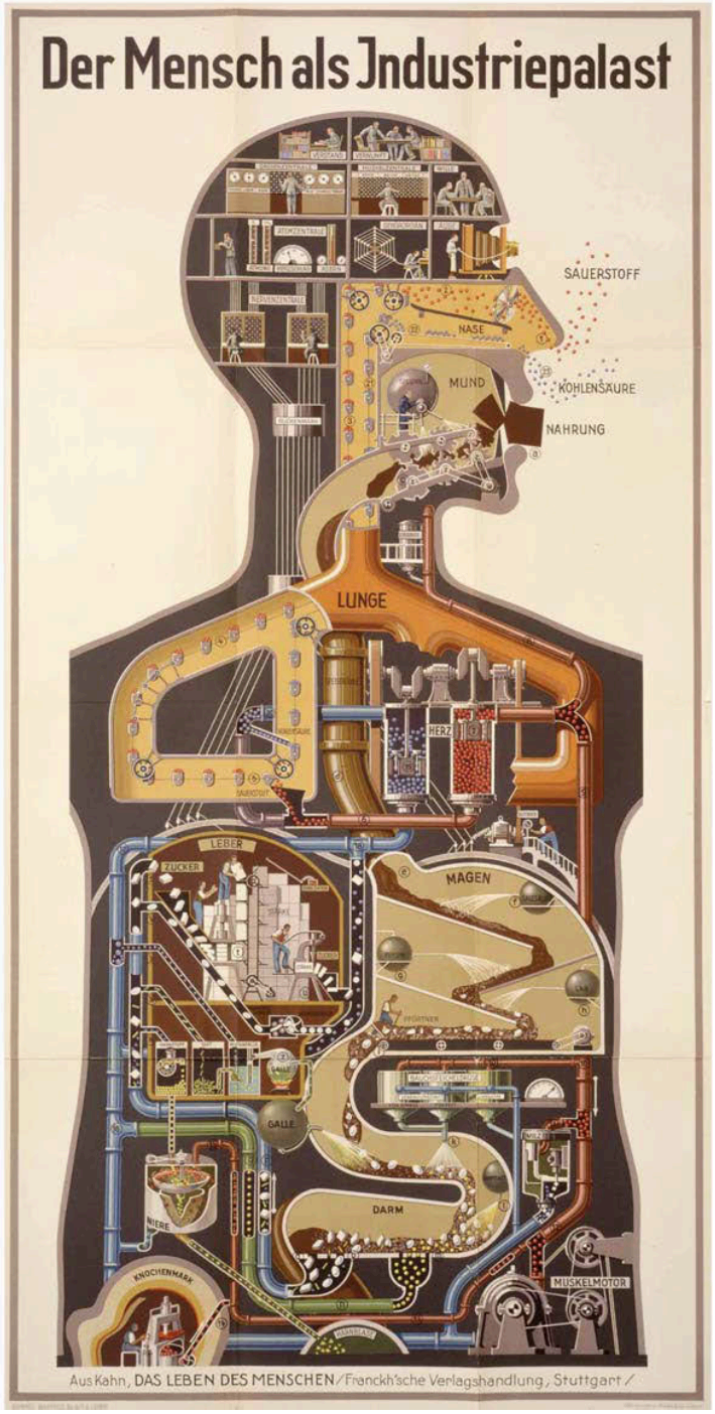


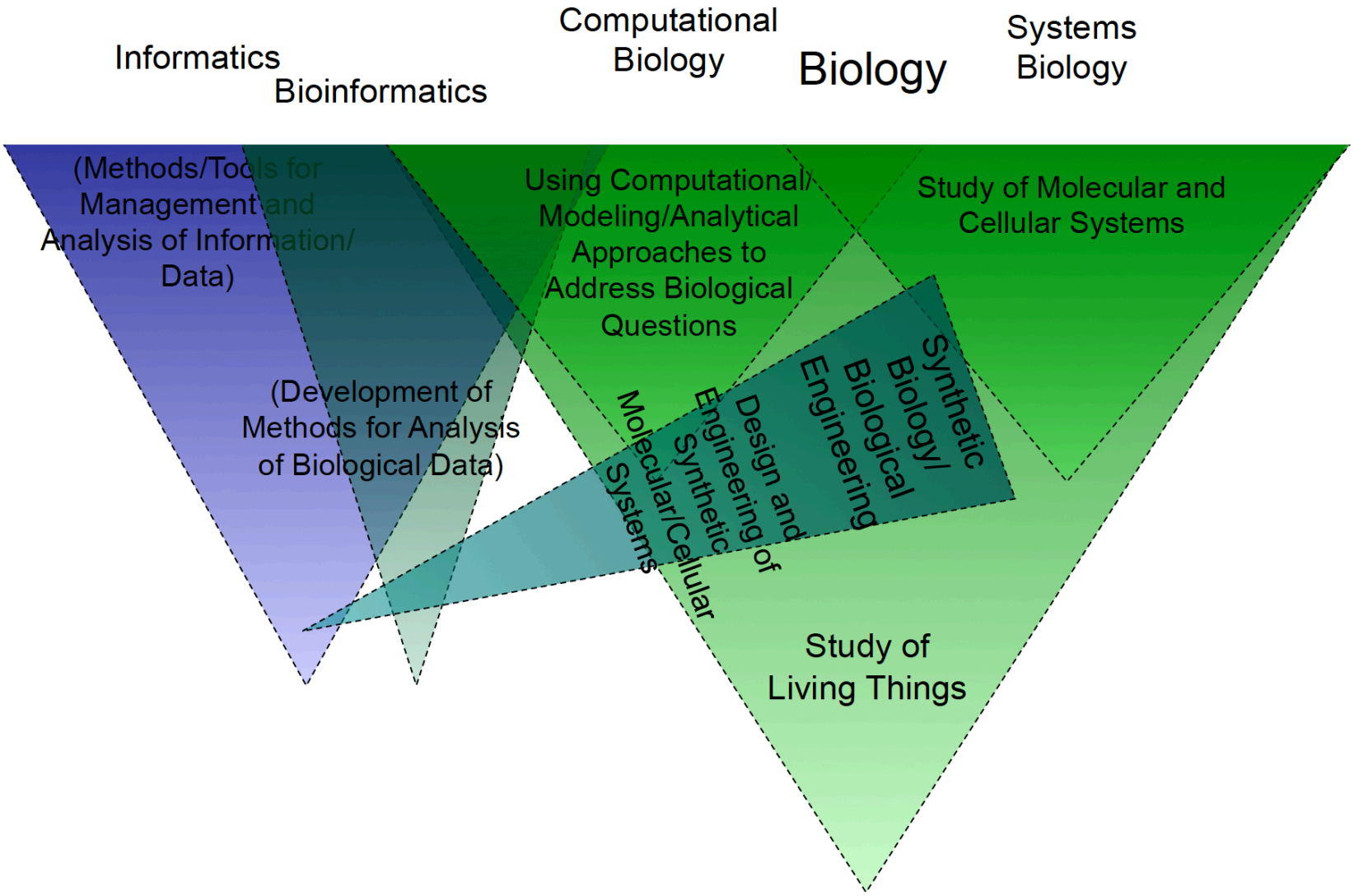
COMP 5970/6970: Computational Biology

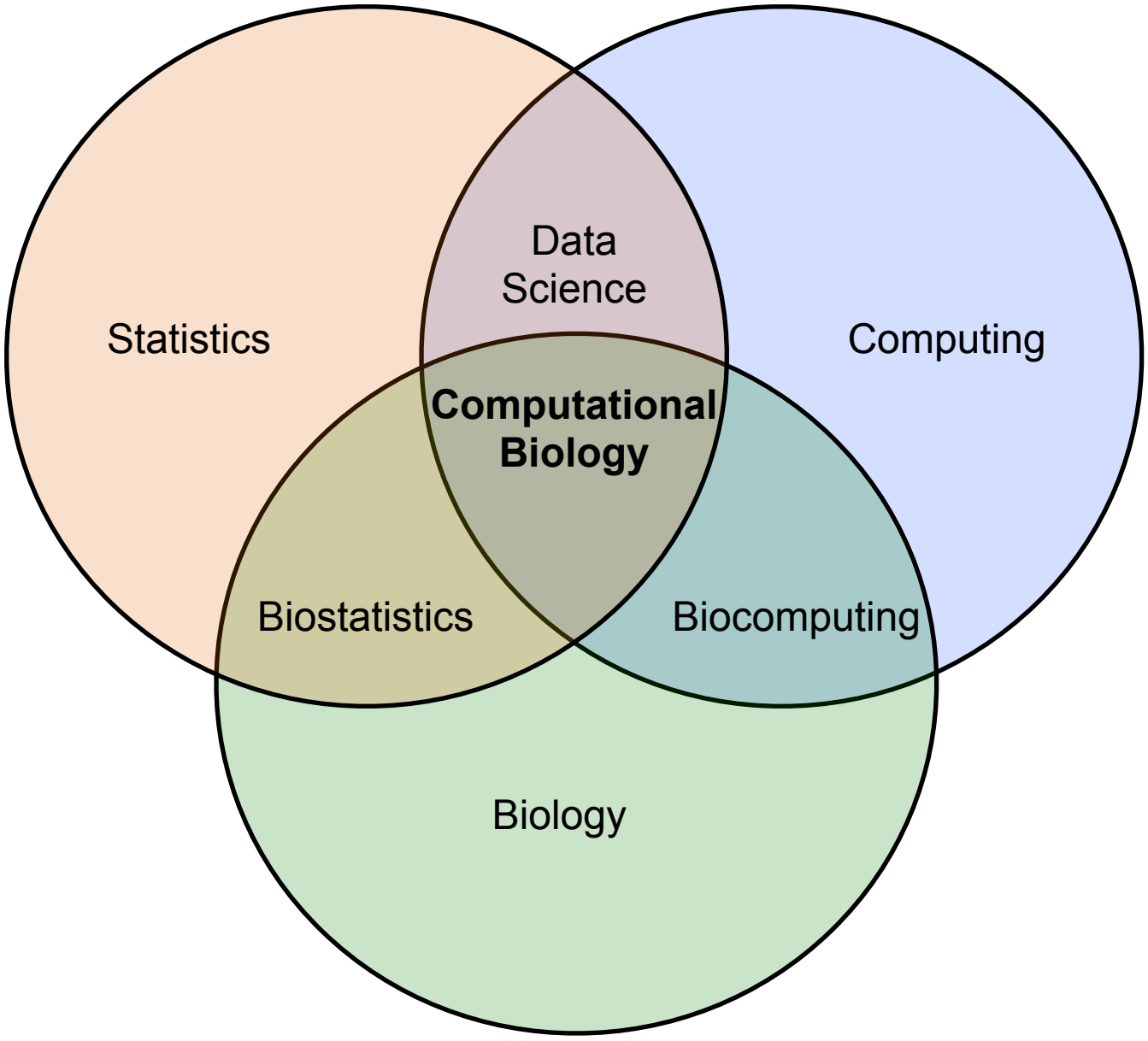
Debswapna Bhattacharya, PhD
Assistant Professor
Computer Science and Software Engineering
Auburn University

What, Why, How?

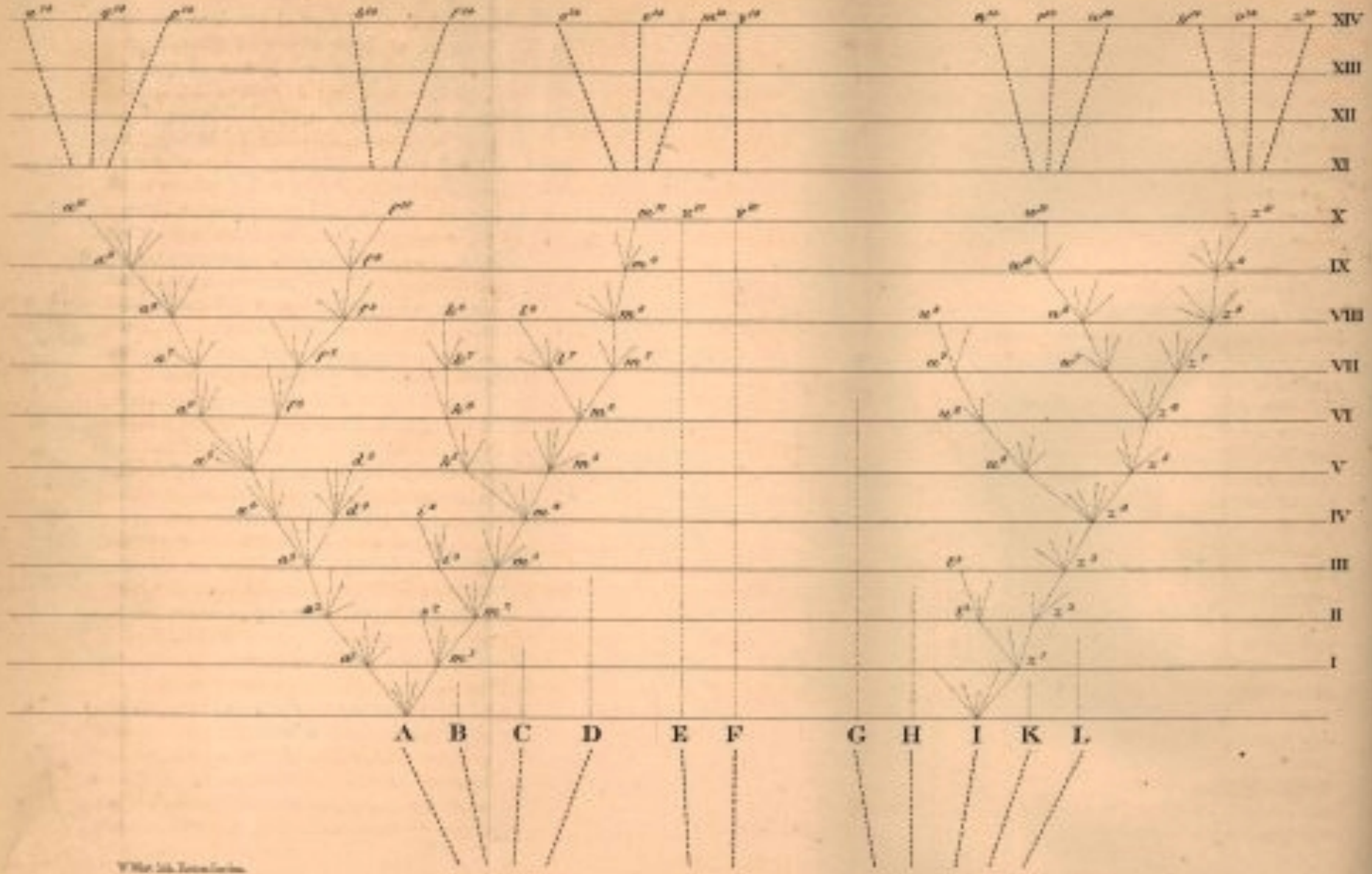
What?

Overlapping Fields

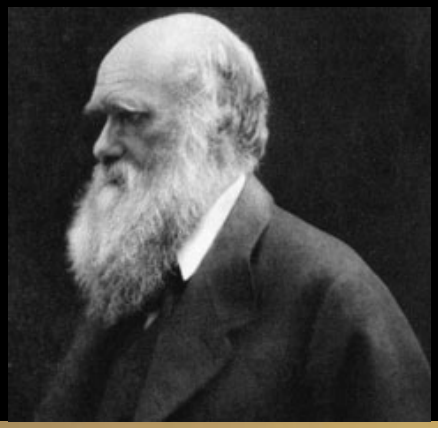



Statistics
Data
Science
Computing
Computational
Biology
Biostatistics
Biocomputing
Biology

The Inception …



“Principle of Divergence”



ON

THE ORIGIN OF SPECIES

BY MEANS OF NATURAL SELECTION,

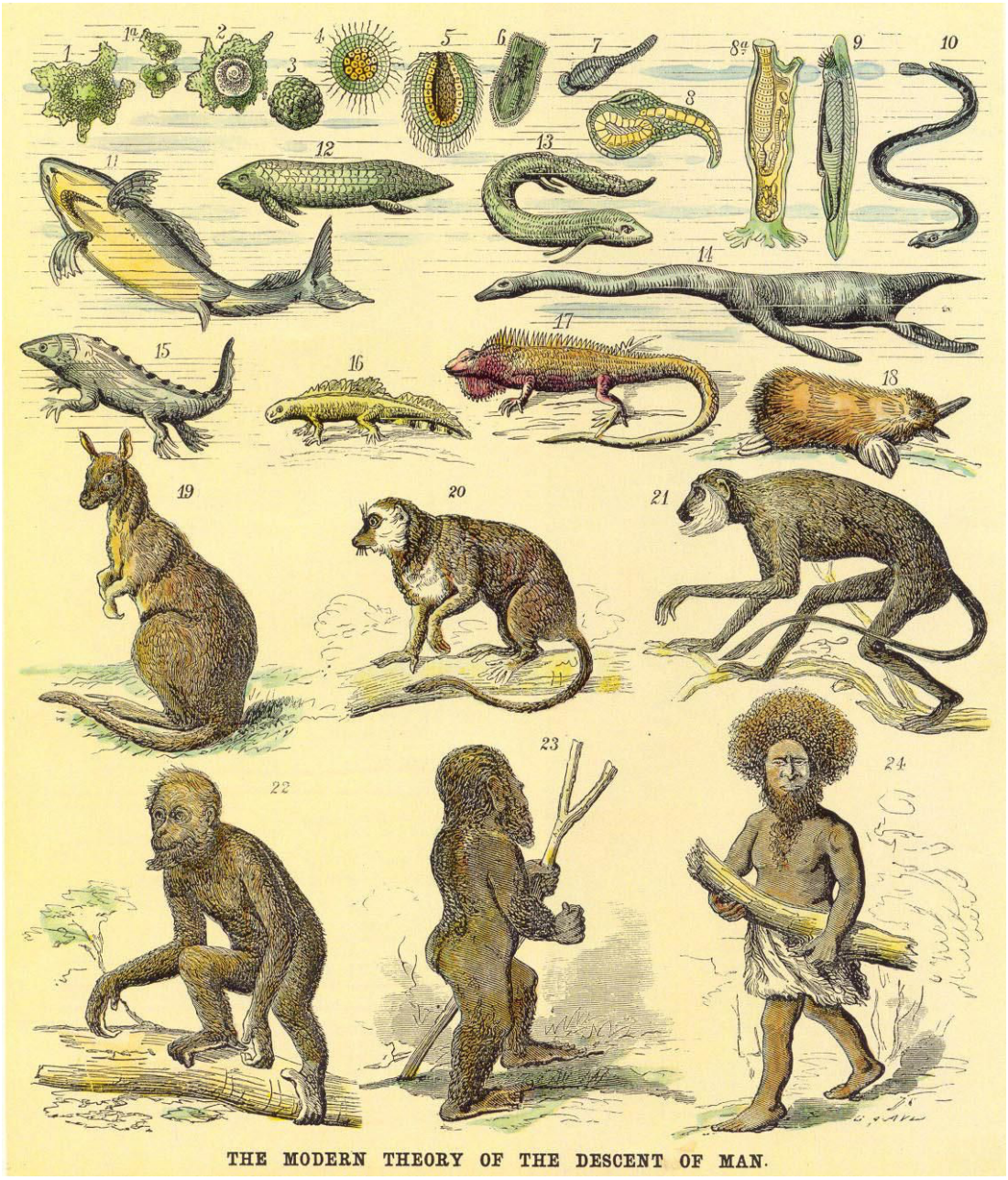
OR THE

PRESERVATION OF FAVOURED RACES IN THE STRUGGLE FOR LIFE.

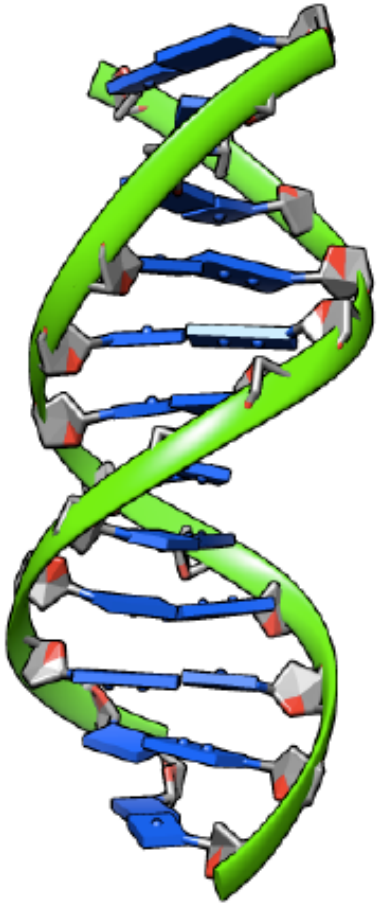
By CHARLES DARWIN, M.A.,

1859

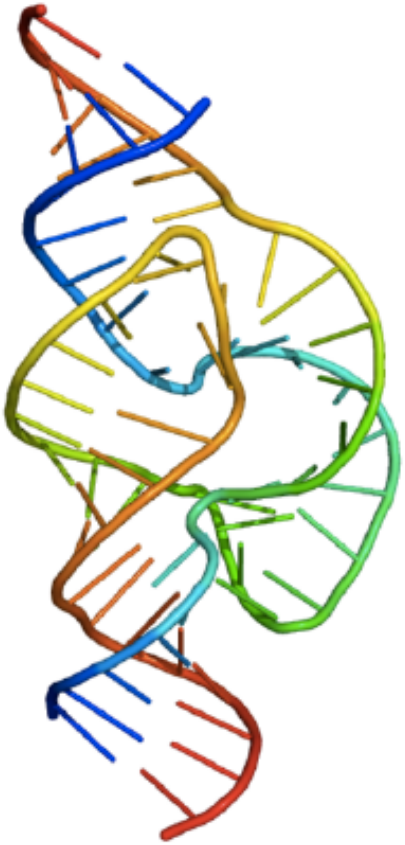
Life at the Macro Level



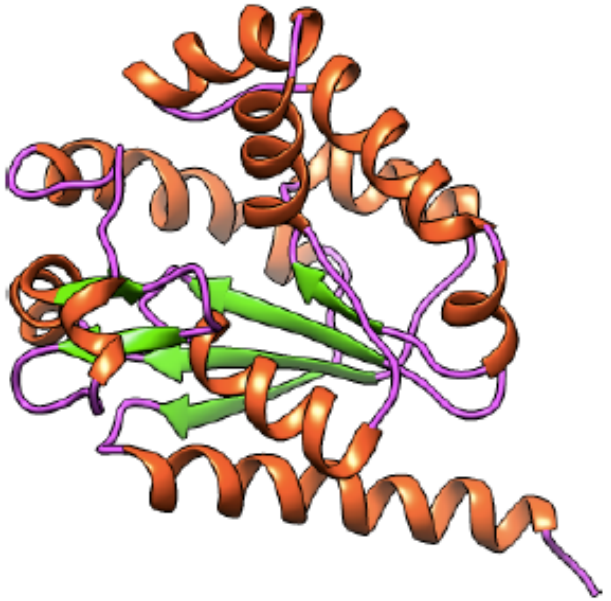
Life at the Molecular Level



DNA



RNA



Protein

The Inner Life of the Cell: BioVisions @Harvard U

https://www.youtube.com/watch?v=wJyUtbn0O5Y

A Brief, Anecdotal History

The 1970s and Earlier: Molecular Evolution Sequence Databases, Similarity Matrices,…

How do protein sequences evolve?

How should similarity between two proteins be scored to most accurately detect homology?

- First protein sequence databases / protein family classification
- PAM matrices for protein sequence comparisons (still used!)

What can molecular sequences tell us about organismal evolution?

- Molecular classification of life
- Molecular clocks

The 1980s: Sequence Alignment/Search

Which specific residues/positions in a pair of proteins are homologous?

- Smith-Waterman alignment algorithm

What RNA secondary structure has minimum folding free energy?

- Nussinov algorithm
- Zuker algorithm

Politicians Learn to Search PubMed

NCBI Director David Lipman (far left) coaches Vice President Gore (seated) as he searches PubMed.
NIH Director Harold Varmus (center) and NLM Director Donald Lindberg (far right) look on.

The ’90s: HMMs, Ab Initio Protein Structure Prediction, Genomics, Comparative Genomics

How to identify domains in a protein?

How to identify genes in a genome?

Hidden Markov Models as a framework for such problems

How to study gene expression globally, infer gene function from expression?

- Microarrays and clustering

How to predict protein function by comparing genomes?

- gene fusions, phylogenetic profiling, etc.

How to predict protein structure directly from primary sequence?

- Rosetta algorithm

The 2000s Part 1:

The human genome is sequenced, assembled, annotated

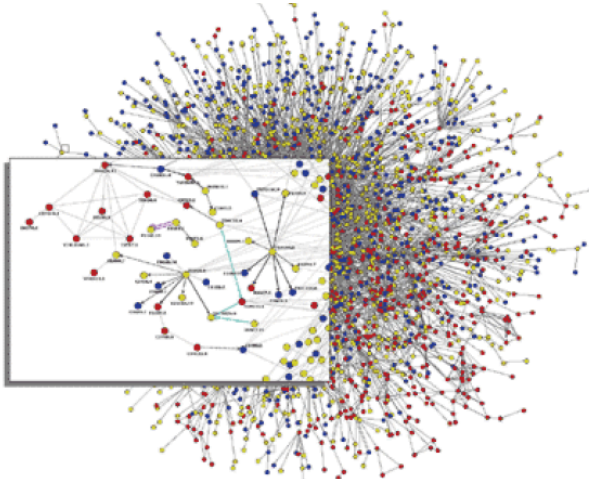
genomics becomes fashionable

16 Feb, 2001

15 Feb, 2001

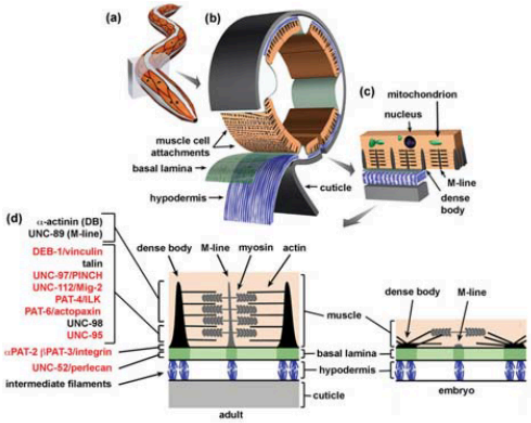
The 2000s Part 2: Biological Experiments Become High-Throughput, Computational Biology Becomes more Biological

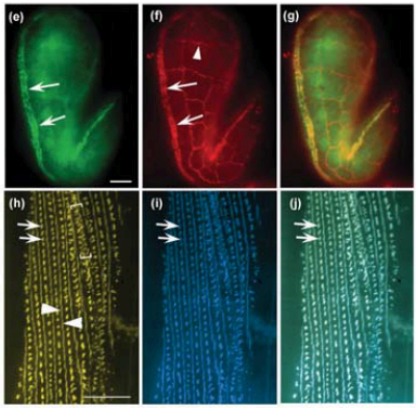
Massively parallel data collection - transcriptomics, proteomics, interactomics, metagenomics



Using sequence and array data to address fundamental questions about transcription, splicing, microRNAs, translation, epigenetics, protein structure/function, development, evolution, disease, etc.

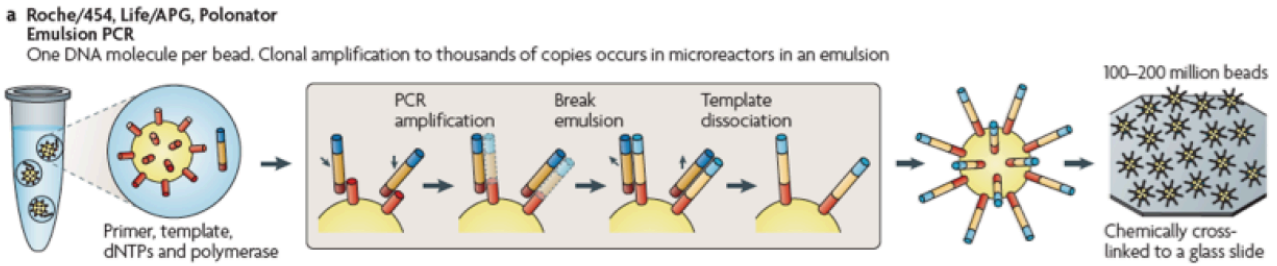
Integrated computational/experimental approaches

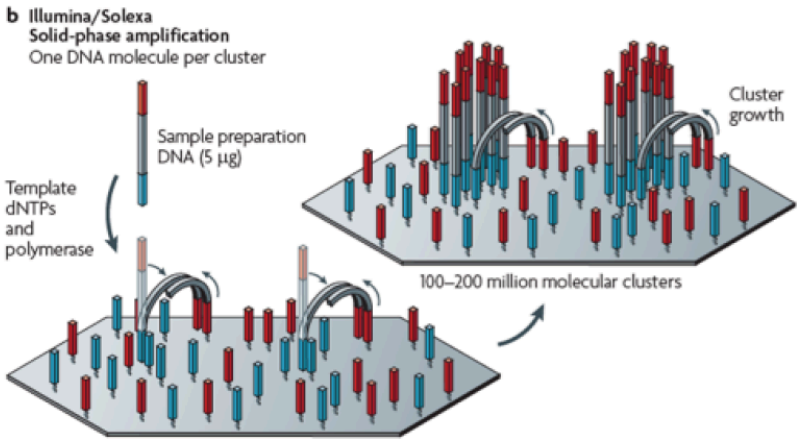




Rise of bioimage informatics

Late 2000s / Early 2010s





Next-gen sequencing finds applications across biology

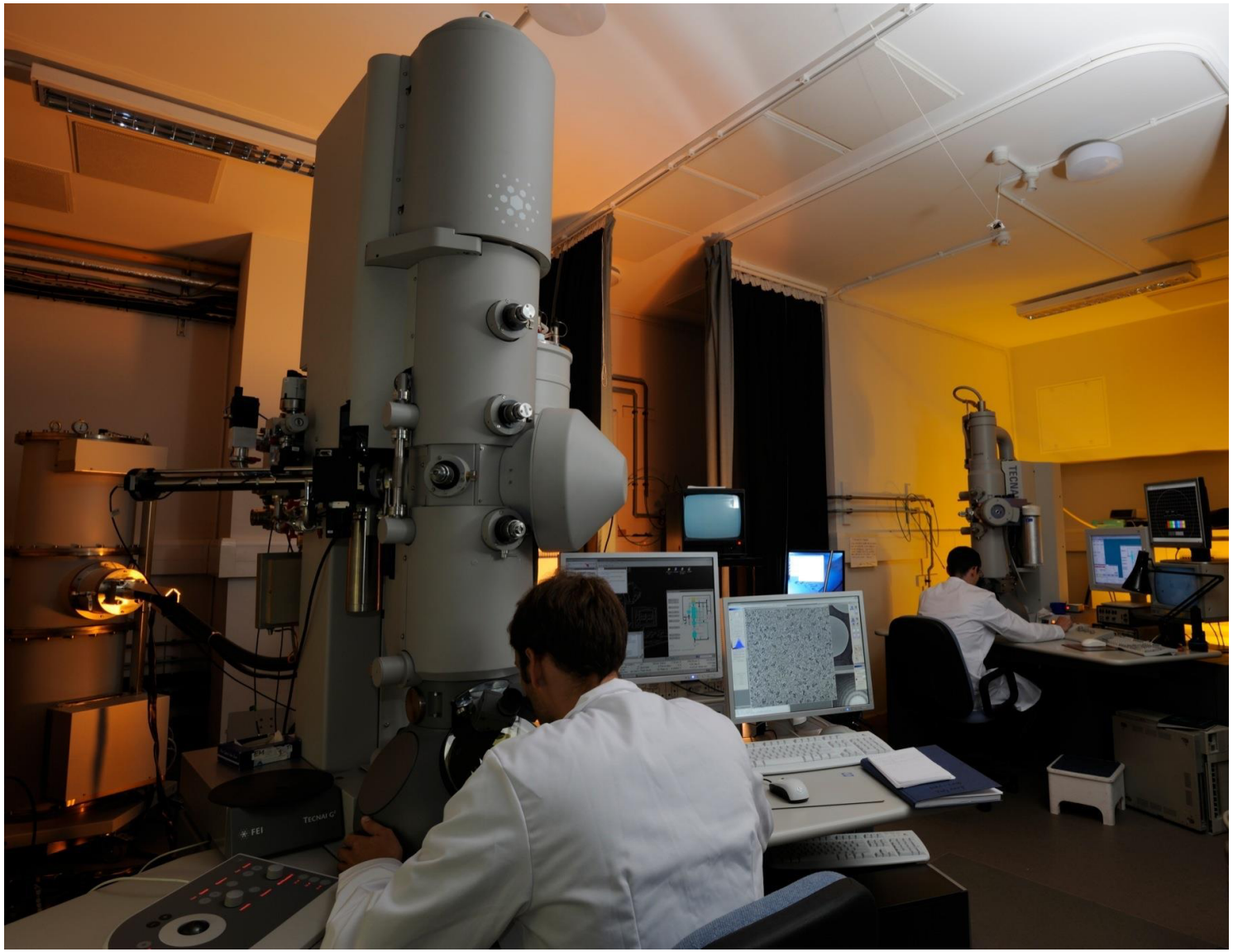
Genome sequencing

Transcriptome sequencing

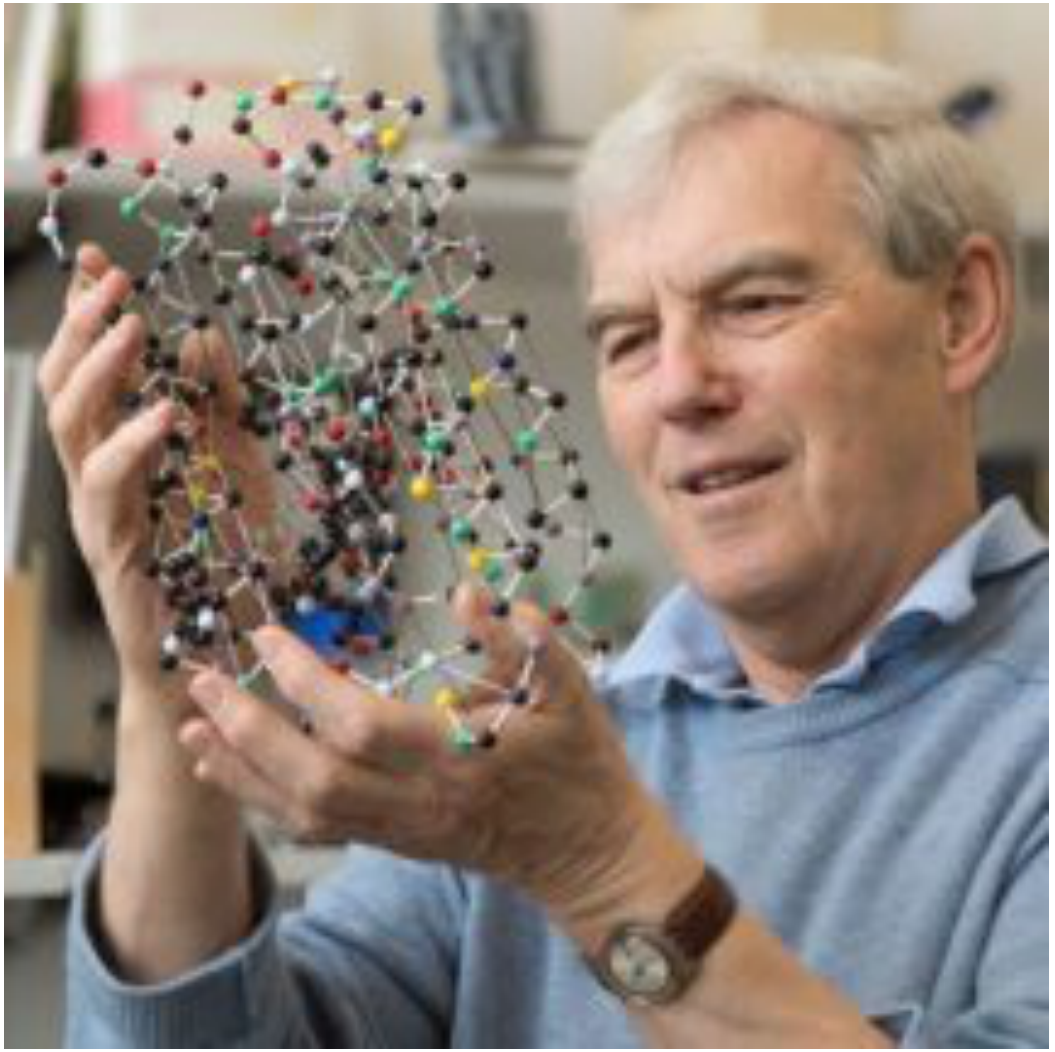
Protein-DNA intrxns (ChIP-seq)

Protein-RNA intrxns (CLIP-seq)

Very Recently : Cryo-electron microscopy for high-resolution structure determination of biomolecules in solution

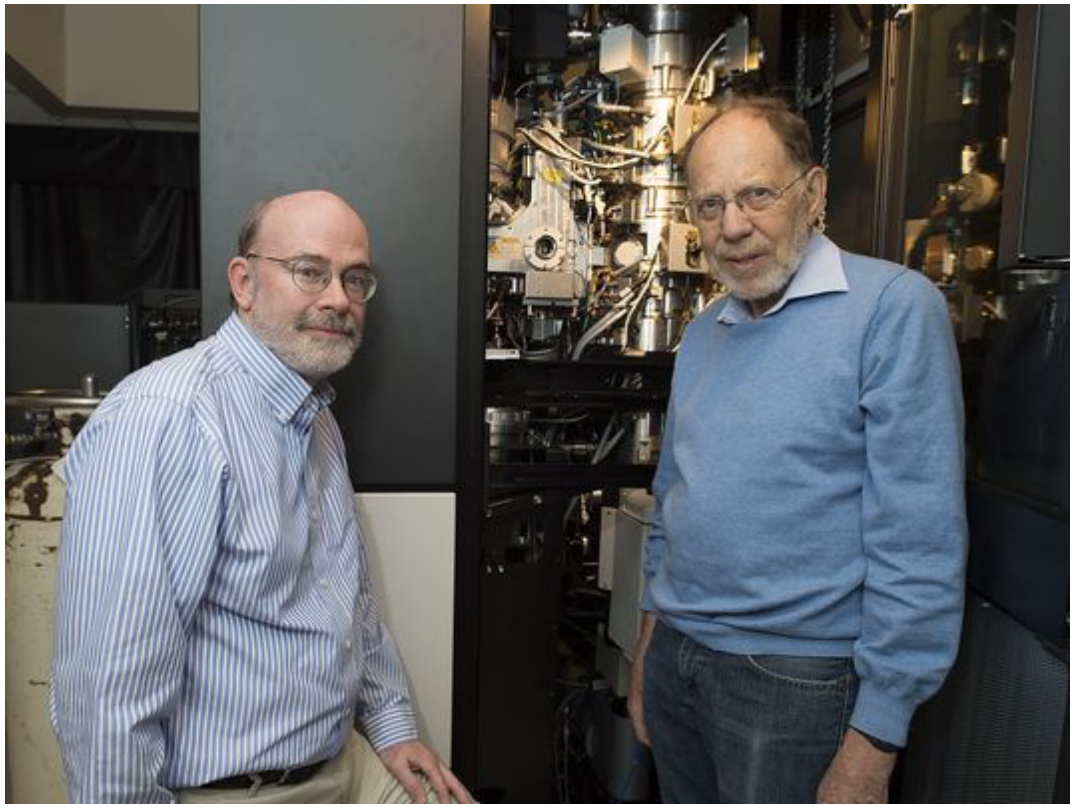


2017

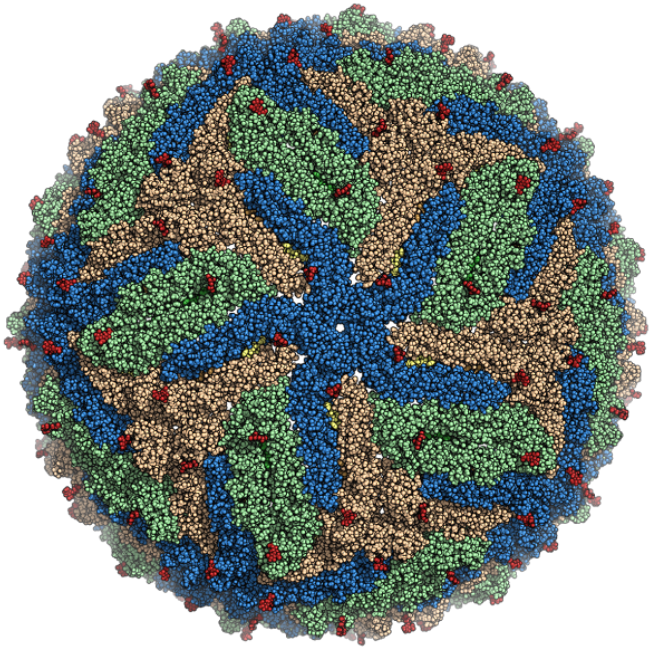


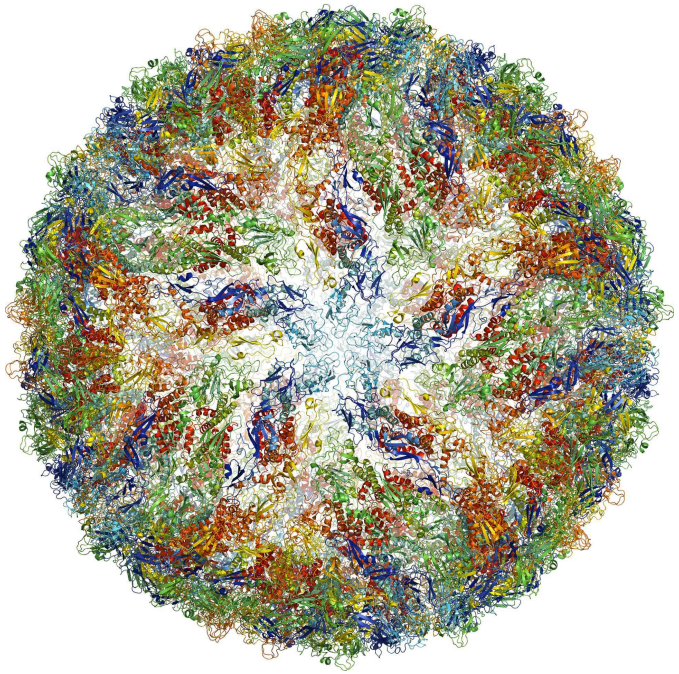
Richard Henderson @ LBM wins 2017 Nobel Prize for Chemistry

2018

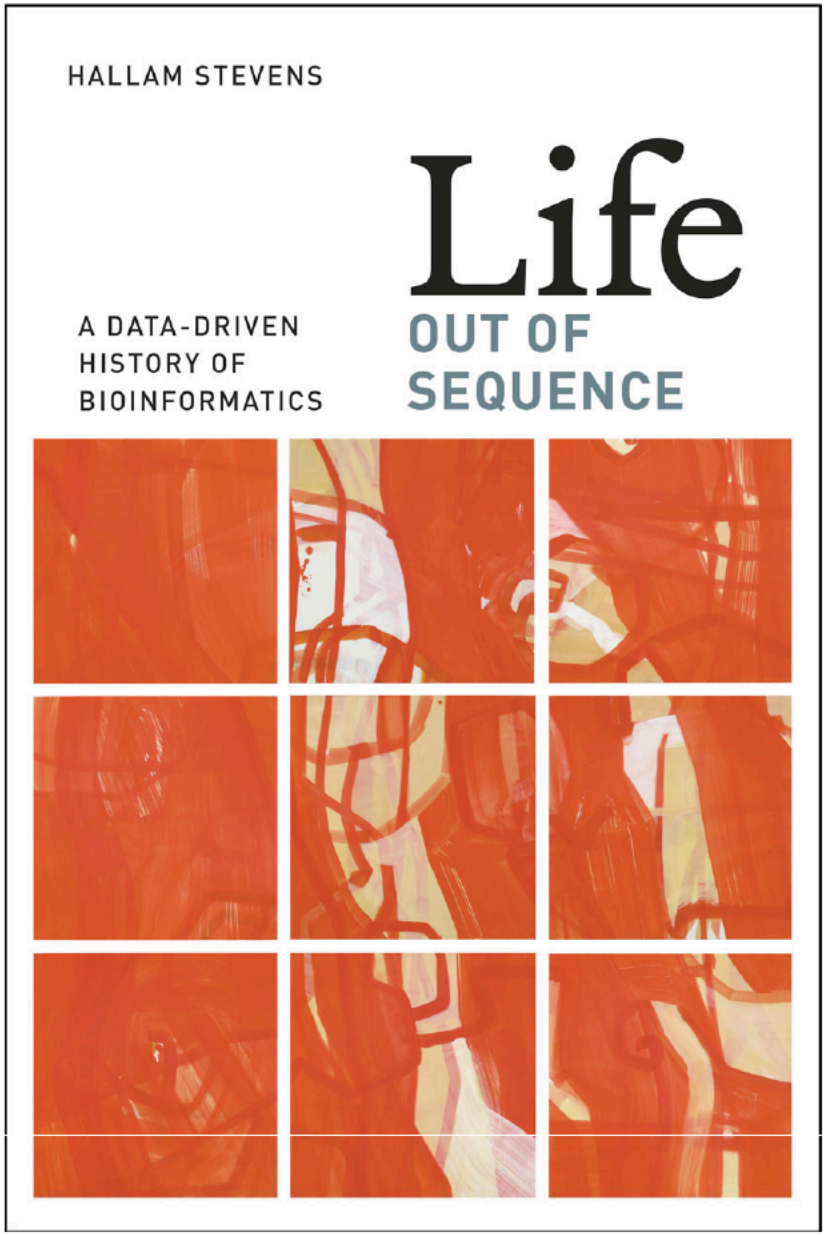


Michael G. Rossmann @ Purdue solves atomic level structure of Zika virus





For those who would like a proper history of the field



Why?

We are living in a Genomic Era

The Genomic Era

Genome Sequencing Revolution

Analysis Revolution

Medicine Revolution

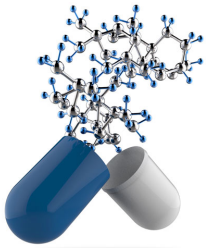
10 years ago

TODAY

the future

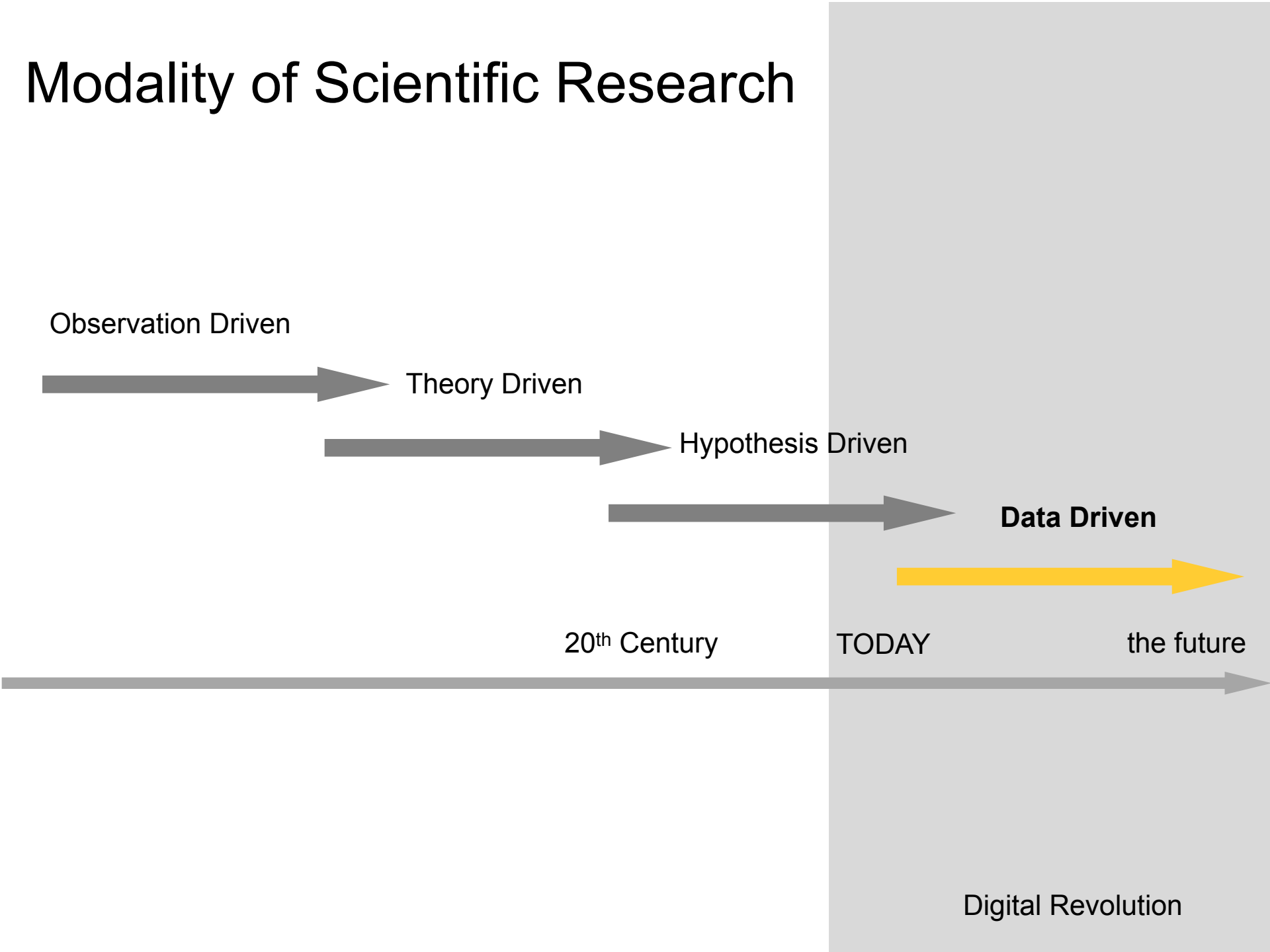
No prognosis
Treat symptoms

General preventive recommendations



Personalized medicine

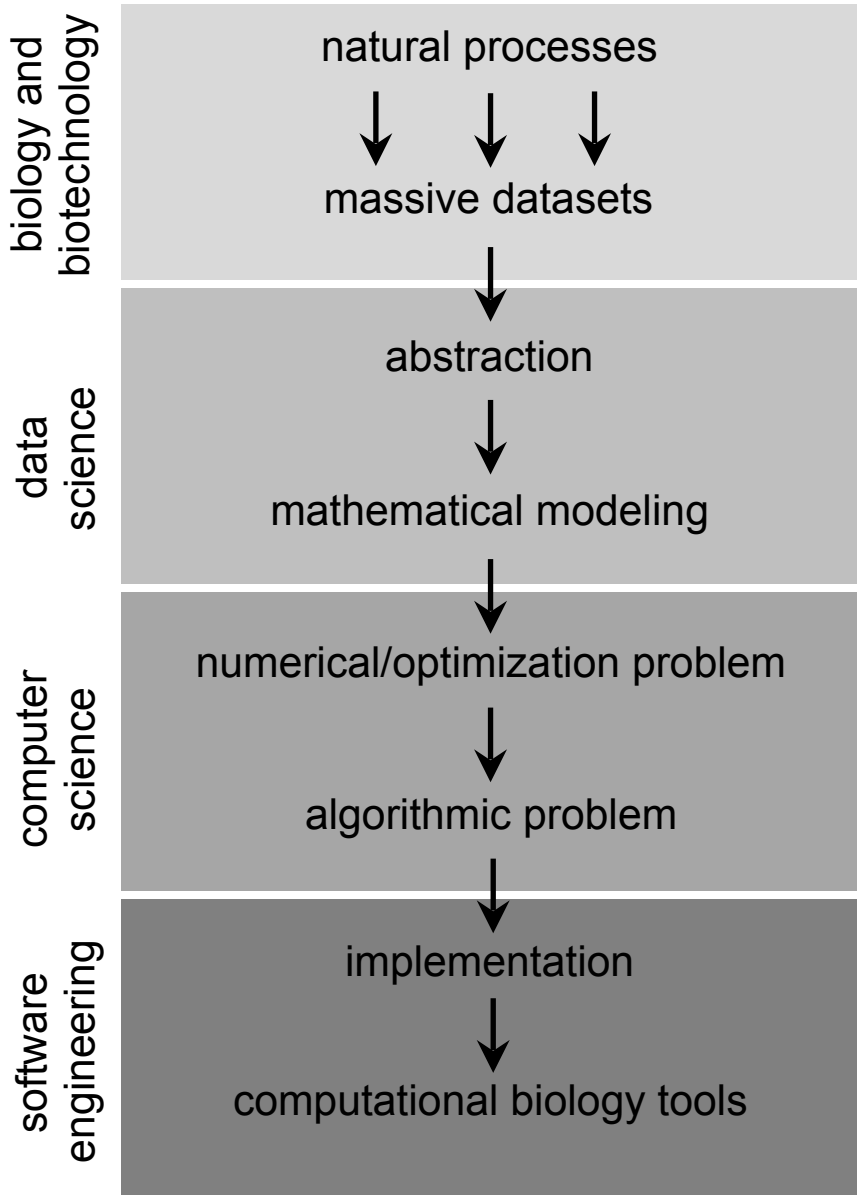
Modality of Scientific Research

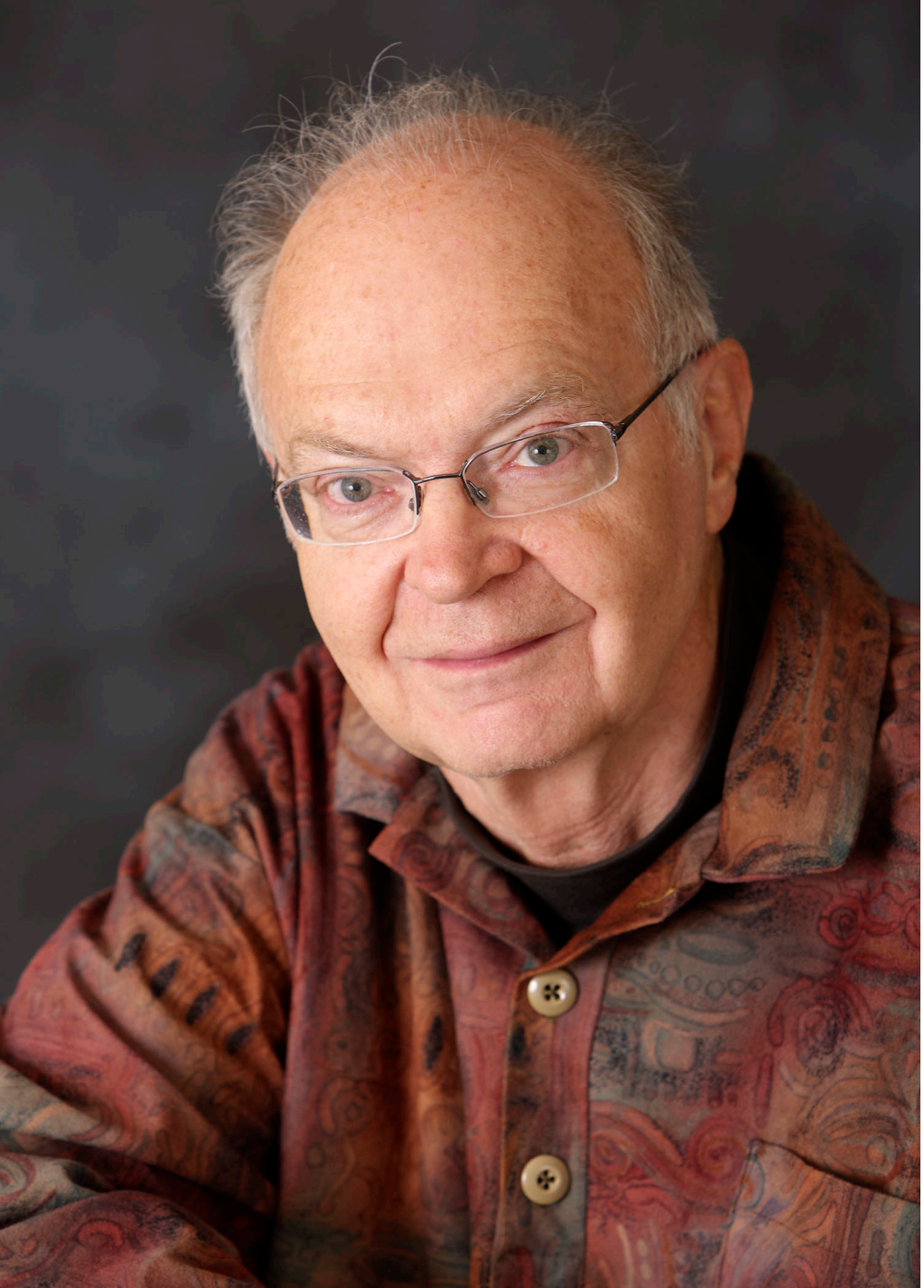


How?

Computational Biology:
Information theoretic way to analyze and cognize massive biological data

Computational Biology Workflow





“

I can’t be as confident about computer science as I can about biology. Biology easily has 500 years of exciting problems to work on. It’s at that level.

”

Donald Knuth

Course Logistics

What exactly will we study?

- We will study theories of Algorithms, Machine Learning, Data Science:
  - Dynamic Programming
  - Inductive Inference, Function Approximation
  - Decision Trees
  - Probabilistic Estimation: MLE, MAP
  - Bayesian Inference
  - Logistic Regression
  - Stochastic Optimization
  - Linear Regression
  - Graphical Models
- And their applications to solve computational biology problems
  - Sequence Analysis
  - Solvent Exposure
  - Secondary Structure
  - Spatial Contacts
  - 3D Structural Similarity

Prerequisite

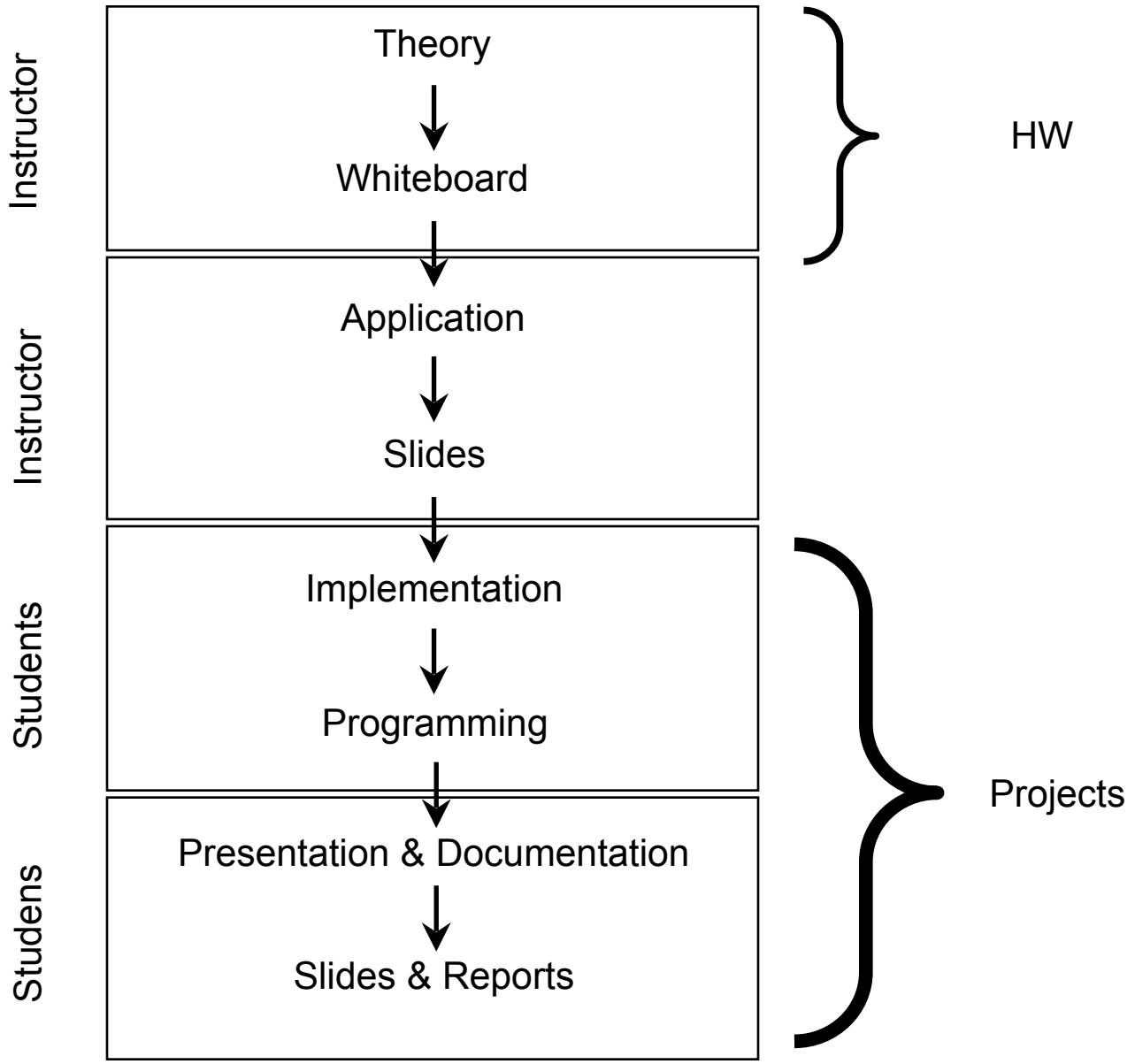
The prerequisites for this course will be a course in algorithms (COMP 3720 or equivalent) or a strong background in biology.

Textbook

NONE!

The lecture notes provided will serve as the primary reference. Reference materials will be provided as appropriate.

This course follows an Active Learning Approach



Homeworks

Theoretical in Nature

Will test your understanding of the materials…No memorization is needed and will NOT be time consuming.

Programming Language

Python

A brief intro to Python programming will be provided…

We will use standard Python only. Special purpose libraries (e.g. scikit-learn, biopython) are not allowed.

The four parts in detail

- Theory

  Mathematical concepts of complex systems, deterministic and probabilistic interference, stochastic optimization, their related algorithms, and their tradeoffs.

- Application

  The related computational biology problems cherry picked to match the theory just covered.

- Implementation

  Application of the theory to solve and analyze interesting computational biology problems.

- Presentation and Documentation

  Communicating effectively in verbal and written forms about your study, challenges faced and your findings.

Grading

NO Midterm Exams!

Not even finals!!

5 Homework Assignments (5% each)

5 Programming Projects (15% each)

Grading Algorithm

Grade(points: real) returns char

temp=ceiling(points)

if temp ≥ 90 then return "A"

else if temp ≥ 80 then return "B"

else if temp ≥ 65 then return "C"

else if temp ≥ 50 then return "D"

else if temp < 50 return "F"

What will I learn?

- Understand the theories of Algorithms, ML, Data Science relevant to Computational Biology
- Given a computational biology problem to solve, be able to specify it at a level that allows you to develop a computational solution
- Given a problem abstraction, be able to come up with advanced computational algorithms for solving it
- Be able to turn those algorithms into correct and efficient computer programs and evaluate its performance
- Analyze and interpret the results
- Present your findings to an audience
- Document your study in the form of a technical report

What should you expect?

Somewhat abstract theoretical material that requires knowledge about algorithms, probability, and statistics…

Mathematical…but not rocket science!

Substantial exposure to programming (Python)

Not easy (so plan to spend a lot of time and keep up with the materials; if not, you won't be able to catch up later)

How to get help?

GTA
Md Hossain Shuvo
Email: mzs0149@auburn.edu
Office: Shelby 3136
Office Hours: TR 9:30 am -10:30 am

GTA is the Primary Point of Contact for Grading

Instructor
Debswapna Bhattacharya, PhD
Email: bhattacharyad@auburn.edu
Office: Shelby 3104
Office Hours: MWF 10:00 am -11:00 am

Late Submission Policy

No Late Submission!

Submissions marked late by Canvas and emailed submissions will get a zero.

Regrading

All regrading requests should be made to GTA **within one week the grades are posted**.

NOT to the instructor.

How do I make an A?

- Attend all classes
- Review the lecture notes after each class and before the next class — understand the concepts
- Participate and ask questions in class
- Start the projects early — DO NOT wait till the last minute
- Prepare clean slides for project presentation and practice your talk
- Document your project in a well-formatted report (template will be provided)
- Make use of office hours – instructor/TA available every day of the week
- **Do all HW assignments yourself**
- **Work regularly, not at the last minute!**

Attendance Policy

NONE!

I do not take attendance. But you need to show up during the project presentations.

Please do not...

send me an email like this (that actually happened):

“I finished with an 88 in your course. I am currently a Junior and have a 4.0 GPA. This would be my first B at Auburn. Is it possible that you can round my grade up to an A?”

“I am 2 percentage points away from a C in your class and would like to know if there is any way to round up my grade. I'm sorry to ask this of you, but I would greatly appreciate anything that can be done.”

I do not curve grades

My rules

Except in emergencies, I must be informed **beforehand** if you are going to miss something...post-hoc excuses not accepted!

Assignments must be turned in on time. **Submissions marked late by Canvas and emailed submissions will get a zero.**

If you want an assignment re-graded, you must ask **within** a week of when the grades are posted.

Cheating of any kind is unacceptable!

No food or drink in the class.

Cell phones off or vibrate mode…answer calls outside class

You may use laptops for taking notes or reviewing class materials; tweeting, facebooking, etc. will be injurious to your grade!

Sleeping is permitted but no snoring…

Syllabus in Canvas

Assignment Schedule (subject to change)	No late assignments will be accepted!
HW 1 Released	1.18
Due Date & Time (upload to Canvas)	1.25 11:59 PM
Project 1 Released	1.25
Due Date & Time (upload to Canvas)	2.13 11:59 PM
HW 2 Released	2.1
Due Date & Time (upload to Canvas)	2.8 11:59 PM
Project 2 Released	2.8
Due Date & Time (upload to Canvas)	2.27 11:59 PM
HW 3 Released	2.15
Due Date & Time (upload to Canvas)	2.22 11:59 PM
Project 3 Released	2.22
Due Date & Time (upload to Canvas)	3.8 11:59 PM
HW 4 Released	3.27
Due Date & Time (upload to Canvas)	4.3 11:59 PM
Project 4 Released	3.22
Due Date & Time (upload to Canvas)	4.10 11:59 PM
HW 5 Released	4.12
Due Date & Time (upload to Canvas)	4.19 11:59 PM
Project 5 Released	4.5
Due Date & Time (upload to Canvas)	4.24 11:59 PM