CellPress
REVIEWS

Scientific Life

Computational and Systems Immunology: A Student's Perspective

Zinaida Good,[1,2,3,4,7,*] Jacob Glanville,[1,3,4,8] Marvin H. Gee,[1,5,9] Mark M. Davis,[3] and Purvesh Khatri[6,*]

The big data revolution has transformed the landscape of immunology research. As inaugural students of Stanford's new Computational and Systems Immunology PhD track, we share our experiences and advice with other institutions considering a similar program.

An Evolving Field of Immunology

Technological revolution in immunology is producing ever-increasing amounts of data [1]. Translating these data into insights requires advanced statistics, data mining, and machine learning skills. It also requires a strong background in immunology to ask suitable questions, design appropriate experiments, and recognize discoveries in data trends. The paucity of researchers with a combination of these skills has created a bottleneck in systems-level immunology research. Training a new generation of computational immunologists will provide opportunities for discoveries that simply have not been possible with traditional approaches [2,3].

As immunology PhD students at Stanford University, we praise our faculty for anticipating this need and establishing a Computational and Systems Immunology (CSI) track within the PhD Program in Immunology in 2012. Initiated by Mark Davis and Atul Butte (the latter now at University of California, San Francisco), the CSI track runs in parallel with the Molecular, Cellular, and Translational Immunology (MCTI) track. To our knowledge, Stanford University remains the only academic institution to offer a formal graduate program in computational immunology.

As CSI students, we are grateful for the skills we received and the opportunities that this training has opened. Here, we describe our program and enthusiastically recommend that other institutions offer a similar opportunity to their trainees. In collaboration with the CSI faculty and Immunology leadership, we further provide recommendations on creating and continuously refining a successful computational immunology program.

The Need for a Computational Immunology Program

As novel multiplexed technologies accelerate high-throughput interrogation of immune phenomena, these data accumulate in large public repositories, such as the NCBI Gene Expression Omnibus, NIAID-funded ImmPort, and ImmGen. An increasing number of studies are utilizing data from these repositories to identify hitherto unknown immunology [4–6]. A broad spectrum of problems that require computational immunology skills include immune repertoire analysis [7,8], structural basis of antigen recognition [9,10], single-cell differentiation trajectory construction [11,12], novel cell subset detection [13], simulation of immune processes [14], and clinical outcome prediction [4–6,15]. Thus, the field needs computational immunologists who are able to integrate their understanding of technology platforms and databases with expertise in immunology and data analysis.

Currently, computationally savvy immunologists are often self-taught, making it easy to cultivate bad habits and be swayed by a 'method of the week'. Collaborations with bioinformaticians are important, but without an expert with hybrid training, there can be delays, mix-ups, and failures to recognize data inconsistencies or unanticipated discoveries. Thus, immunology trainees would benefit from a structured curriculum designed to master computational immunology problems.

Learning Objectives in Stanford's CSI Track

The Stanford PhD program in Immunology offers two tracks: MCTI and CSI. Our CSI program accommodates incoming students skilled in either immunology or programming and provides an opportunity to cover the remaining material in year 1 (Figure 1A). In the first year, students in both MCTI and CSI tracks rotate in research laboratories and take many of the same courses. In addition to traditional MCTI course work that builds expertise in immunology (with a reduction of one core course and one MCTI elective), CSI students obtain computational skills through courses in computer science, statistics, and bioinformatics. As we do not select a track until the end of the first year, undecided students are able to try CSI courses and a computational rotation without committing to the track. In the second year, all students define a thesis project, take a qualifying exam, and serve as teaching assistants. CSI students also complete the remaining core courses, including a newly designed series focused specifically on computational immunology problems (Figure 1B). The series consists of seminars and courses focused on understanding, practicing, and solving problems specific to computational immunology (https://med.stanford.edu/immunol/phd-program/resources/curriculum.html). After building a strong foundation through coursework and defining a thesis project, CSI students take two electives in years 3+ to refine project-specific skills, which typically include courses in advanced statistics, bioinformatics, or machine learning.

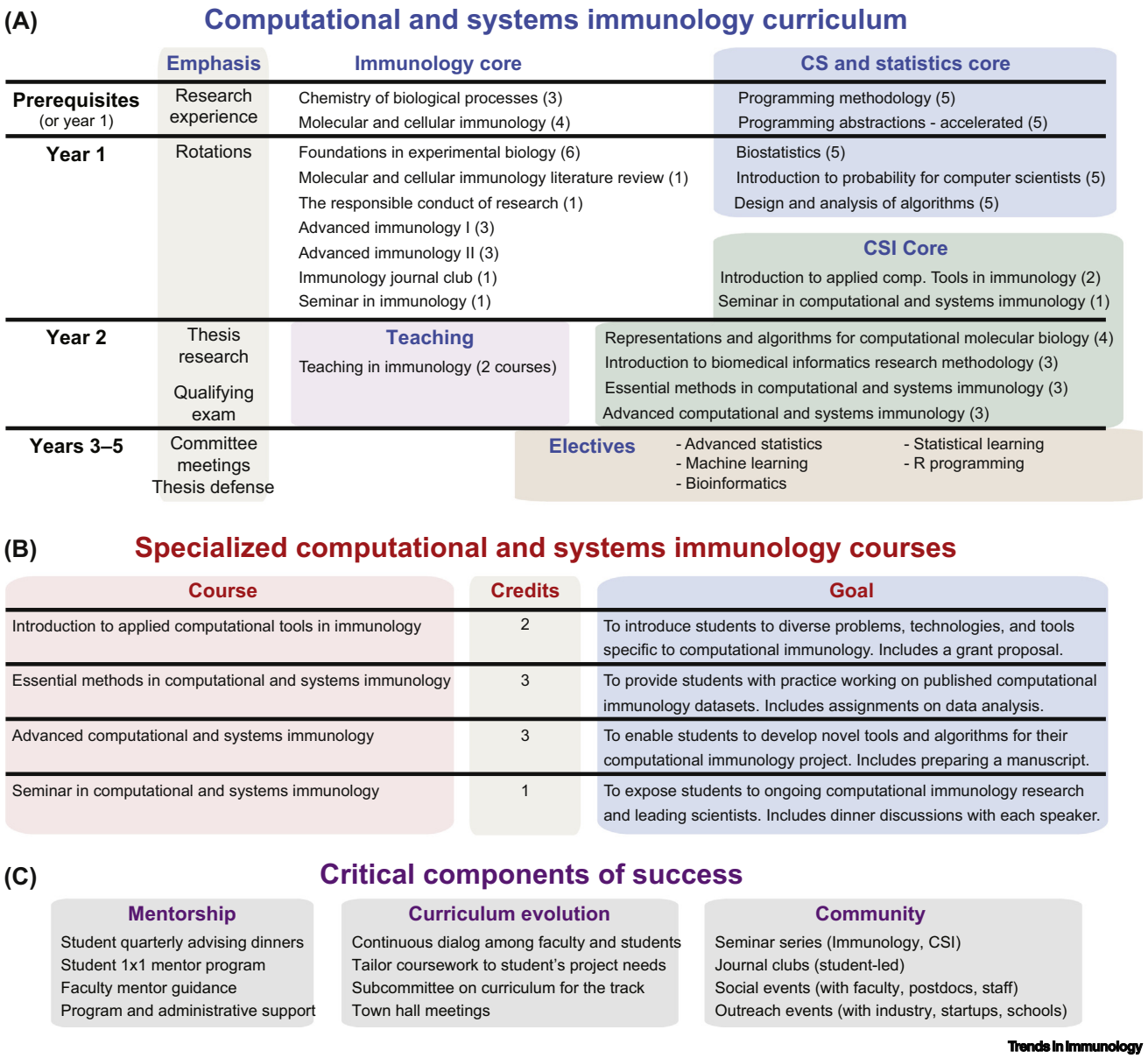

Figure 1. Stanford's Computational and Systems Immunology (CSI) Curriculum and Critical Components. (A) CSI track timeline, research emphasis, and coursework, as of 2018–2019. The number of credits for each course are shown in parentheses. This curriculum has evolved since 2012 and may be developed further (see Curriculum Evolution). (B) Specialized CSI courses. Full descriptions are available on https://med.stanford.edu/immunol/phd-program/resources/curriculum.html. (C) Critical components that have enabled the success of the CSI track at Stanford University. Abbreviations: CS, computer science.

Since 2012, 11 of 57 students began this training, and three (J.G., M.H.G., and Z.G.) have graduated. At this time, the CSI track takes 4–5 years (mean 4.6 years); comparable, or perhaps slightly less than a graduation mean of 5.7 years for the Stanford Immunology PhD program. In addition, students focusing on computational immunology prior to the formation of the CSI track also graduated in 4–6 years. These limited data suggest that students in both MCTI and CSI tracks graduate in a similar timeframe despite the intensive CSI coursework. Finally, computational immunology training has resulted in innovative published work [7–10,12,15].

Recommendations to Future Computational Immunology Programs

Based on Stanford's program, in this Scientific Life article, we advise other institutions to foster a culture of continuous learning through coursework (if not already incorporated). Specifically, creating a

formal computational immunology program enables students to take a heavy course load without taking time away from research. We posit that the time invested in immunology, computer science, and CSI courses has so far yielded a high return by accelerating thesis research. Other interdisciplinary PhD programs (e.g., biomedical informatics) have also shown that combining coursework requirements across multiple subject areas is certainly feasible.

Next, we recommend developing a rigorous core curriculum. The core course work should emphasize advanced immunology, statistics, computer science, and dynamic modeling. A set of dedicated courses should then cover specialized problems, data formats, and methodologies specific to computational immunology (e.g., CSI core; Figure 1B). Flexible electives can maximize the relevance of coursework to student theses. Thus, in addition to fostering support for students taking courses, developing an effective track requires faculty to dedicate time to developing and teaching these new courses.

To reduce the burden on existing faculty and to provide sufficient guidance to students, institutions should be encouraged to recruit additional faculty and postdocs working in computational immunology. Another possibility is a joint mentorship of a given student by both experimental and computational faculty members. For example, of the immunology students who started their PhD program between 2012 and 2017, 8/11 (73%) CSI students were coadvised, in contrast to 7/46 (15%) MCTI students. Although co-mentorship offers numerous benefits to a student (advisor support, access to laboratory expertise and resources, and more independence), arguably it has some drawbacks (twice the laboratory-related activities, and potentially conflicting expectations from coadvisors). Thus, although co-mentoring can be beneficial, a decision as to whether a student should be co-mentored needs to be carefully considered on a case-by-case basis.

A successful program relies on the self-motivation of graduate students, but also requires providing access to advising, as needed (Figure 1C). Program advisors might potentially tailor an individual student's curriculum to ensure that incoming PhD students are challenged at just the right level, learn skills directly applicable to their work, and hence, be able to graduate in a reasonable time. Finally, a continuous two-way dialog between students and faculty is key to ensuring collaborative and continuous improvements in the curriculum and training.

Building a Computational Immunology Community

Aside from the course work, building a scientific community within Stanford's CSI track was a key to its early success (Figure 1C). In the program, senior students advise junior students through formal quarterly advising dinners, one-on-one mentoring programs, and informally. Community support ensures that nobody 'falls through the cracks', and individual stories help break the 'it's too late for me to learn how to code' barrier. Postdocs, faculty, and administrators set norms, advise students, and provide the necessary resources and support. Through CSI seminars, students learn about the emerging computational immunology concepts and obtain career advice during informal dinner discussions with each speaker.

Concluding Remarks

Within the past decade, big (voluminous), deep (high-dimensional), and multiomics data have become commonplace in immunology and other areas of biomedical research, such as neuroscience, cancer biology, and developmental biology. As a new generation of scientists, we are expected to build and utilize these resources to pose and answer outstanding questions in immunology. We are thus grateful to programs such as Stanford's new CSI track, preparing us for the future. We strongly encourage other immunology graduate programs to offer a similar option to their students. The concepts presented here may be applicable to designing extended education programs for clinical fellows, postdocs, and other trainees.

Acknowledgments

We thank David R. Glass, Geoffrey T. Ivison, Aditya M. Rao, Graham L. Barlow, Erika Bongen, Sean C. Bendall, Garry P. Nolan, Patricia Jones, Olivia M. Martinez, Nikesh Kotecha, Parag Mallick, Dara M. Strauss-Albee, Matthew H. Spitzer, Michael H. Birnbaum, Bryan J. Xie, Eden Maloney, Sarah Kongpachith, Daniel Lu, Thomas Keller, Dmitry Tebaykin, Otavio B. Good, Jonathan E. Wosen, Maureen Panganiban, Andrew J. Gentles, and Robert Tibshirani for insightful discussions and comments on the earlier versions of this manuscript. We also acknowledge Atul Butte, Patricia Jones, Olivia M. Martinez, Nikesh Kotecha, and Parag Mallick for their roles in helping to develop Stanford's CSI track. The PhD Program in Immunology at Stanford Immunology is supported by NIH training grant 5T32 AI007290-34. Z.G. and M.M.D are members of the Parker Institute for Cancer Immunotherapy, which supported this work.

[1]Computational and Systems Immunology track, PhD Program in Immunology, Stanford University, Stanford, CA, USA
[2]Baxter Laboratory in Stem Cell Biology, Stanford University, Stanford, CA, USA
[3]Department of Microbiology and Immunology, Stanford University, Stanford, CA, USA
[4]Department of Pathology, Stanford University, Stanford, CA, USA
[5]Department of Molecular and Cellular Physiology, Stanford University, Stanford, CA, USA
[6]Department of Medicine, Center for Biomedical Research, Stanford University, Stanford, CA, USA
[7]Current address: Stanford Cancer Institute and Department of Biomedical Data Science, Stanford, CA, USA
[8]Current address: Distributed Bio, South San Francisco, CA, USA
[9]Current address: 3T Biosciences, Menlo Park, CA, USA

*Correspondence:
zinaida@stanford.edu (Z. Good) and
pkhatri@stanford.edu (P. Khatri).

https://doi.org/10.1016/j.it.2019.06.006

References

1. Davis, M.M. *et al.* (2017) Systems immunology: just getting started. *Nat. Immunol.* 18, 725–732
2. Spreafico, R. *et al.* (2015) Training the 21st century immunologist. *Trends Immunol.* 36, 283–285
3. Schultze, J.L. (2015) Teaching 'big data' analysis to young immunologists. *Nat. Immunol.* 16, 902–905

4. Andres-Terre, M. *et al.* (2015) Integrated, multi-cohort analysis identifies conserved transcriptional signatures across multiple respiratory viruses. *Immunity* 43, 1199–1211
5. Chowdhury, R. *et al.* (2018) A multi-cohort study of the immune factors associated with *M. tuberculosis* infection outcomes. *Nature* 560, 644–648
6. Scott, M.K.D. *et al.* (2019) Increased monocyte count as a cellular biomarker for poor outcomes in fibrotic diseases: a retrospective, multicenter cohort study. *Lancet Respir. Med.* 7, 497–508
7. Glanville, J. *et al.* (2017) Identifying specificity groups in the T cell receptor repertoire. *Nature* 547, 94–98
8. Han, A. *et al.* (2014) Linking T-cell receptor sequence to functional phenotype at the single-cell level. *Nat. Biotechnol.* 32, 684–692
9. Gee, M.H. *et al.* (2018) Antigen identification for orphan T cell receptors expressed on tumor-infiltrating lymphocytes. *Cell* 172, 549–563 e516
10. Gee, M.H. *et al.* (2018) Stress-testing the relationship between T cell receptor/peptide-MHC affinity and cross-reactivity using peptide velcro. *Proc. Natl. Acad. U. S. A.* 115, E7369–E7378
11. Schiebinger, G. *et al.* (2019) optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell* 176, 928–943 e922
12. Good, Z. *et al.* (2019) Proliferation tracing with single-cell mass cytometry optimizes generation of stem cell memory-like T cells. *Nat. Biotechnol.* 37, 259–266
13. Villani, A.C. *et al.* (2017) Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* 356 Published online April 21, 2017. https://doi.org/10.1126/science.aah4573
14. An, G. *et al.* (2017) Optimization and control of agent-based models in biology: a perspective. *Bull. Math. Biol.* 79, 63–87
15. Good, Z. *et al.* (2018) Single-cell developmental classification of B cell precursor acute lymphoblastic leukemia at diagnosis reveals predictors of relapse. *Nat. Med.* 24, 474–483

Spotlight

Transposons to V(D)J Recombination: Evolution of the RAG Reaction

Michael R. Lieber[1,*]

Evolutionarily, how RAG endonucleases in vertebrate immune systems could shed dangerous transposon-like propensities, and instead, support the organized assembly of antigen receptor variable domains, has been unclear. Recent structural work by Schatz and colleagues (Nature, 2019) identifies features of the RAG endonuclease deemed to be key in supporting this critical change in vertebrate advancement.

RAG endonuclease complexes in vertebrates cut into the antigen receptor genomic locus at **variable domain** (see Glossary) segments. These cuts lead to the formation of the variable domain exon that encodes the variable domain binding pocket of antibodies. This assembly requires cutting the DNA at the edges of the variable (V), diversity (D), and joining (J) gene segments in a manner directed by heptamer–nonamer nucleotide **recombination signal sequences (RSSs)**, bearing either a 12- or 23-bp spacer [1]. This endonuclease action is identical to the direct transesterification chemistry by which the transposases of all transposons function (Figure 1).

However, the choreography of the DNA ends varies significantly for the RAG reaction in vertebrates relative to transposons, which is why antigen receptor gene rearrangement superficially appears to be so different from a transposon excising from one genomic location and integrating into a new DNA location. For antigen receptor gene rearrangement in vertebrate B and T cells, RAG cutting had to change in fundamental ways.

First, the V, D, and J coding segment DNA ends could no longer be repaired by an assortment of pathways, as donor ends are handled when prokaryotic and eukaryotic transposons move from one location to another [2,3] (Figure 1). Instead, in the vertebrate immune system, the DNA ends of these V, D, and J segments must be generated and repaired in a consistent manner (with a terminal DNA hairpin), which is then processed by the nonhomologous DNA end joining (NHEJ) repair pathway, yielding what is called **junctional diversity** [4]. The strategy of junctional

Glossary

Amphioxus ProtoRAG transposase: each transposon genome encodes its own enzyme (transposase), which cuts the transposon DNA out of the DNA in which it was integrated, and then inserts it somewhere else. The amphioxus (lancelot) ProtoRAG transposase appears to be the forerunner of the RAG enzyme in all jawed vertebrates.
Artemis:DNA-PKcs: hairpin opening at the V, D, or J coding ends is achieved by the structure-specific endonuclease Artemis, activated by the kinase DNA-PKcs. The kinase activity of DNA-PKcs is activated when it binds to a DNA end, including DNA hairpinned ends. Artemis and DNA-PKcs form a tight protein complex. Artemis and DNA-PKcs appear on the evolutionary stage roughly at the invertebrate to vertebrate transition, suggesting that efficient hairpin opening might have been a key selective advantage towards generating a diverse immune repertoire.
Junctional diversity: diversity in the immune system in vertebrates consists of two facets: combinatorial diversity due to the rearrangement of different V, D, and J segments with one another; and junctional diversity due to base-pair addition and removal, intrinsic to the NHEJ process.
Recombination signal sequence (RSS): DNA sequence at which a RAG enzyme binds (RAG1 or RAG2). It consists of two parts: a palindromic heptamer and an AT-rich nonamer, separated by either a 12- or 23-bp spacer. One V(D)J recombination reaction requires one RSS with a 12- and another with a 23-bp spacer, also known as the 12/23 rule.
Variable domain: portion of the heavy or light chain of an antibody (or either chain of the T cell receptor) that forms the binding pocket allowing it to bind to an antigen (or a peptide–MHC complex in the case of T cell receptors). The variable domain does not exist as such in the genome, but rather must be assembled from V, D, and J segments (heavy chain) or V and J segments (light chain). The assembly occurs via RAG endonuclease double-strand break formation at RSS sites (see main text) adjacent to the V, D, and J segments. The joining of the DNA ends requires opening of the RAG-generated coding end hairpins by the Artemis:DNA-PKcs complex and then joining by the NHEJ double-strand break repair pathway.