terms, we examine the fundamental trade-off in molecular mechanics: balancing physical accuracy with computational tractability.

L3.3 - Energy Minimization *(Mar 5)*

Experimental structures often contain high-energy artifacts that must be relaxed before simulation. We treat molecular geometry as an optimization problem and use gradient-based algorithms to navigate the potential energy surface. We distinguish between local and global minima to resolve steric clashes and prepare stable structures for dynamics.

L3.4 - Molecular Dynamics Engine *(Mar 17)*

We focus on Molecular Dynamics (MD), the iterative integration of Newton's equations of motion. We cover integration algorithms and the necessity of femtosecond time steps to resolve atomic vibrations. You will learn that computational cost is the limiting factor, as simulating biological timescales requires billions of calculation steps.

L3.5 - Atomistic Insights *(Mar 19)*

A single snapshot offers limited insight; the statistical distribution of states drives function. We bridge simulations to statistical mechanics by introducing the thermodynamic ensemble. We move beyond visual inspection to quantitative metrics for stability and flexibility, analyzing trajectories to identify functional protein motions.

L3.6 - Protein Structure Prediction *(Mar 24)*

We address the protein folding problem: predicting 3D structure from sequence. We contrast classical template-based approaches with modern deep learning methods. We deconstruct how evolutionary covariance is transformed into spatial constraints and emphasize the importance of confidence metrics in evaluating predicted structures.

L3.7 - Protein Engineering *(Mar 26)*

We transition from observation to intervention, using physics-based scores and machine learning to predict mutation effects. We apply these principles to protein engineering, focusing on optimizing stability and binding affinity. You will learn how computational mutagenesis is used to design therapeutic candidates with improved target interactions.

Module 4 - Cheminformatics

L4.1 - The Language of Chemistry *(Apr 2)*

We introduce linear notation systems to digitize small molecules. We explore how molecules are represented as graphs and use canonicalization algorithms to ensure unique identifiers. You will learn to manipulate these strings programmatically, a prerequisite for searching the massive chemical spaces used in drug discovery.

L4.2 - Molecular Descriptors *(Apr 7)*

We cover molecular descriptors, extracting numerical features from chemical graphs to quantify properties like hydrophobicity and molecular weight. We introduce drug-likeness heuristics used to estimate bioavailability. By reducing complex structures to feature vectors, we prepare chemical data to predict biological activity.

L4.3 - Molecular Similarity *(Apr 9)*

We formalize the neighborhood behavior principle using molecular fingerprints and similarity metrics. We demonstrate how bit-vectors enable rapid database screening. You will learn that similarity is relative to the chosen descriptor, a concept critical for scaffold hopping to find structurally novel compounds with desired activities.

L4.4 - Chemical Libraries *(Apr 14)*

Drug discovery is a search problem within an effectively infinite space. We discuss the construction and management of chemical libraries, distinguishing between diversity and focused collections. Crucially, we address synthesizability, teaching you to filter out theoretical molecules that cannot be physically manufactured.

L4.5 - Protein-Ligand Docking *(Apr 16)*

We treat molecular docking as an optimization problem: finding the ligand orientation that minimizes binding free energy. We dissect the interaction between search algorithms and scoring functions. You will learn to use docking scores as ranking metrics to prioritize compounds for experimental testing rather than as precise affinity measurements.

L4.6 - Lead Optimization *(Apr 21)*

Turning a hit into a drug requires multiparameter optimization. We cover the Lead Optimization phase, discussing ADMET properties. We examine how structural modifications are made to balance improved efficacy with reduced toxicity.

Appendices

These materials supplement the syllabus and primarily provide additional context.

Supplemental

These resources can be used to deepen your understanding of various topics, but due to the nature of this course, are not required.

BMC	Bishop, C. M. (2006) *Pattern recognition and machine learning*. Spring Science+Business Media, LLC.
BSM	Brown, S. M. (2013) *Next-generation DNA sequencing informatics*. Cold Spring Harbor Laboratory Press.
CHS	Chan, S. H. (2021) *Introduction to probability for data science*. Michigan Publishing.
CLRS	Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (2022) *Introduction to Algorithms* (4th Ed.). MIT Press.
CMS	Coumar, M. S. (2021) *Molecular docking for computer-aided drug design: Fundamentals, techniques, resources and applications*. Elsevier Inc.
CS	Chateau, A. Salson, M. (2022) *From sequences to graphs: Discrete methods and structures for bioinformatics*. John Wiley & Sons, Inc.
DBR	Donald, B. R. (2011) *Algorithms in structural molecular biology*. MIT Press.
DEKM	Durbin, R., Eddy, S. R., Krogh, A., Mitchison, G. (1998) *Biological sequence analysis: Probabilistic models of protein and nucleic acids*. Cambridge University Press.
DTW	Deonier, R. C., Tavaré, S., Waterman, M. S. (2005) *Computational genome analysis: An introduction*. Springer Science.
FDA	Forero, D. A. (2022) *Bioinformatics and human genomics research*. CRC Press.
GPA	Gagniuc, P. A. (2021) *Algorithms in bioinformatics: Theory and implementation*. John Wiley & Sons, Inc.
GS	Gervasio, F. L., Spiwok, V. (2019) *Biomolecular simulations in structure-based drug discovery*. Wiley-VCH Verlag GmbH & Co.
HA	Heifetz, A. (2020) *Quantum mechanics in drug discovery*. Springer Nature.
KA	Khamis, A. (2024) *Optimization Algorithms: AI techniques for design, planning, and control problems*. Manning Publications.
KT	Kleinberg, J. Tardos, E. (2005) *Algorithm Design*. Pearson.
KW	Kochenderfer, M. J. Wheeler, T. A. (2025) *Algorithms for Optimization*. MIT Press.

LA	Levitin, A. (2021) *Introduction to the Design and Analysis of Algorithms* (3rd Ed.). Pearson.
LS	Luke, S. (2013) *Essentials of Metaheuristics* (2nd Ed.). Lulu.
LAM	Lesk, A. M. (2014) *Introduction to bioinformatics* (4th Ed.). Oxford University Press.
MPK	Murphy, K. P. (2012) *Machine learning: A probabilistic perspective*. MIT Press.
MWH	Majoros, W. H. (2007) *Methods for computational gene prediction*. Cambridge University Press.
NW	Nocedal, J., Wright, S. J. (2006) *Numerical Optimization* (2nd Ed.). Springer-Verlag New York, Inc.
PJ	Pevzner, P. A., Jones, N. C. (2004) *An introduction to bioinformatics algorithms*. MIT Press.
PS	Pal, S. (2020) *Fundamentals of molecular structural biology*. Elsevier Inc.
RSM	Ross, S. M. (2014) *Introduction to Probability Models* (11th Ed.). Academic Press.
SR	Schwartz, R. (2008) *Biological modeling and simulation: A survey of practical models, algorithms, and numerical methods*. MIT Press.
SSS	Skiena, S. S. (2008) *The Algorithm Design Manual* (2nd Ed.). Springer Nature.
SW	Sedgewick, R., Wayne, K. (2011) *Algorithms* (4th Ed.). Addison-Wesley Professional
YDC	Young, D. C. (2009) *Computational drug design: A guide for computational and medicinal chemists*. John Wiley & Sons, Inc.

Exam Format

In scientific disciplines, "knowing" is not a binary state. There is a profound difference between recognizing a technical term and possessing the ability to deploy that concept to solve a novel problem or diagnose a complex system failure. Traditional examinations often mix these cognitive levels randomly, leading to grades that do not clearly reflect your actual depth of skill. In this course, we utilize a Tiered Assessment Framework to ensure transparency. This structure is designed so that you will always understand exactly what cognitive skill is being tested and how to prepare for it.

Unlike hurdle systems in bundled grading, where you must pass one section to unlock the next, this course uses even weighting. Every question on the exam is worth the same amount of points (approx. 3.125% each). However, the mix of questions is carefully curated to ensure that your final score reflects your level of mastery.

If you master …	You will likely get …	Grade range
D & C tiers	~ 22/32 questions	~ 69%
+ B tier	~ 27/32 questions	~ 84%
+ A tier	~ 30/32 questions	~ 94%
+ S tier	~ 32/32 questions	~ 100%

This creates a natural sorting mechanism based on your depth of understanding. If you have only mastered the definitions and basic concepts (D and C Tiers), you will be able to answer the majority of the questions and pass the course, but you will hit a "ceiling" when you encounter the higher-tier questions. To earn a B, you must demonstrate the ability to simulate dynamic processes (B Tier). To earn an A, you must demonstrate the ability to diagnose complex system failures (A Tier) and synthesize new models (S Tier). This structure ensures that a high grade is not just a measure of how much you remember, but a measure of how effectively you can think.

Tier Descriptions

D-Tier (Definitions) The Definitions tier strictly tests your proficiency in the vocabulary of the field through one-to-one association. Questions at this level require you to map a specific term, structure, or entity to its corresponding name or category without any further manipulation or reasoning. The mental process here is purely lookup; you are simply identifying the identity of an object in isolation. When you encounter these questions, you should ask yourself if the answer is simply the label of the thing provided. These questions ensure you are fluent in the terminology required to communicate scientific ideas.

C-Tier (Concepts) The Concepts tier tests your understanding of static properties and mechanisms. Unlike the Definitions tier, which asks "What is this?", the Concepts tier asks "How does this work?" or "Why is this the way it is?". In this tier, you must identify intrinsic characteristics, functional roles, or underlying explanations of a single entity or principle in its natural state. A strict rule for this tier is that it describes a system as it exists; you are not asked to change variables, manipulate data, or predict new outcomes.

B-Tier (Broadening) The move from Concepts to Broadening is the move from a static image to a motion picture. This tier tests your ability to handle dynamic processes and linear logic. It is distinct from the Concepts tier because it strictly involves a change, a sequence, or a comparison. You are required to mentally change a specific variable within a known system, rule, pathway, or procedure to determine a result. These questions effectively ask you to run a mental simulation of a process to generate a specific output. If a question asks "If X happens, then what?", or requires you to order

a series of events, it belongs to this tier. Success here requires you to "trace" the logic of the system step-by-step to predict the immediate downstream consequence.

A-Tier (Analytical) While the Broadening tier looks at linear cause-and-effect, the Analytical tier tests your ability to manage system state and interaction. This is the domain of debugging and diagnosis. The mental process required here is checking multiple conditions simultaneously to determine the state of a complex system. A strict rule for this tier is that the question must involve logic gates (e.g., "If X is high AND Y is low, then what?") or require you to validate a list of potential outcomes where more than one answer is correct. You are no longer looking at a single variable; you are looking at how multiple variables or distinct concepts interact. This requires you to hold the entire system in your head and evaluate how conflicting rules resolve to determine why a specific outcome occurred.

S-Tier (Synthesis) The Synthesis tier represents the highest level of cognitive performance because it tests novelty and evaluation. Here, you are asked to generate a prediction for a scenario that was not explicitly covered in class, or to evaluate competing models to choose the "best" one. The mental process is "Modeling", you must build a mental picture of a new system based on the principles you have learned. Unlike lower tiers, where the answer is clearly "right or wrong" based on a rule, S-tier questions often ask which explanation is "most likely" or "best supported." You might be presented with a novel scenario or a set of anomalous data and asked to propose a hypothesis or modify an existing model to fit the new context. To succeed, you cannot simply recall a fact; you must understand the design choices of the system well enough to critique and modify them.

Logistics

Every exam will have 32 questions, distributed as follows, with each question worth 3.125% of the exam grade.

Tier	D	C	B	A	S
Number of questions	6	16	5	3	2

Core Topic Descriptions

The field of computational biology is vast, spanning from the study of single chemical bonds to the analysis of entire populations. To help you navigate this complexity, this appendix provides a detailed breakdown of the field's core pillars. There is not enough time in this course to cover everything, so we detail them here for your exploration.

As you progress through the semester, use these descriptions as a roadmap for your professional development. If you are interested in a career in drug discovery, look into cheminformatics and structural biology; if you are drawn to precision medicine and clinical diagnostics, the fields of NGS and Statistical Genetics are important.

Next-Generation Sequencing (NGS)

Next-Generation Sequencing (NGS) represents the technological shift from sequencing single DNA fragments to massively parallel sequencing of millions of fragments simultaneously. In this domain, computational biologists move beyond laboratory chemistry to focus on the data lifecycle: how raw signals from a sequencer are converted into high-fidelity digital strings (FASTQ). Mastery of this field involves the algorithmic logic behind high-speed mappers and variant callers—tools used for millions of short reads within a 3-billion-base-pair human genome. Understanding NGS is foundational for biotechnology, as it serves as the primary readout for nearly all modern biological experiments. Expertise includes manipulating large datasets with Python and familiarity with industry-standard file formats.

RNA-seq and Transcriptomics

While the genome is a static blueprint, the transcriptome is a dynamic snapshot of biological activity. RNA-seq enables quantification of gene expression across different tissues, disease states, or drug treatments. This field centers on the Count Matrix pipeline, which transforms aligned reads into numerical values representing the activity levels of specific genes. Key technical competencies include the application of normalization methods and the statistical rigors of differential expression analysis to ensure biological signals are distinguished from statistical noise. Beyond bulk analysis, single-cell RNA-seq enables scientists to examine the transcriptomes of individual cells, revealing hidden diversity within complex tissues such as tumors and brain regions. Professional proficiency in this area often requires using Python libraries to perform clustering and dimensionality reduction.

Statistical Genetics

Statistical Genetics is the study of how minute differences in DNA lead to observable traits or diseases. A primary tool in this field is the genome-wide association study (GWAS), which scans the entire genome of thousands of individuals to find statistical correlations between specific variants and disease risk. Advanced applications involve the concepts of linkage disequilibrium and polygenic risk scores (PRS), which aggregate thousands of small genetic effects to predict a person's overall health trajectory. For a computational biologist, this field represents a masterclass in handling big data. Professional roles require the ability to parse files containing millions of variants while applying population-level filters to account for ancestry and family structure.

Cheminformatics and Computer-Aided Drug Design

Cheminformatics bridges the gap between digital data and the physical world of small molecules. In this context, molecules are treated as computational objects: graphs where atoms are nodes and bonds are edges. Using industry-standard libraries like RDKit, scientists represent chemicals as SMILES strings, calculate molecular fingerprints,

and perform similarity searches. This discipline also applies heuristics to determine if a molecule possesses the properties required to become a successful therapeutic. Computer-aided drug design extends these principles into predictive simulations. Through virtual screening, millions of molecules are digitally docked into a protein's binding pocket to identify potential hits efficiently.

Computational Structural Biology

Biology is inherently three-dimensional. Computational Structural Biology focuses on the physical shape and motion of proteins. This field encompasses the algorithms behind protein structure prediction and the parsing of 3D coordinate files. Technical expertise in this area involves using computational tools to calculate distance matrices and identify the specific amino acids that form the active site where a drug might bind. A key component is molecular dynamics, which uses the laws of physics to simulate how a protein moves over time. By calculating the forces between thousands of atoms, researchers observe how a protein changes shape when it encounters a drug.

Systems Biology and Metabolic Modeling

Systems Biology moves away from looking at single genes or proteins in isolation and instead views the cell as a complex, interconnected circuit. Graph theory is used to model these interactions and identify the network's hubs, which are most critical for biological function. Proficiency in this domain involves using Python's `NetworkX` to visualize protein-protein interaction networks and identify functional modules that may be dysregulated in disease. A major focus is flux balance analysis, a mathematical approach for simulating the metabolism of an entire cell.

Lipidomics and Metabolomics

While the genome and transcriptome indicate potential activity, the metabolome and lipidome provide a readout of what is currently occurring in the cell. Metabolomics is the study of small molecules (metabolites) like glucose or amino acids, while Lipidomics focuses on the structural and signaling molecules (lipids) that form cell membranes. These fields are closest to the phenotype, providing a direct readout of a cell's current chemical state. Computationally, these fields present unique challenges because metabolites and lipids are chemically diverse and often require mass spectrometry (MS) for identification. Technical skills in this area include handling peak lists and m/z (mass-to-charge) data in Python, and mapping these chemical signatures to biological pathways using databases such as KEGG and HMDB.