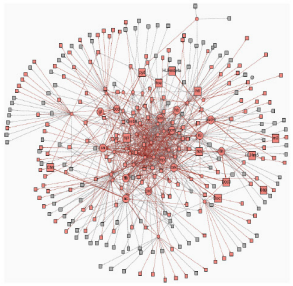
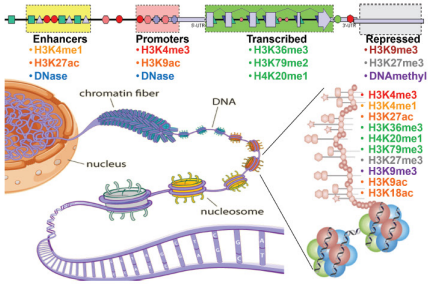
MIT 6.874/6.802/20.390/20.490/HST.506

Deep Learning in the Life Sciences

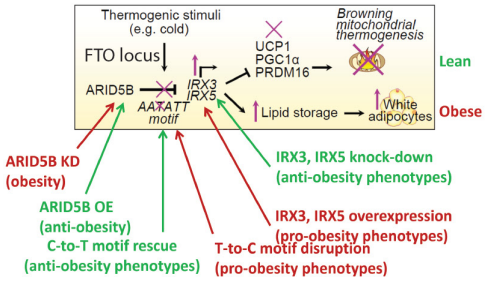
DNA motifs



Gene networks

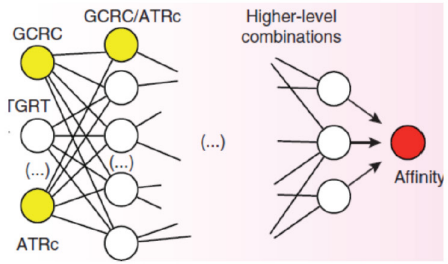


Epigenomics

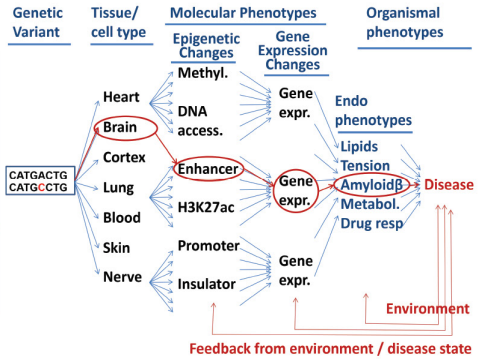


Manipulate disease circuitry

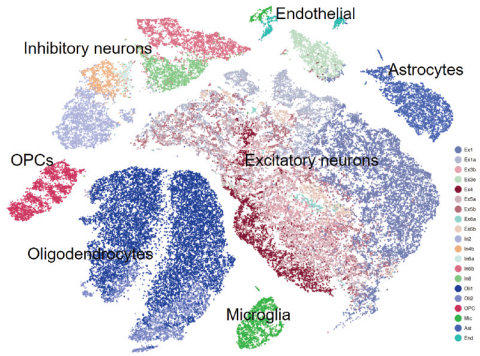
Multi-phenotype



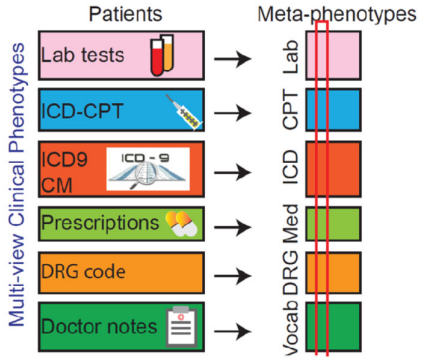
Deep learning



Mediation analysis/QTLs



Single-cell dissection



Medical record models

Prof. Manolis Kellis

Spring 2021

Course Logistics

This course was initially developed by
Prof. David K Gifford

Manolis & David co-taught it in Spring 2020
but David is on sabbatical this Spring

David
Gifford

Manolis
Kellis

Many of the lectures build on courses
that David and Manolis have taught for many years now
at the interface of machine learning and biology
(and many other theoretical and applied courses)

Course Staff and Meeting Times

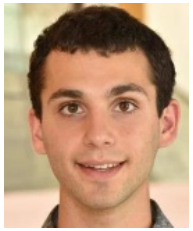
An introduction to the **application of machine learning** to tasks in the **life sciences**

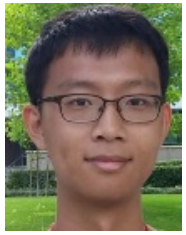
Subject components:
Lectures: Tuesday and Thursday 1pm - 2:30pm
Recitations: Friday 3pm
Mentoring sessions: Friday 4pm (most weeks, optional)
Office hours: posted on website


Manolis Kellis

Your teaching staff

- Manolis Kellis <manoli@mit.edu> (lectures)
- Dylan Cable <dcable@mit.edu>
- Zheng Dai <zhengdai@mit.edu>
- Tess Gustafson <tgust21@mit.edu>
- Jackie Valeri <valerij@mit.edu>


Dylan Cable


Zheng Dai


Tess Gustafson

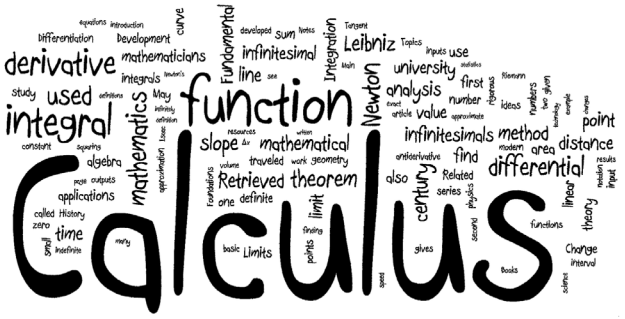

Jackie Valeri

Online resources
compbio.mit.edu/6874
mit6784.github.io (links to Piazza, Stellar, Tutorials)

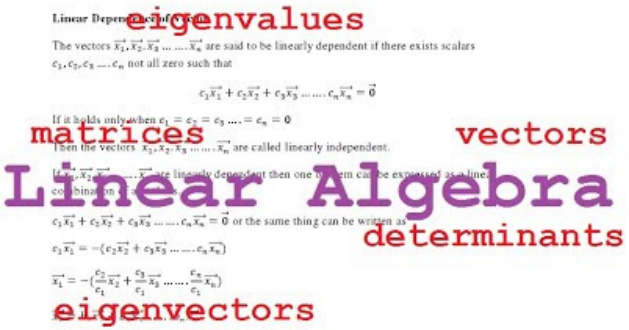
Requests to teaching staff
6.874staff@mit.edu

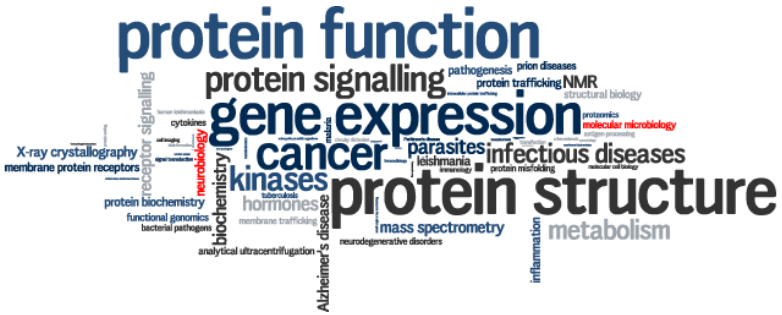
Your background

- Calculus, Linear Algebra
- Probability, Programming
- Introductory Biology




```
def loadstable(ver):
    return _loadversion(ver, prefix="_stable_")

def loadunstable(ver):{...}
def loadexact(ver):{...}
def _loadversion(ver, prefix):
    targetname = prefix + ver.replace('.', '_')
    mainpackage = _original__import__("tools", globals(), locals(),
        [targetname])
    global currentversion
    currentversion = getattr(mainpackage, targetname)

    # Let users change versions after choosing this one
    currentversion.loadstable = loadstable
    currentversion.loadunstable = loadunstable
    currentversion.loadexact = loadexact

    return currentversion

currentversion = None
```





Grade contributions

- Five Problem Sets (drop worst one) (30%)
  - Individual contribution
  - Done using Google Cloud, Jupyter Notebook
- One quiz (1.5 hours), one sheet of notes (25%)
- Final Project (35%)
  - Done in teams of 2-3 students
- Scribing + Participation (10%)
  - Mentoring sessions, brainstorming, guest lecturers, diving into primary literature

compbio.mit.edu/6874

mit6874.github.io

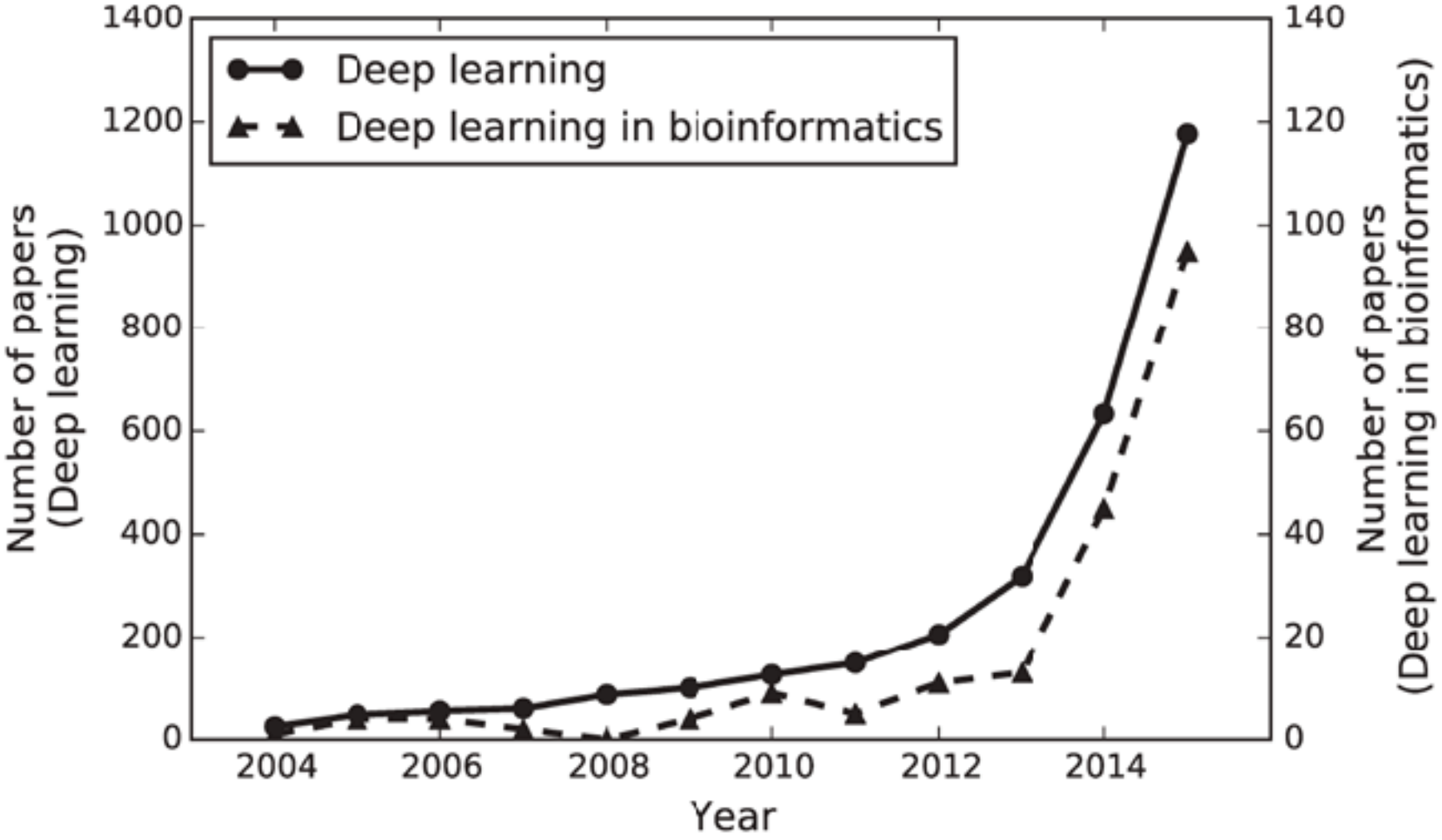
6.874staff@mit.edu

Please use Piazza or the staff email for any questions

You will be receiving your Google Cloud coupon URL in your email

Why Deep Learning in the Life Sciences

Approximately 8% of deep learning publications are in bioinformatics



Why Deep Learning in Life Sciences

- Enabled by the convergence of three things
  - Inexpensive, high-quality, collection of large data sets (sequencing, imaging, etc.)
  - New machine learning methods (including ensemble methods)
  - High-performance Graphics Processing Unit (GPU) machine learning implementations
- Result is completely transformative

Computational Biology Courses at MIT

- 6.047 / 6.878 Computational Biology: Genomes, Networks, Evolution
- 6.S191: Introduction to Deep Learning
- 6.S897/HST.956: Machine Learning for Healthcare (2:30pm 4-270)
- 8.592 Statistical Physics in Biology
- 7.09 Quantitative and Computational Biology
- 7.32 Systems Biology
- 7.33 Evolutionary Biology: Concepts, Models and Computation
- 7.57 Quantitative Biology for Graduate Students
- 18.417 Introduction to Computational Molecular Biology
- 20.482 Foundations of Algorithms and Computational Techniques in Systems Biology

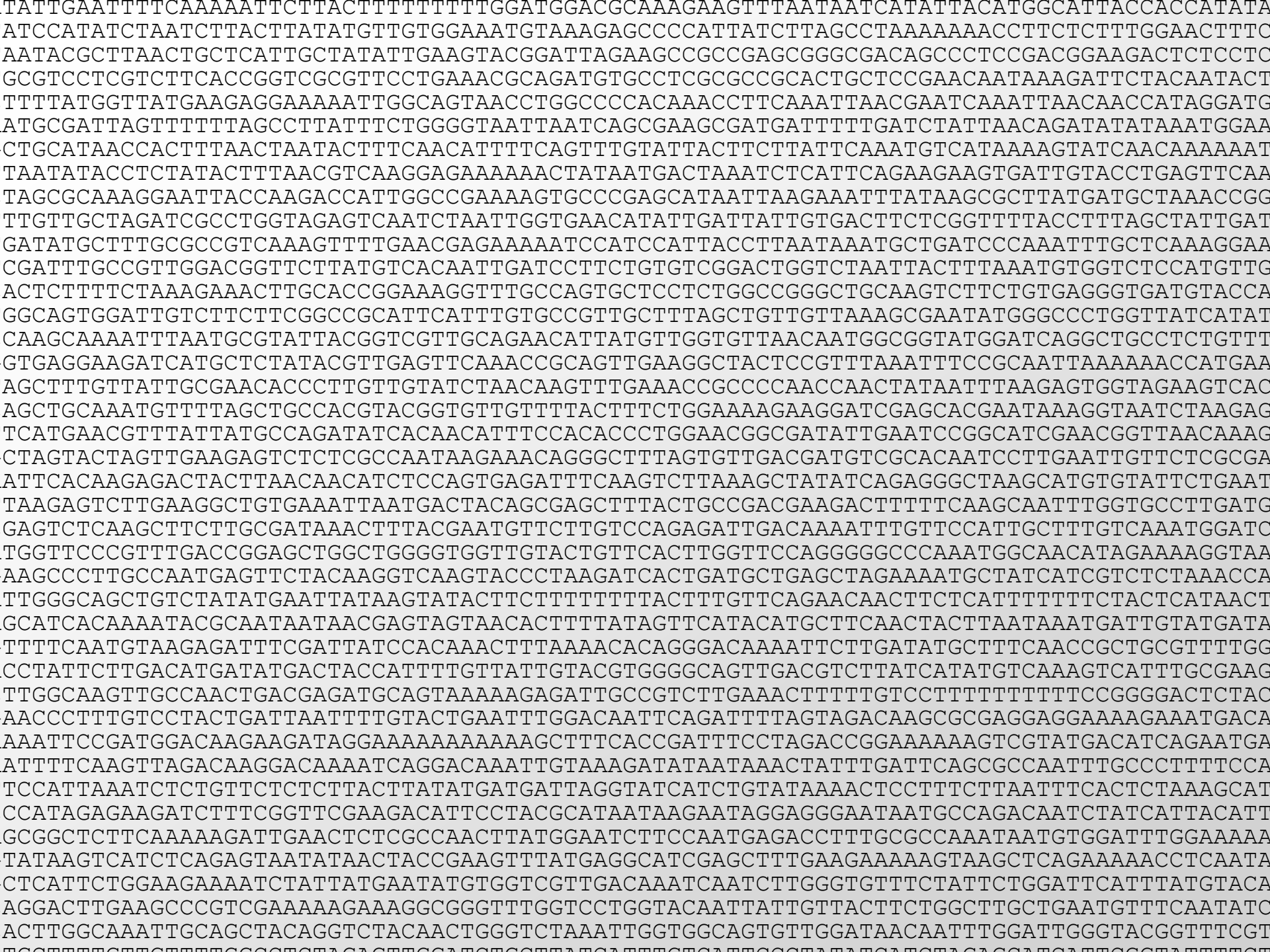
Why Computational Biology ?

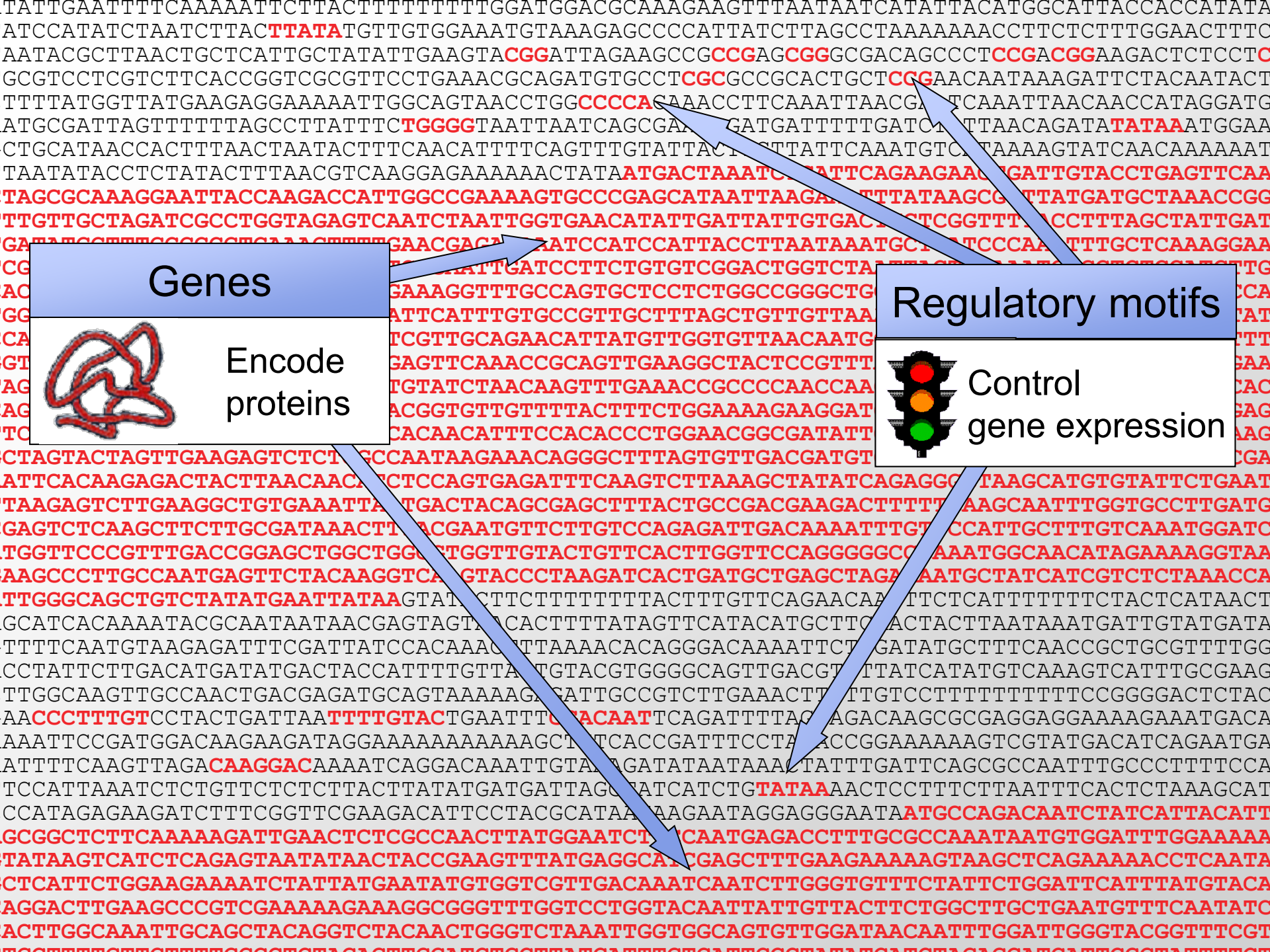
Why Computational Biology: Last year's answers

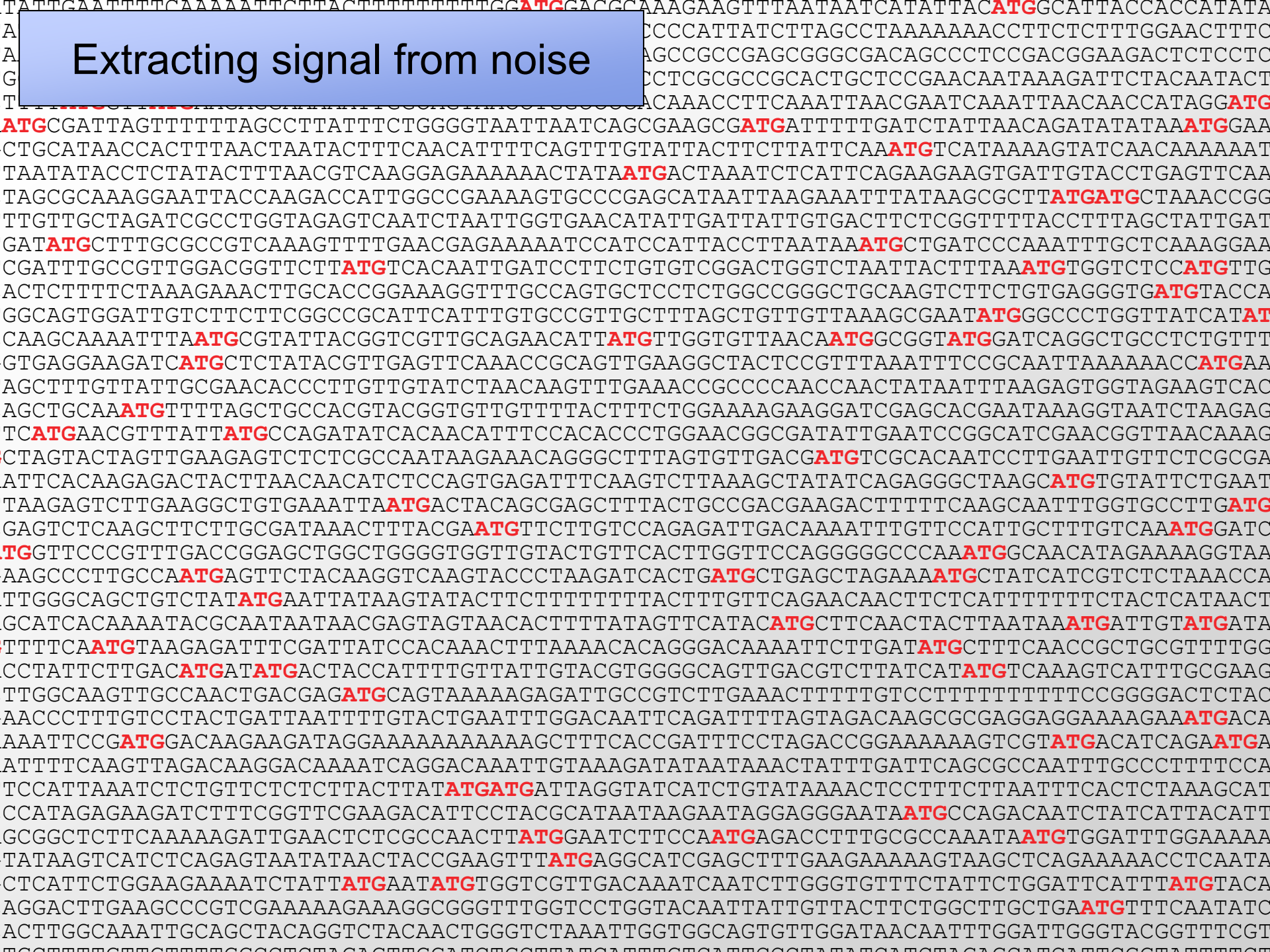
- Lots of data (* lots of data)
- There are rules
- Pattern finding
- It's *all* about data
- Ability to visualize
- Simulations, temporal relationships
- Guess + verify (generate hypotheses for testing)
- Propose mechanisms / theory to explain observations
- Networks / combinations of variables
- Efficiency (reduce experimental space to cover)
- Informatics infrastructure (ability to combine datasets)
- Correlations, higher-order relationships
- Cycle from hypothesis generation to testing condensed
- Life itself is digital. Understand cellular instruction set

Why Computational Biology: Live in Zoom Chat F20

- Data-rich in a historically data-poor domain (Matthew West)
- potential to do whatever you want without waiting for experiments (Stuti Khandwala)
- DNA is a massive dataset (Pablo X Villalobos)
- More efficient and in depth way to explore biology (Lilly K Edwards)
- There're tons of biological datasets waiting to be analyzed (Hieu Q Dinh)
- Because you can use other people's datasets and then get good research done on a budget (Ari)
- Might be the biggest frontier of computing today (Erez Kaminski)
- More and more sequencing data are coming out (Evelyn Tong)
- New technologies - lots of data - (Manu Ponnapati)
- Biology benefits from approximation (Thomas Xiong)
- The need to integrate multi-omics data to gain more insights (Kathleen Sucipto)
- Its interesting and new (Daniel R Gutierrez)
- Can use expertise from other engineering fields to impact health (Swathi Manda)
- Complex patterns in biological data (Farhan Khodaee)
- impact real human lives, important applications (Lucy Zhang)
- answers questions not easily solvable by traditional experimental biology (Andrew D Hennes)
- Expands our horizons in asking biological questions (Dylan McCormick)
- Computational biology and simulations can help deconvolve results from experiments (Raina Thomas)




Genes
Encode proteins
Regulatory motifs
Control gene expression

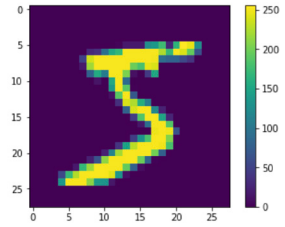

Extracting signal from noise

Deep Learning Problem Sets and compute platform

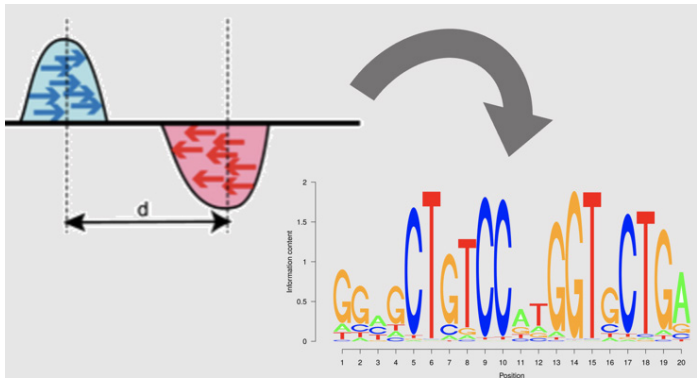
6.874/6.802/20.390/20.490/HST.506 Deep Learning in the Life Sciences Spring 2021 Lec:Tue/Thu 1-2:30pm Rec:Fri 3-4pm Proj:Fri 4-5pm

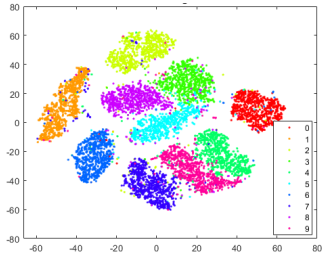
Psets	Date	Module	Week	Lec/R	Description
PS0: Set up Environment (Due Monday 2/22)	Tuesday, February 16, 2021	Module 1: ML models and interpretation	1	L01	Course Intro + Overview Foundations
	Thursday, February 18, 2021			L02	ML foundations
	Friday, February 19, 2021			R01	ML Review
	Friday, February 19, 2021			Proj1	Intro video + personal profile
PS1: Softmax warmup (MNIST) (out: Tue 2/23, due: Wed 3/10)	Tuesday, February 23, 2021		2	L03	Convolutional Neural Networks CNNs
	Thursday, February 25, 2021			L04	RNNs, GNNs
	Friday, February 26, 2021			R02	Neural Networks Review
	Friday, February 26, 2021			Proj2	Research Mentors Introductions and Breakouts
	Tuesday, March 2, 2021		3	L05	Interpretability, Dimensionality Reduction, tSNE
	Thursday, March 4, 2021			L06	Generative Models, GANs, VAEs
	Friday, March 5, 2021			R03	Interpreting ML Models
	Friday, March 5, 2021			Proj3	Research Team Building Breakout Rooms
PS2: CNN for TF binding prediction (out: Tue 3/16, Due: Mon 3/29)	Tuesday, March 9, 2021	Module 2: Gene Regulation	4		No Class (Monday Schedule)
	Thursday, March 11, 2021			L07	DNA accessibility, Promoters and Enhances
	Friday, March 12, 2021			R04	Chromatin and gene regulation, dimensionality reduction
	Friday, March 12, 2021			Proj4	Initial Ideas 1-slide presentations (teams, or individual)
	Tuesday, March 16, 2021		5	L08	Transcription factors, DNA methlyation
	Thursday, March 18, 2021			L09	Gene Expression, Splicing
	Friday, March 19, 2021			R05	RNA-seq, Splicing
	Friday, March 19, 2021			Proj5	Meet with potential mentors (optional, asynchronous)
	Tuesday, March 23, 2021		6		No Class (Student Holiday)
	Thursday, March 25, 2021			L10	Single-cell RNA-sequencing
	Friday, March 26, 2021			R06	scRNA-seq
	Friday, March 26, 2021			Proj6	Full Project Proposals Due (pdf, slides, team video)
PS3: scRNA-seq tSNE analysis (out: Tue 3/30, due Mon 4/12)	Tuesday, March 30, 2021	Module 3: Genetic Variation / Disease	7	L11	Genetics and Variation
	Thursday, April 1, 2021			L12	GWAS and Rare variants
	Friday April 2, 2021			R07	Genetics
	Friday, April 2, 2021			Proj7	Meet with your mentors (optional, asynchronous)
	Tuesday, April 6, 2021		8	L13	eQTLs
	Thursday, April 8, 2021			L14	Electronic health records and patient data
	Friday April 9, 2021			R08	ML for health data
	Friday, April 9, 2021			Proj8	End-to-End pipeline demo (team video)
PS4: Graph Neural Networks (Out: Tue 4/13, Due: Wed 4/28)	Tuesday, April 13, 2021	Module 4: Graphs and Proteins	9	L15	Protein-protein interactions and graph analysis
	Thursday, April 15, 2021			L16	Protein Structure
	Friday April 16, 2021			R09	Protein Structure Prediction
	Tuesday, April 20, 2021		10		No Class (Student Holiday)
	Thursday, April 22, 2021			L17	Drug Development
	Friday April 23, 2021			R10	Drug Development
	Friday, April 23, 2021			Proj9	Meet with your mentors (optional, asynchronous)
PS5: Image Analysis (Out: Wed 4/28, Due: Mon 5/10)	Tuesday, April 27, 2021	Module 5: Imaging	11	L18	Therapeutics
	Thursday, April 29, 2021			L19	Imaging, Morphology
	Friday, April 30, 2021			R11	Therapeutics, 3D structure, imaging
	Friday, April 30, 2021			Proj10	Midcourse report (google doc)
	Tuesday, May 4, 2021		12	L20	Imaging applications in healthcare
	Thursday, May 6, 2021			L21	Video processing, structure determination
	Friday May 7, 2021				No Class (Student Holiday)
Finalize Projects	Tuesday, May 11, 2021	Module 6: Frontiers	13	L22	Text applications in healthcare, clinical decision making
	Thursday, May 13, 2021			L23	Neuroscience
	Friday, May 14, 2021			R12	How to Present
	Friday, April 30, 2021			Proj11	How to Present
	Monday, May 17, 2021		14	Proj12	Final Reports due (Google doc + pdf)
	Tuesday, May 18, 2021			L24	Cancer and Infectious Disease
	Wednesday, May 19, 2021			Proj13	Final Presentations (slides, team video)
	Thursday, May 20, 2021			L25	Final Presentations

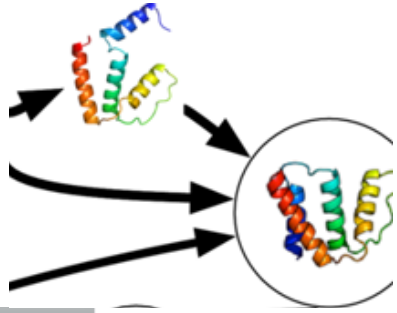
Course website: http://compbio.mit.edu/6874/

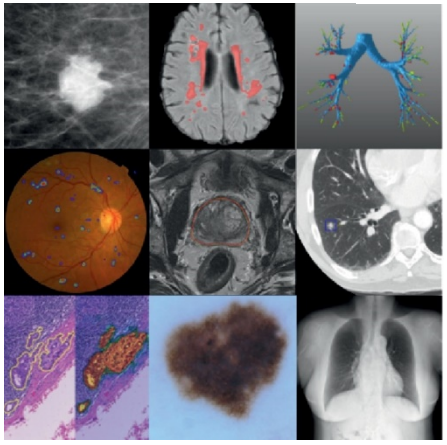


ground truth: 5









Collage of some medical imaging applications in which deep learning has achieved state-of-the-art results.

From top-left to bottom-right:

1. mammographic mass classification
2. segmentation of lesions in the brain,
3. leak detection in airway tree segmentation,
4. diabetic retinopathy classification
5. prostate segmentation,
6. nodule classification,
7. breast cancer metastases detection,
8. skin lesion classification
9. bone suppression

Deep Learning Problem Sets and compute platform

Your programming environment

Problem 2

In this problem, we wish to use CNN to learn the motif of CTCF from sequences with similar di-nucleotide frequency. The positive samples are 101bp sequences centered at CTCF ChIP-seq peaks from GM12878 cell line. The negative sequences are generated by permuting the nucleotides in the positive sequences while keeping the di-nucleotide frequency.

We will provide functions for loading data, training and testing. You will:

- implement a CNN model with given specifications
- specify the initilization of parameters in the model
- train the model and evaluate on the test set

All the places where you need to fill in begins with "TODO" and ends with "END OF YOUR CODE".

In [1]:
```
import tensorflow as tf, sys, numpy as np, h5py
from os.path import join,dirname,basename,exists,realpath
from os import makedirs
from tensorflow.examples.tutorials.mnist import input_data
from sklearn.metrics import roc_auc_score
```

In [2]:
```
data_folder = '../data/motif_disc'
batch_size = 128
valid_size = 2000
epochs = 20
best_model_file = join('../output',basename(data_folder),'best_model.ckpt')
if not exists(dirname(best_model_file)):
    makedirs(dirname(best_model_file))
```

In [3]:
```
# Function to load the data embedded in the previous problem and their labels
def load_data(mydir):
    train = h5py.File(join(mydir,'train.h5'),'r')
```


Your computing resourc

Cloud Platform Education Grants

Use credits provided to you via the Google Cloud Platform Education Grants program to access Google Cloud Platform. Get what you need to build and run your apps, websites and services.

Thank you for your interest in Google Cloud Platform Education Grants. Please fill out the form below to receive a coupon code for credit to use on Google Cloud Platform.

First Name

Last Name

School Email

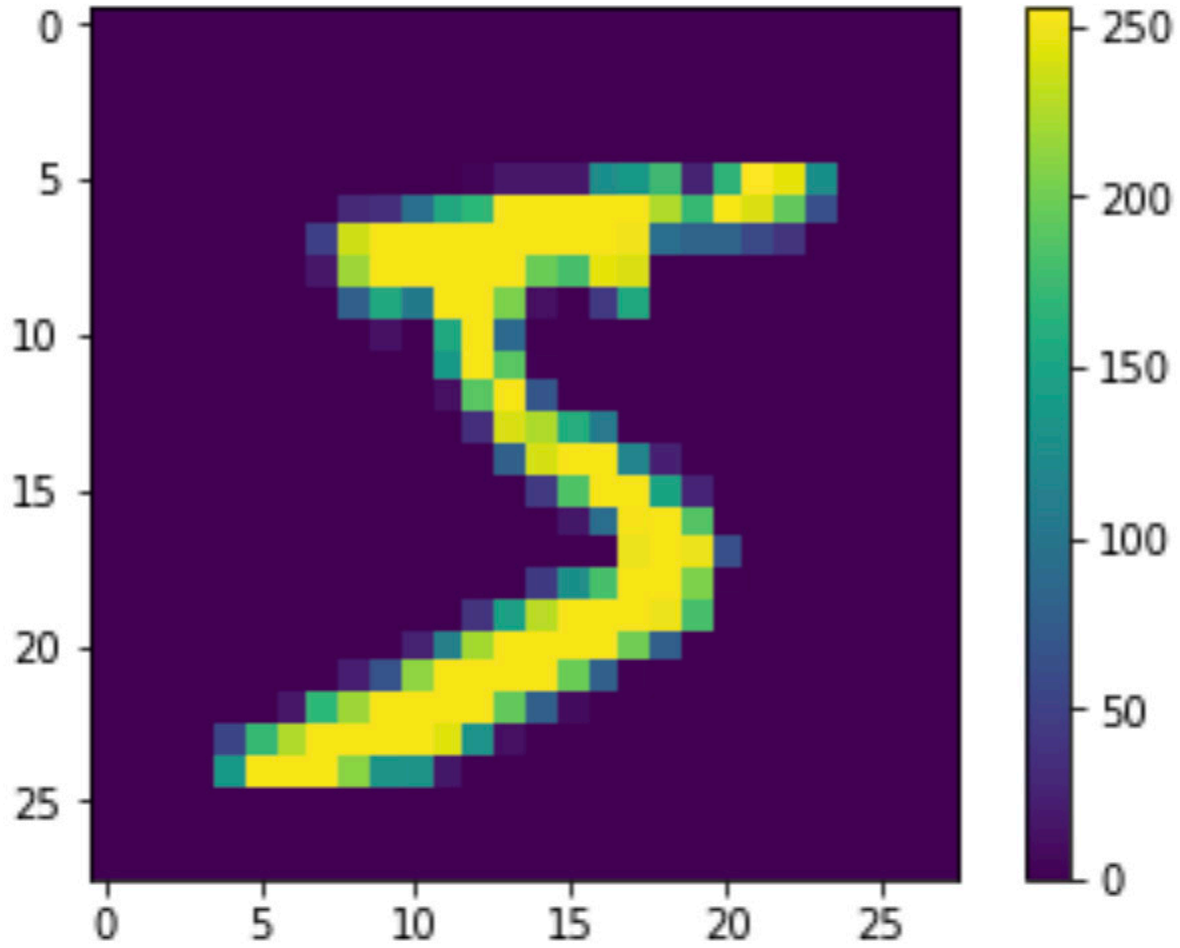
@mit.edu

If you do not see your domain listed, please contact your course instructor: gifford@mit.edu

By clicking "Submit" below, you agree that we may share the following information with your educational institution and course instructor (gifford@mit.edu): (1) personal information that you provide to us on this form and (2) information regarding your use of the coupon and Google Cloud Platform products.

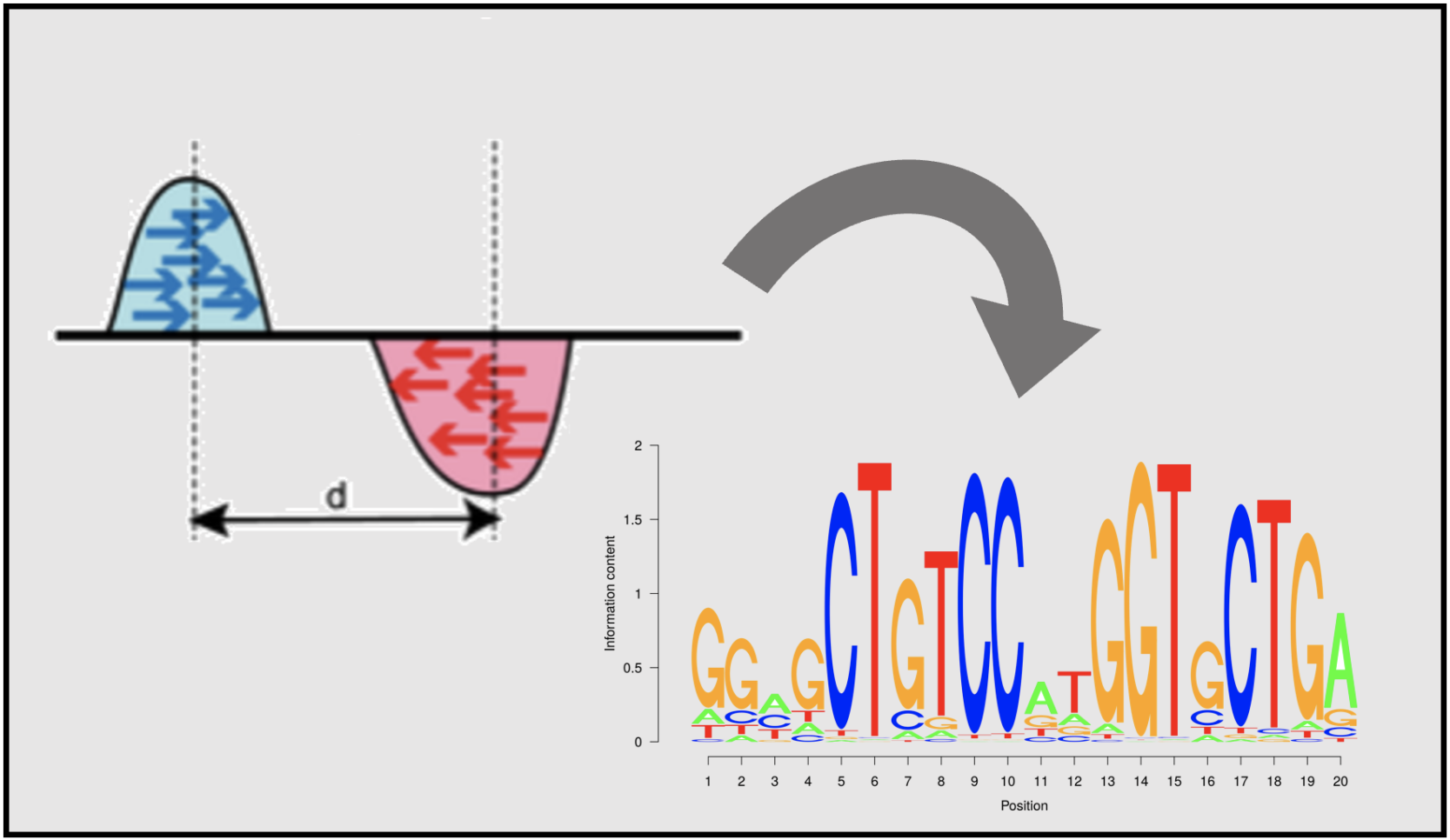
Submit

PS 1: Tensor Flow Warm Up

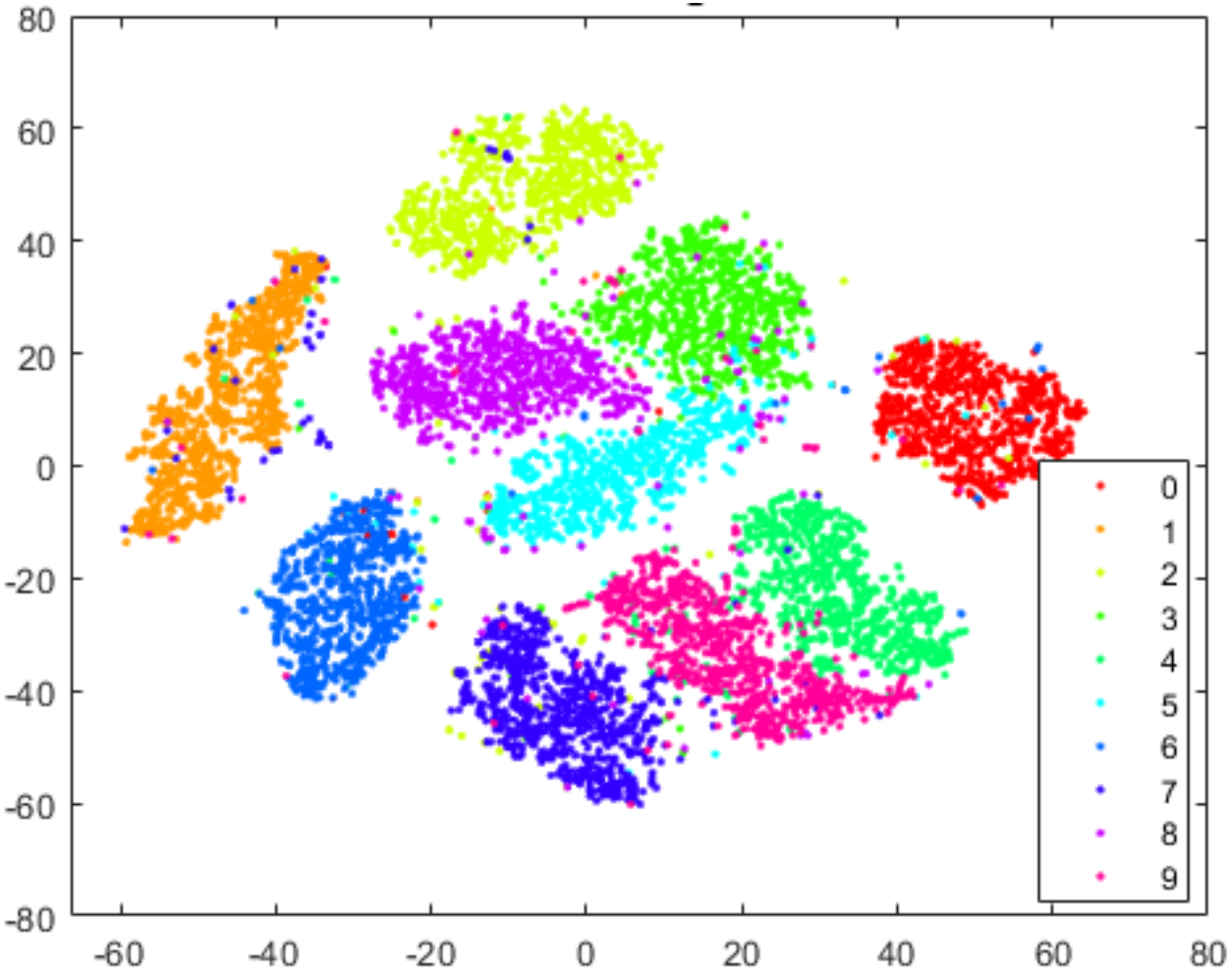


ground truth: 5

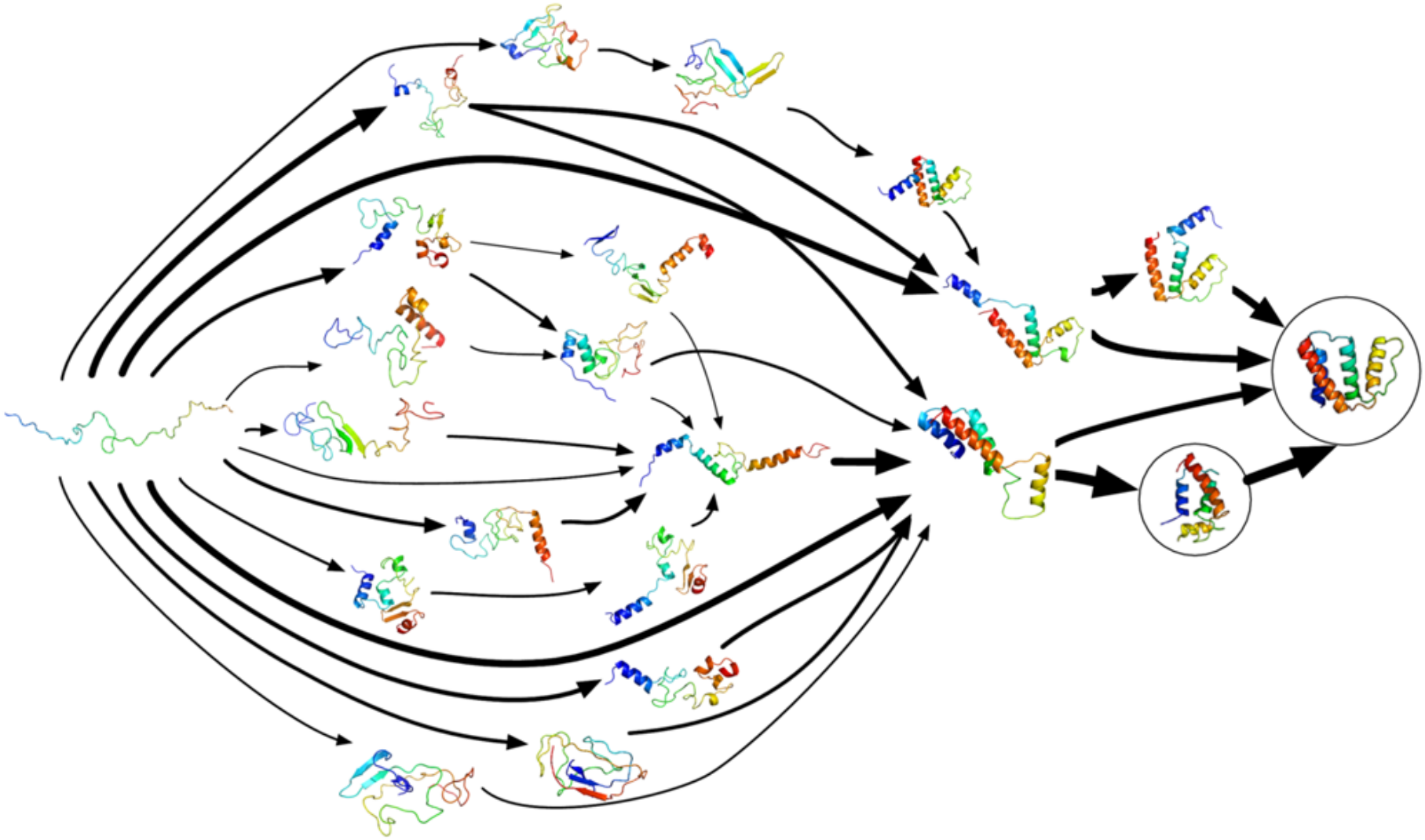
PS 2: Genomic regulatory codes



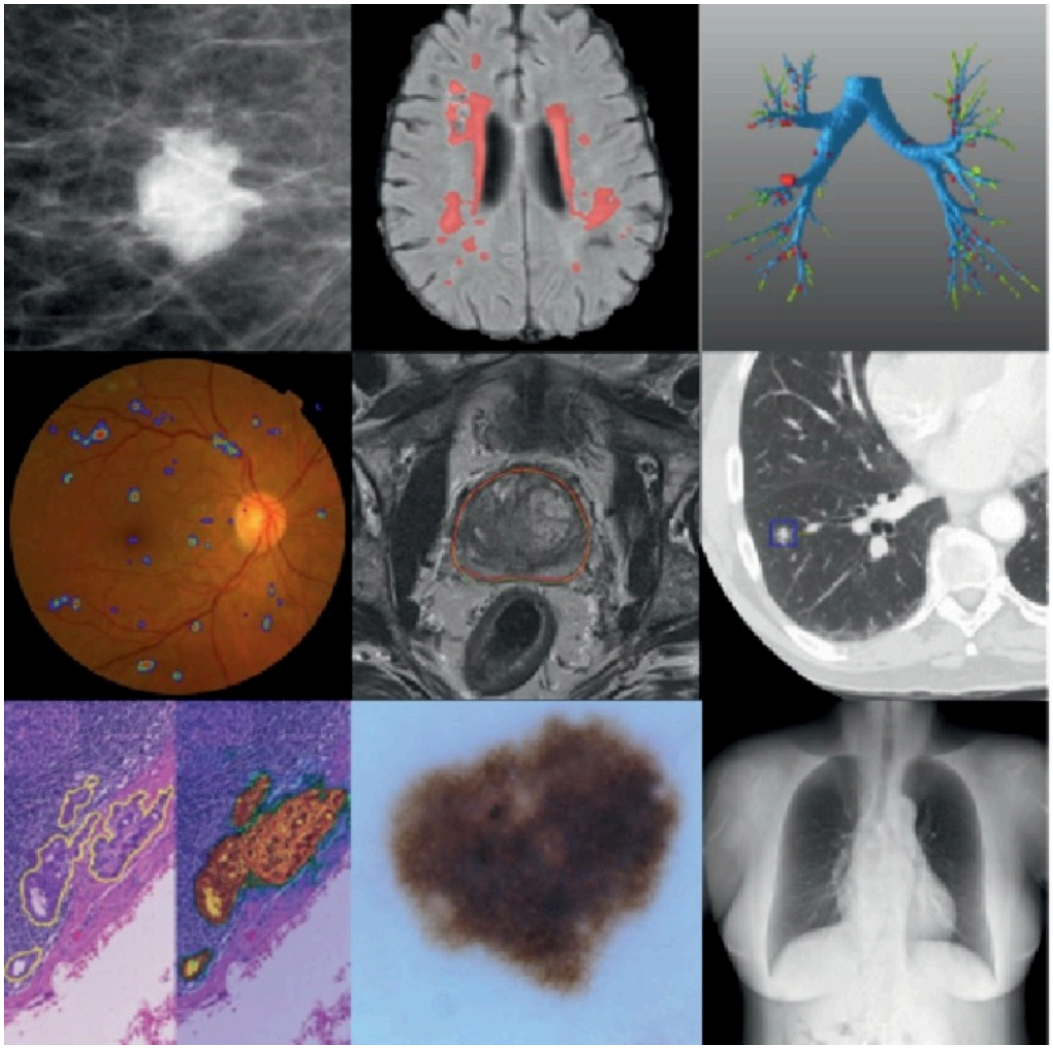
PS 3: Parametric tSNE
Single Cell RNA-seq data



PS 4: Protein Folding + Drug Design



PS 5: Medical Image Analysis



Collage of some medical imaging applications in which deep learning has achieved state-of-the-art results.

From top-left to bottom-right:

1. mammographic mass classification
2. segmentation of lesions in the brain,
3. leak detection in airway tree segmentation,
4. diabetic retinopathy classification
5. prostate segmentation,
6. nodule classification,
7. breast cancer metastases detection,
8. skin lesion classification
9. bone suppression

Deep Learning Problem Sets and compute platform

Lectures and Scribing

- Each lecture will have a dedicated scribe who will take notes on the lecture
  - Please sign up to scribe for lecture on the sheet being passed around
- Build on notes from previous years
  - Available on course website
- Final draft of scribe notes due 6 days after lecture
  - Your grade depends on the improvement from previous year and completeness
- Some lectures need more work: multiple scribes
- Some tasks are better-suited to you than just scribing
  - E.g. figures, references, layout, macros, let us know!

Details on the 1.5 hour quiz

- It's not a midterm, and it's not a final exam
  - It's a quiz, friendly, fun, interesting, cute, fuzzy
- Demonstrate mastery of the material in 4 modules
  - Understand key points emphasized in lecture
  - Understand subtleties revealed in the psets
  - Ability to apply new skills to solve practical problems
- Types of questions
  - Knowledge questions: T/F justify, multiple choice
  - Deeper understanding questions: short answers
  - Practical problems: work through simple algorithm
  - Design problem(s): new/modified algorithm, need both knowledge and new idea, argue correctness

Guest Lectures: A living breathing field

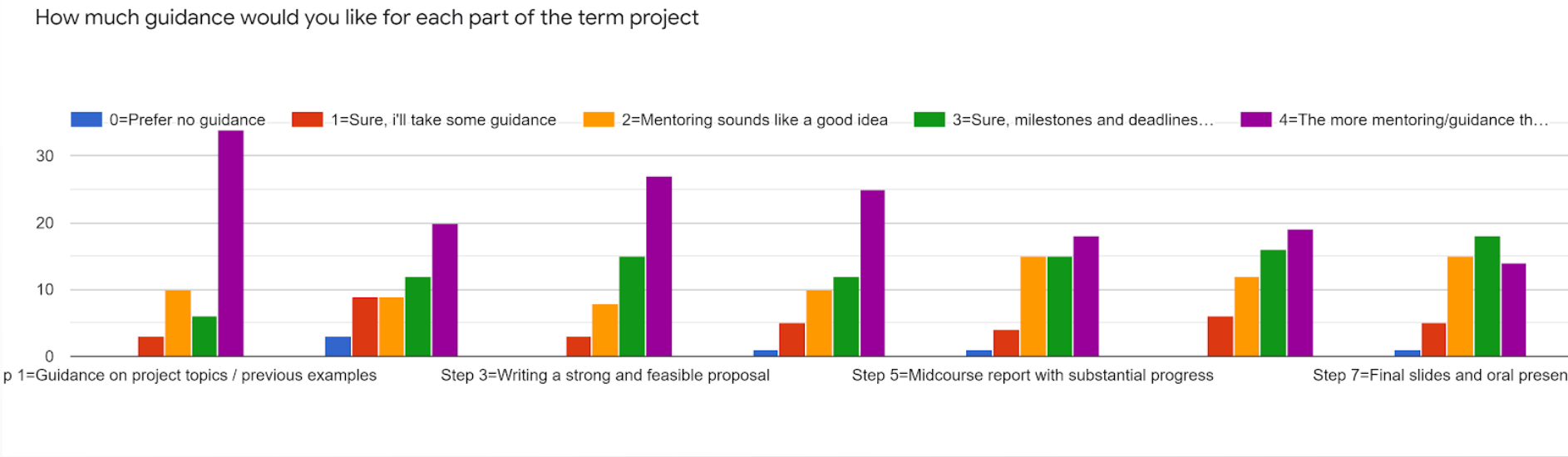
- For several lectures, we will invite 1-2 guest lecturers for short (25 + 5) presentations on key papers in the field
- We will introduce the area and foundational material at the beginning of these lectures (1:05pm-1:30pm)
- Then we'll have a guest lecture (1:30pm-1:55pm) or two (also 2pm-2:25pm), with the ability to ask questions and dive deeply on their papers
- We'll have a short Q&A with each guest lecturer
- We'll discuss the presentations and papers in more detail at the corresponding recitation section
- We'll build on them for research project directions at the mentoring sessions
- You'll have a chance to diver more deeply in your own projects working with your mentors

Team Projects: Original Research

Final Project: Original Research in Comp Bio

- A major aspect of the course is preparing you for original research in computational biology.
  - Framing a biological problem computationally
  - Gathering relevant literature and datasets
  - Solving it using new algorithms, machine learning
  - Interpreting the results biologically
- Also ability to present your ideas and research
  - Crafting a research proposal (fellowships/grants)
  - Working in teams of complementary skill sets
  - Review peer proposals, find flaws, suggest imprvmts
  - Receiving feedback and revising your proposal
  - Writing up your results in a scientific paper format
  - Presenting a research talk to a scientific audience
- Term project experience mirrors this process

Milestones



- Round 1: Self-introduction video and form (due Week 1 Friday)
- Round 2:
- Round 4: Team formation and initial project proposal (due Week 4 Friday)
- Round 5: Peer Review (due Week 5 Friday)
- Round 6: Revised proposals (due Week 6 Friday)
- Round 7: Initial end-to-end pipeline (due Week 7 Friday)
- Round 8: Midcourse report (due Week 11, Friday)
- Round 9: Project presentations (due Week 14 Wednesday (Day 18)
- Round 10: Final report, due Week 4 Friday (Day 20)

It's a team project

- Please make an effort to meet your peers!
- Form teams early with complementary expertise

Details on the final project

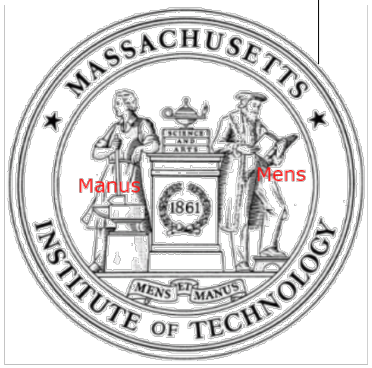
- Milestones ensure sufficient planning / feedback
  - Set-up: find project matching your skills and interests
  - Team: common interests and complementary skills
  - Inspiration: last year's projects, and recent papers
  - Proposal: establish milestones, deliverables, expectations
  - Midcourse: see endpoint, outline report, methods, figures
- Periodic mentoring sessions
  - Senior students and postdocs can serve as your mentors
  - Group discussions to share ideas, guidance, feedback
  - Peer-review: think critically about peer proposals, receive feedback/suggestions, respond to critiques, adjust course
- Real-world experience, condensed in a single term
  - Grant/fellowships proposals, peer review, yearly reports, budget time/effort, collaboration, paper writing, give talk

Final Project at a Glance

Week	Date	Milestone	Description
1	Friday, February 19, 2021	Proj1	Intro video + personal profile
2	Friday, February 26, 2021	Proj2	Research Mentors Introductions and Breakouts
3	Friday, March 5, 2021	Proj3	Research Team Building Breakout Rooms
4	Friday, March 12, 2021	Proj4	Initial Ideas 1-slide presentations (teams, or individual)
5	Friday, March 19, 2021	Proj5	Meet with potential mentors (optional, asynchronous)
6	Friday, March 26, 2021	Proj6	Full Project Proposals Due (pdf, slides, team video)
7	Friday, April 2, 2021	Proj7	Meet with your mentors (optional, asynchronous)
8	Friday, April 9, 2021	Proj8	End-to-End pipeline demo (team video)
10	Friday, April 23, 2021	Proj9	Meet with your mentors (optional, asynchronous)
11	Friday, April 30, 2021	Proj10	Midcourse report (google doc)
13	Friday, May 14, 2021	Proj11	How to Present
14	Monday, May 17, 2021	Proj12	Final Reports due (Google doc + pdf)
14	Wednesday, May 19, 2021	Proj13	Final Presentations (slides, team video)

Grading and Estimated Hours spent: Mens et Manus

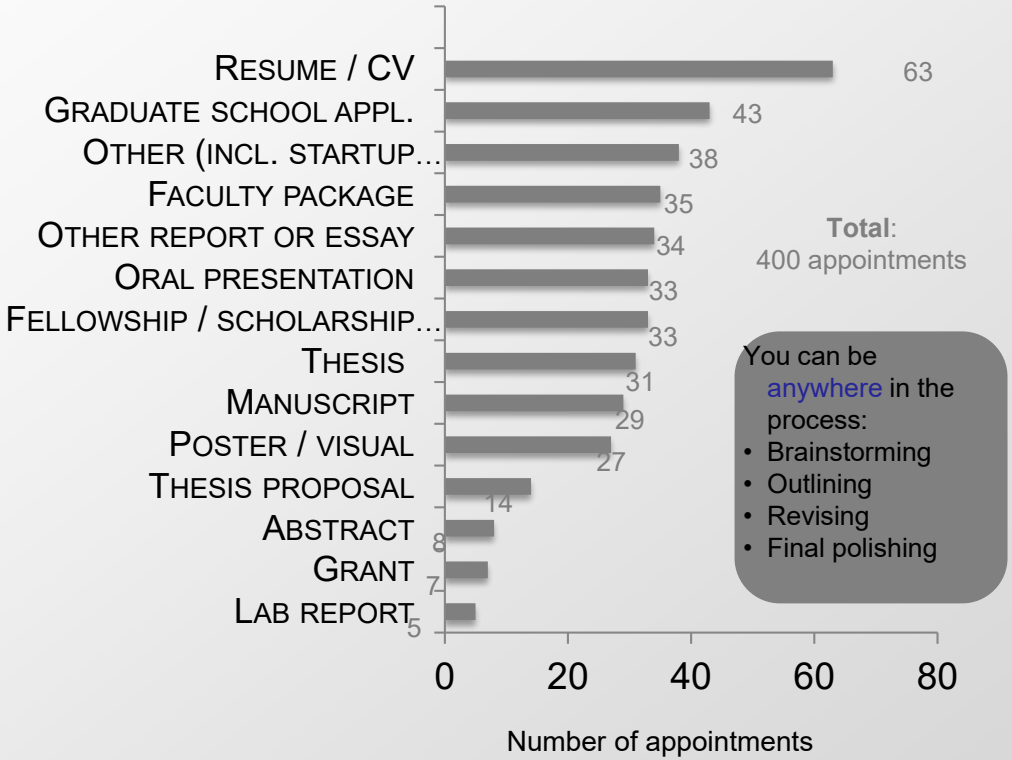
	Activity	Points	Hours	Hours
Learning	Attending Lectures	[factored below]	36.0	
	Attending Recitations	[factored below]	14.0	
	Understanding lecture/recitation	[factored below]	24.0	**84.0**
	Studying for Quiz	30	10.0	
	Working on Psets	20	30.0	
Doing	Mentoring session attendance	[factored below]	7.0	
	Self-Intro (R1)	3	0.5	
	Paper presentation (R2)	7	2.0	
	Top 3 ideas (R3)	5	1.0	
	Proposal (Initial/Revised, R4/R6)	10	3.5	
	Peer-Reviewing (R5)	10	6.0	**84.0**
	Initial Pipeline (R7)	5	1.5	
	Midcourse Report (R8)	5	3.0	
	Final Slides (R9)	5	2.0	
	Final Presentation (R9)	5	2.0	
	Final Report (R10)	15	4.0	
	Working on project (R4-R10*team)	[factored above]	21.5	
	Total	**120**	**168.0**	
		Hours per week	**12.000**	



Comm Lab: Help communicating your research!

A free resource for peer feedback from trained EECS grad students and postdocs.

Why people come to CommLab:



"Very, very valuable. Thank you!"
—Elena Glassman, EECS PhD alumna

"I strongly encourage students to schedule a session; it's a very impressive resource."
—Dirk Englund, professor

"The experience and coaching helped me apply successfully for an important fellowship this year."
—Joel Jean, EECS grad

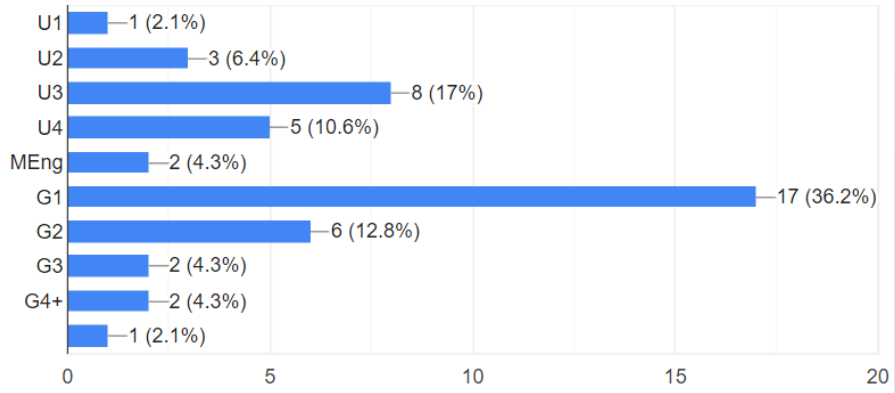
Finding a research mentor / research advisor

- Chance to meet faculty at MIT/Broad/Harvard:
  - Through guest lectures and mentoring
  - Topics and papers covered in the lectures
  - Experts on: (1) human comparative genomics, (2) lincRNAs, (3) metabolic modeling, (4) disease mapping, selection, evolution and ecology (following four modules)
- Chance to meet senior students and postdocs:
  - On: coding genes, ncRNAs, regulatory motifs, networks, epigenomics, phylogenomics (again on each module)
  - Mentorship sessions with entire MIT CompBio group
- Your own personal research experience:
  - collaborators, datasets
  - learn active research directions, frontiers
  - living, breathing changing field

First Day Survey

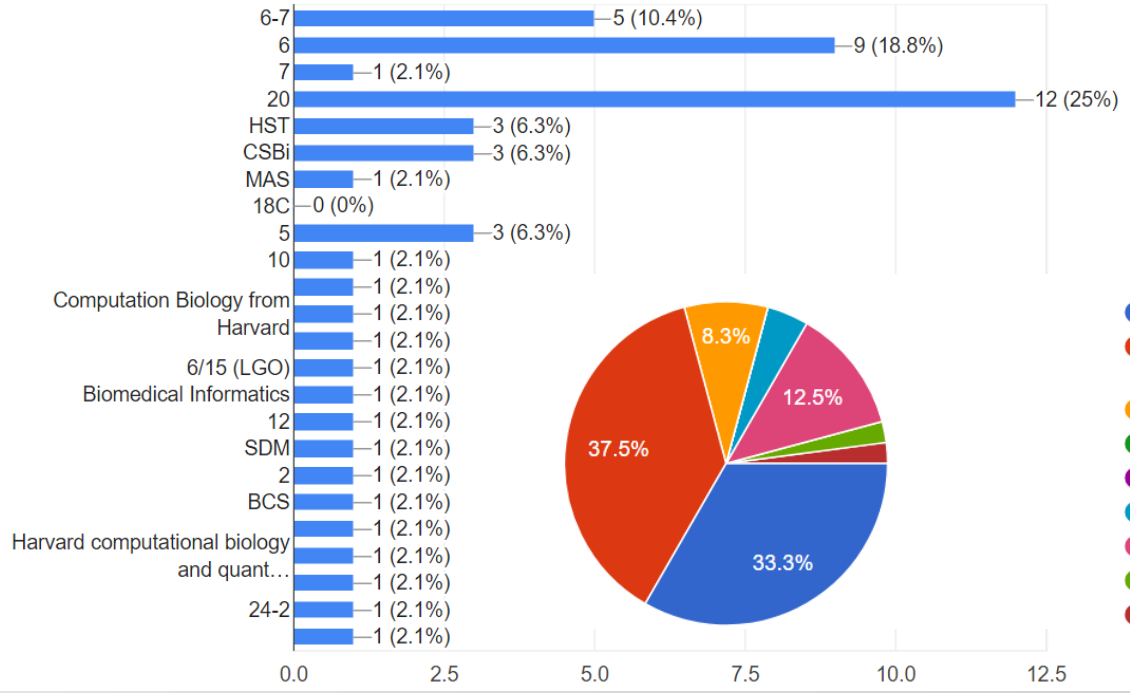
What year are you?

47 responses

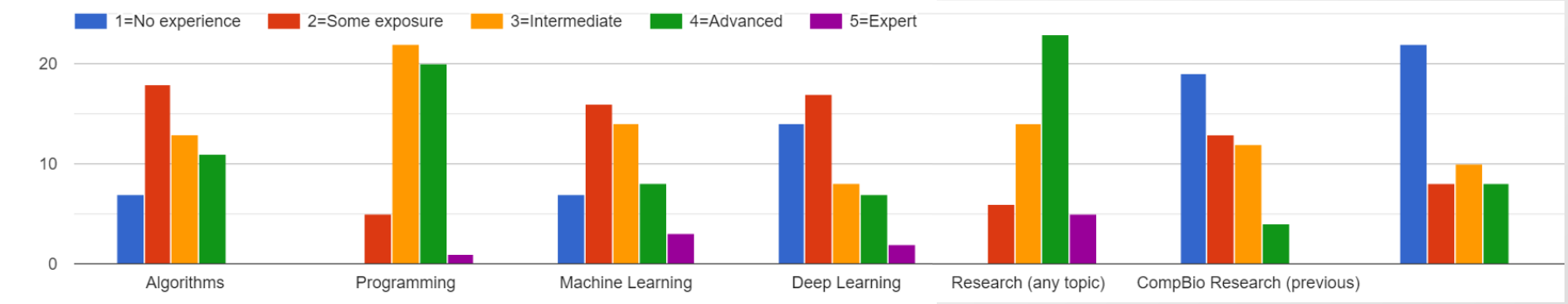


What major are you?

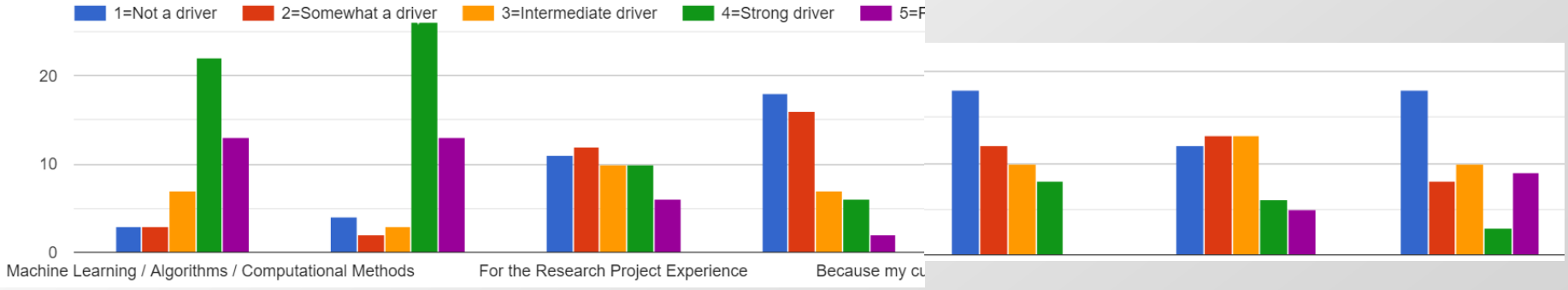
48 responses



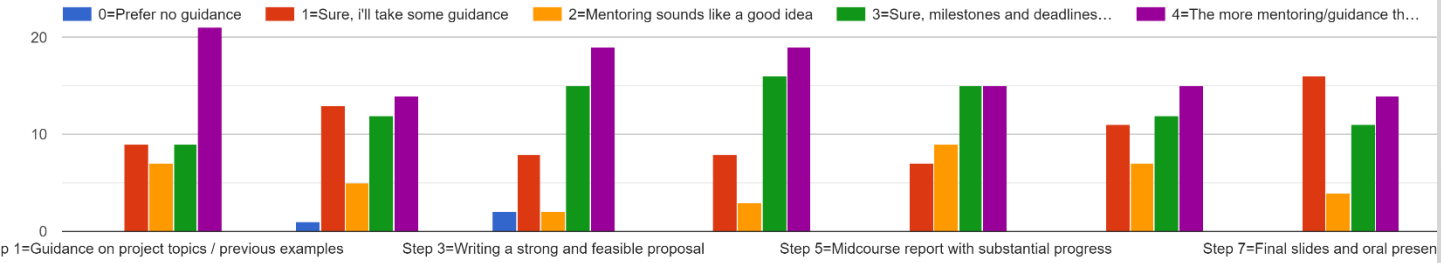
Please score your background in each area



Score your primary drivers/reasons for taking this class



How much guidance would you like for each part of the term project



Intro to Machine Learning

What is artificial intelligence? (AI)

Hard for Machines

General intelligence

Manipulation

Driving

Images

Faces

Language

Jeopardy!

'Deep' Learning

Prime factorization

Knowledge

Planning

Logic

Reasoning

Remembering

'Classical' AI

Math

Integration

Go

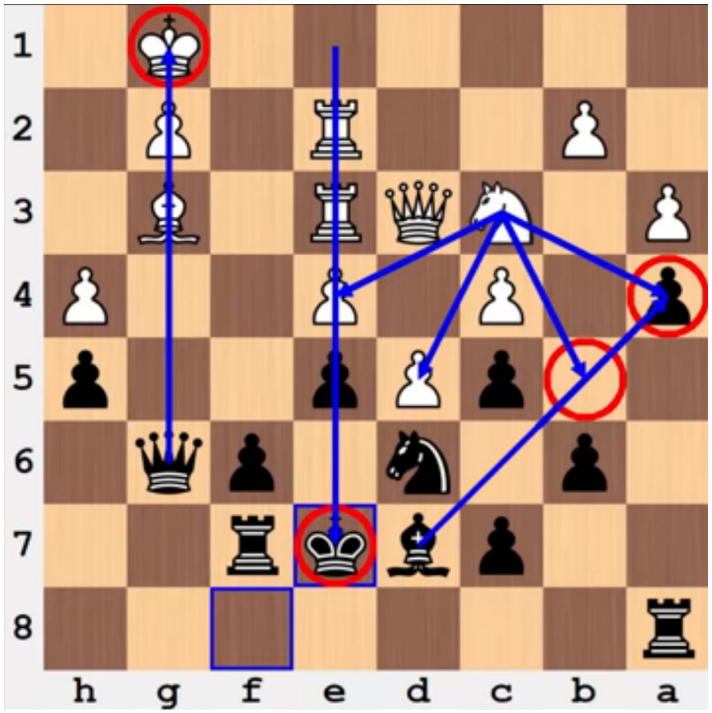
Chess

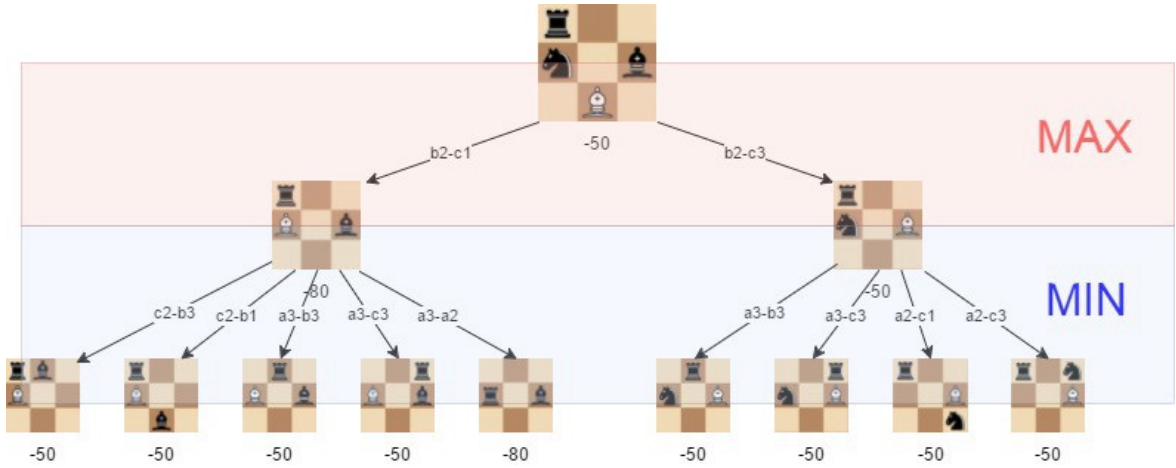
Easy for Machines

Easy for humans

Hard for humans

How do machines play chess?





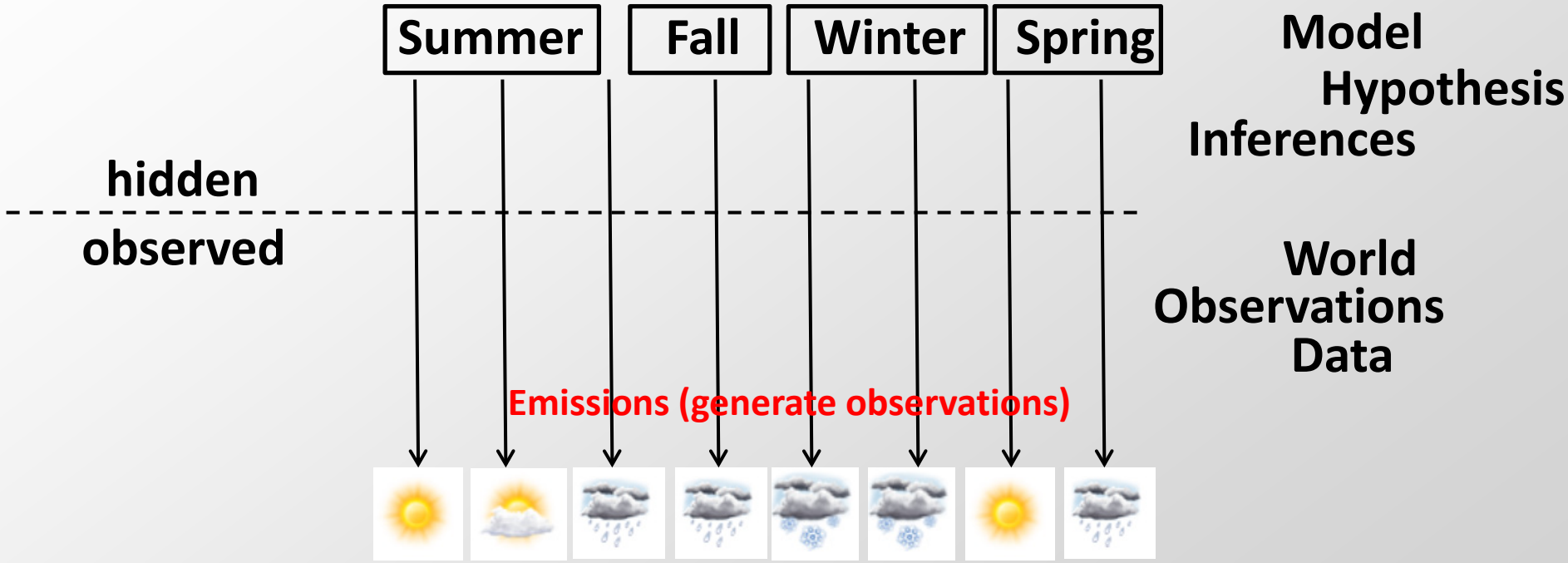
‘Classical’ AI approach (rule-based, tree search):

1. Human: Program in all the rules of chess
2. Human: Hand-craft a scoring function for each position
3. Search all moves that you can make (max score)
4. Search all moves that opponent can make (min score)
5. Repeat for many iterations
6. Choose move that gives best score

What do *you* see?

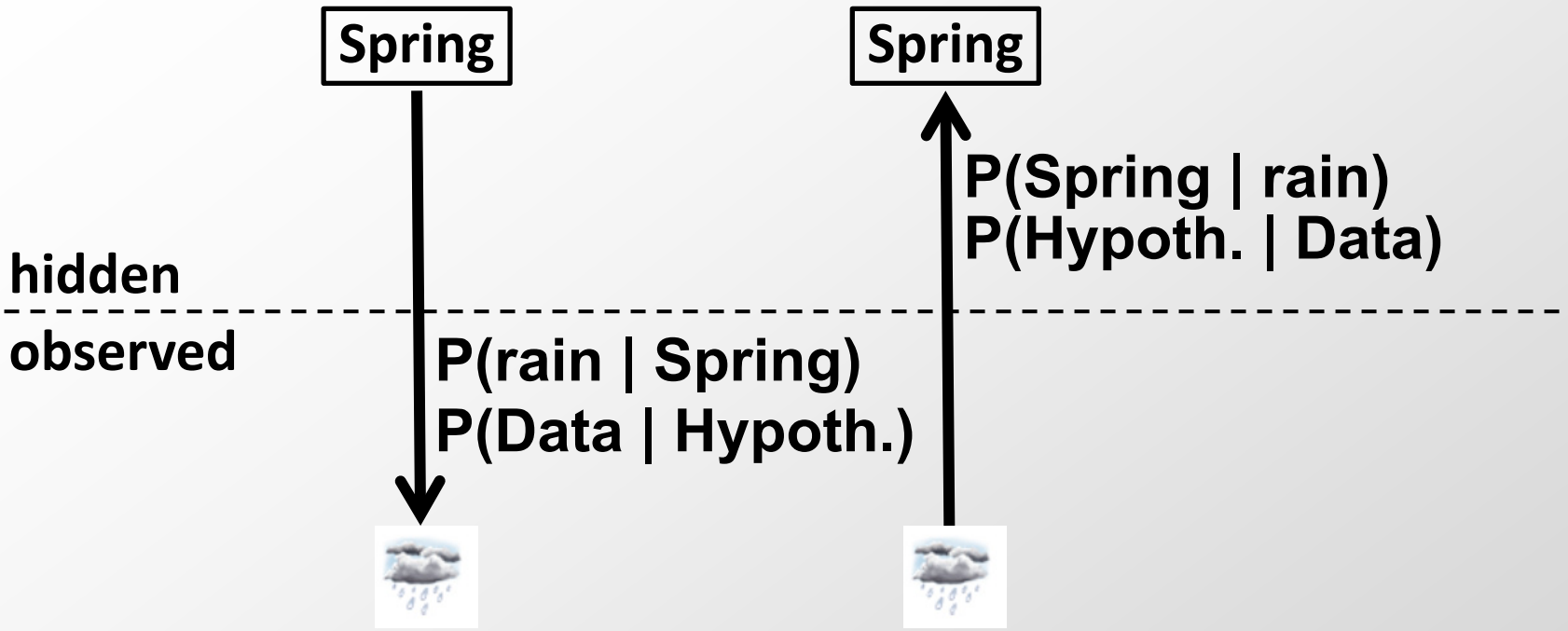
Making inferences about the world

- Generative models:
  Express **forward** probability of an event,
  given the **hidden** state of the world



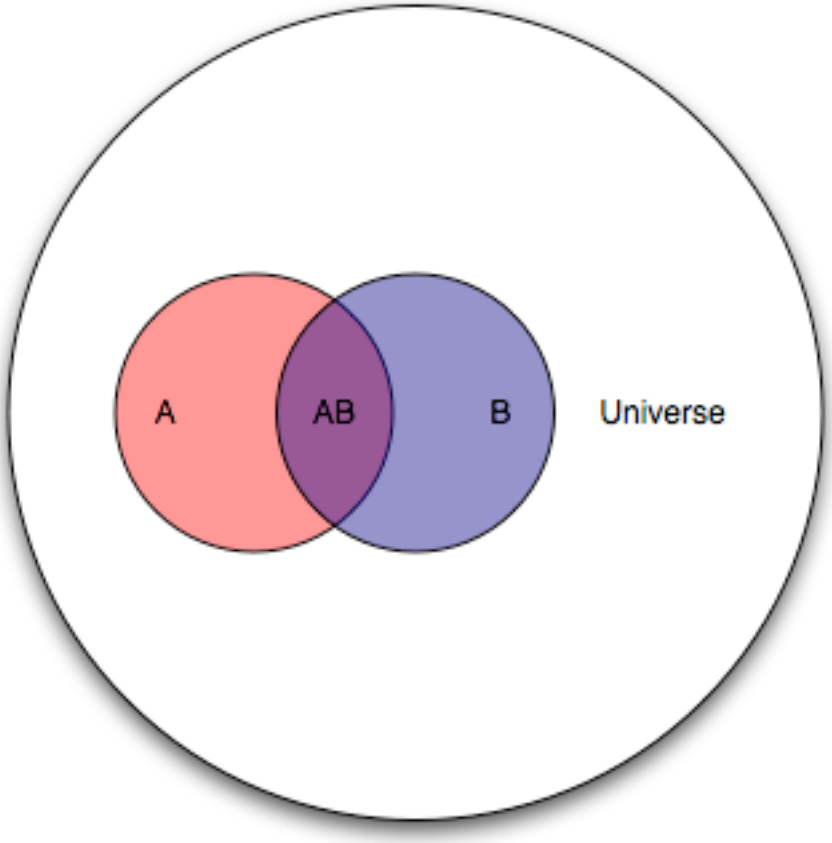
- We can estimate:
  - P(snow | winter), P(observation | season)

"Reversing the arrows"



- Goal: P(D|H) ➔ P(H|D)
- Bayes' Rule allows us to do this:
- $$P(H|D) = \frac{P(D|H) * P(H)}{P(D)}$$

Proving Baye's Rule



$$P(A|B)P(B) = P(B|A)P(A)$$

$$P(A|B) = P(B|A)P(A)/P(B)$$

$$P(B|A) = P(A|B)P(B)/P(A)$$

Bayes' rule

Bayes Theorem

Likelihood
Probability of collecting this data when our hypothesis is true

Prior
The probability of the hypothesis being true before collecting data

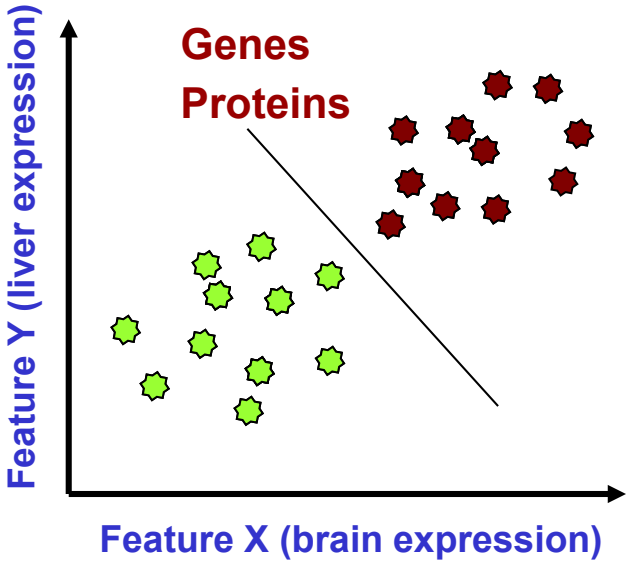
$$P(H|D) = \frac{P(D|H)\,P(H)}{P(D)}$$

Posterior
The probability of our hypothesis being true given the data collected

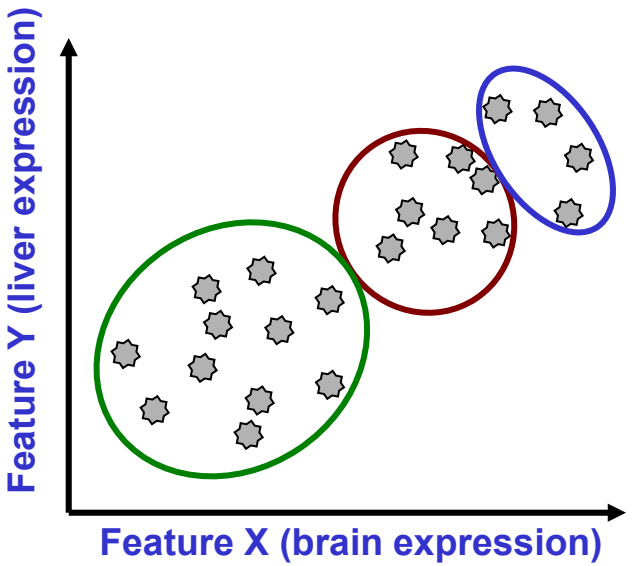
Marginal
What is the probability of collecting this data under all possible hypotheses?

Clustering vs Classification

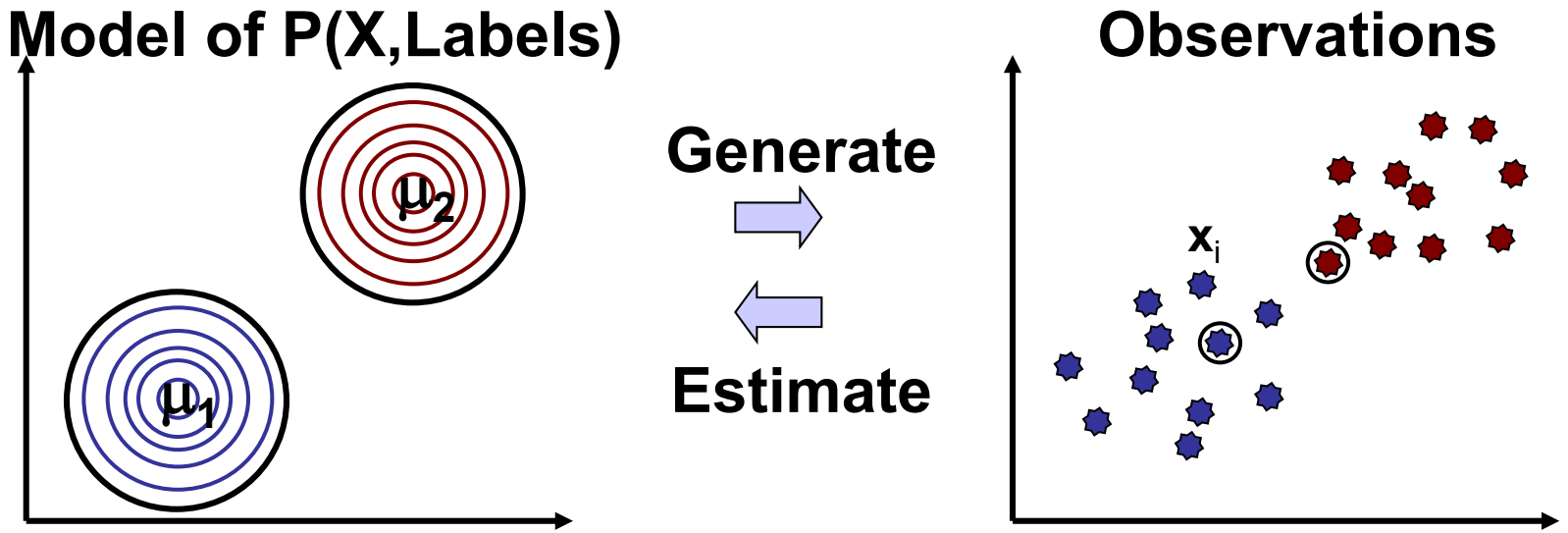
- Objects characterized by one or more features
- **Classification (supervised learning)**
  - Have labels for some points
  - Want a "rule" that will accurately assign labels to new points
  - Sub-problem: Feature selection
  - Metric: Classification accuracy



- **Clustering (unsupervised learning)**
  - No labels
  - Group points into clusters based on how "near" they are to one another
  - Identify structure in data
  - Metric: independent validation features



Expectation Maximization.
Iterative estimation of generative model parameters

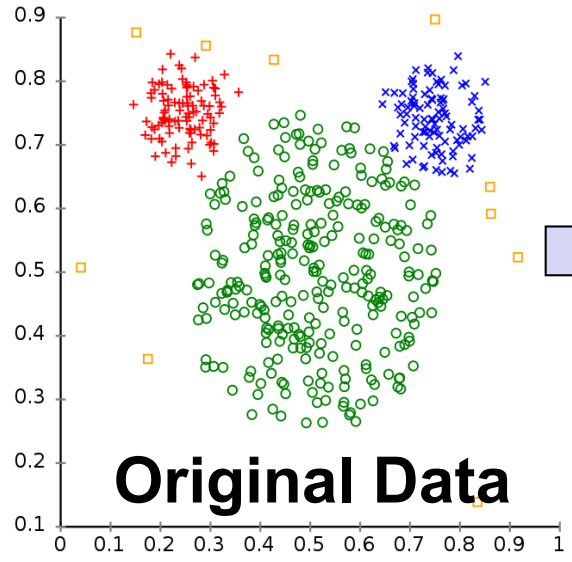


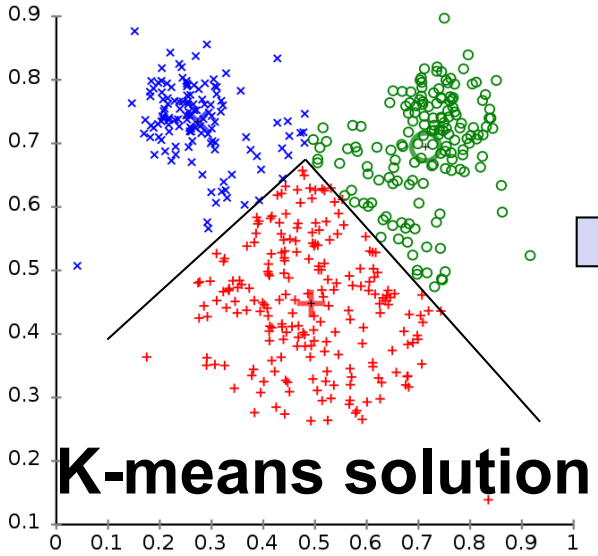
Samples drawn from normal distributions with unit variance - a *Gaussian Mixture Model*

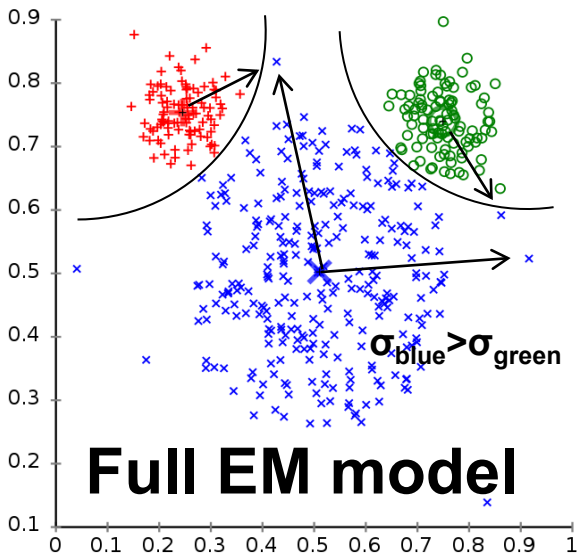
$$P\left(\mathbf{x}_i \mid \mathbf{u}_j\right) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{\left(\mathbf{x}_i - \mathbf{u}_j\right)^2}{2}\right\}$$

Given only samples, how do we estimate max lik model params: (1) centroid definitions, (2) point assignments?

EM vs. K-means vs. fuzzy K-means







	K-means solution	EM generalization
Cluster sizes	**Uniform** priors	Class priors $P(class_i)$
Spread of points	**Unit** distance function	$Gaussian\ (\mu_i, \sigma_i)$
Cluster shape	**Symmetric**, x-y indpt	Co-variance matrix $q_{jk} = \frac{1}{N}\sum_{i=1}^{N}(x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)$
Label assignment	K-means: Pick **max** Fuzzy: Full **density**	EM: Full **density** Gibbs: **sample** posterior

Support Vector Machines for Classification

We define a vector **w** normal to the separating line

Assume all data satisfy the following:

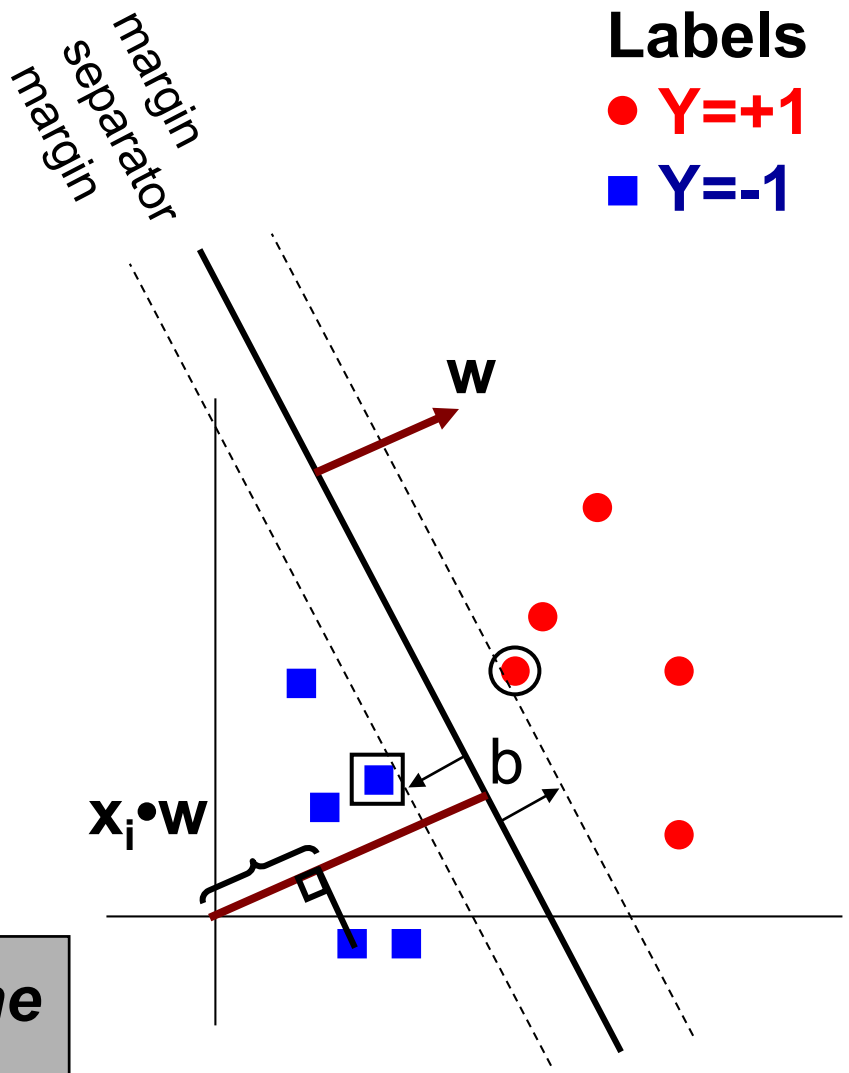
$$\mathbf{x_i} \bullet \mathbf{w} - b \geq +1 \quad \text{for } y_i = +1$$

$$\mathbf{x_i} \bullet \mathbf{w} - b \leq -1 \quad \text{for } y_i = -1$$

$$\Downarrow$$

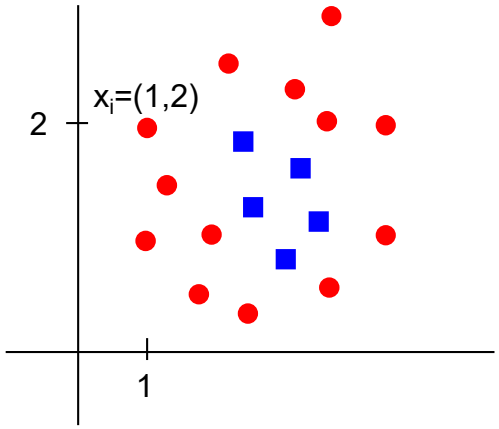
$$y_i \left(\mathbf{x_i} \bullet \mathbf{w} - b \right) \geq 1$$

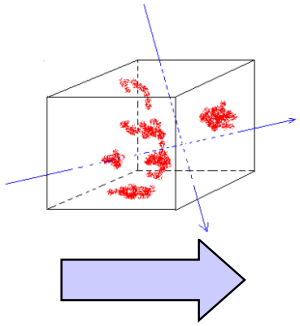
Find the separator with the largest margin

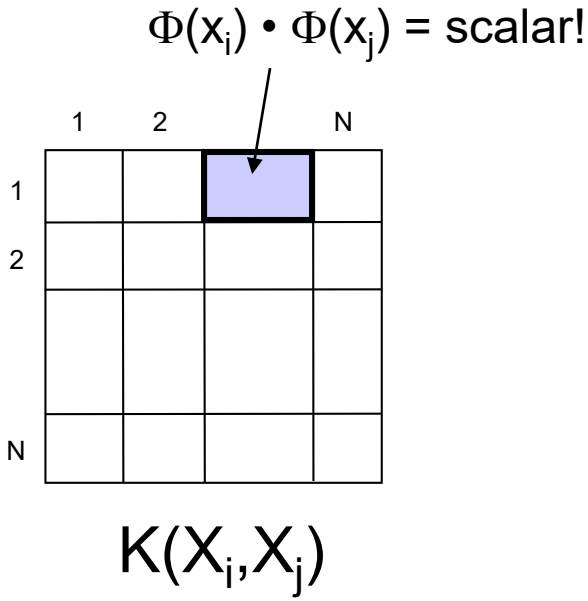


Kernel mapping higher-dimensional embedding

So the key step is to take your input data and transform it into a kernel matrix





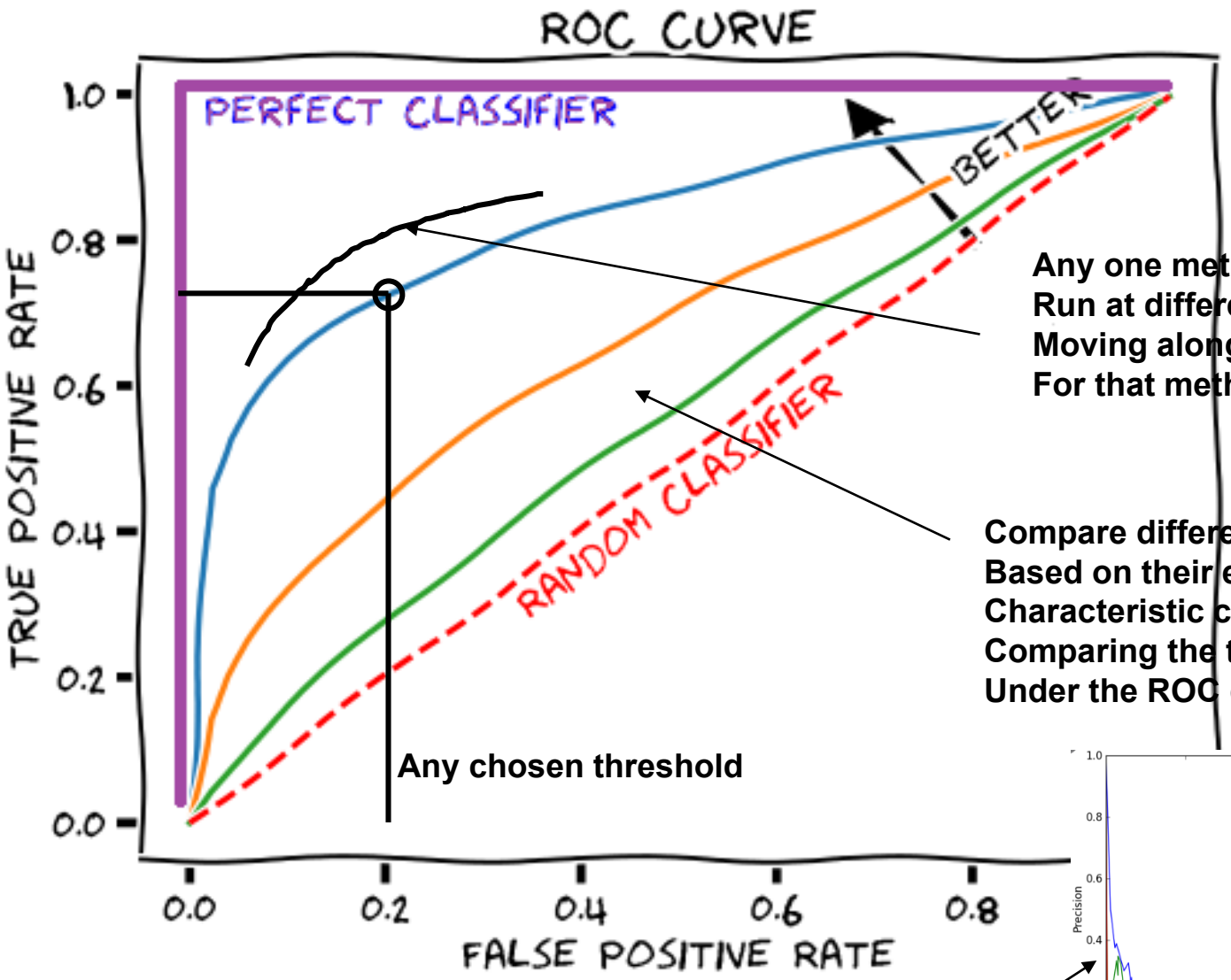


We have then done two very useful things:

1. Transformed X into a high (possibly infinite) dimensional space (where we hope are data are separable)
2. Taken dot products in this space to create scalars

Classification performance at different thresholds

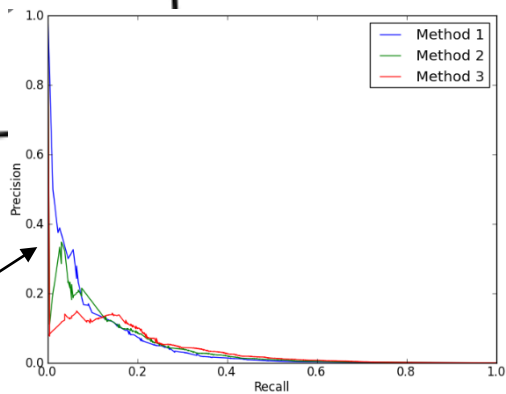
Receiver Operating Characteristic (ROC) Curve



Any one method can be Run at different thresholds, Moving along the ROC line For that method

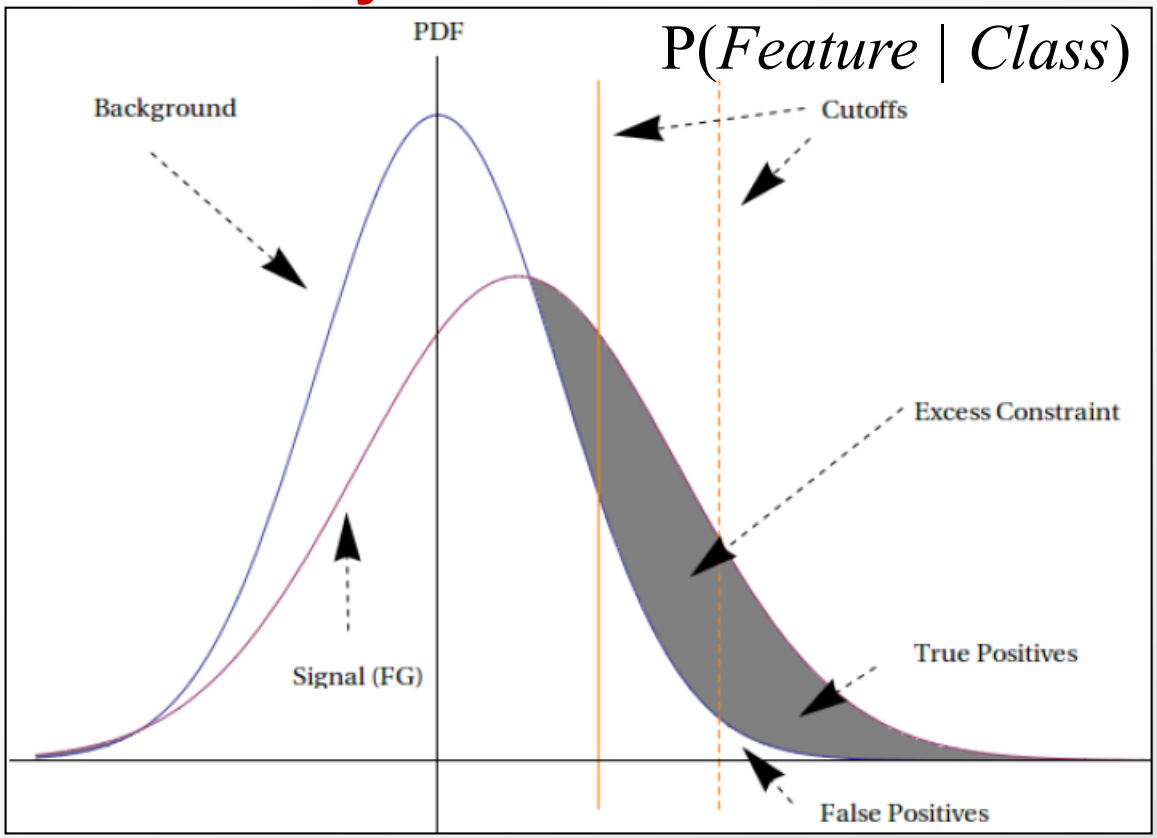
Compare different methods Based on their entire operating Characteristic curve, by Comparing the total Area Under the ROC curve (AUROC)

Any chosen threshold



When datasets are imbalanced, instead use a Precision Recall Curve

Bayesian classification with a single feature



Ex 1: DNA repair genes show higher expression during stress

Ex 2: Protein-coding regions show higher conservation levels

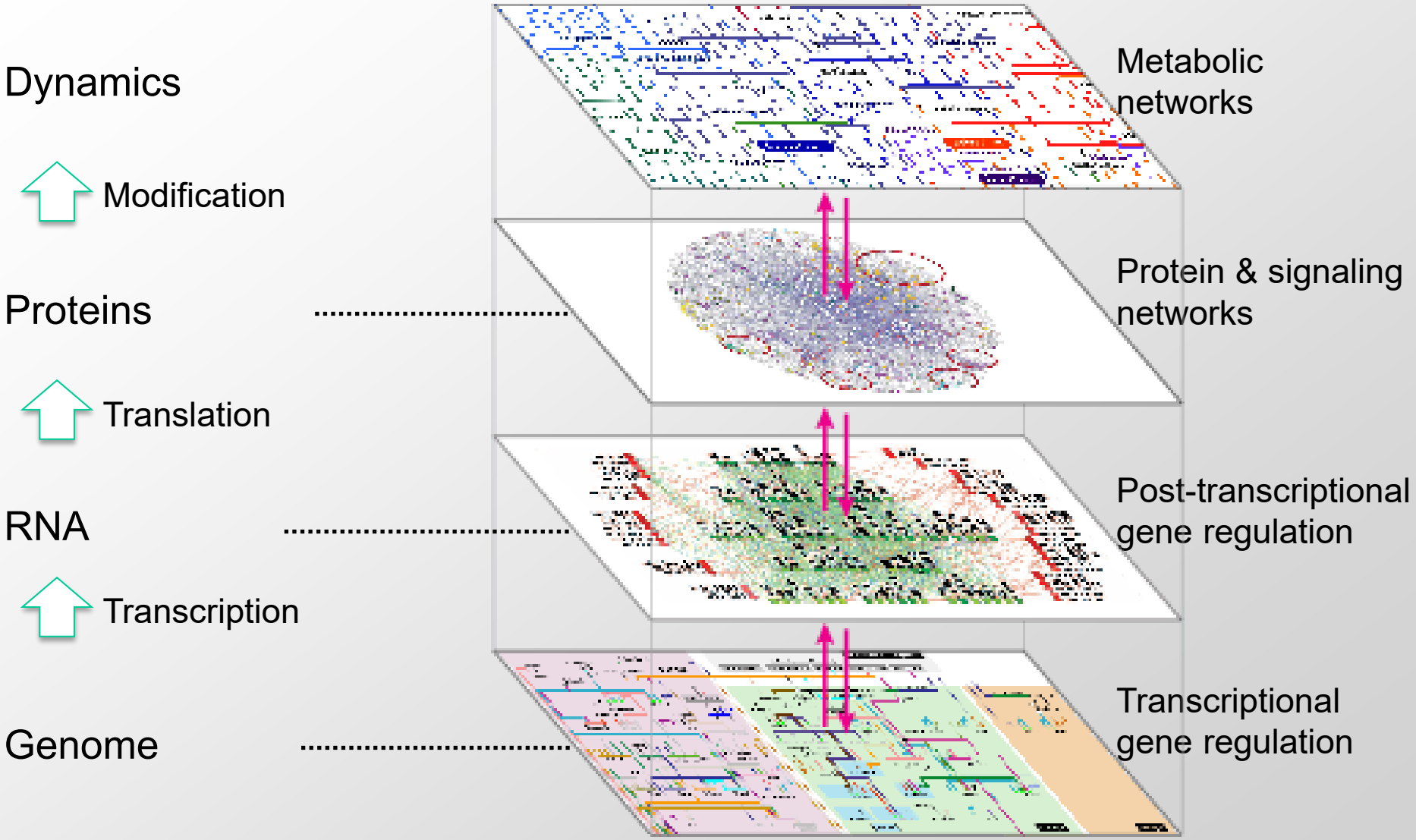
Ex 3: Regulatory regions show higher GC-content

In general: foreground signal vs. background

1. If you know both distributions, how to classify a new example
   - Picking a cutoff. Minimizing classification error. Maximizing posterior prob.
2. If you have many classified examples, how to estimate model params.
   - Parametric vs. non-parametric models. Class-conditional distributions. Priors
3. Bayes' Rule:
   - P(C|F) from P(F|C)
   - Take probability ratios

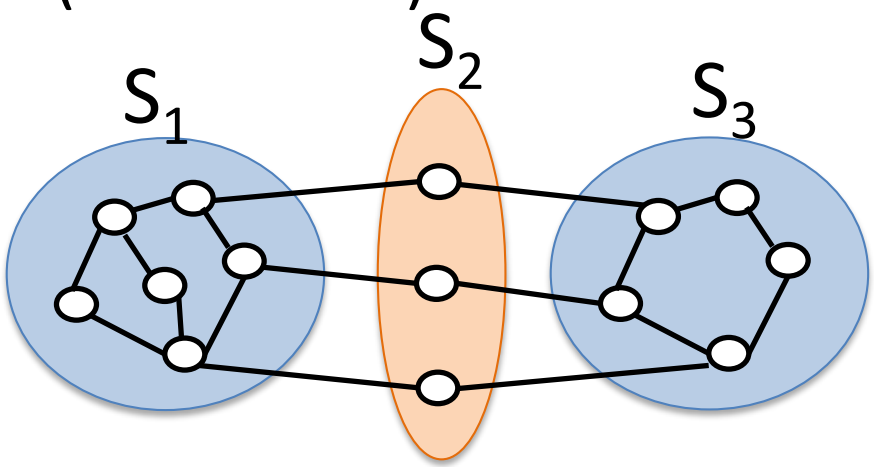
$$\underbrace{P(Class \mid Feature)}_{\textbf{Posterior}} = \frac{\overbrace{P(Feature \mid Class)}^{\textbf{Likelihood}}\overbrace{P(Class)}^{\textbf{Prior}}}{\underbrace{P(Feature)}_{\textbf{Evidence}}}$$

Network analysis



Probabilistic Network Analysis

- There are several types of networks, with different meanings, and different applications
- Networks as graphical models:
  - modeling joint probability distribution of variables using graphs
  - Bayesian networks (directed), Markov Random Fields (undirected)

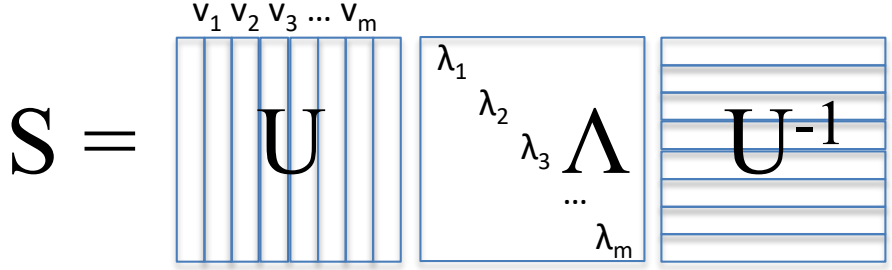


Next Lecture!

$$X_{S_1} \perp\!\!\!\perp X_{S_3} | X_{S_2}$$

Eigen/diagonal Decomposition

- Let $\mathbf{S} \in \mathbb{R}^{m \times m}$ be a **square** matrix with ***m*** **linearly independent eigenvectors** (a "non-defective" matrix)



- **Theorem**: Exists an **eigen decomposition**

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$$

diagonal

Unique for distinct eigen-values

  - (cf. matrix diagonalization theorem)

- Columns of ***U*** are **eigenvectors** of ***S***

- Diagonal elements of $\mathbf{\Lambda}$ are **eigenvalues** of $\mathbf{S}$

$$\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m), \quad \lambda_i \geq \lambda_{i+1}$$

L1 (Lasso) vs. L2 (Ridge) regularization

- L1 regularization (lasso)

Linear

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_j \left(t(\mathbf{x}_j) - \sum_i w_i h_i(\mathbf{x}_j) \right)^2 + \lambda \sum_{i=1}^{k} |w_i|$$

- L2 regularization (ridge)

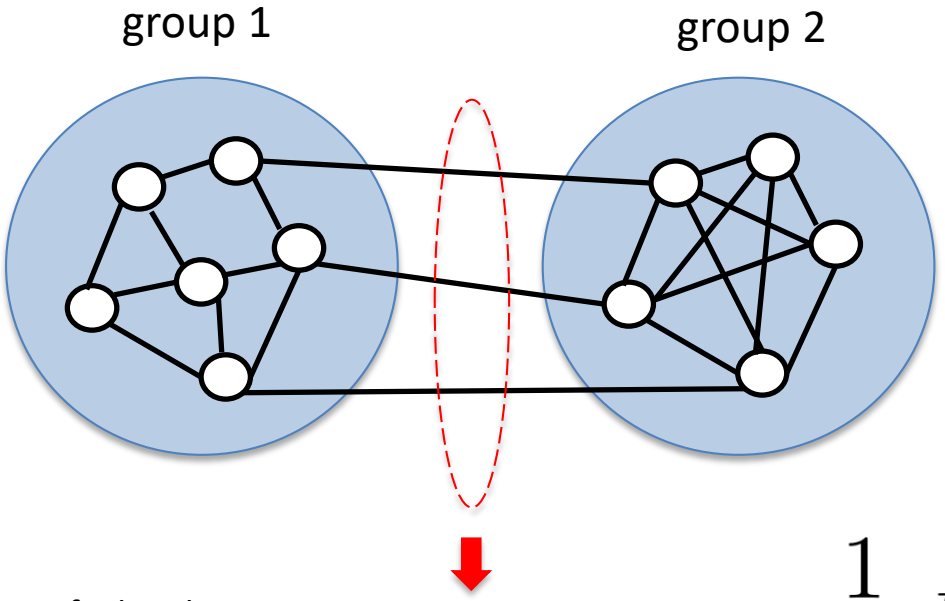
Quadratic

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_j \left(t(\mathbf{x}_j) - \sum_i w_i h_i(\mathbf{x}_j) \right)^2 + \lambda \sum_{i=1}^{k} w_i^2$$

- L1 pros:
  - More robust (outliers don't matter more than small diffs)
- L1 cons:
  - Less stable gradient ascent
  - Multiple solutions

- L2 pros:
  - Always one solution
  - More stable gradient ascent
- L2 cons:
  - Less robust (square diffs, outliers have strong effect)

Laplacian graph clustering

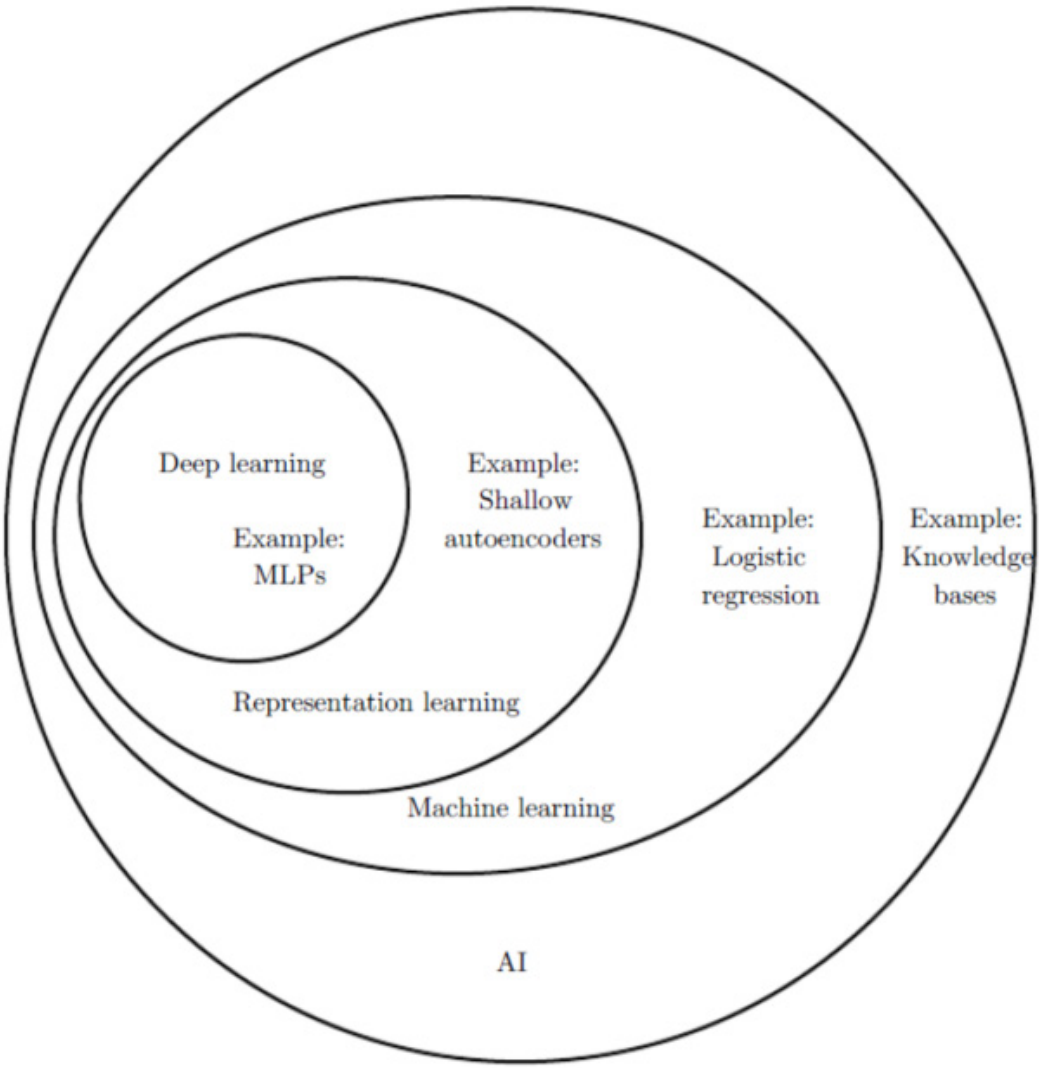


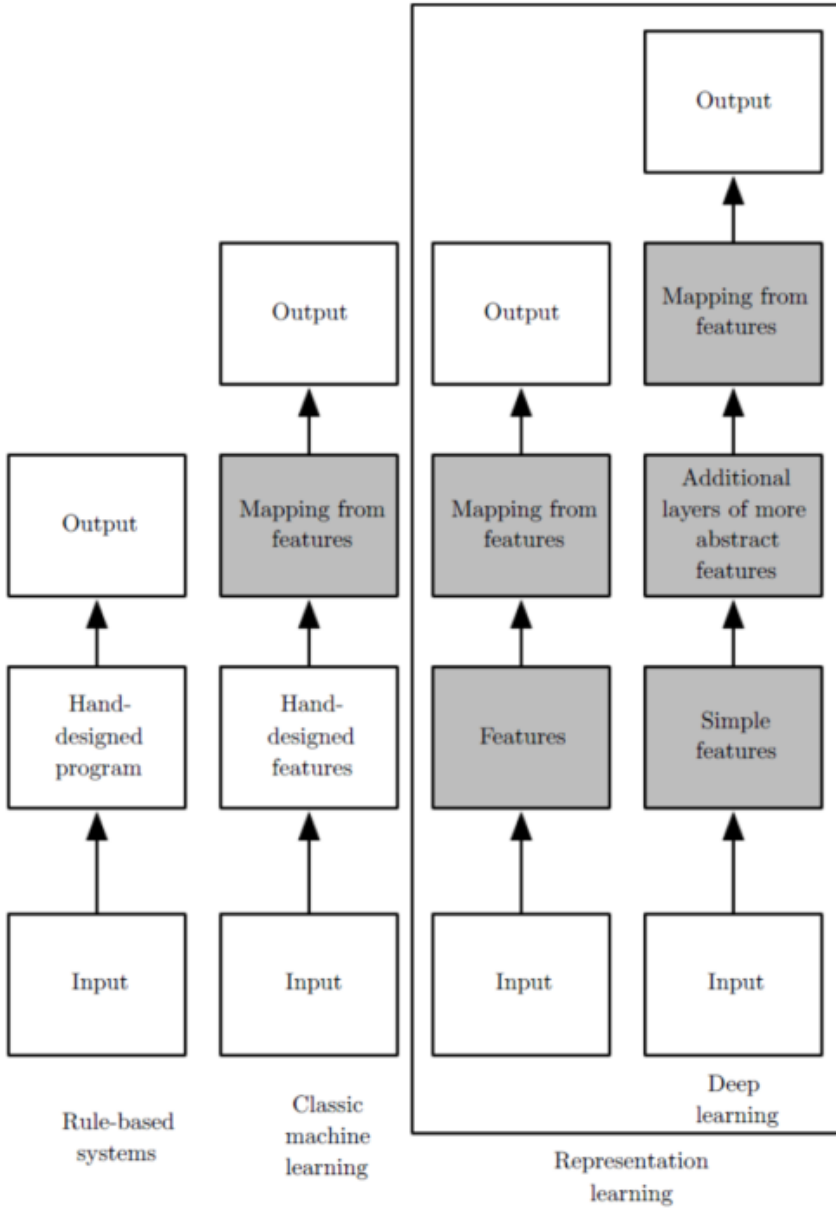
node *i* in group 1 => s_i=1
node *i* in group 2 => s_i=-1

# of edges between groups = $\frac{1}{4}\mathbf{s}^t L\mathbf{s}$

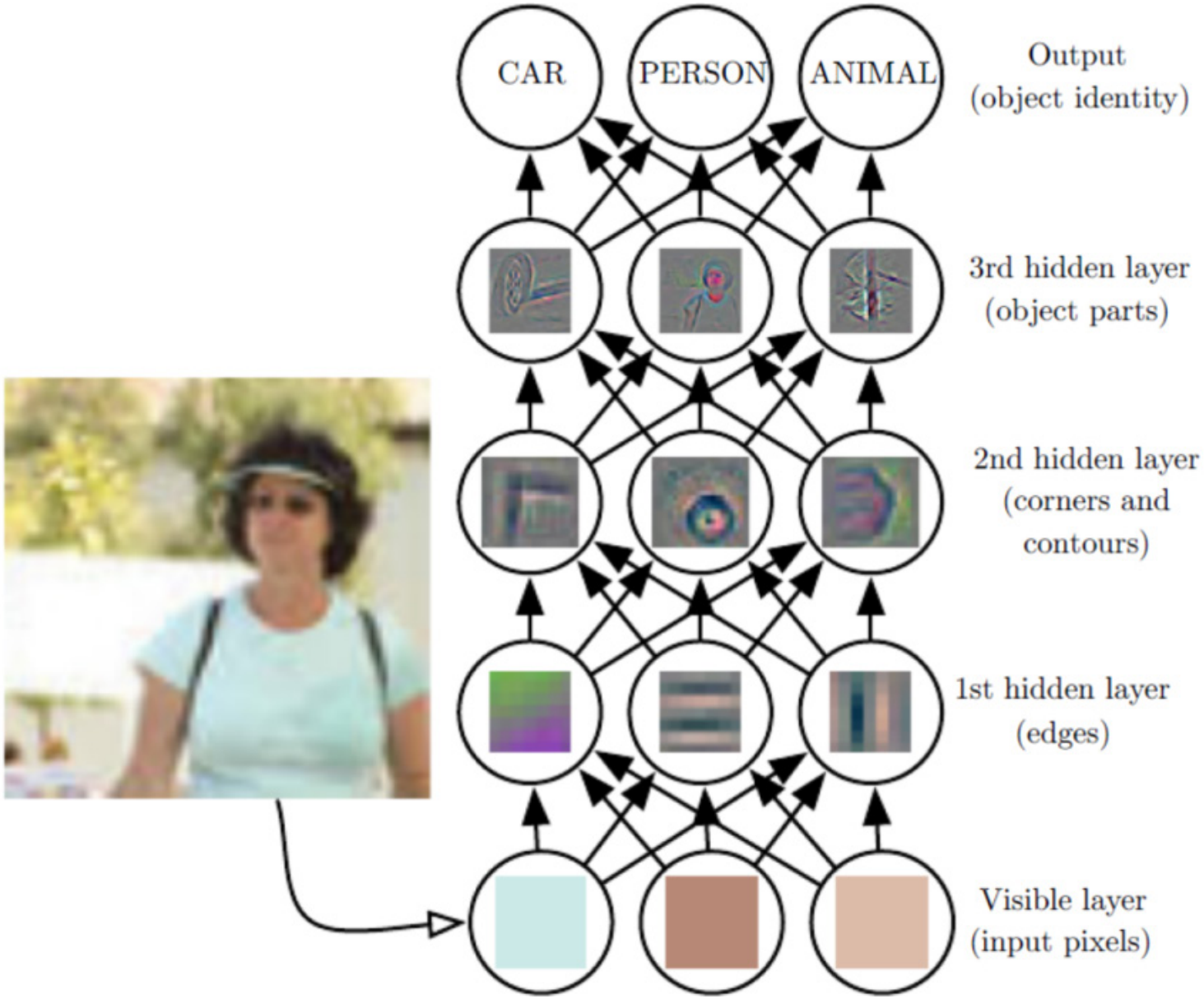
- Choose vector *s* to minimize the error term
- A trivial solution: if s=(1,1,…,1), error is zero.
- *A non-trivial solution: s parallel to the second eigenvector of L (why?)*

Not all “learning” is “deep”



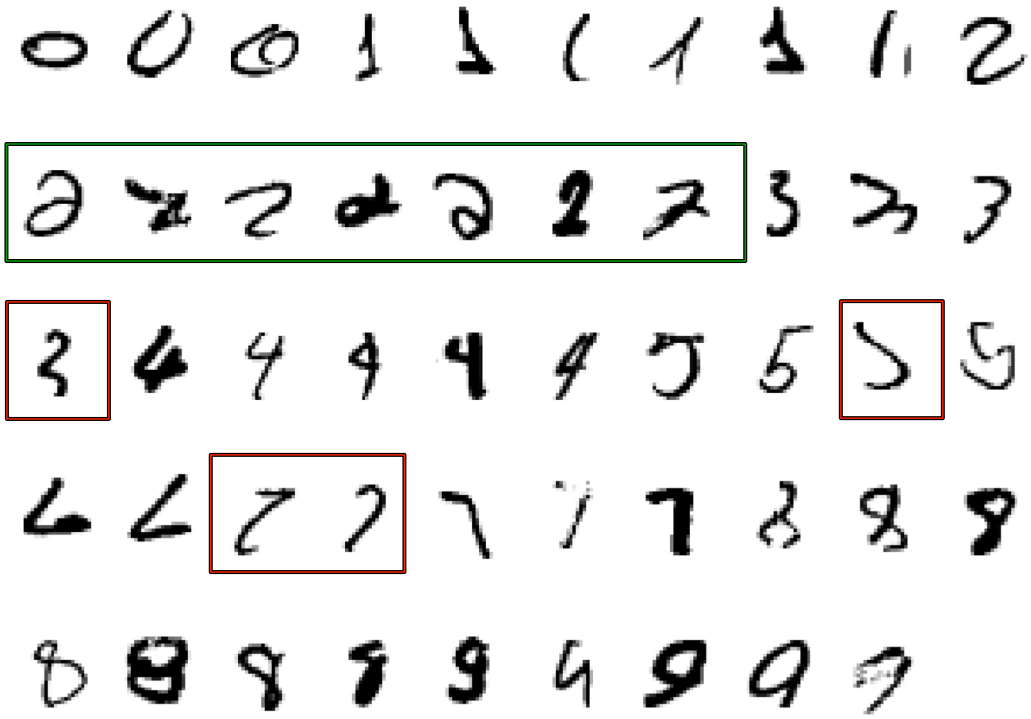


Deep learning ➔ many **layers** of **abstraction**

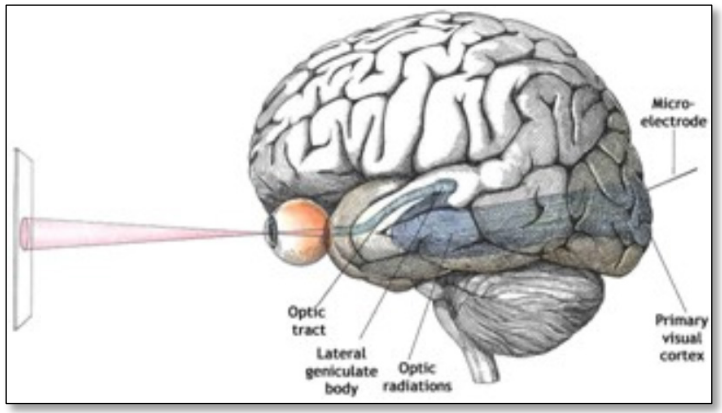


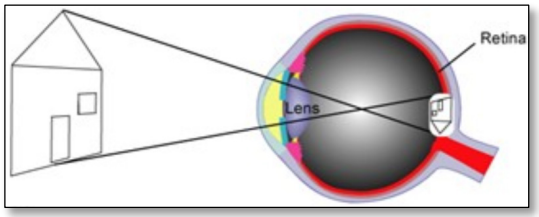
How do ‘we’ recognize patterns?

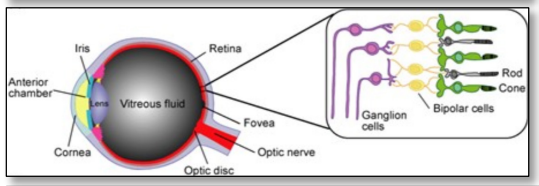
For example, it’s easy for us to recognize a hand-written ‘2’ but not so trivial for machines:

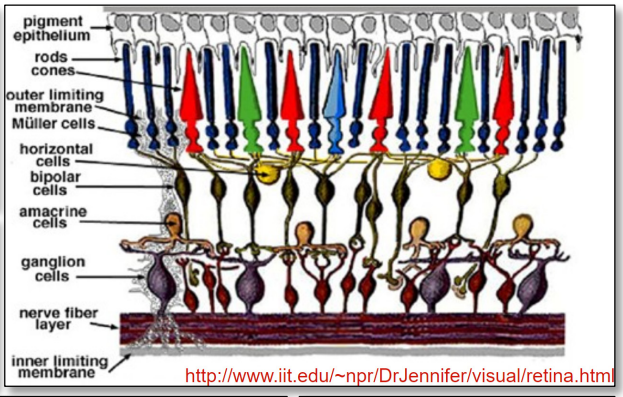


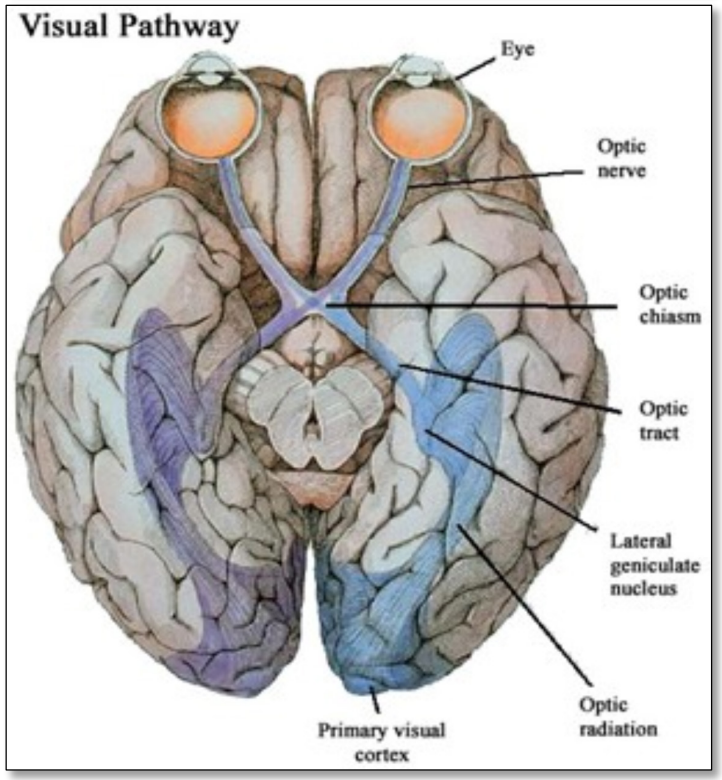
Inspiration: human/animal visual cortex

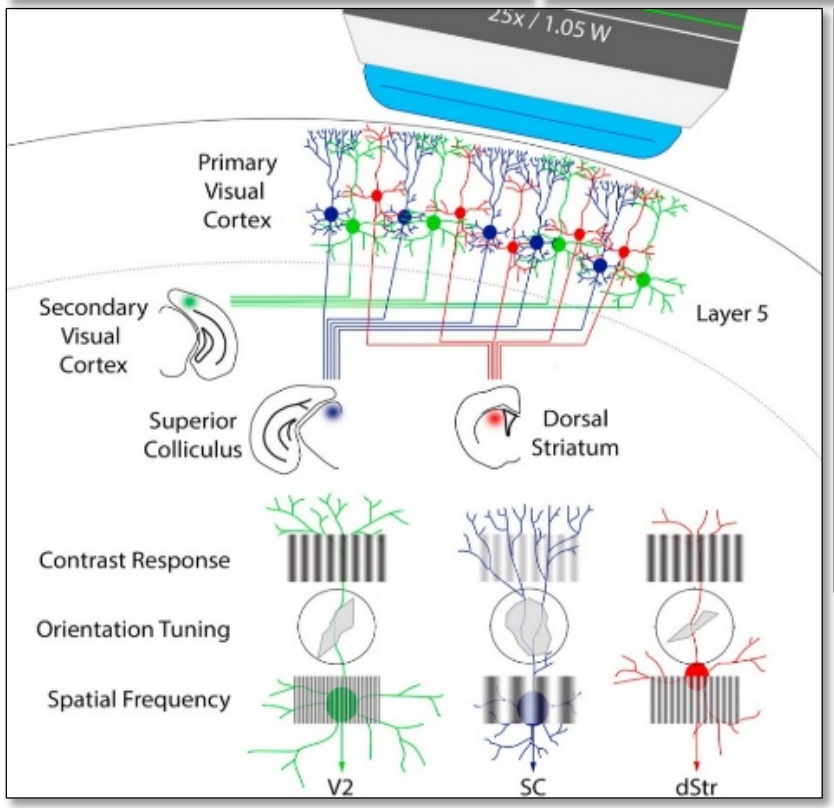


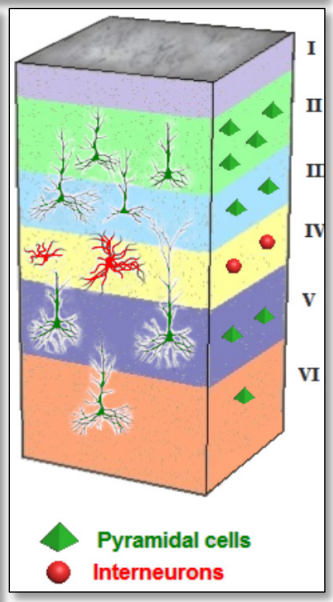






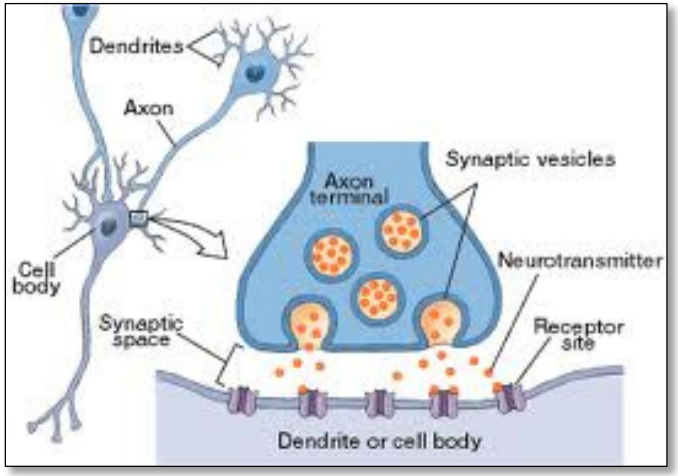


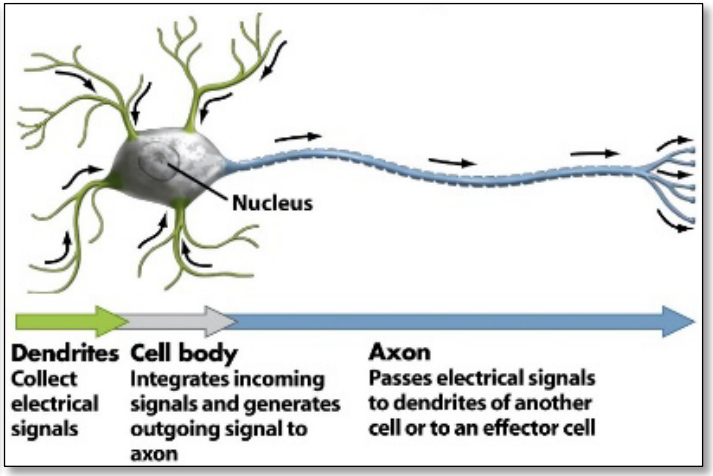


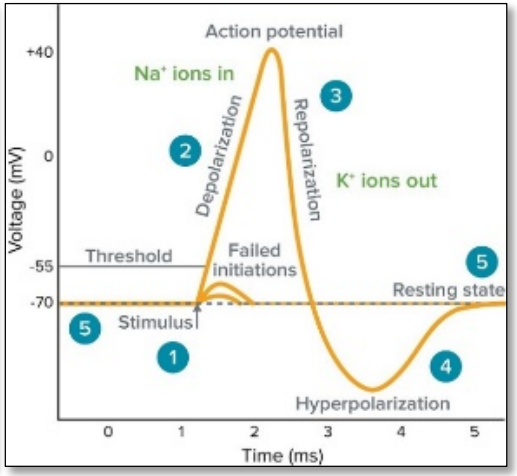


- **Layers of neurons: pixels, edges, shapes, primitives, scenes**
- **E.g. Layer 4 responds to bands w/ given slant, contrasting edges**

Primitives: Neurons & action potentials



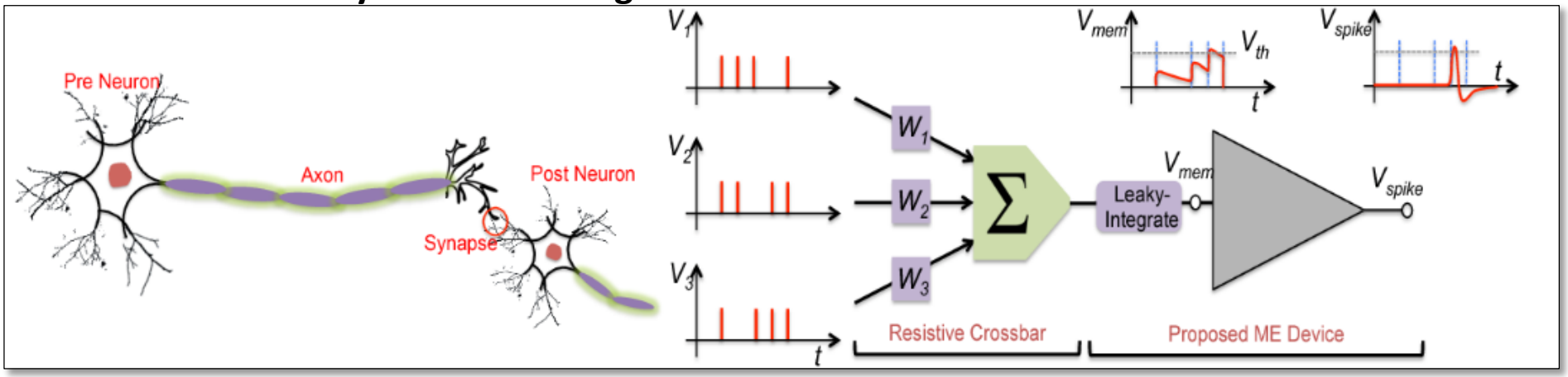




- Chemical accumulation across dendritic connections
- Pre-synaptic axon
  → post-synaptic dendrite
  → neuronal cell body

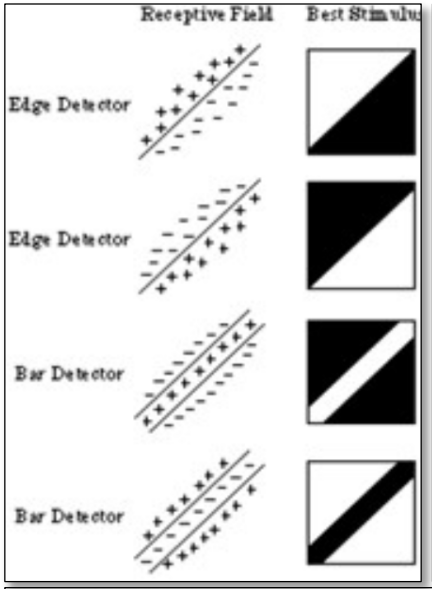
- Each neuron receives multiple signals from its many dendrites
- When threshold crossed, it fires
- Its axon then sends outgoing signal to downstream neurons

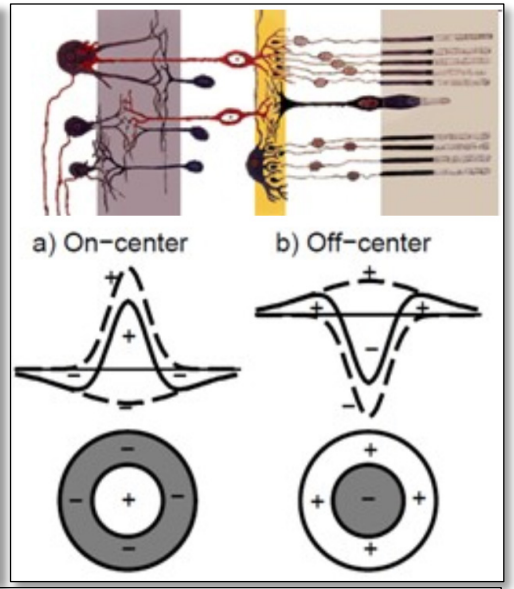
- Weak stimuli ignored
- Sufficiently strong cross activation threshold
- Non-linearity within each neuronal level

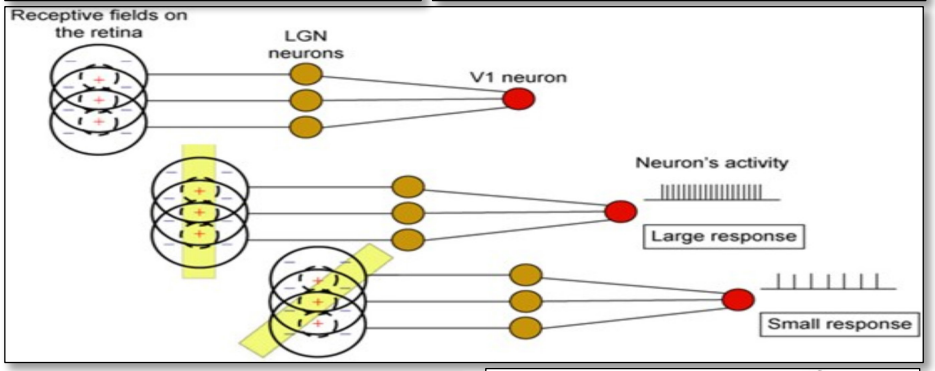


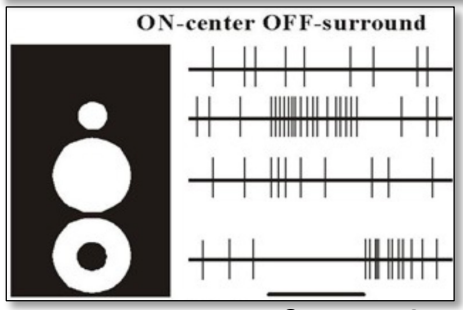
- Neurons connected into circuits (neural networks): emergent properties, learning, memory
- Simple primitives arranged in simple, repetitive, and extremely large networks
- 86 billion neurons, each connects to 10k neurons, 1 quadrillion (10^{12}) connections

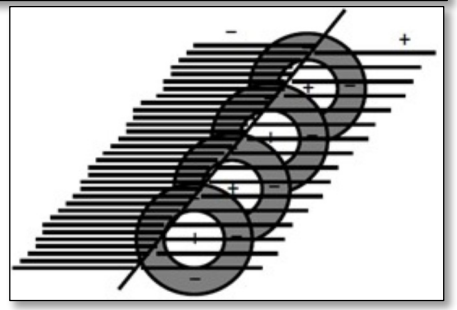
Abstraction layers: edges, bars, dir., shapes, objects, scenes

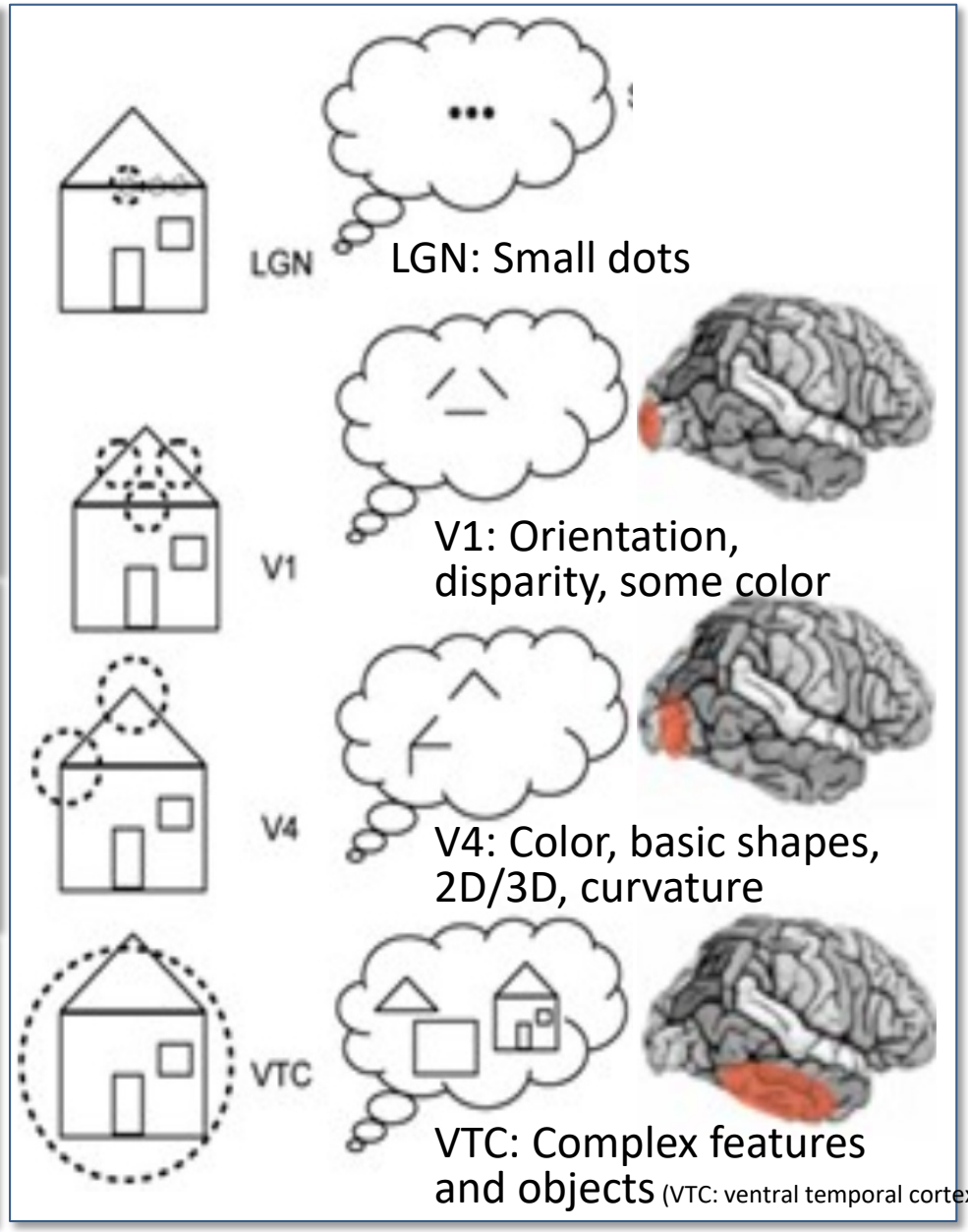






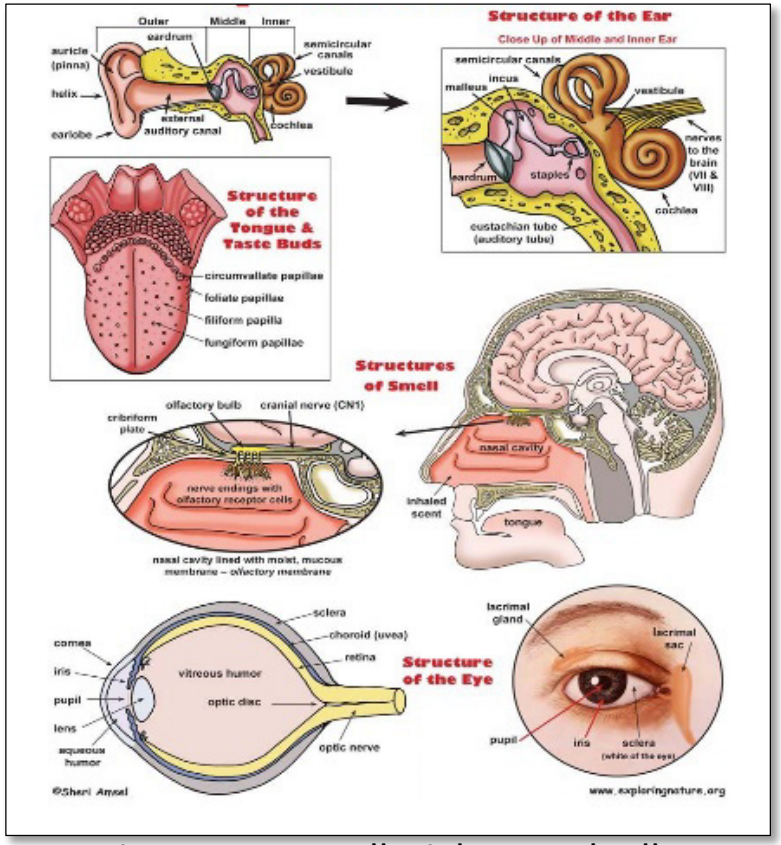




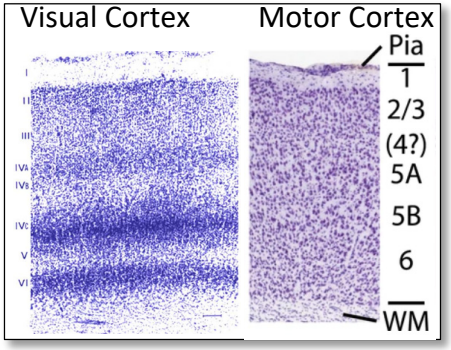


- **Primitives of visual concepts encoded in neuronal connection in early cortical layers**
- **Abstraction layers ⇔ visual cortex layers**
- **Complex concepts from simple parts, hierarchy**

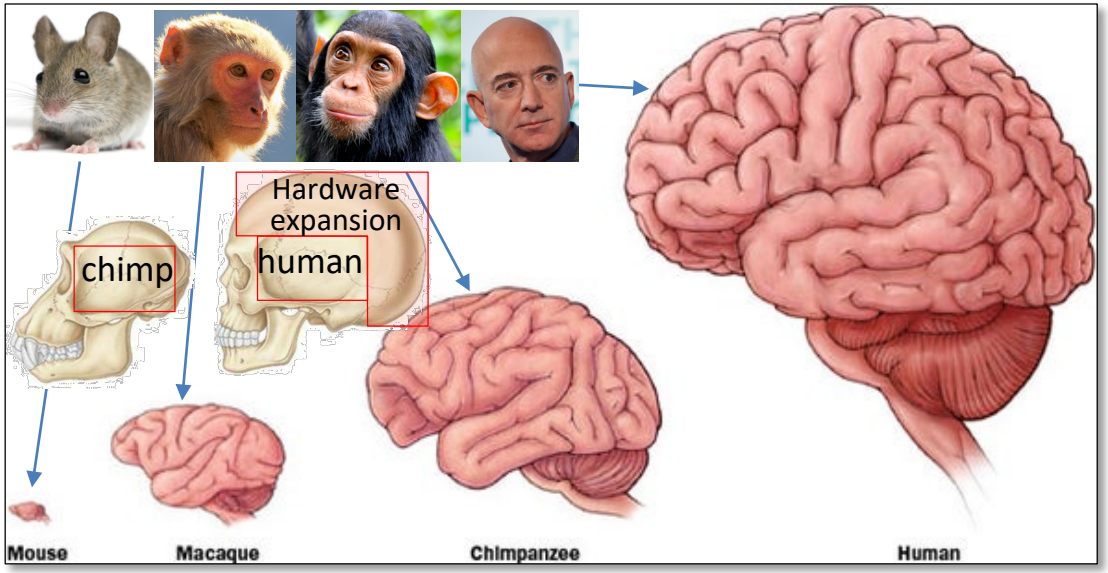
General "learning machine", reused widely



- Hearing, taste, smell, sight, touch all re-use similar learning architecture



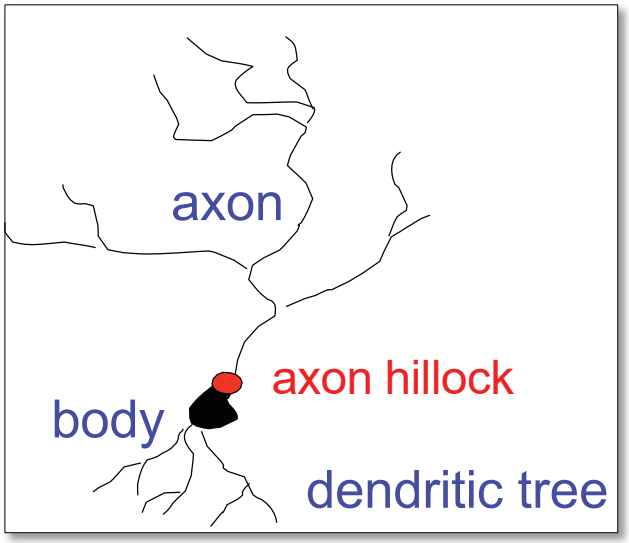
- Interchangeable circuitry
- Auditory cortex learns to 'see' if sent visual signals
- Injury area tasks shift to uninjured areas



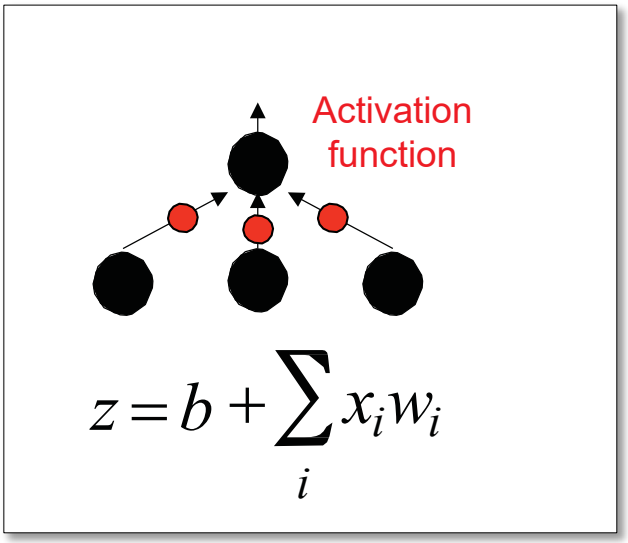
- Massive recent expanse of human brain has re-used a relatively simple but general learning architecture

- Not fully-general learning, but well-adapted to our world
- Humans co-opted this circuitry to many new applications
- Modern tasks accessible to any homo sapiens (<70k years)
- ML primitives not too different from animals: more to come?

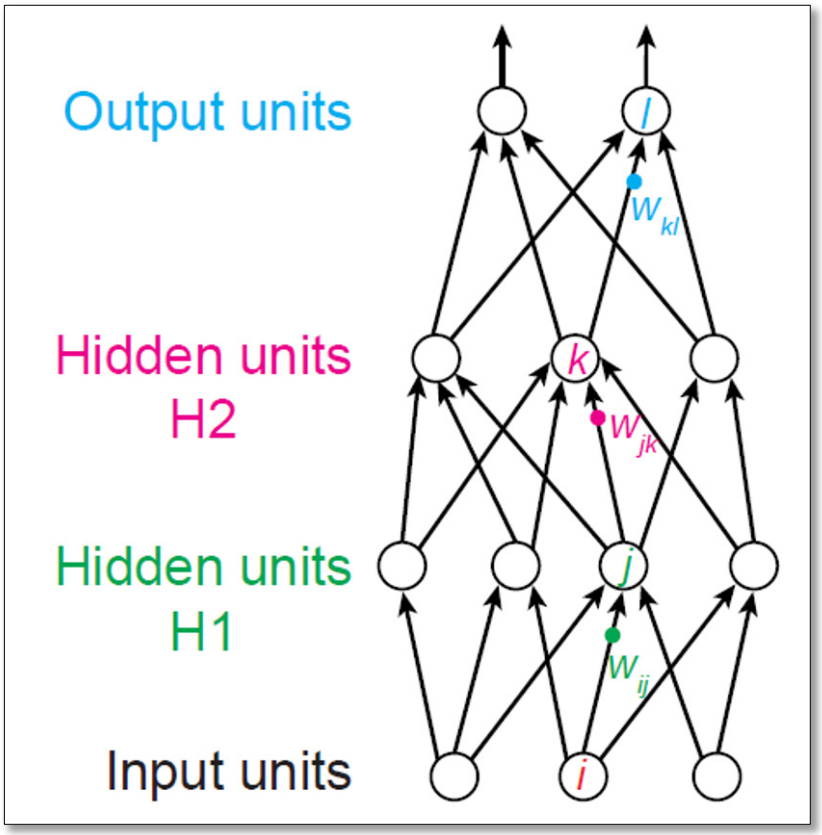
How the brain works inspired artificial “neural" networks



Biological neuron



Artificial perceptron



Neural Network (e.g. 4-layers ‘deep’)

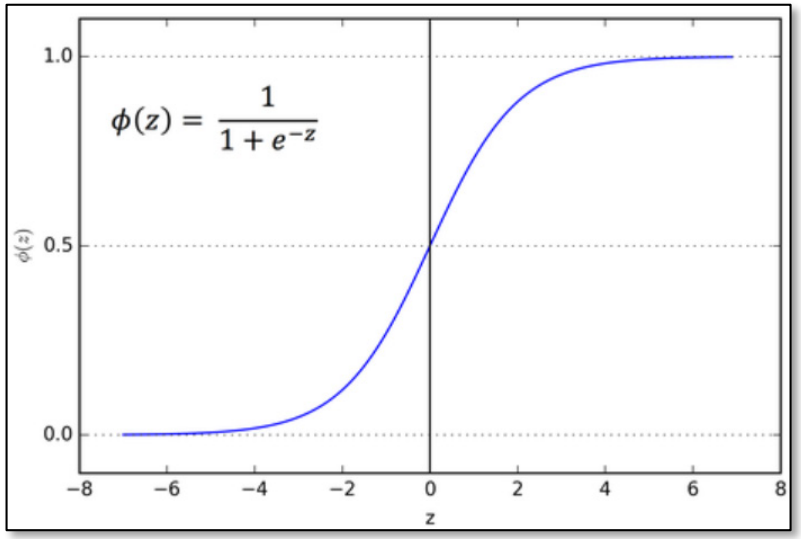
Deep multi-layer neural networks can ‘learn’ almost any function

Learning non-linear functions: non-linear "activation" units

Sigmoid unit

$$S(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1}.$$

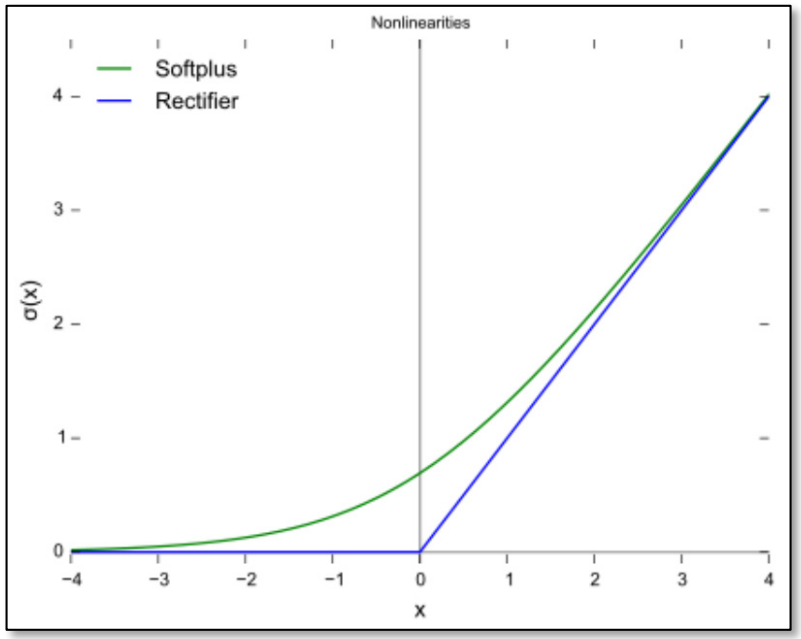
Maps to [0,1], saturation



Softplus unit

$$\operatorname{softmax}(\boldsymbol{x})_i = \frac{\exp(x_i)}{\sum_{j=1}^{n} \exp(x_j)}.$$

No saturation, smooth transition

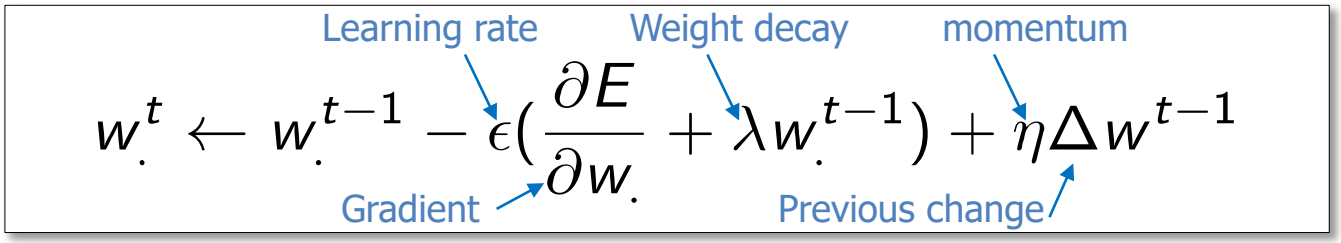


Rectified linear unit (ReLU)

$$f(x) = x^+ = \max(0, x)$$

Easy to optimize, generalizes well

Gradient-based learning: use derivative to update weights



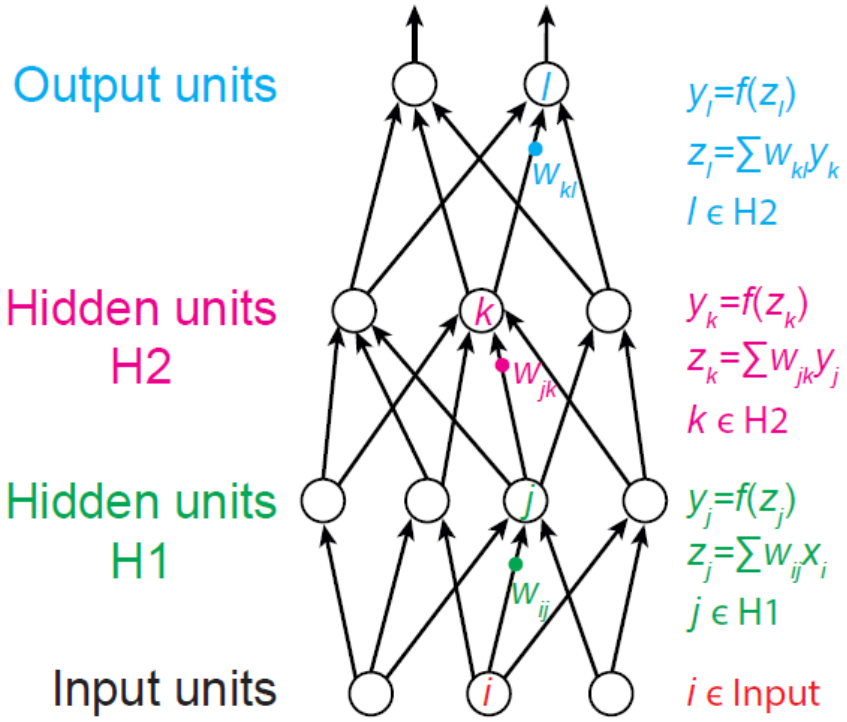
where

- Gradient descent: **∂E/∂w** = partial derivative of error E wrt w
- ϵ = **learning rate** (e.g. <0.1), needed to not overshoot the optimal solution
- λ = **weight decay**, penalizes large weights to prevent overfitting
- η = **momentum**, based on magnitude+sign of previous update (Δw^{t-1}); when direction of update is consistent ➔ faster convergence

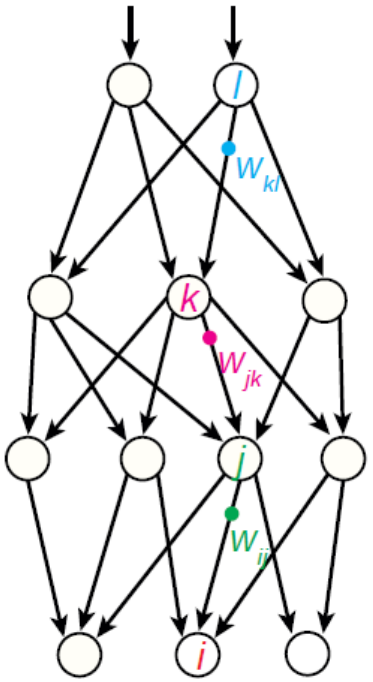
Using only a subset of samples at a time:

- ***Stochastic*** **gradient descent (SGD)**: speed up computation
  - Randomly sample subset of samples
  - Update the weights using only that subset
- **On-line learning:** Update gradient using only 1 training data point each time

Back-propagation of error across multiple layers



Forward pass



Backpropagation

$$\frac{\partial E}{\partial y_l} = y_l - t_l$$

$$\frac{\partial E}{\partial z_l} = \frac{\partial E}{\partial y_l}\frac{\partial y_l}{\partial z_l}$$

$$\frac{\partial E}{\partial w_{kl}} = \frac{\partial E}{\partial z_l}\frac{\partial z_l}{\partial w_{kl}}$$

$$\frac{\partial E}{\partial y_k} = \sum_{l \in \text{out}} \frac{\partial z_l}{\partial y_k}\frac{\partial E}{\partial z_l}$$

$$\frac{\partial E}{\partial z_k} = \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial z_k}$$

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial z_k}\frac{\partial z_k}{\partial w_{jk}}$$

$$\frac{\partial E}{\partial y_j} = \sum_{k \in \text{H2}} \frac{\partial z_k}{\partial y_j}\frac{\partial E}{\partial z_k}$$

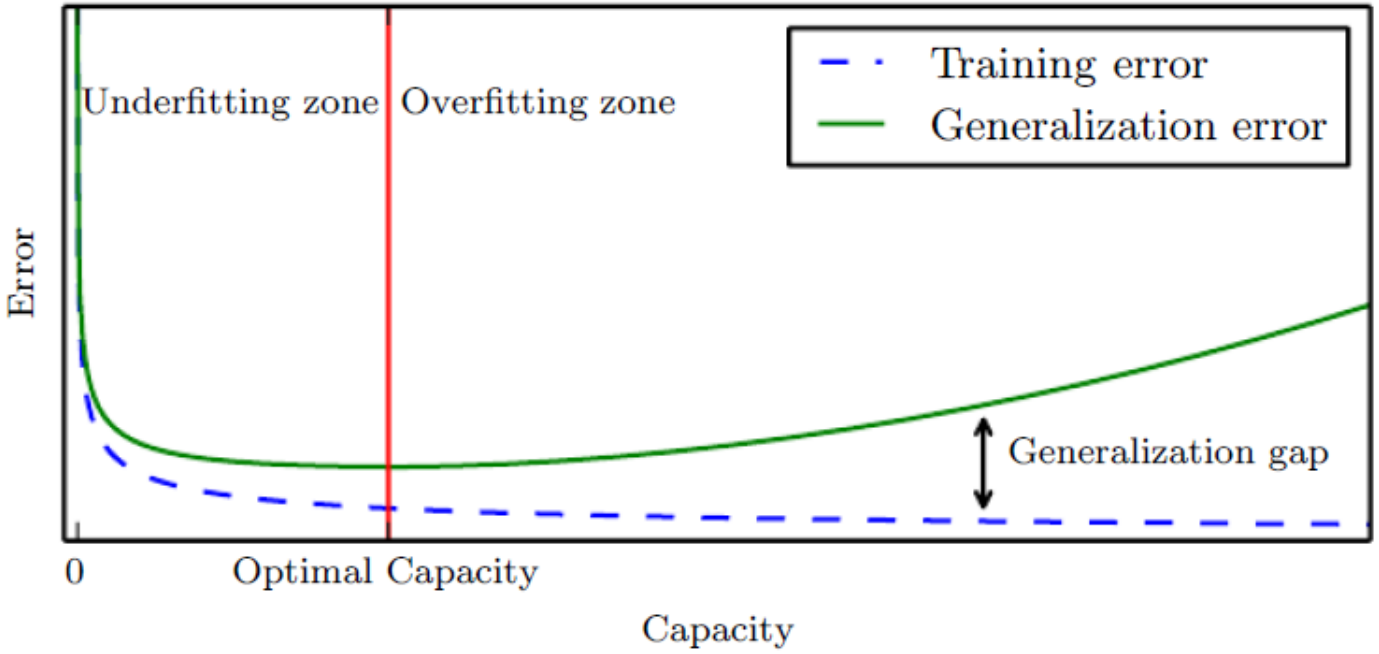
$$\frac{\partial E}{\partial z_j} = \frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial z_j}$$

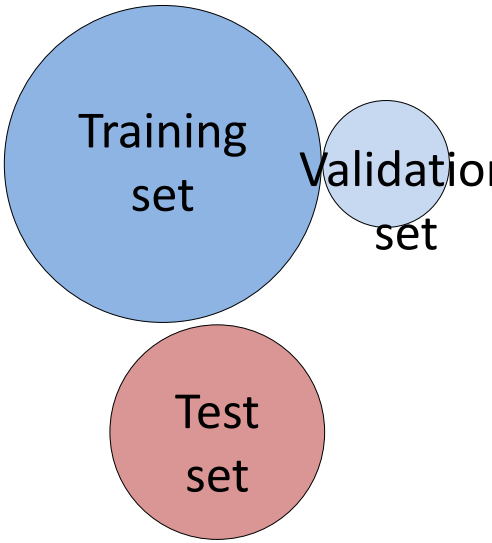
$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial z_j}\frac{\partial z_j}{\partial w_{ij}}$$

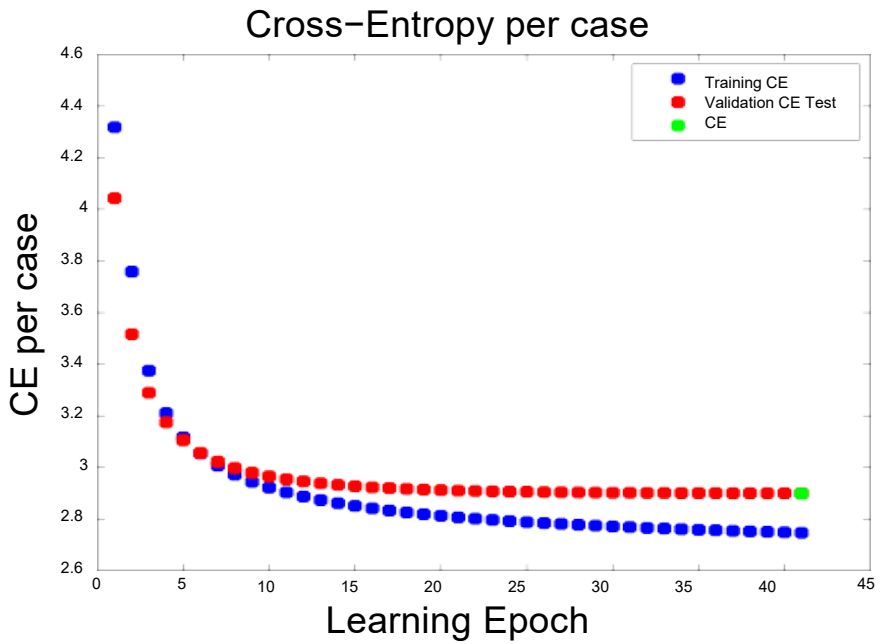
$$\text{Update: } w_{\cdot}^{t} \leftarrow w_{\cdot}^{t-1} - \epsilon\left(\frac{\partial E}{\partial w_{\cdot}} + \lambda w_{\cdot}^{t-1}\right) + \eta \Delta w^{t-1}$$

[Rumelhart and Hintont, 1986, LeCun et al., 2015]

Overfitting and its remedies: validation set, early stopping





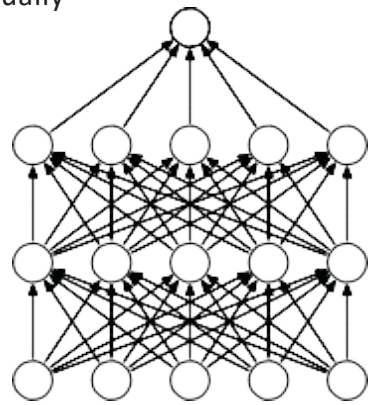


Leave out small "validation set"

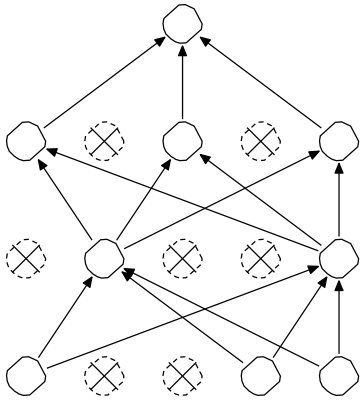
- not used to train the model
- used to evaluate model at each epoch/iteration (VCE, Validation cross-entropy)
- Stop when VCE increases, prevent overfitting

Avoid overfitting: Regularization, Dropout training

Conceptually

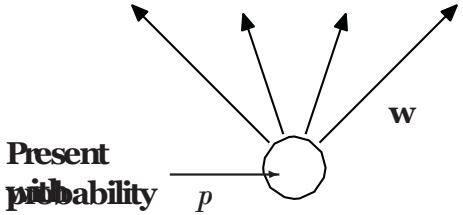


(a) Standard Neural Net

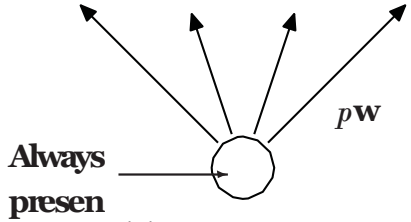


(b) After applying dropout.

In practice



(a) At training time

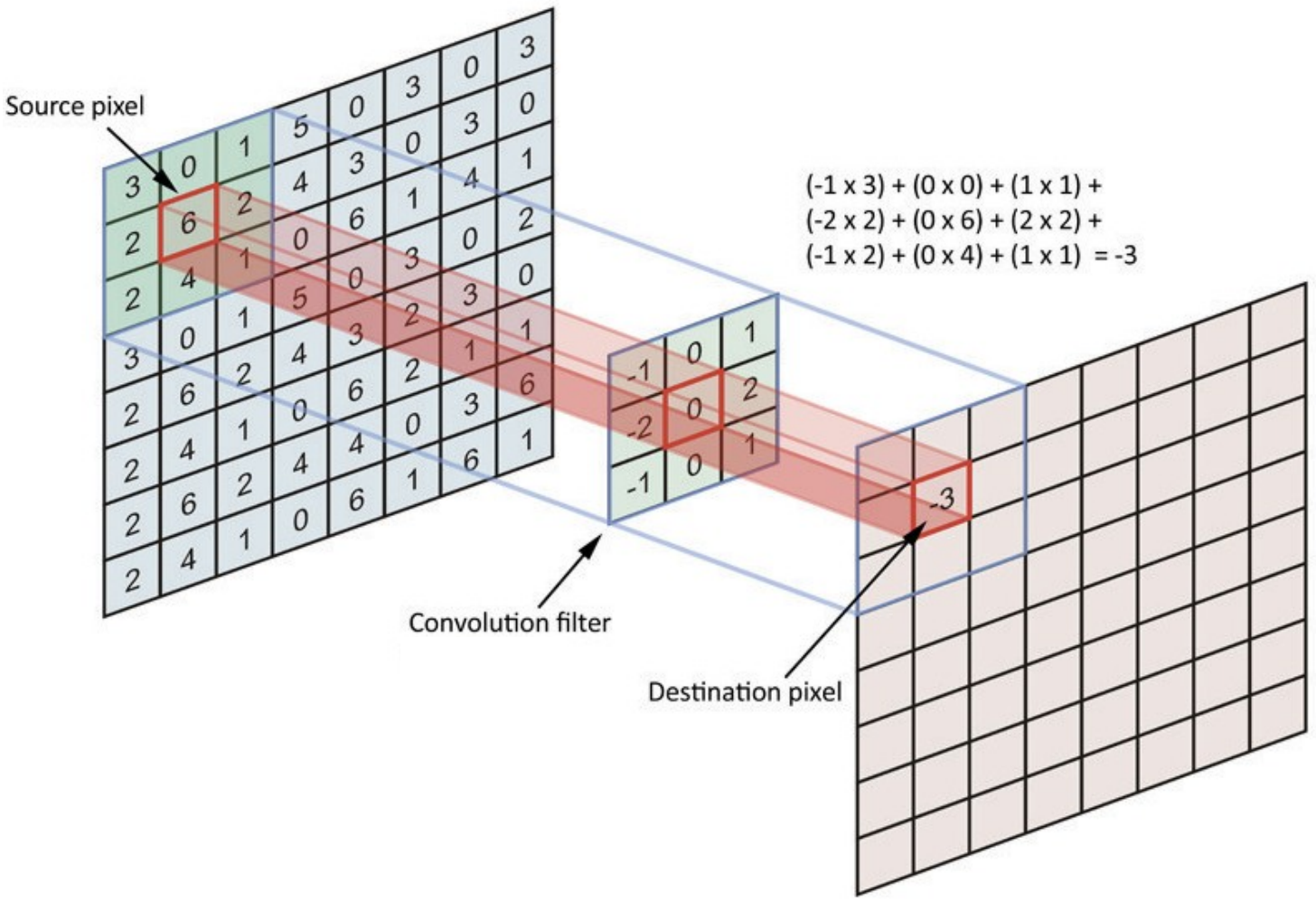


(b) At test time

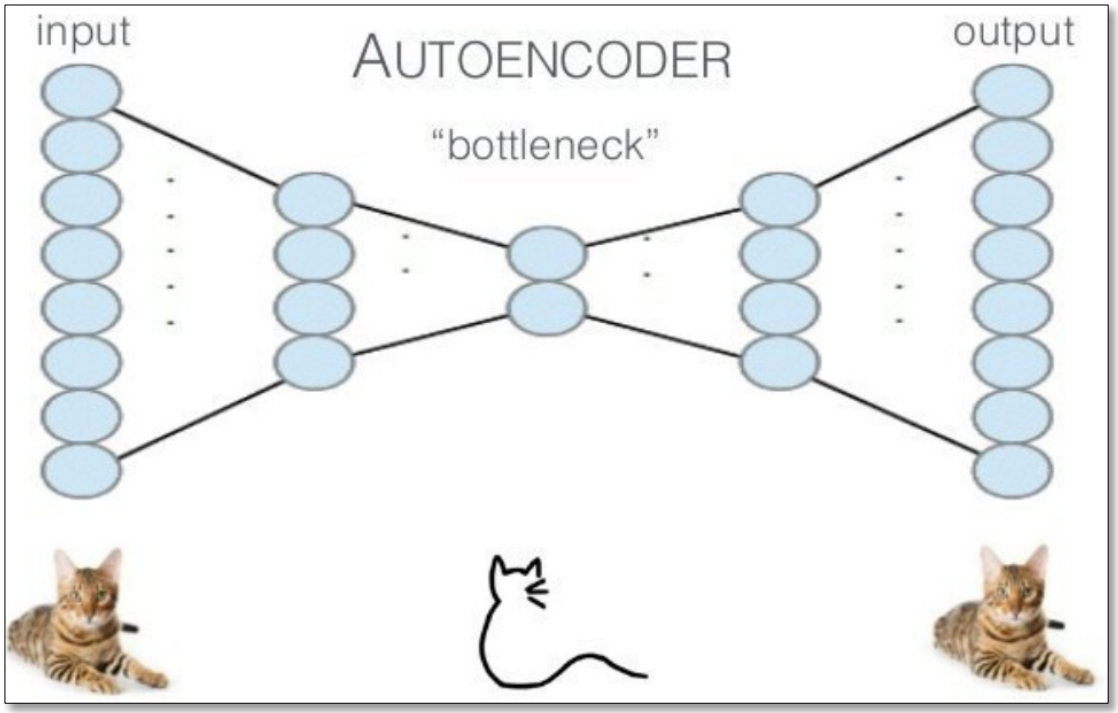
[Srivastava et al., 2014]

- Regularization: recall linear (L1, lasso), quadratic (L2, ridge) or combination (elastic net) on parameters.
- Dropout is achieves parameter minimization in deep learning, by randomly dropping hidden units for different input points with some probability p.
- Train sub-network by back-propagation as usual.
- Equivalent to bagging (**b**ootstrap **agg**regat**ing**) an exponential number of models, each of which is missing some nodes
- Provides powerful regularization method, avoids overfitting in practice

Convolutional filter



Autoencoder: dimensionality reduction with neural net

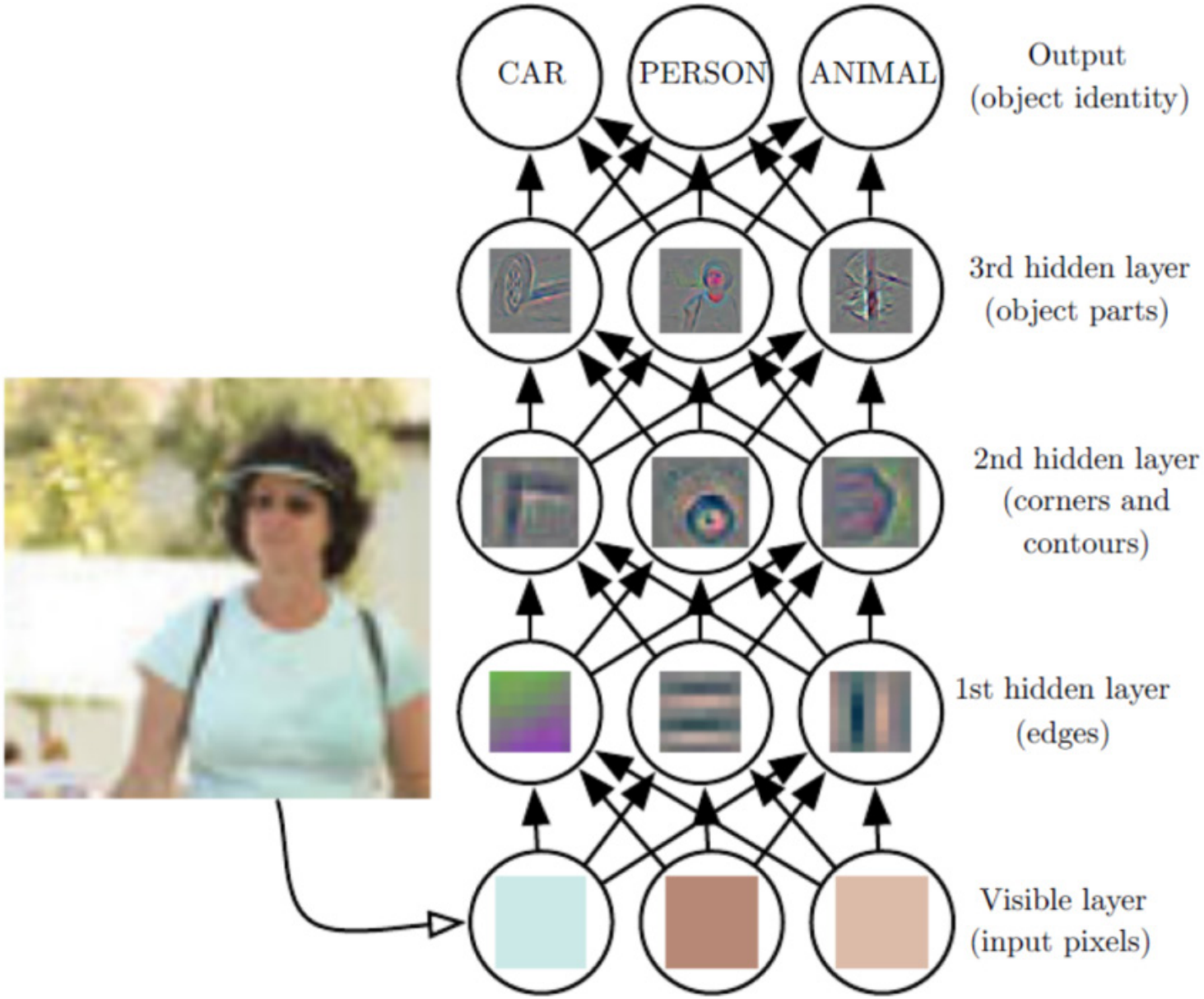


- Tricking a **supervised** learning algorithm to work in **unsupervised** fashion
- Feed input as output function to be learned. **But!** Constrain model complexity

- **Pretraining** with RBMs to learn representations for future supervised tasks. Use RBM output as "data" for training the next layer in stack
- After pretraining, "unroll" RBMs to create deep autoencoder
- Fine-tune using backpropagation

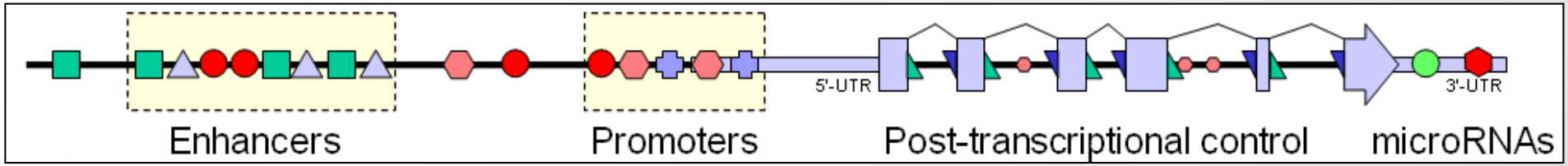
[Hinton *et al*, 2006]

Deep learning ➔ many **layers** of **abstraction**



Intro to Genomics

The components of genomes and gene regulation



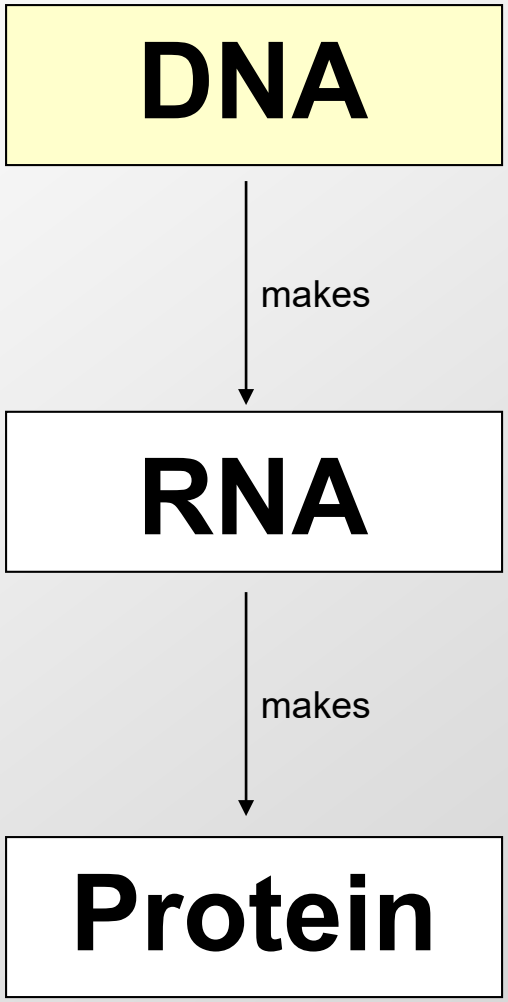
Goal: A systems-level understanding of genomes and gene regulation:

- The genome: Map reads, align genes/genomes, assembly strategies
- The genes: Protein-coding exons, introns, non-coding RNA, RNA folding
- The control regions: Promoters, enhancers, insulators, chromatin states
- The actual words: Regulatory motifs, high-resolution accessibility maps
- The regulators: Transcription factors, chromatin modifiers, nucleosomes
- The dynamics: Changing maps between cell types, across development
- The networks: regulator→enhancer→target, ChIP-seq, correlated activity
- The grammars: TF/motif/mark combinations, predictive models
- Human variation: Human diversity, population genomics, linkage maps
- Evolution: Phylogenetics, phylogenomics, coalescent, human ancestry
- GWAS/QTLs: Genome variation ⇔ organismal/molecular phenotypes
- Disease: Personal (epi)genomics, pharmacogenomics, synthetic biology

Biology primer

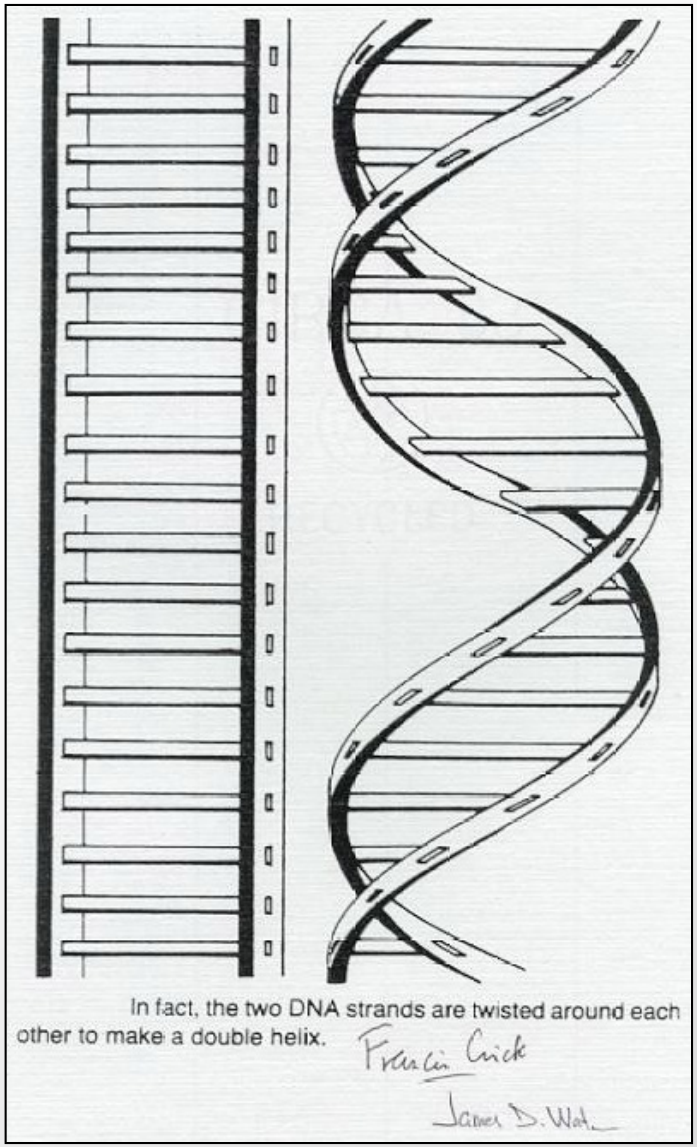
Quick introduction to molecular biology
and information transfer within the cell

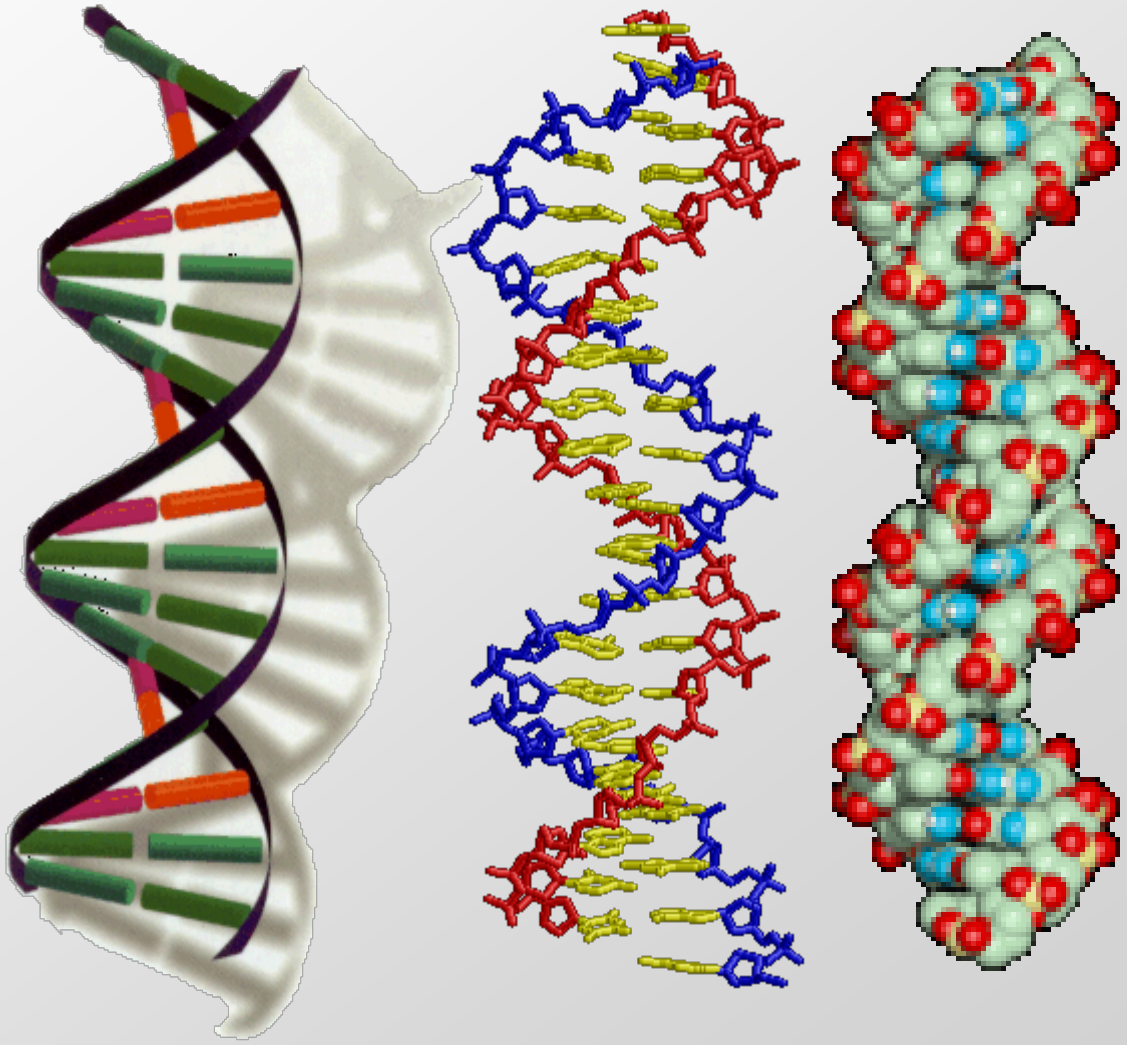
“Central dogma” of Molecular Biology



DNA: The double helix

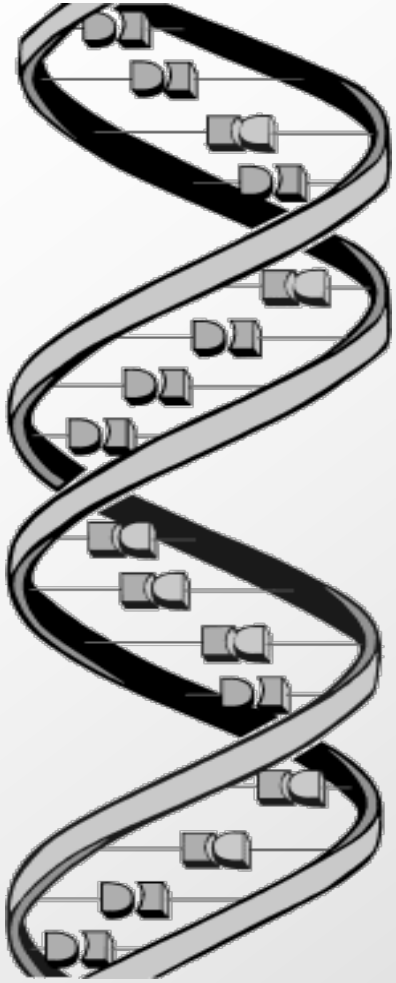
- The most noble molecule of our time

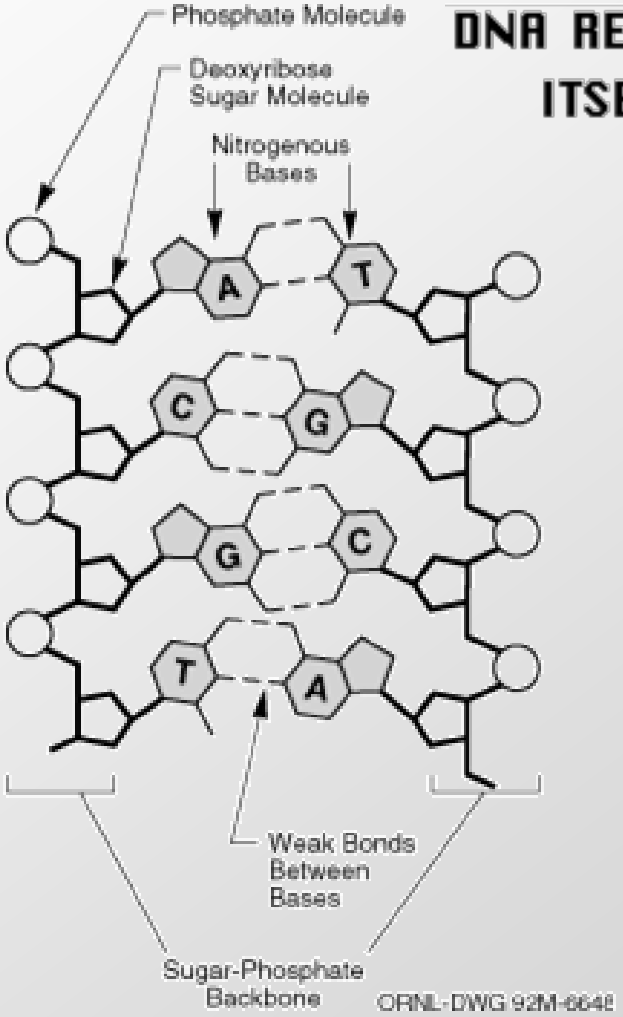


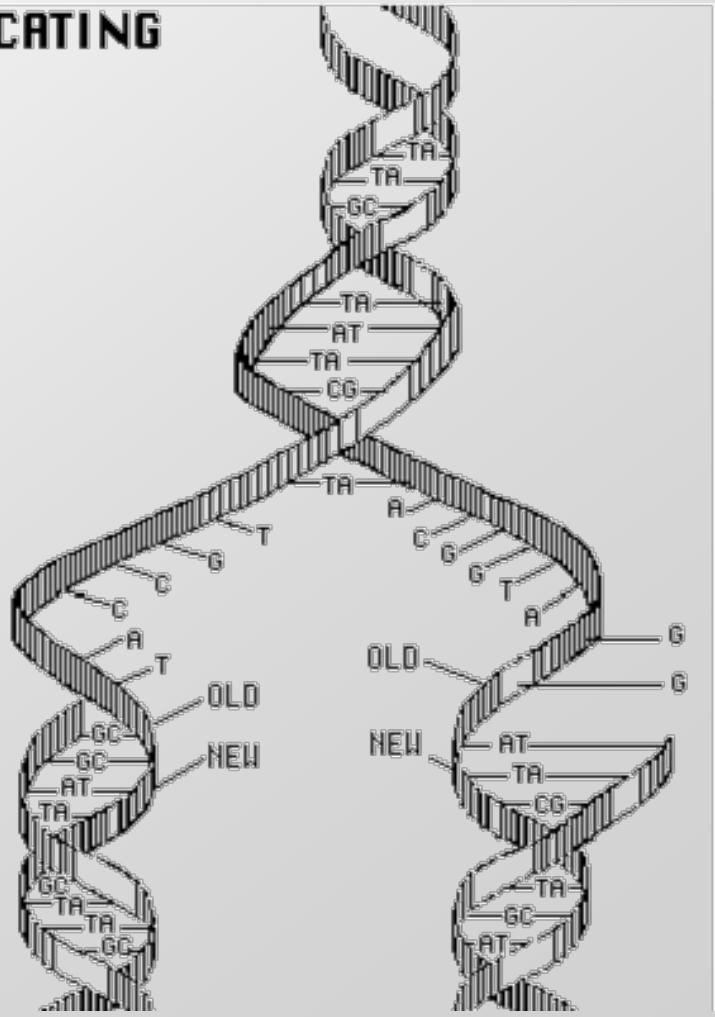


DNA: the molecule of heredity

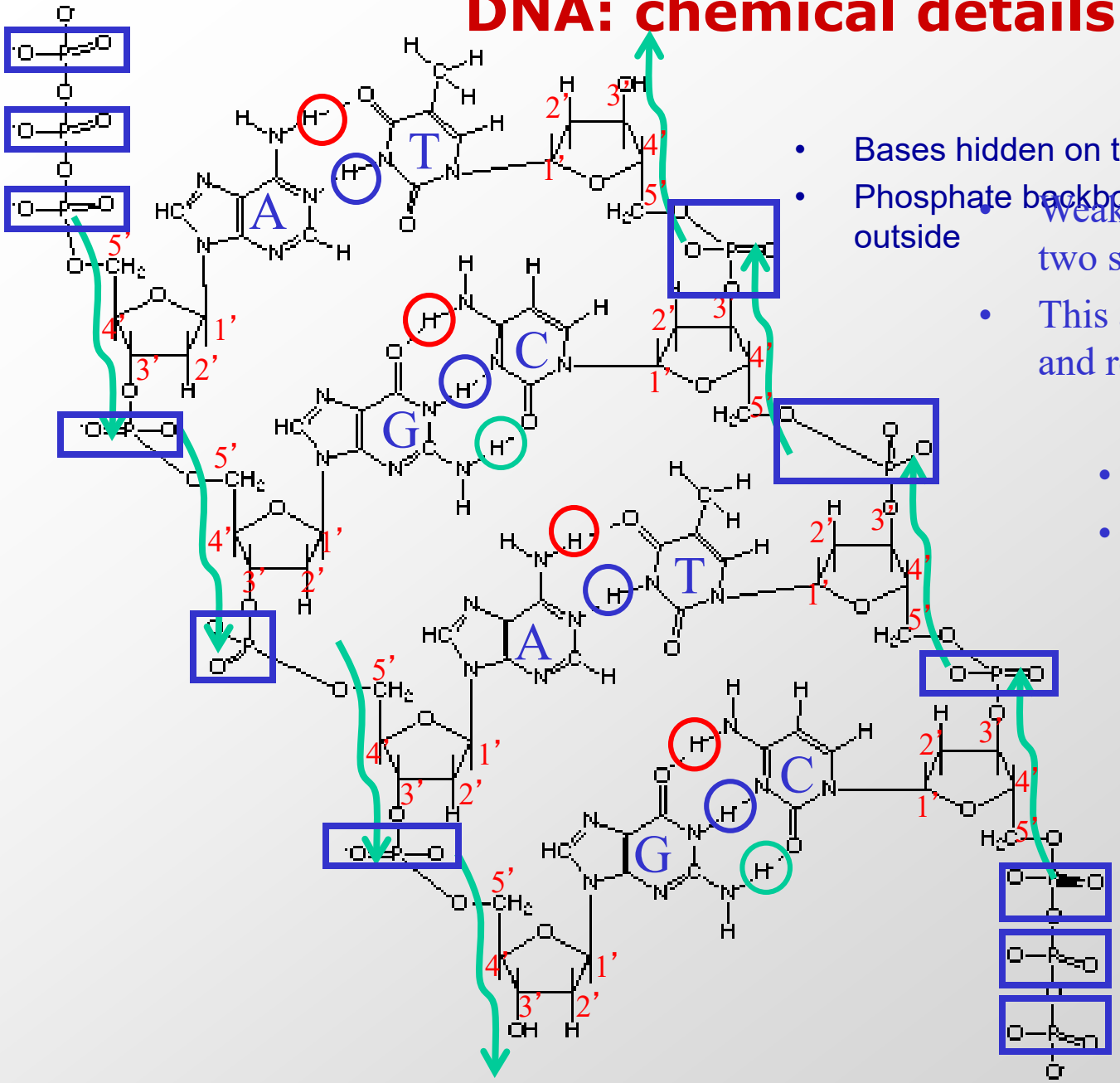
- Self-complementarity sets molecular basis of heredity
  - Knowing one strand, creates a template for the other
  - “It has not escaped our notice that the specific pairing we have postulated immediately suggests a possible copying mechanism for the genetic material.” Watson & Crick, 1953







DNA: chemical details



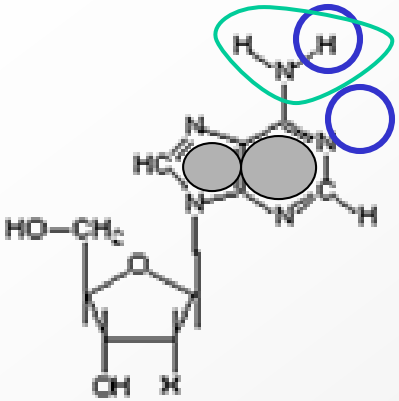
- Bases hidden on the inside
- Phosphate backbone outside
- Weak hydrogen bonds hold the two strands together
- This allows low-energy opening and re-closing of two strands
- Anti-parallel strands
- Extension 5’ →3’ tri-phosphate coming from newly added nucleotide

The only parings are:

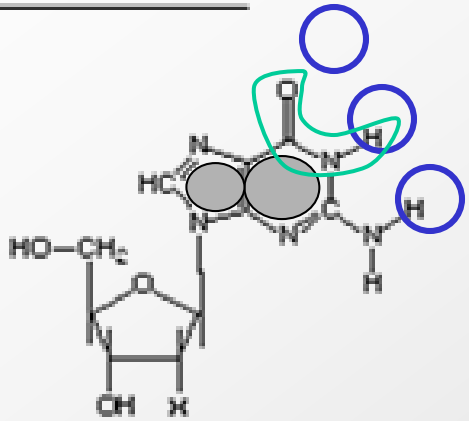
- A with T
- C with G

DNA: the four bases

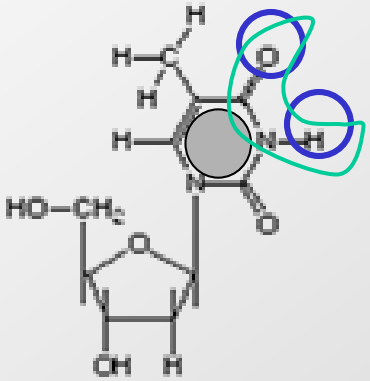
The Nucleotides of DNA



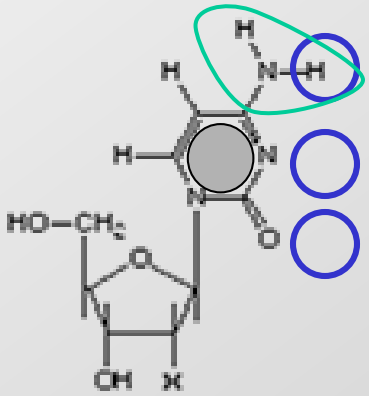
Adenine



Guanosine



Thymine



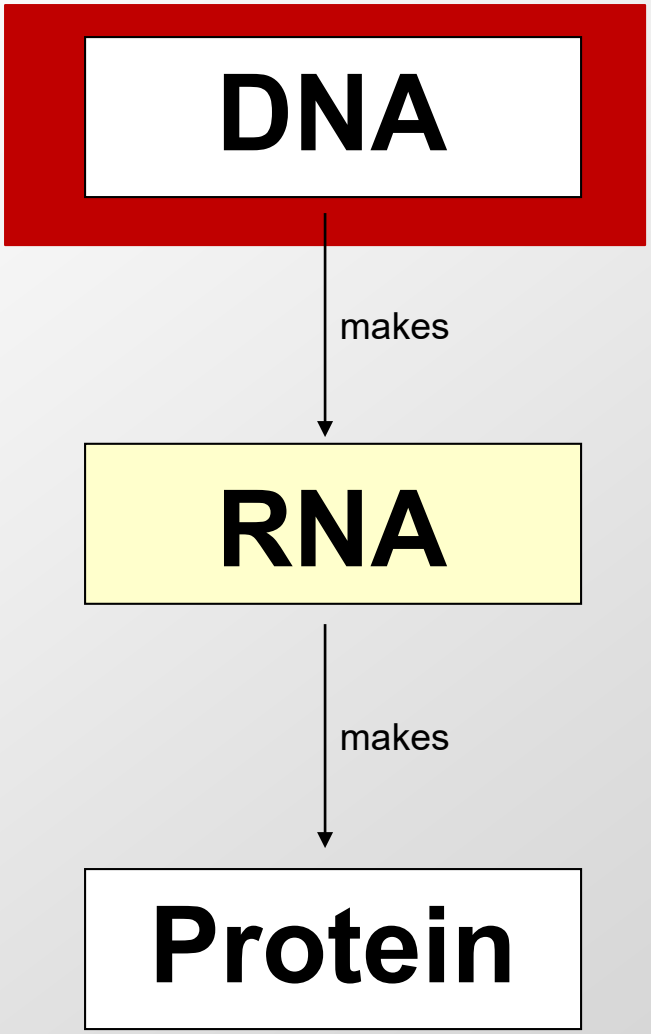
Cytosine

Purine	Purine		
		Pyrimidine	Pyrimidine
Weak		Weak	
	Strong		Strong
Amino			Amino
	Keto	Keto	

Goals for today: Course Introduction

1. **Course overview:**
   - Staff, students, responses to student survey
   - Foundations, frontiers, textbook, homework, quiz
   - Final project: teams, mentorship, challenge, relevance, originality, achievement, presentation
2. **Why Computational Biology;**
   - What makes our field unique
3. **Overview of the main modules**
   - Genomes, Expression, Epigenomics, Networks, Genetics, Evolution, Frontiers
4. **Biology primer (in the context of this course)**
   - Central Dogma of Molecular Biology
   - DNA, Epigenomics, RNA, Protein, Networks
   - Human genetics, evolution

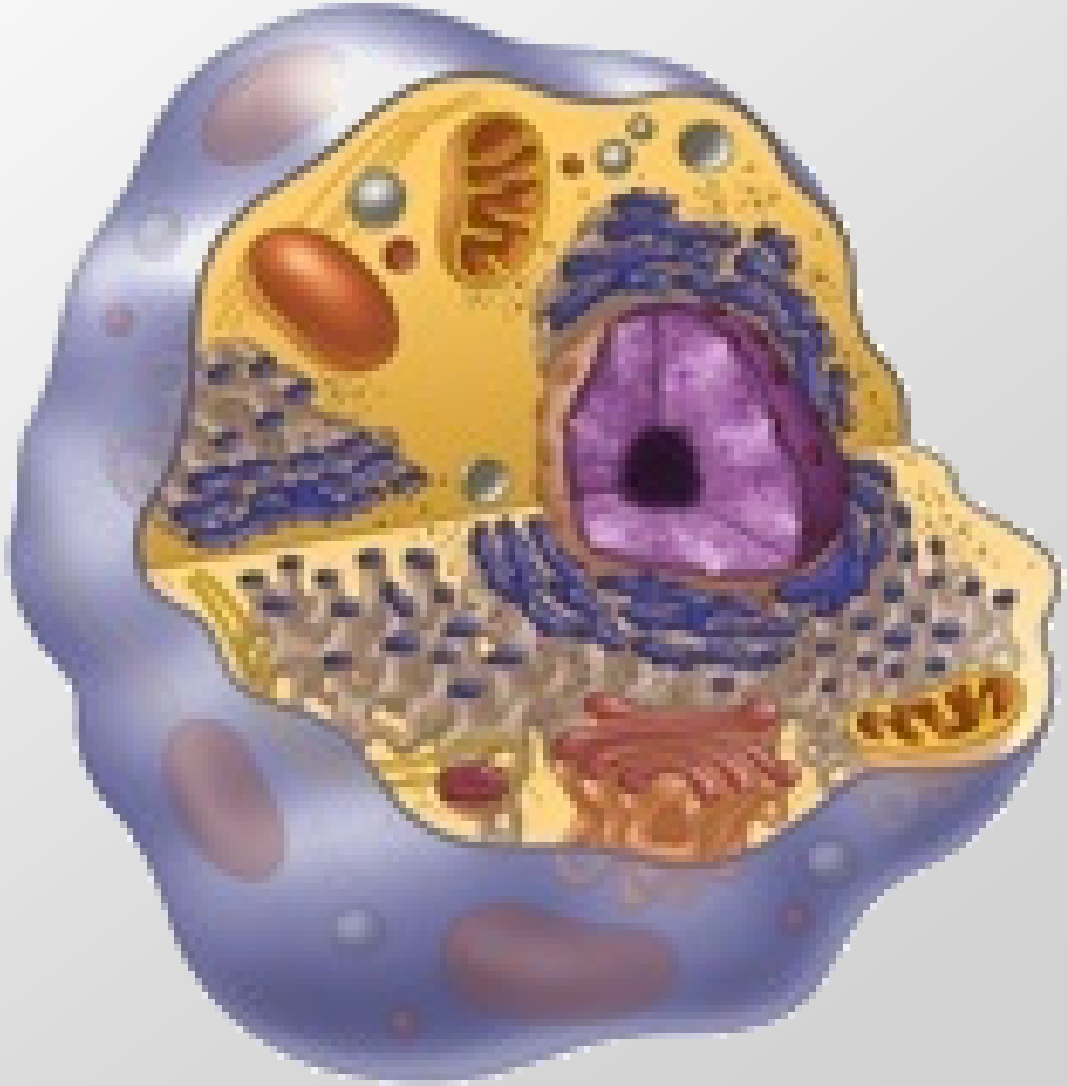
“Central dogma” of Molecular Biology



Epigenomics

Chromosomes inside the cell

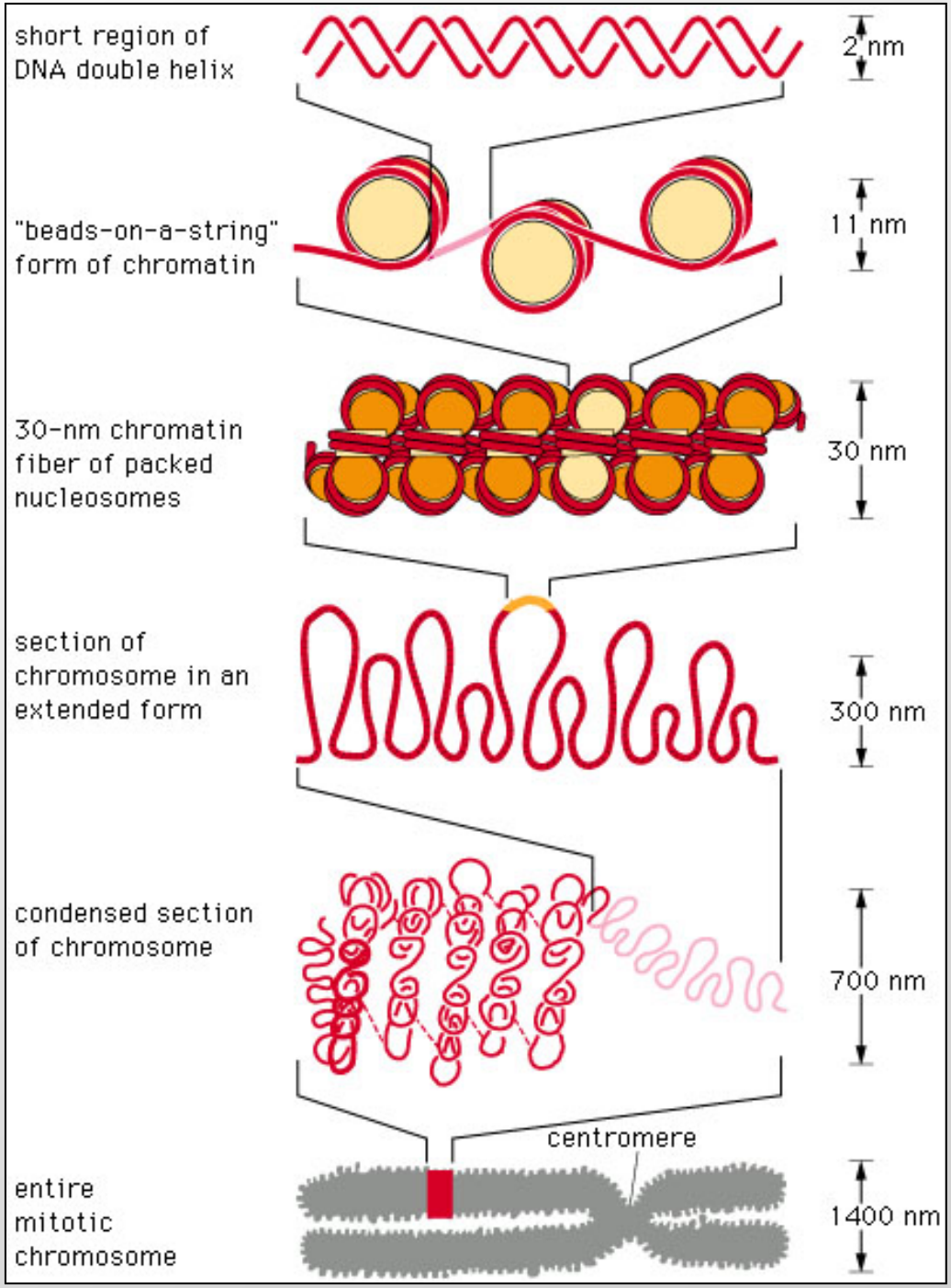
- Eukaryote cell



- Prokaryote cell

DNA packaging

- Why packaging
  - DNA is very long
  - Cell is very small
- Compression
  - Chromosome is 50,000 times shorter than extended DNA
- Using the DNA
  - Before a piece of DNA is used for anything, this compact structure must open locally
- Now emerging:
  - Role of accessibility
  - State in chromatin itself
  - Role of 3D interactions



Diverse epigenetic modifications

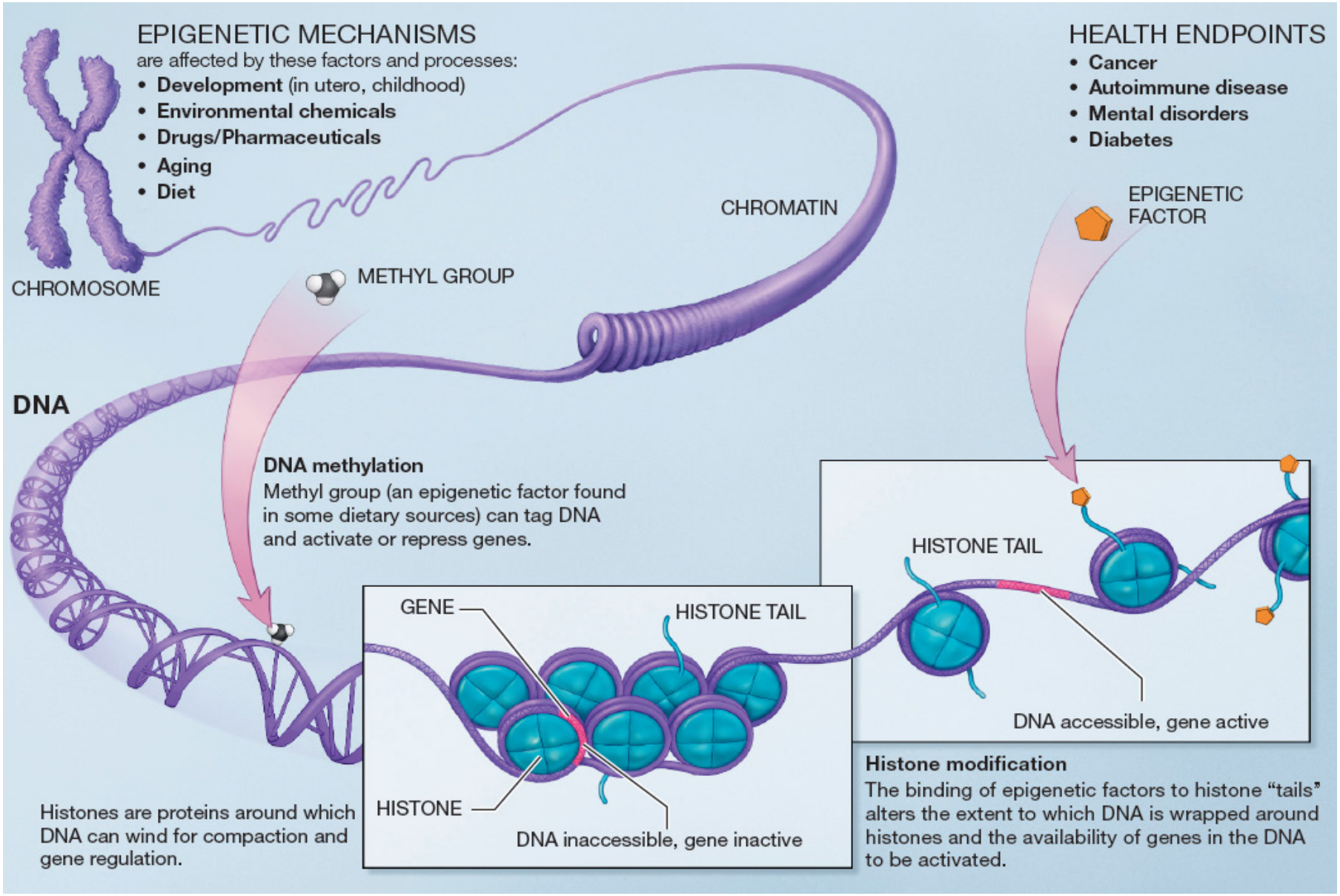
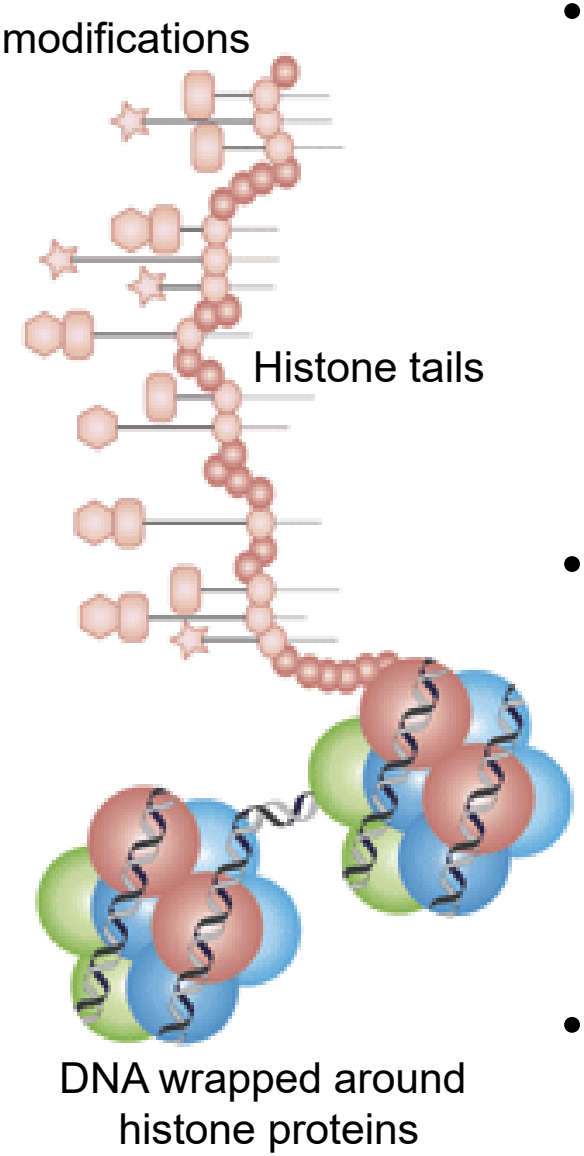


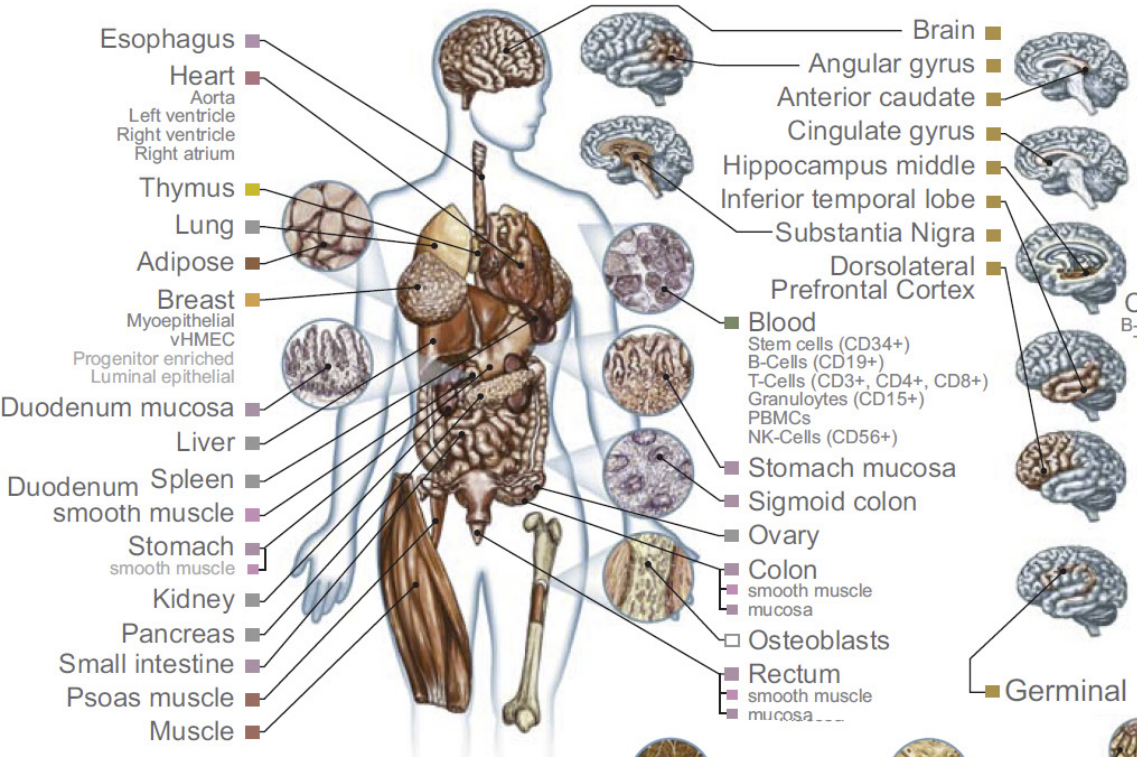
Image source: http://nihroadmap.nih.gov/epigenomics/

Diversity of epigenetic modifications

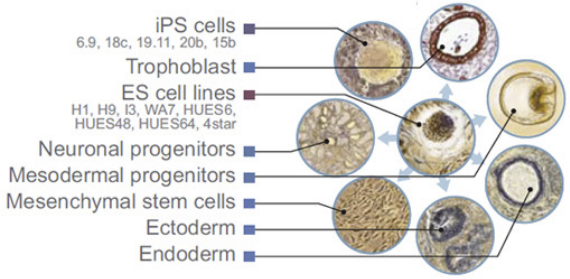


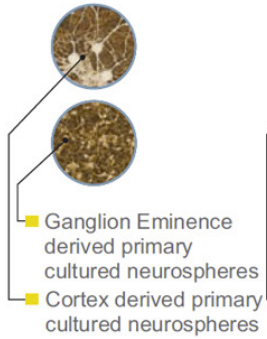
- 100+ different histone modifications
  - Histone protein → H3/H4/H2A/H2B
  - AA residue → Lysine4(K4)/K36…
  - Chemical modification → Met/Pho/Ubi
  - Number → Me-Me-Me(me3)
  - Shorthand: H3K4me3, H2BK5ac
- In addition:
  - DNA modifications
  - Methyl-C in CpG / Methyl-Adenosine
  - Nucleosome positioning
  - DNA accessibility
- The constant struggle of gene regulation
  - TF/histone/nucleo/GFs/Chrom compete

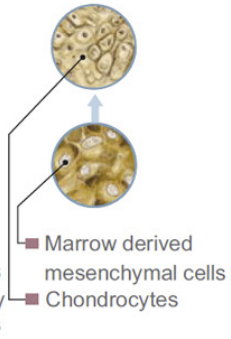
Epigenomics Roadmap across 100+ tissues/cell types

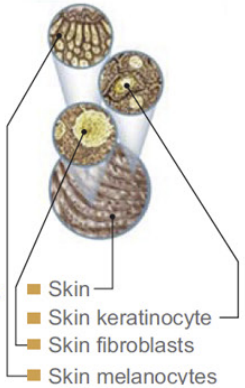


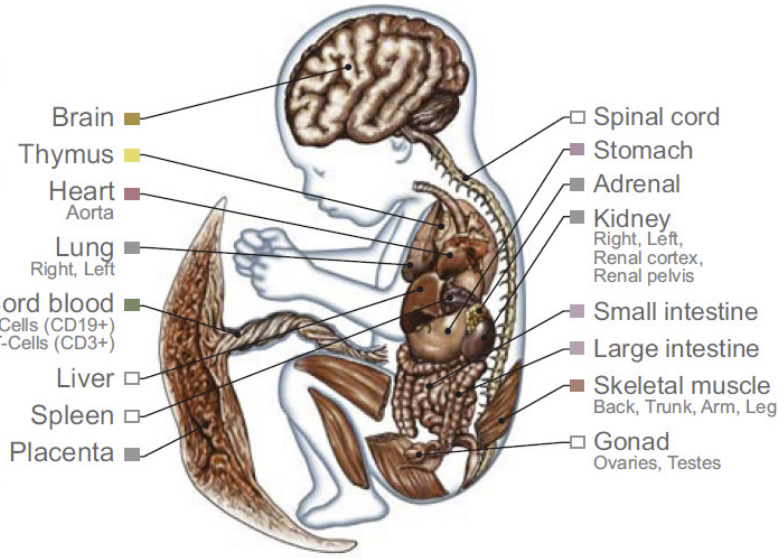
Art: Rae Senarighi, Richard Sandstrom











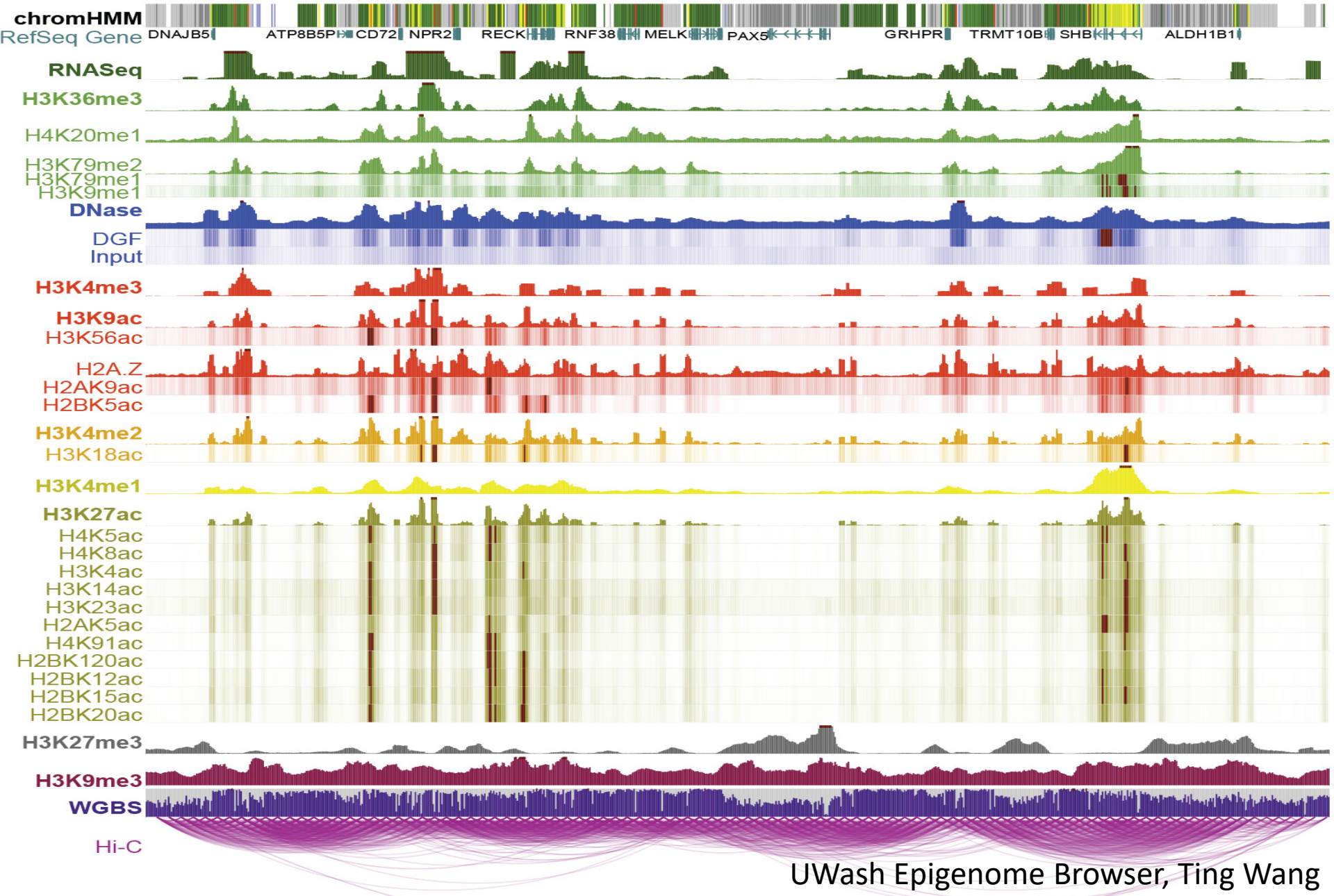
Diverse epigenomic assays:

1. **Histone modifications**
   - **H3K4me3, H3K4me1**
   - **H3K36me3**
   - **H3K27me3, H3K9me3**
   - **H3K27/9ac, +20 more**
2. **Open chromatin**:
   - **DNase**
3. **DNA methylation**:
   - **WGBS, RRBS, MRE/MeDIP**
4. **Gene expression**
   - **RNA-seq, Exon Arrays**

Diverse tissues and cells:

1. **Adult tissues and cells** (brain, muscle, heart, digestive, skin, adipose, lung, blood…)
2. **Fetal tissues** (brain, skeletal muscle, heart, digestive, lung, cord blood…)
3. **ES cells, iPS, differentiated cells** (meso/endo/ectoderm, neural, mesench, trophobl)

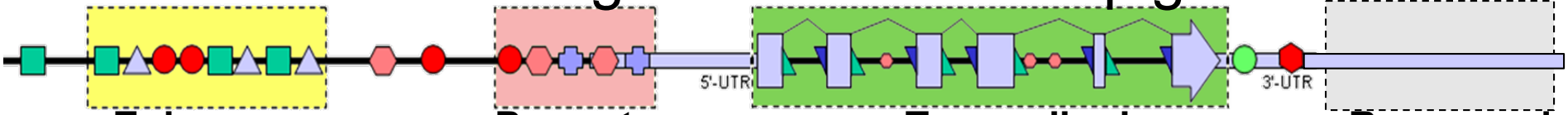
Deep sampling of 9 reference epigenomes (e.g. IMR90)



UWash Epigenome Browser, Ting Wang

Chromatin state+RNA+DNAse+28 histone marks+WGBS+Hi-C

Diverse chromatin signatures encode epigenomic state

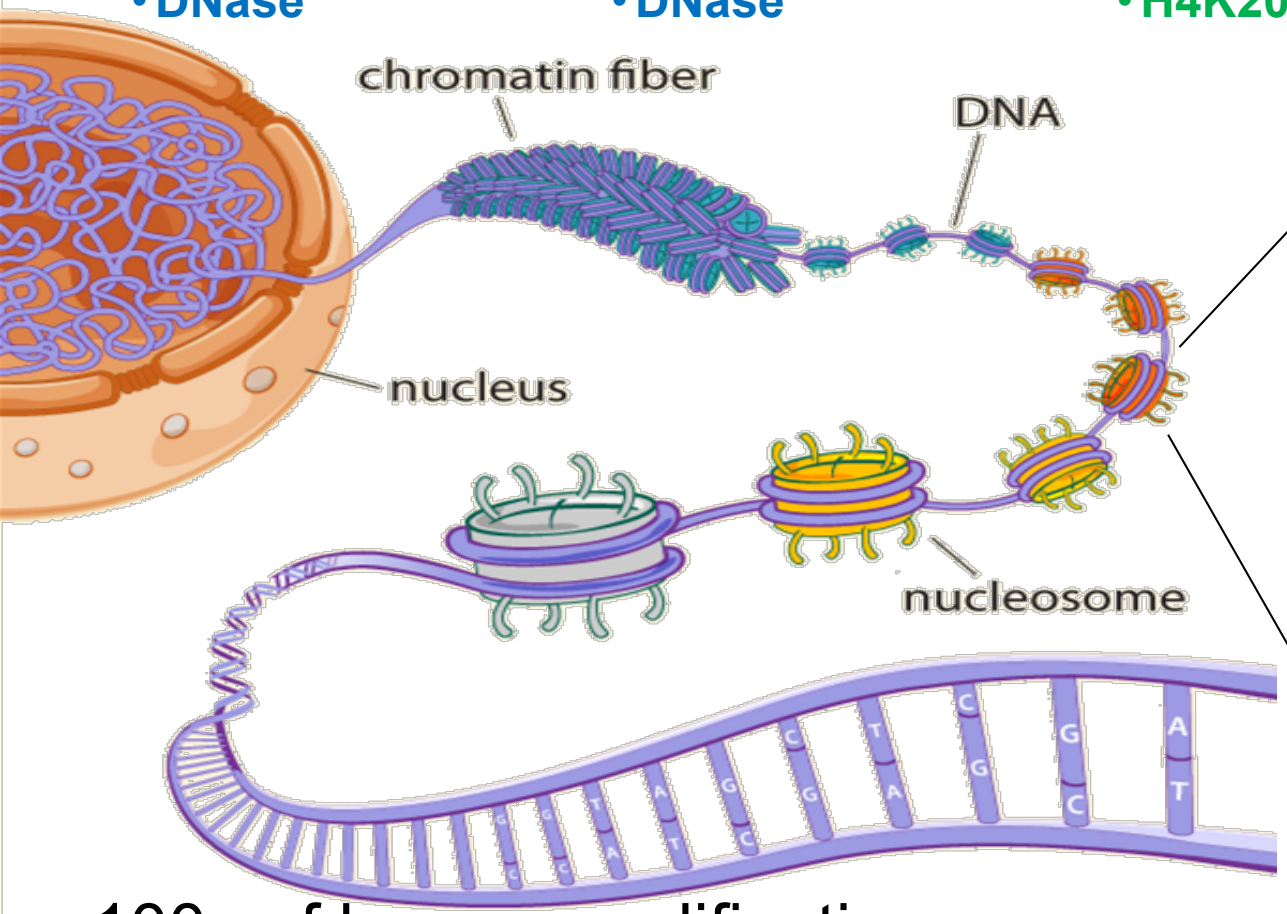


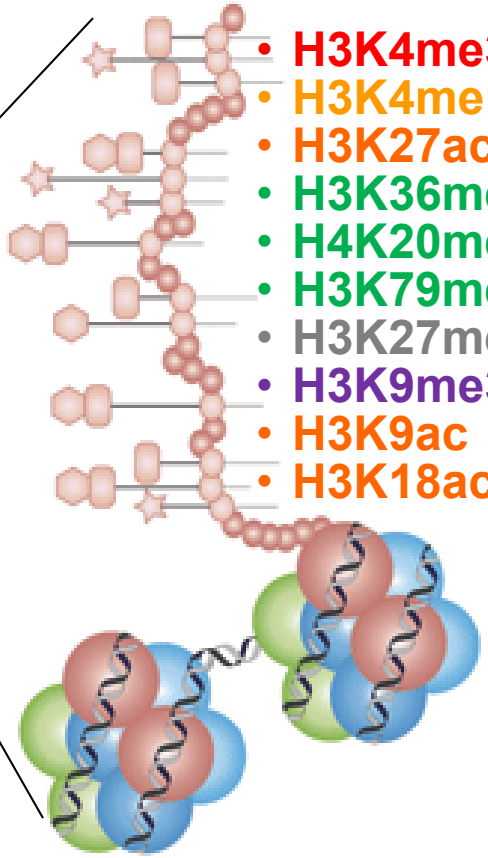
Enhancers
- H3K4me1
- H3K27ac
- DNase

Promoters
- H3K4me3
- H3K9ac
- DNase

Transcribed
- H3K36me3
- H3K79me2
- H4K20me1

Repressed
- H3K9me3
- H3K27me3
- DNAmethyl





- 100s of known modifications, many new still emerging
- Systematic mapping using ChIP-, Bisulfite-, DNase-Seq

Chromatin state annotations across 127 epigenomes

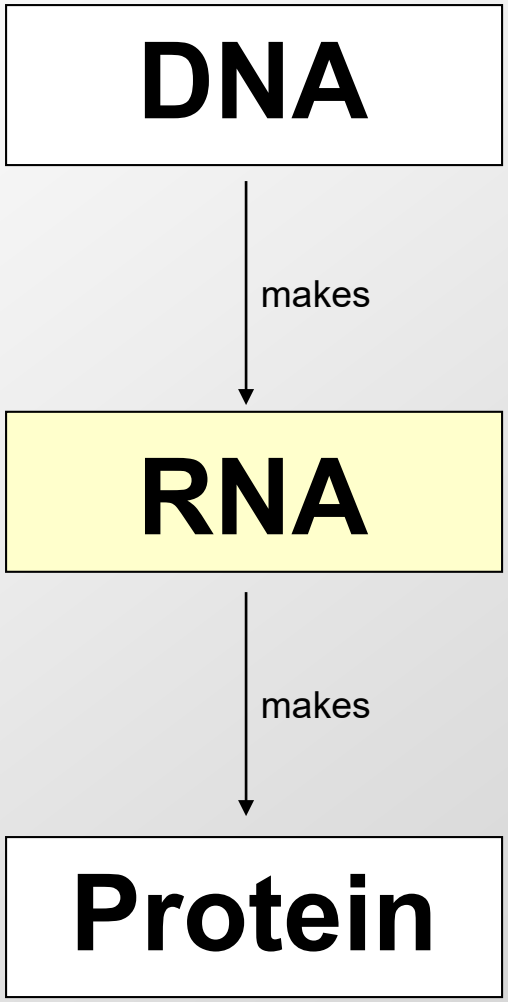
Reveal epigenomic variability: enh/prom/tx/repr/het

Anshul Kundaje

Goals for today: Course Introduction

1. **Course overview:**
   - Staff, students, responses to student survey
   - Foundations, frontiers, textbook, homework, quiz
   - Final project: teams, mentorship, challenge, relevance, originality, achievement, presentation
2. **Why Computational Biology;**
   - What makes our field unique
3. **Overview of the main modules**
   - Genomes, Expression, Epigenomics, Networks, Genetics, Evolution, Frontiers
4. **Biology primer (in the context of this course)**
   - Central Dogma of Molecular Biology
   - DNA, Epigenomics, RNA, Protein, Networks
   - Human genetics, evolution

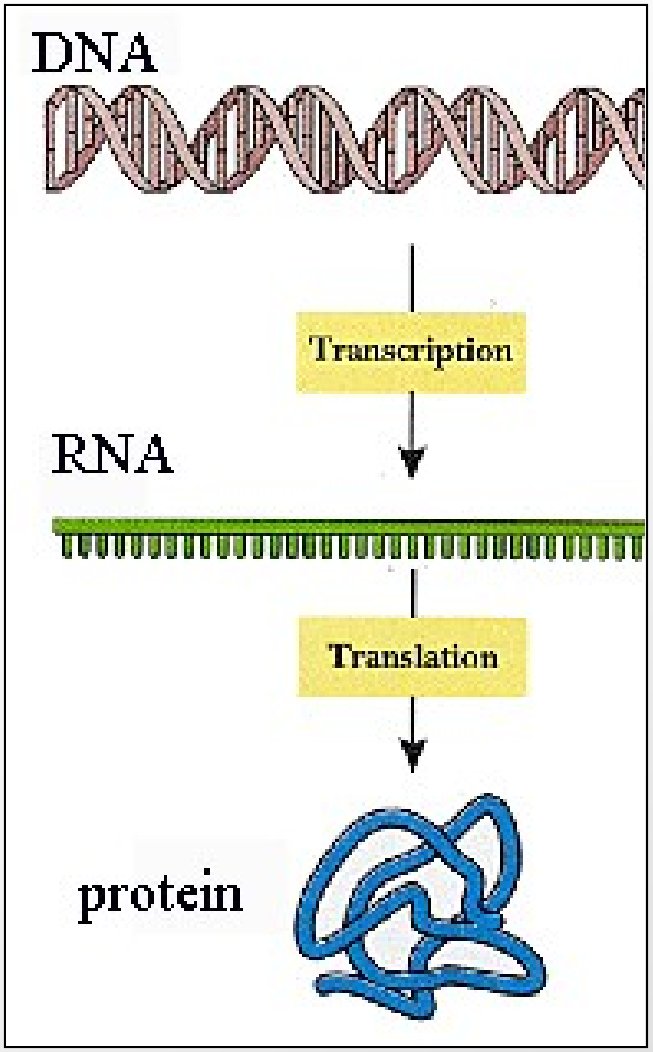
“Central dogma” of Molecular Biology



Genes control the making of cell parts

- The gene is a fundamental unit of inheritance
  - Each DNA molecule ⇔ 10,000+ genes
  - 1 gene ⇔ 1 functional element (one "part" of cell machinery)
  - Every time a "part" is made, the corresponding gene is:
    - Copied into mRNA, transported, used as blueprint to make protein
- RNA is a temporary copy
  - The medium for transporting genetic information from the DNA information repository to the protein-making machinery is an RNA molecule
  - The more parts are needed, the more copies are made
  - Each mRNA only lasts a limited time before degradation

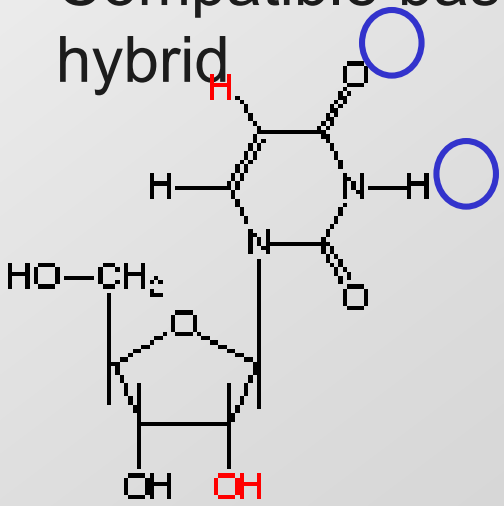
mRNA: The messenger

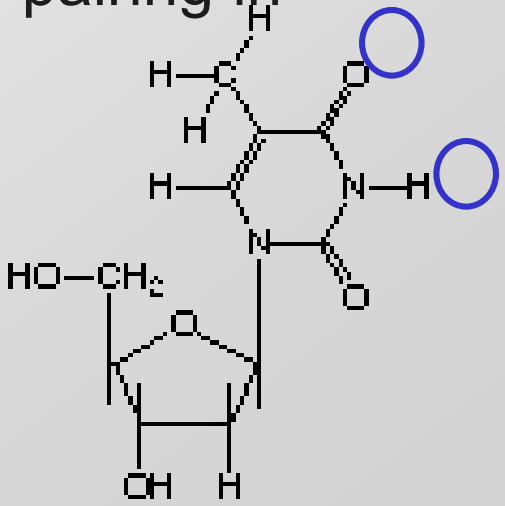


- Information changes medium
  - single strand vs. double strand
  - ribose vs. deoxyribose sugar

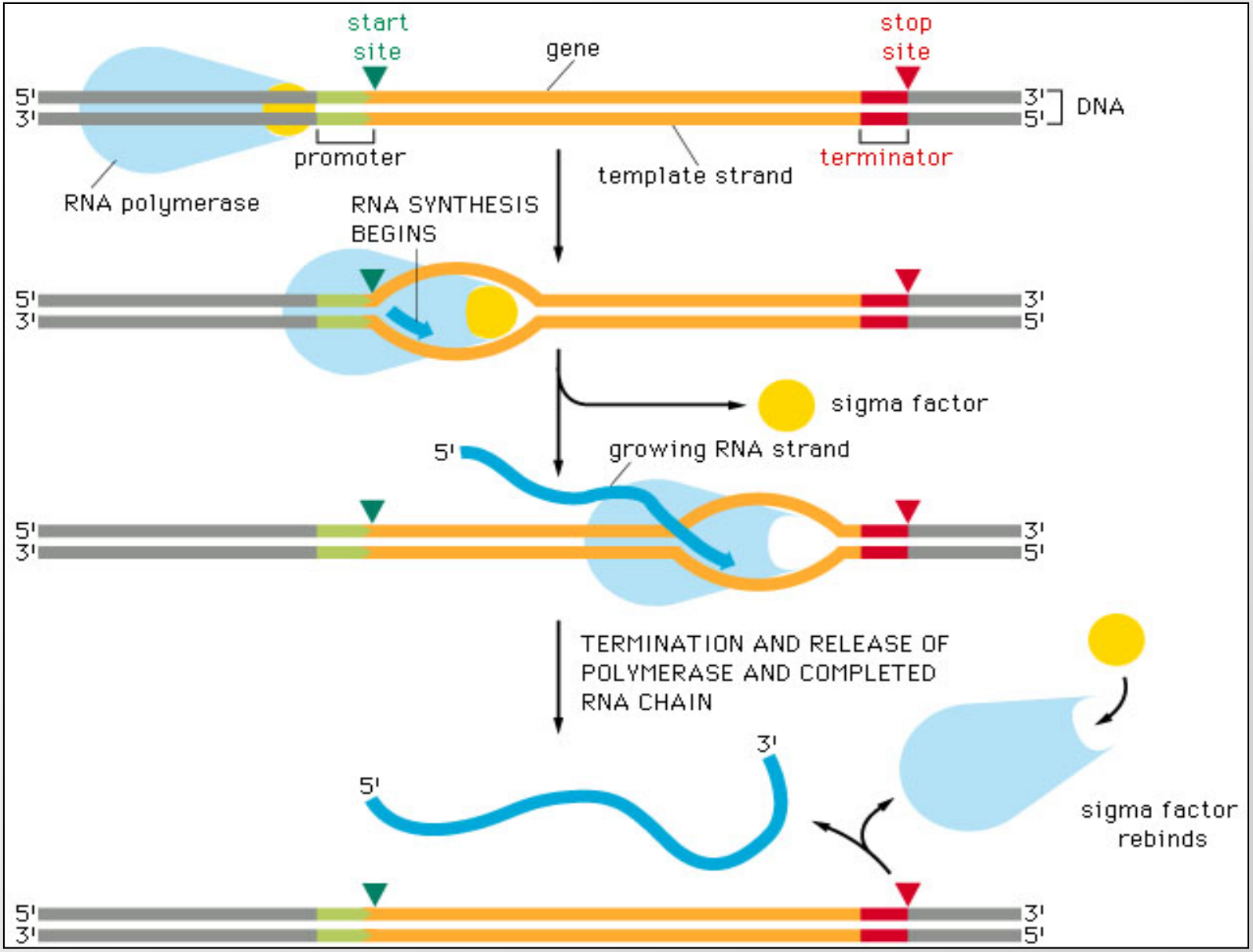
A T T A C G G T A C C G T
U A A U G C C A U G G C A

  - Compatible base-pairing in hybrid



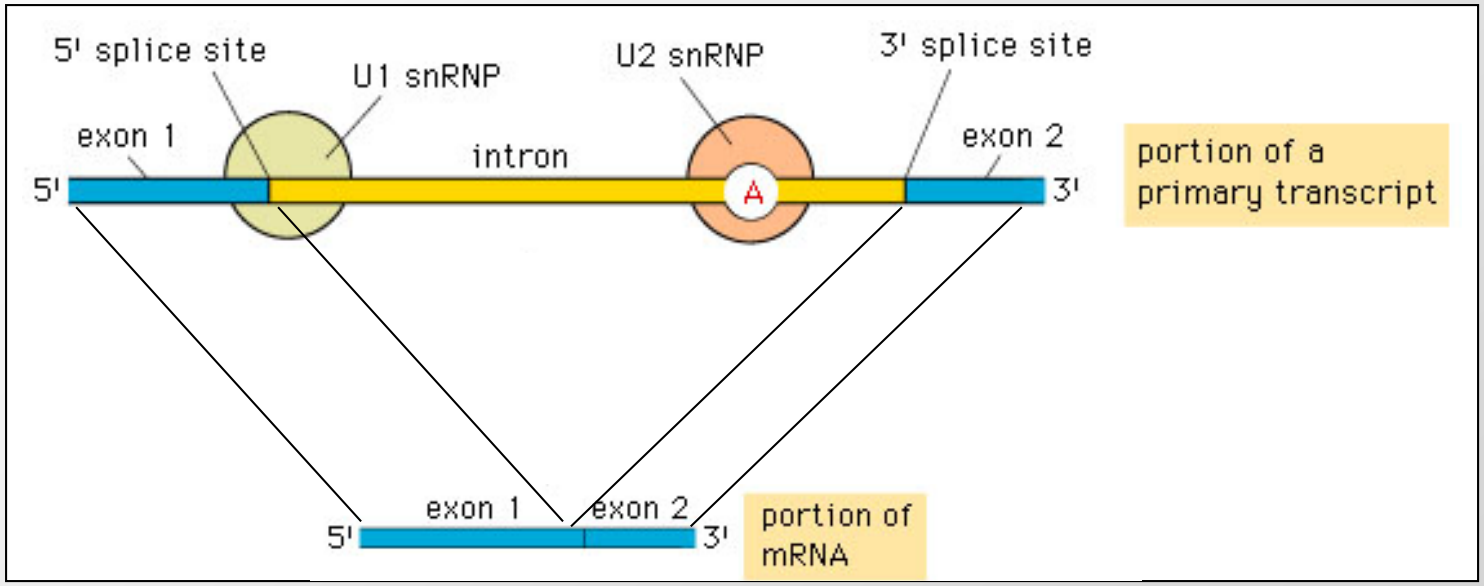


From DNA to RNA: Transcription



From pre-mRNA to mRNA: Splicing

- In Eukaryotes, not every part of a gene is coding
  - Functional exons interrupted by non-translated introns
  - During pre-mRNA maturation, introns are spliced out
  - In humans, primary transcript can be 10^6 bp long

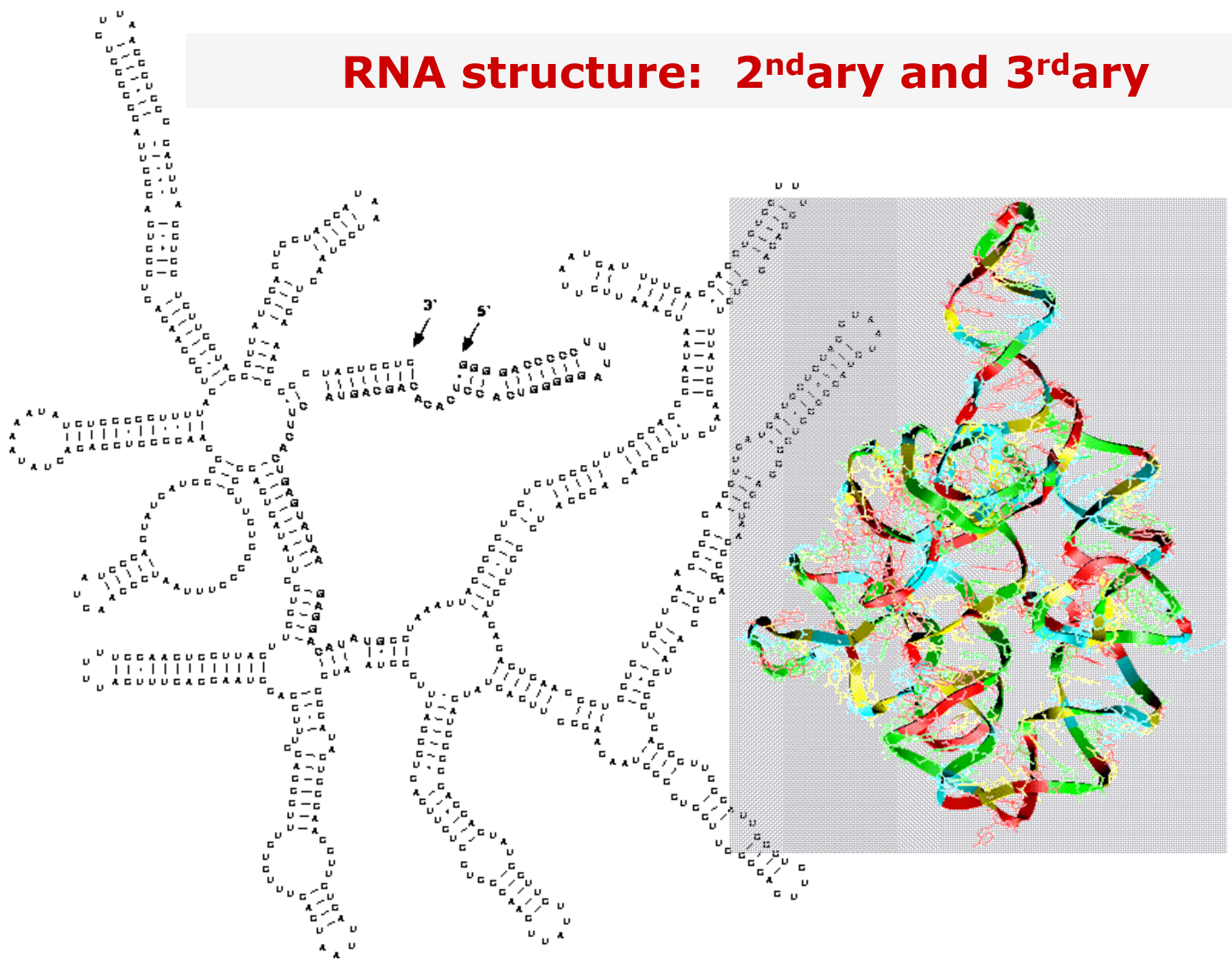


  - Alternative splicing can yield different exon subsets for the same gene, and hence different protein products

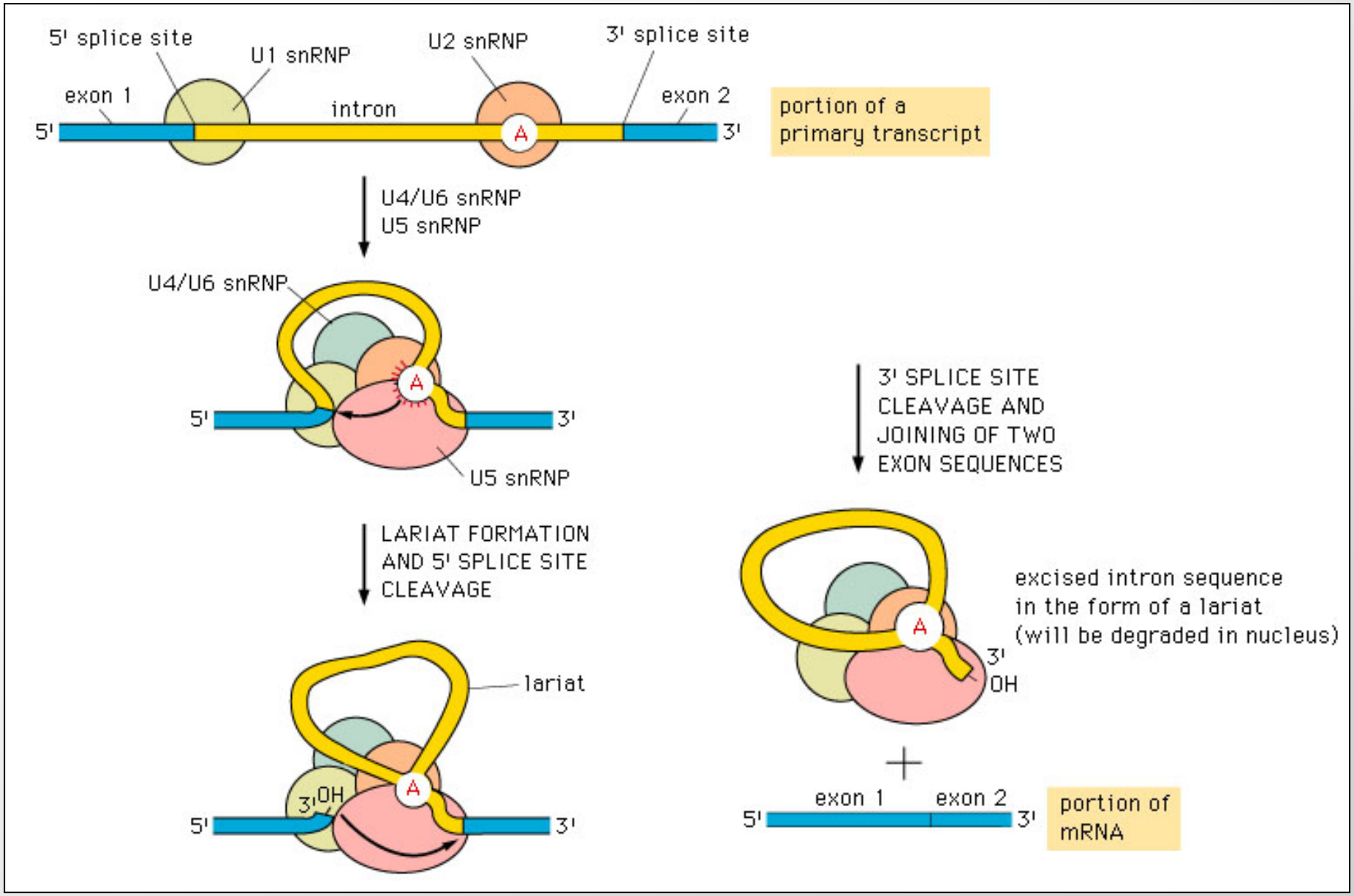
RNA can be functional

- Single Strand allows complex structure
  - Self-complementary regions form helical stems
  - Three-dimensional structure allows functionality of RNA
- Four types of RNA
  - mRNA: messenger of genetic information
  - tRNA: codon-to-amino acid specificity
  - rRNA: core of the ribosome
  - snRNA: splicing reactions
- To be continued…
  - We'll learn more in a dedicated lecture on RNA world
  - Once upon a time, before DNA and protein, RNA did all

RNA structure: 2ndary and 3rdary



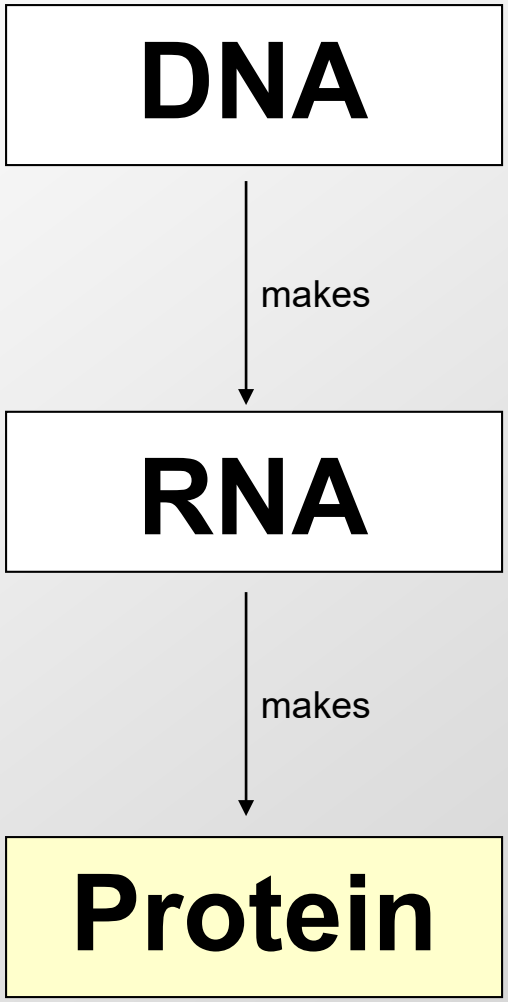
Splicing machinery made of RNA



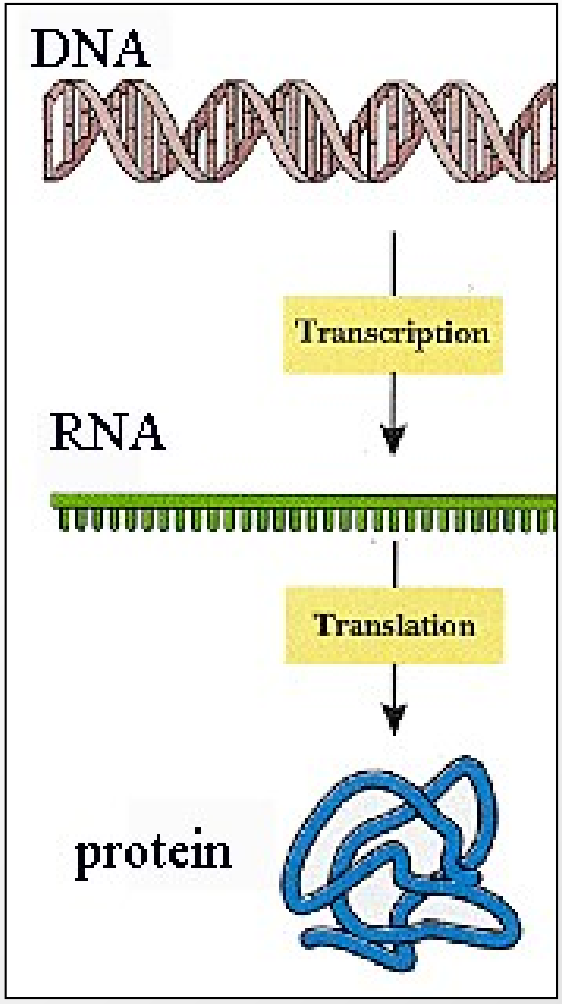
Goals for today: Course Introduction

1. **Course overview:**
   - Staff, students, responses to student survey
   - Foundations, frontiers, textbook, homework, quiz
   - Final project: teams, mentorship, challenge, relevance, originality, achievement, presentation
2. **Why Computational Biology;**
   - What makes our field unique
3. **Overview of the main modules**
   - Genomes, Expression, Epigenomics, Networks, Genetics, Evolution, Frontiers
4. **Biology primer (in the context of this course)**
   - Central Dogma of Molecular Biology
   - DNA, Epigenomics, RNA, Protein, Networks
   - Human genetics, evolution

“Central dogma” of Molecular Biology

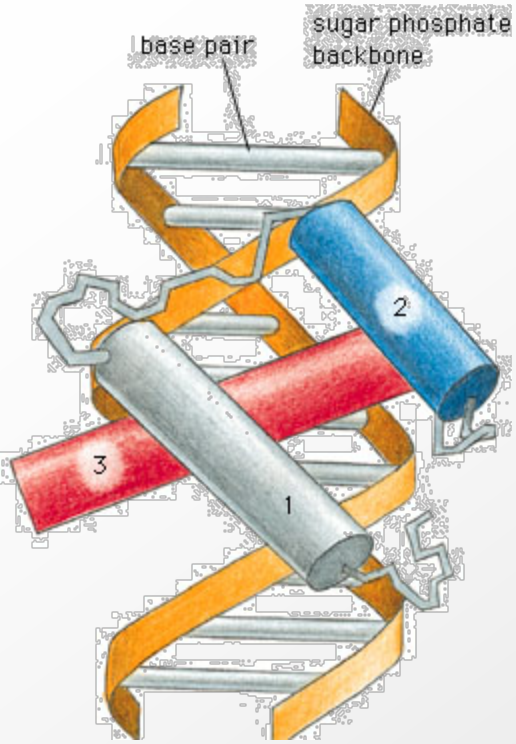


Proteins carry out the cell's chemistry



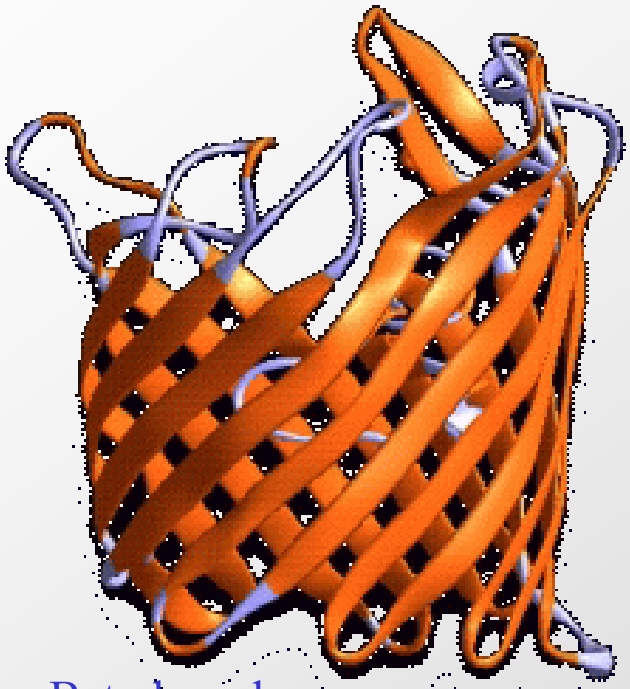
- More complex polymer
  - Nucleic Acids have 4 building blocks
  - Proteins have 20. Greater versatility
  - Each amino acid has specific properties
- Sequence → Structure → Function
  - The amino acid sequence determines the three-dimensional fold of protein
  - The protein's function largely depends on the features of the 3D structure
- Proteins play diverse roles
  - Catalysis, binding, cell structure, signaling, transport, metabolism

Protein structure



Helix-turn-helix

Common motif for DNA-binding proteins that often play a regulatory role as mRNA level transcription factors



Beta-barrel

Some antiparallel b-sheet domains are better described as b-barrels rather than b-sandwiches, for example streptavadin and porin. Note that some structures are

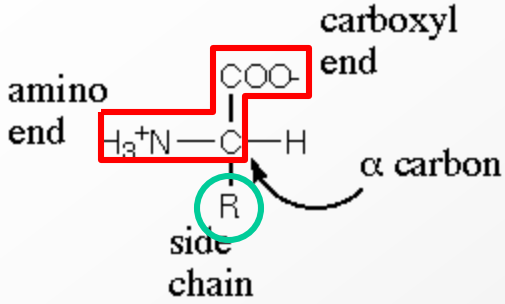
intermediate between the extreme barrel and sandwich arrangements.

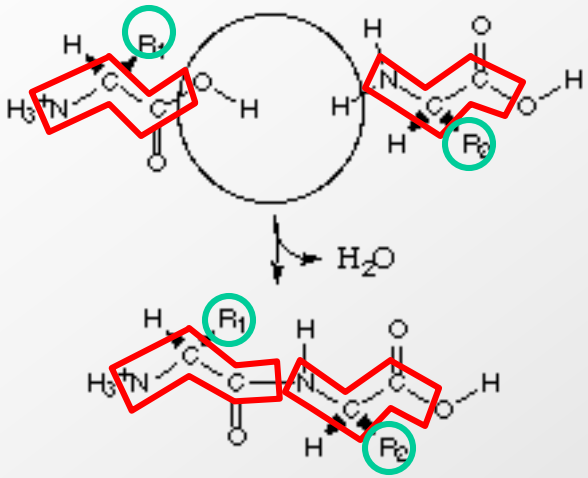
Alpha-beta horseshoe

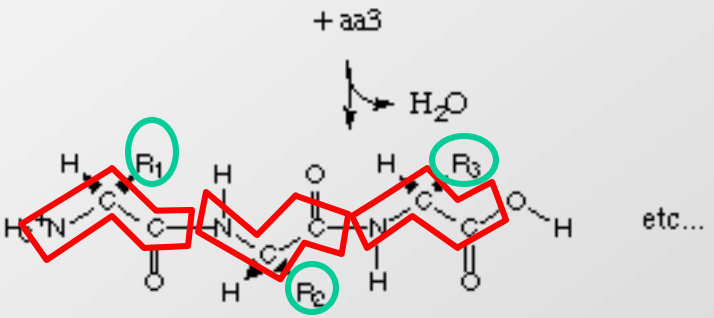
this placental ribonuclease inhibitor is a cytosolic protein that binds extremely strongly to any ribonuclease that may leak into the cytosol. 17-stranded parallel b sheet curved into an open horseshoe shape, with 16 a-helices packed against the outer surface. It doesn't form a barrel although it looks as though it should. The strands are only very slightly slanted, being nearly parallel to the central `axis'.

Protein building blocks

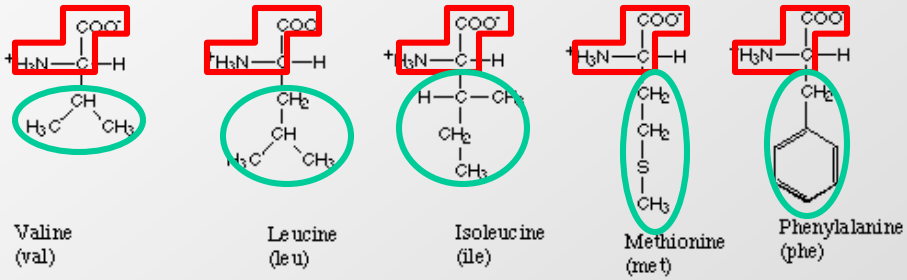
- Amino Acids



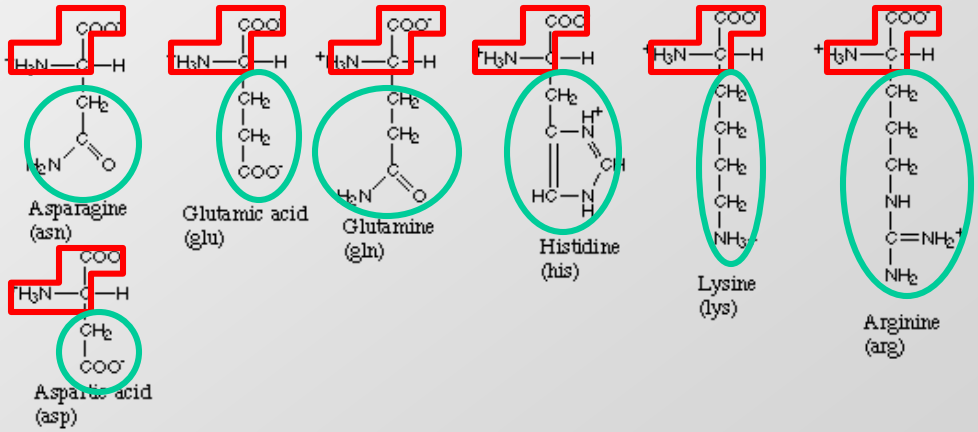




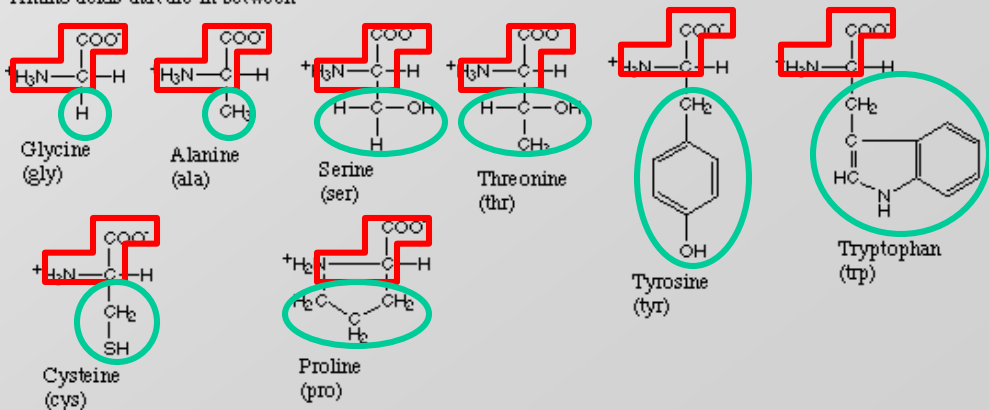
Amino acids with hydrophobic side groups



Amino acids with hydrophilic side groups

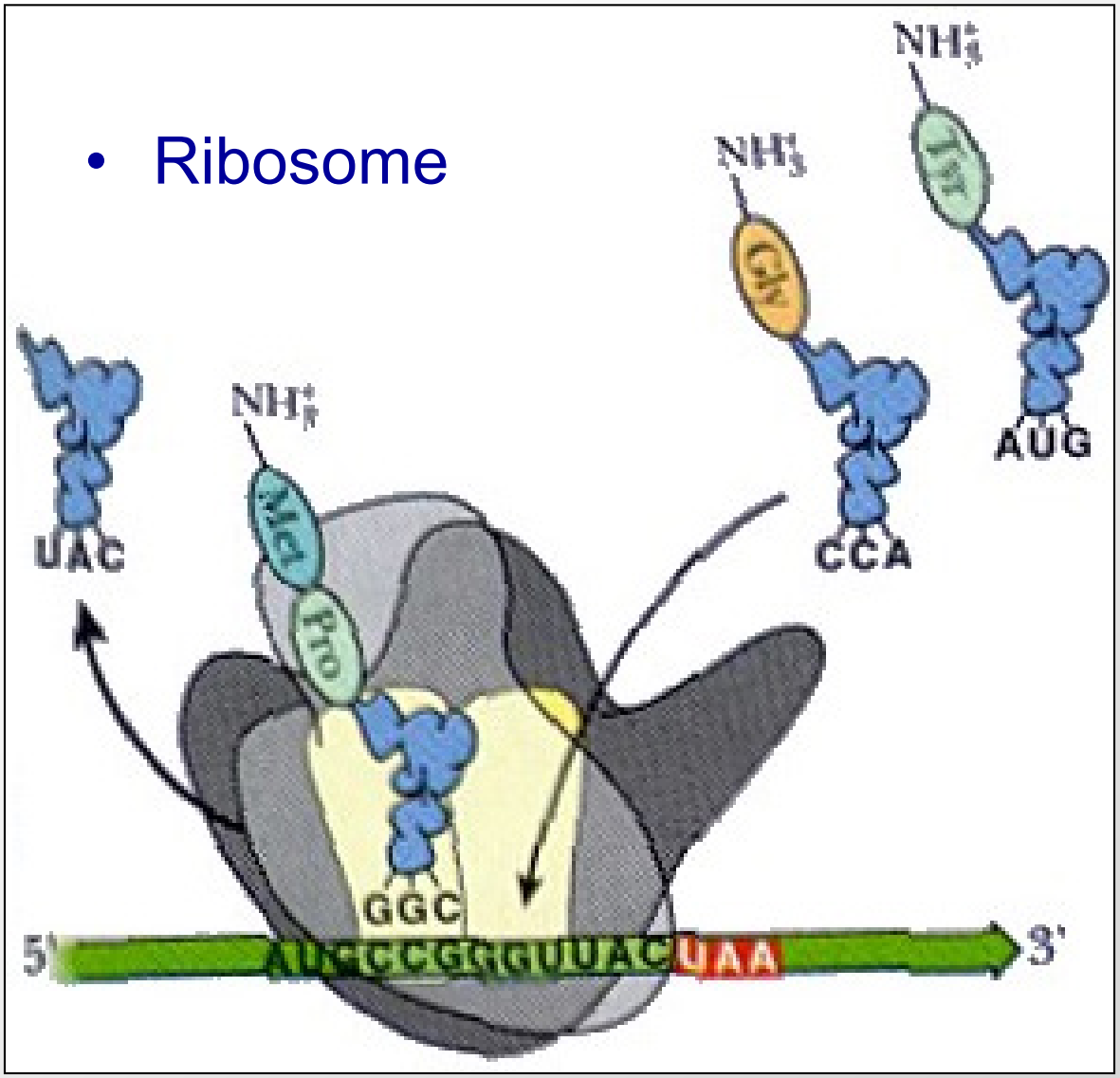


Amino acids that are in between

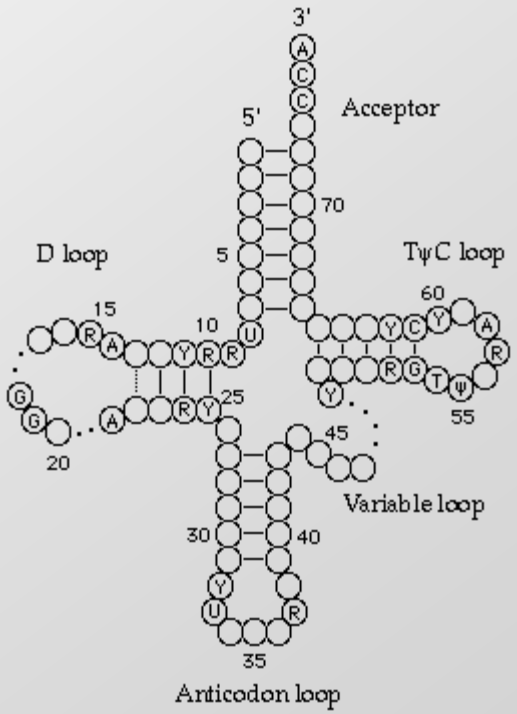


From RNA to protein: Translation

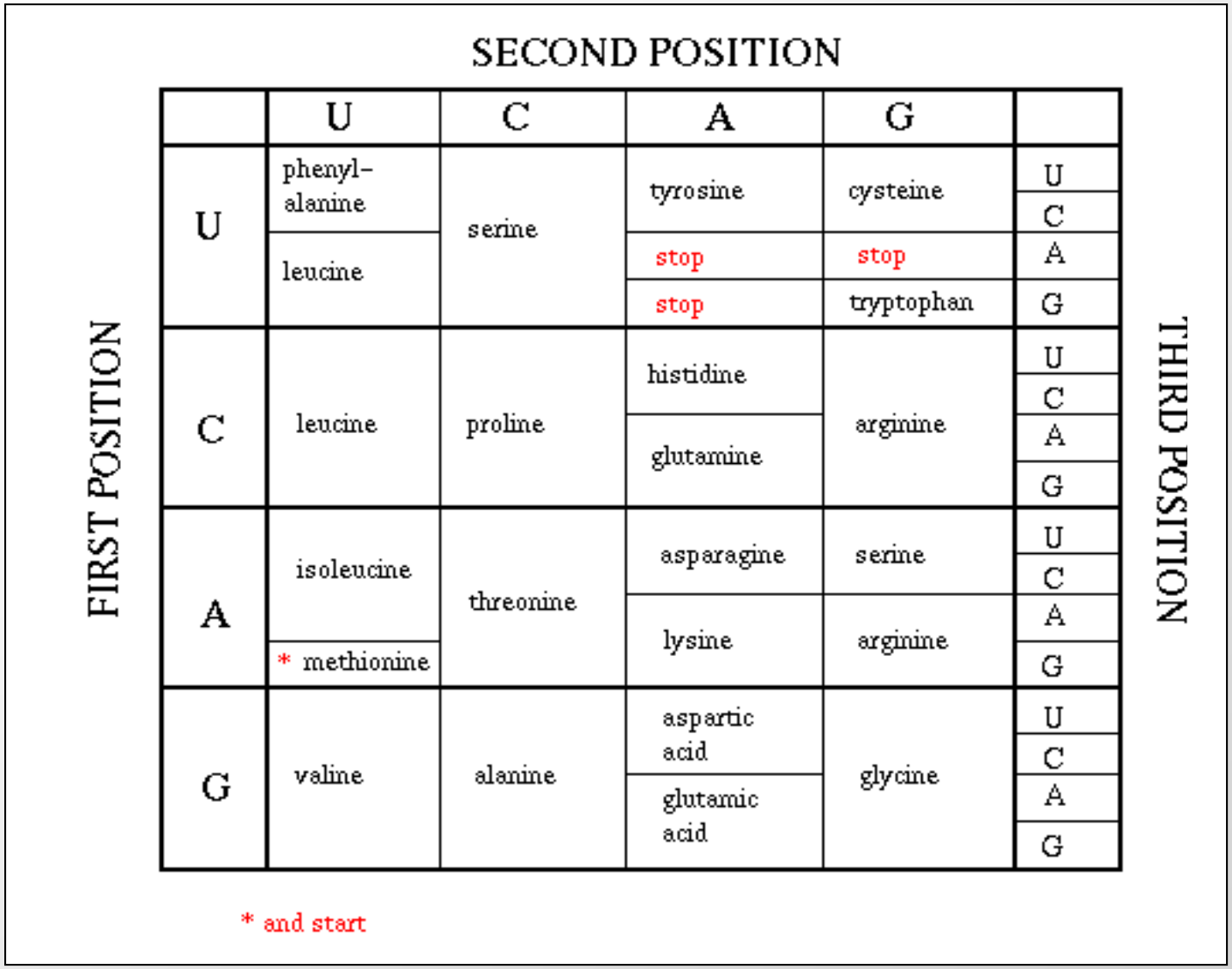
- Ribosome



- tRNA



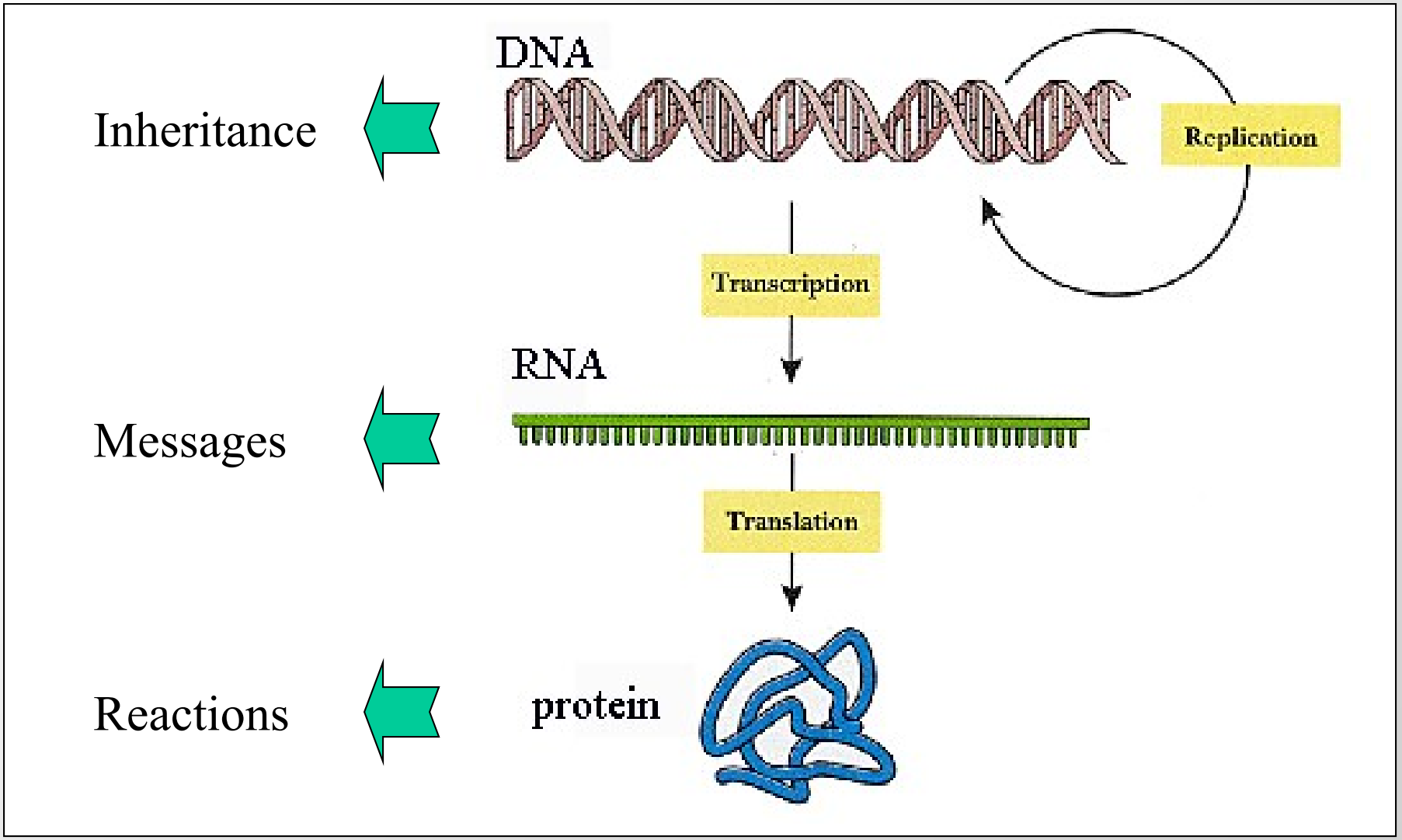
The Genetic Code



➔ Use evolutionary and compositional properties to computationally discover protein-coding genes

Summary: The Central Dogma

DNA makes RNA makes Protein

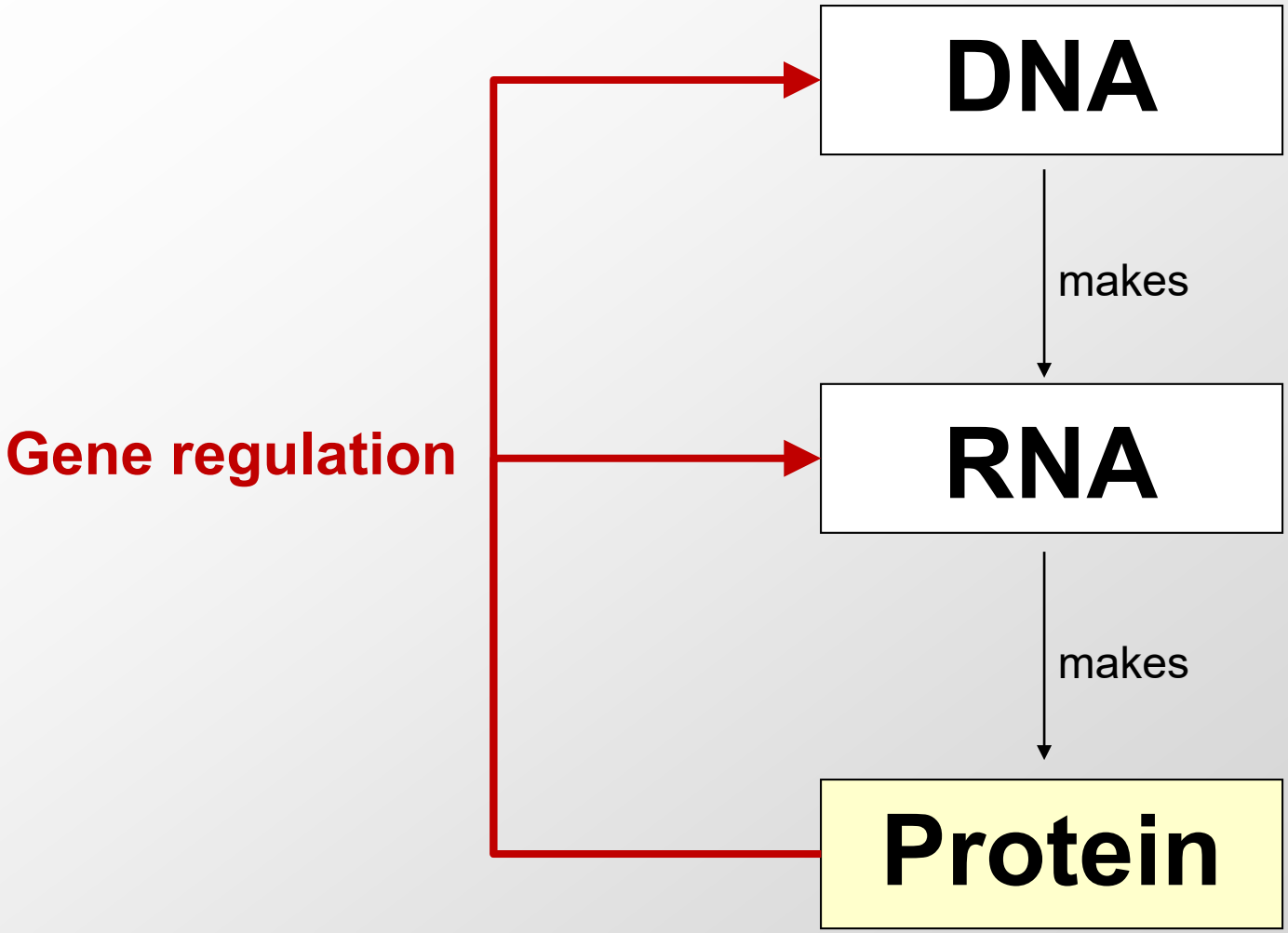


Goals for today: Course Introduction

1. **Course overview:**
   - Staff, students, responses to student survey
   - Foundations, frontiers, textbook, homework, quiz
   - Final project: teams, mentorship, challenge, relevance, originality, achievement, presentation
2. **Why Computational Biology;**
   - What makes our field unique
3. **Overview of the main modules**
   - Genomes, Expression, Epigenomics, Networks, Genetics, Evolution, Frontiers
4. **Biology primer (in the context of this course)**
   - Central Dogma of Molecular Biology
   - DNA, Epigenomics, RNA, Protein, Networks
   - Human genetics, evolution

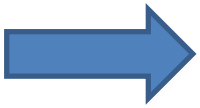
Cellular dynamics and regulation

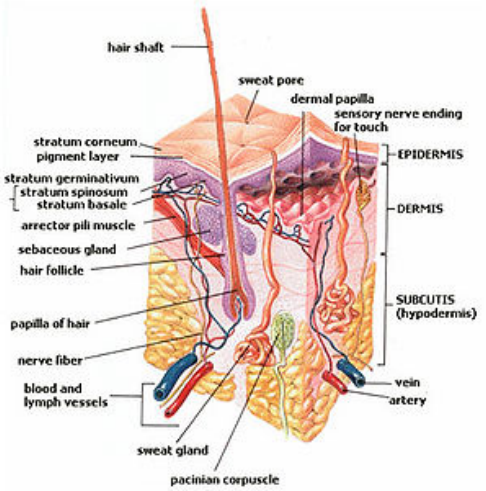
How cells move through this Central Dogma

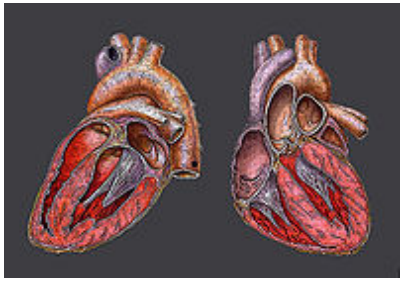


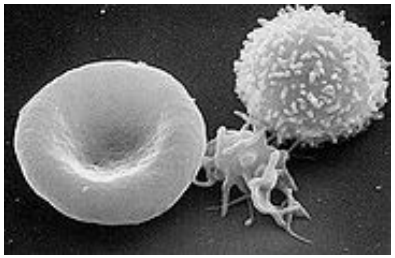
Animal/Human gene regulation: One genome ⇔ Many cell types

ACCAGTTACGACGGTCA
GGGTACTGATACCCCAA
ACCGTTGACCGCATTTA
CAGACGGGGTTTGGGTT
TTGCCCCACACAGGTAC
GTTAGCTACTGGTTTAG
CAATTTACCGTTACAAC
GTTTACAGGGTTACGGT
TGGGATTTGAAAAAAAG
TTTGAGTTGGTTTTTTC
ACGGTAGAACGTACCGT
TACCAGTA









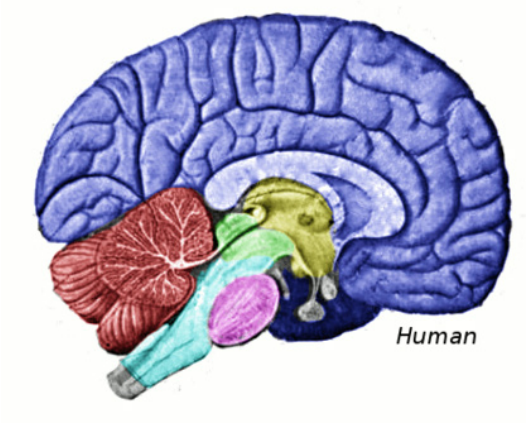
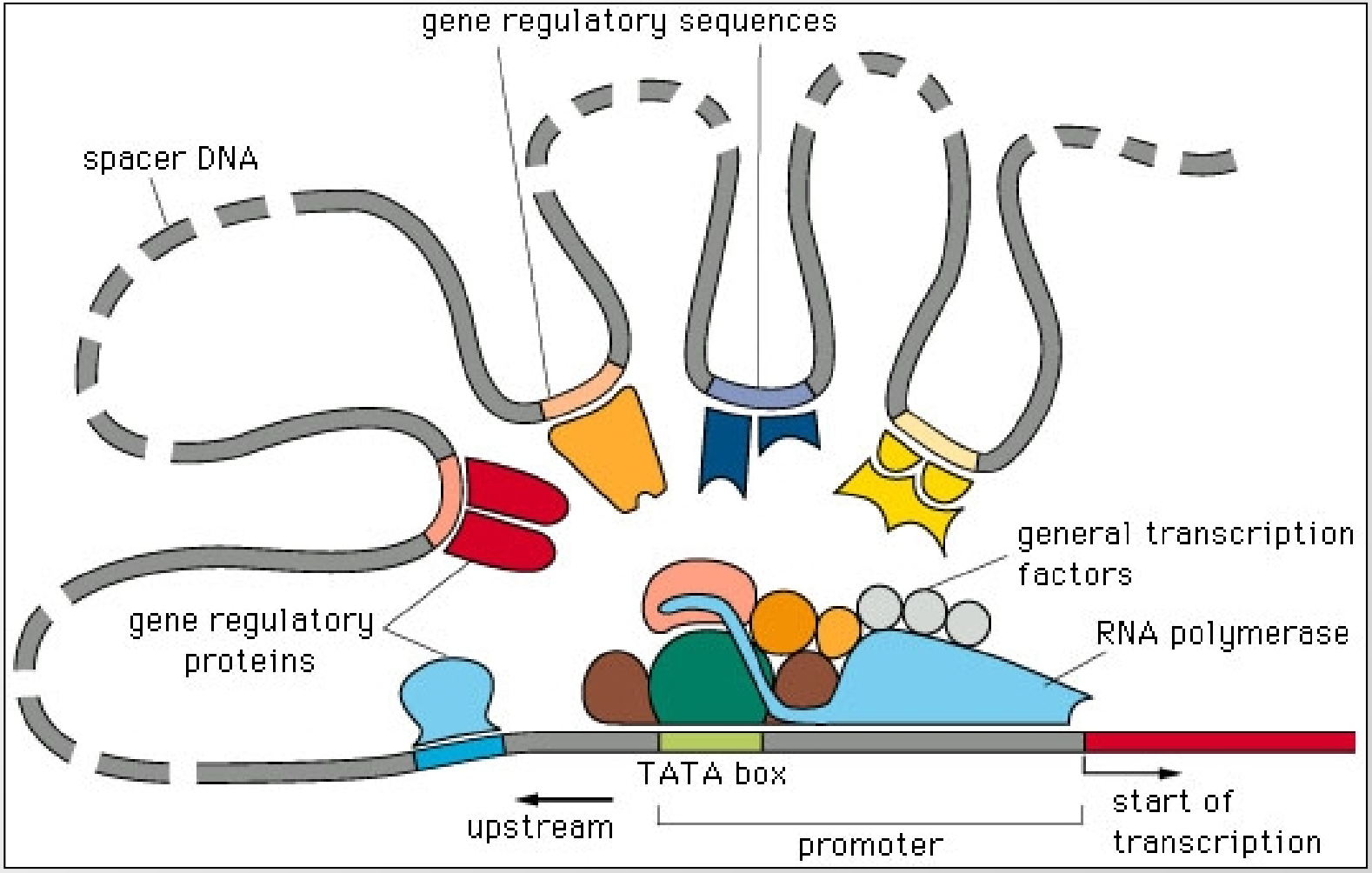


Image Source wikipedia

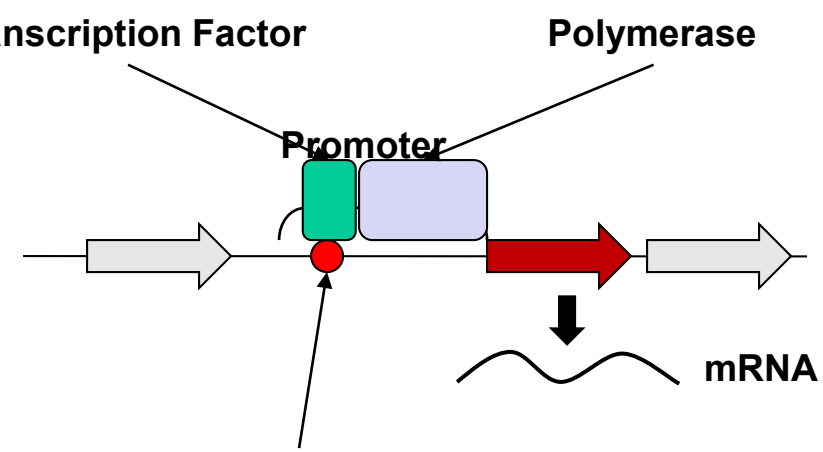
Eukaryotic Gene Regulation



Diverse roles for regulatory non-coding RNAs

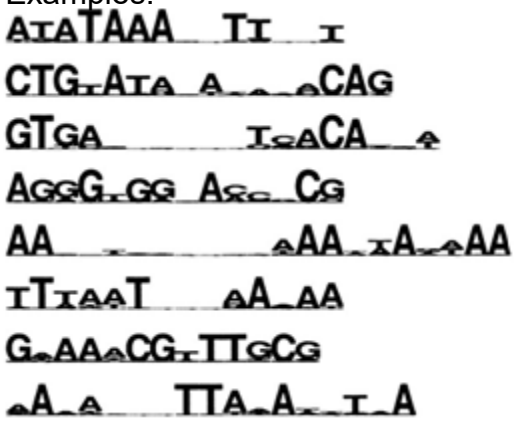
- **Small RNA pathways (18-21 nt)**
  - microRNAs:
    - Repress genes by targeting their 3’ UTRs by complementarity
    - Double-stranded RNA is then recognized and degraded
    - Recently found to also target promoter regions in rare cases
  - piwiRNAs
    - Target and repress transposable elements in germline
  - snoRNAs
  - 21U-RNAs
- **Long non-coding RNAs (1000s nt, many exons)**
  - Scaffolds for protein/TF binding
  - Scaffolds for 3D structure of RNA

Regulation of Gene Expression



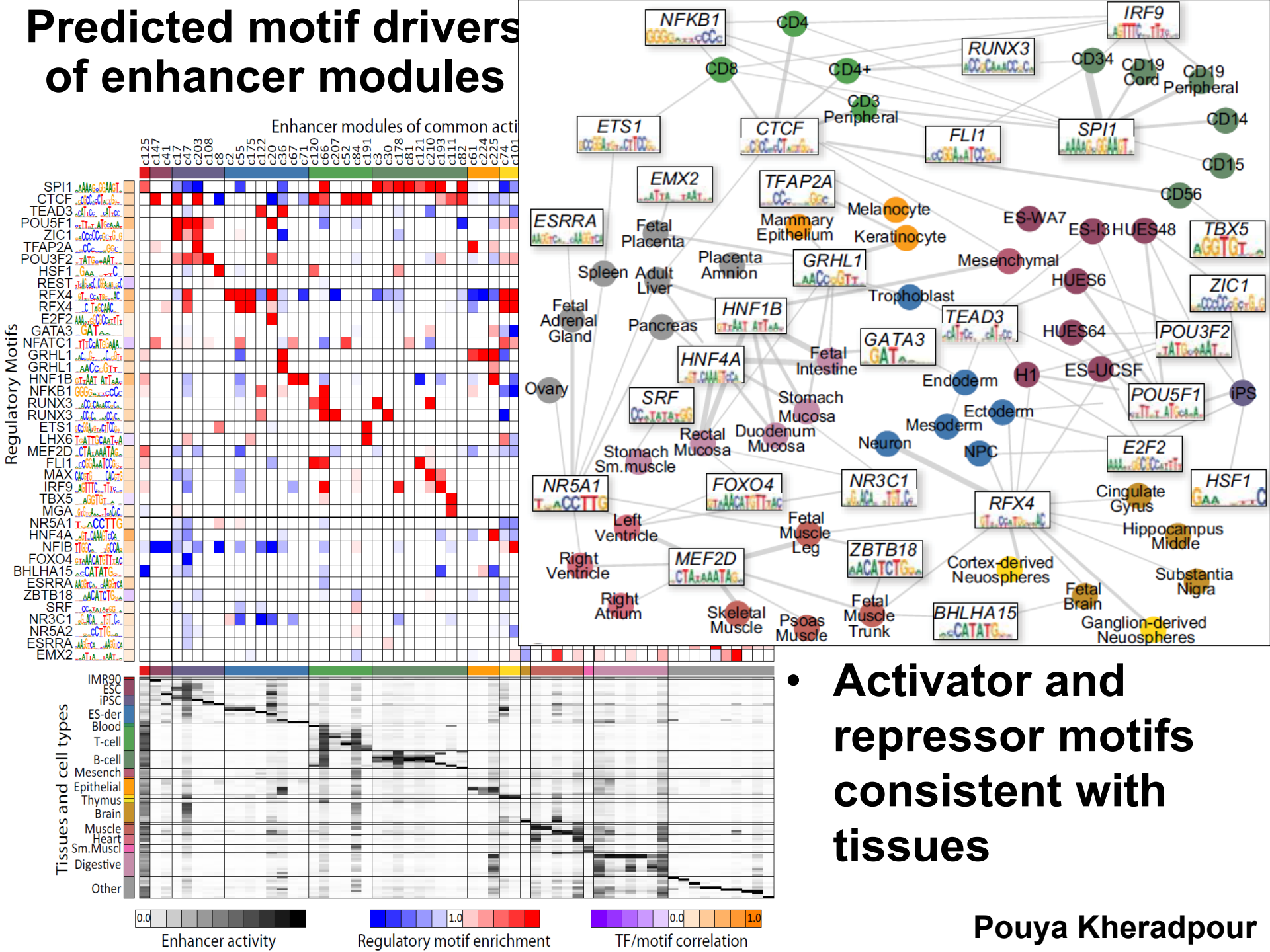
Transcription Factor Binding Site

Examples:



- Upstream of genes are *promoter* regions
- Contain promoter sequences or *motifs*
- *Transcription factors* (TFs) bind to motifs
- TFs recruit *RNA polymerase*
- Gene transcription

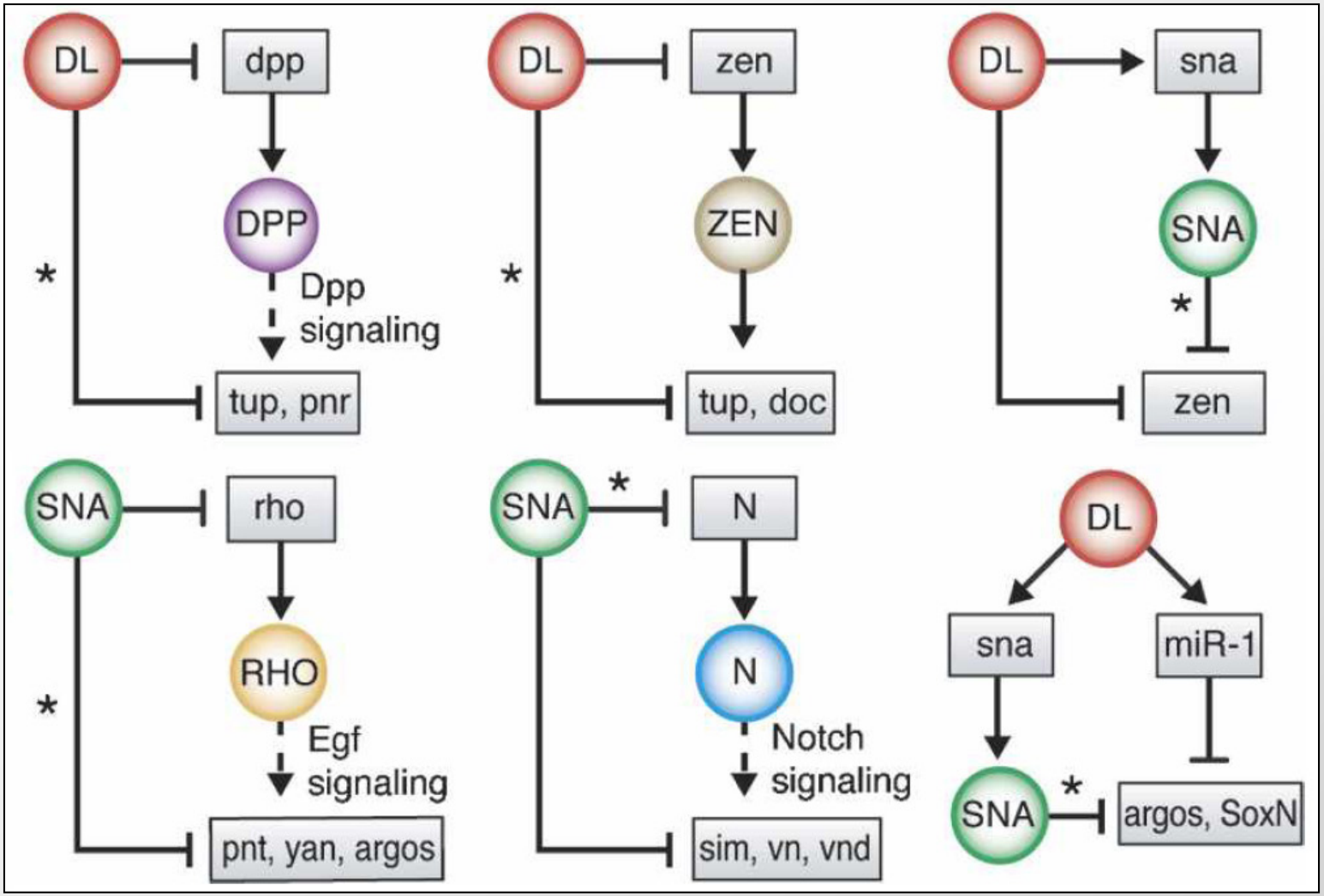
Predicted motif drivers of enhancer modules



- **Activator and repressor motifs consistent with tissues**

Pouya Kheradpour

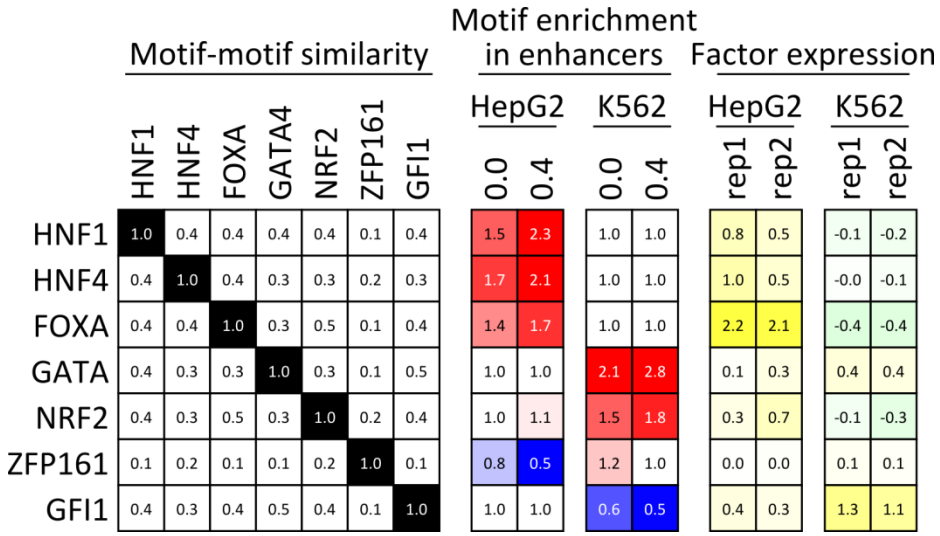
Network components reveal functional modules

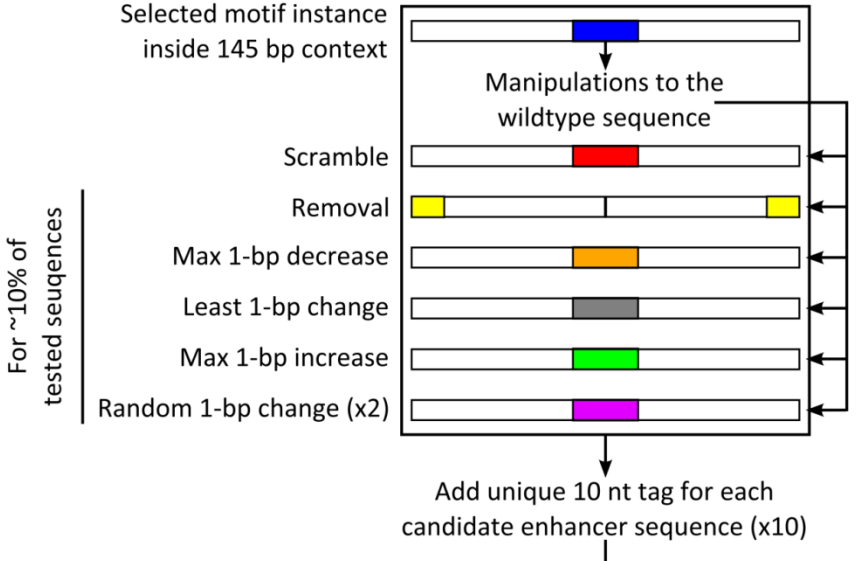


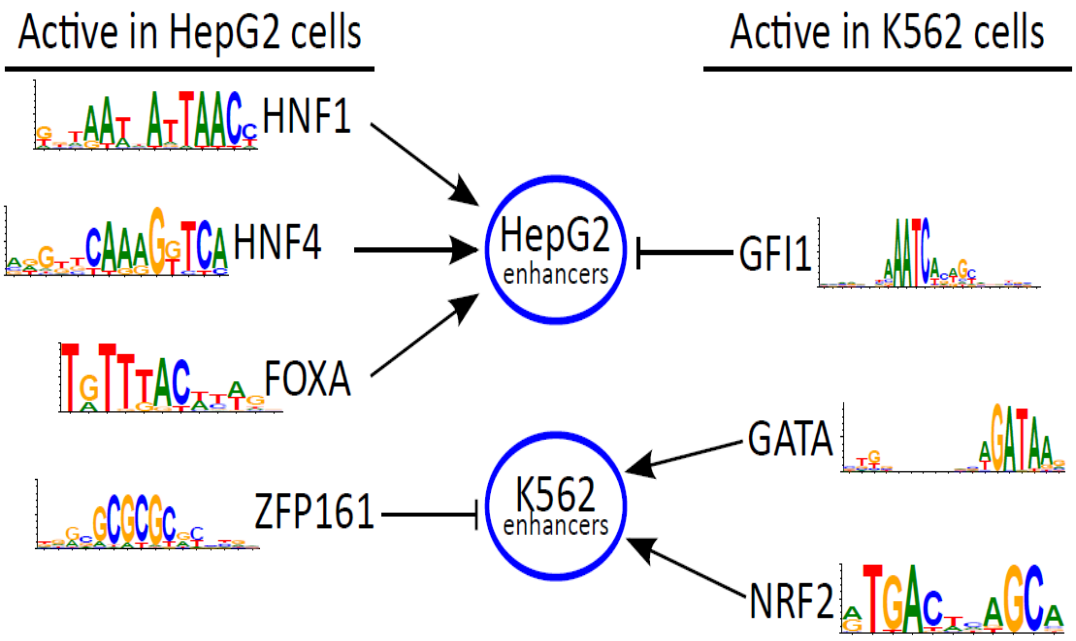
- Feed-forward loops in developmental patterning
- Cooperation of master reg. & downstream reg.

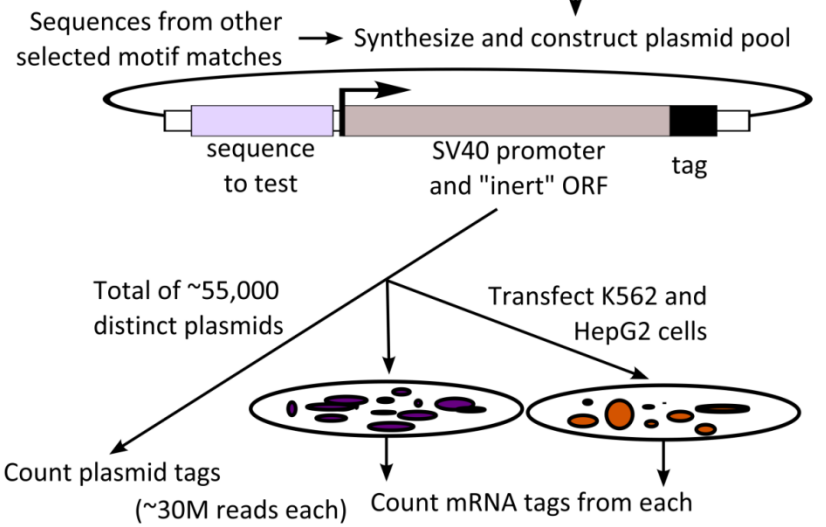
Zeitlinger *et al*, Genes & Development 2007

Systematic motif dissection in 2000 enhancers: 5 activators and 2 repressors in 2 cell lines





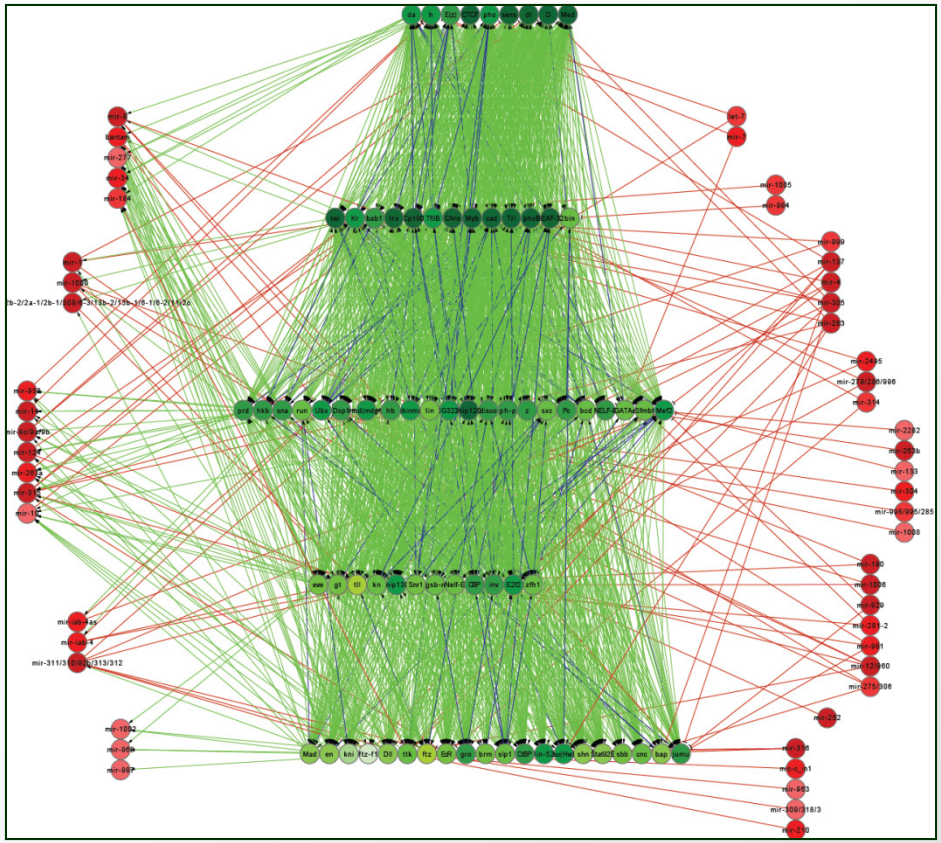


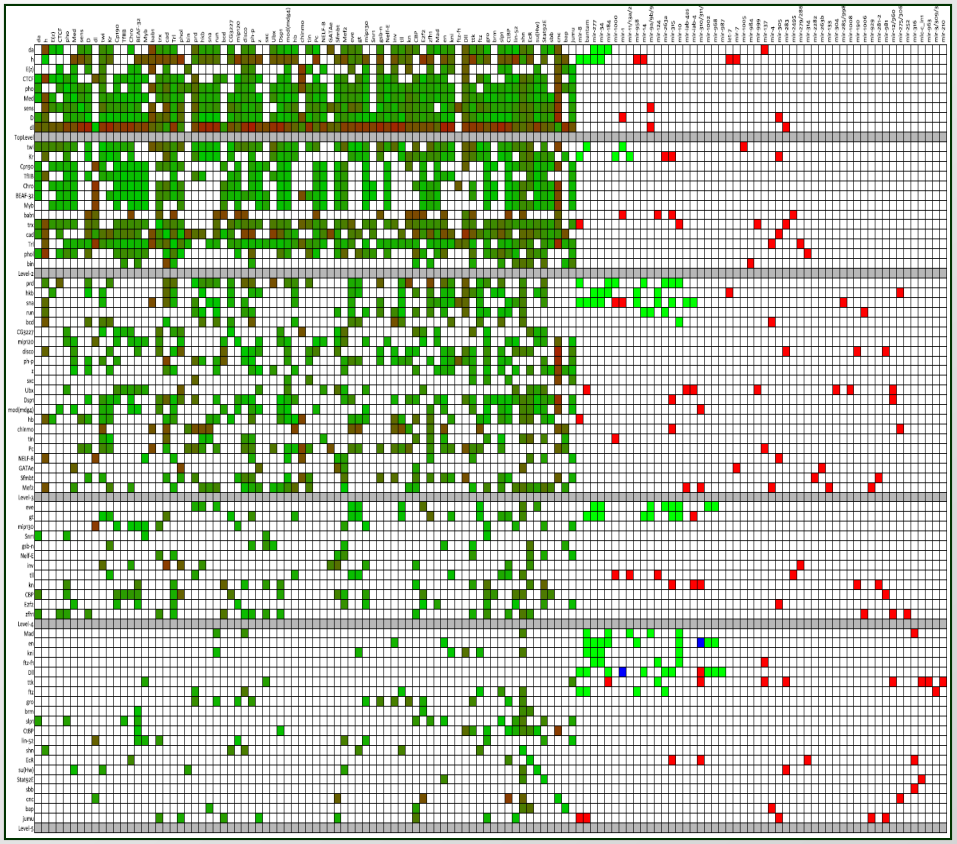


54000+ measurements (x2 cells, 2x repl)

Kheradpour *et al* Genome Research 2013

Emerging properties of regulatory networks

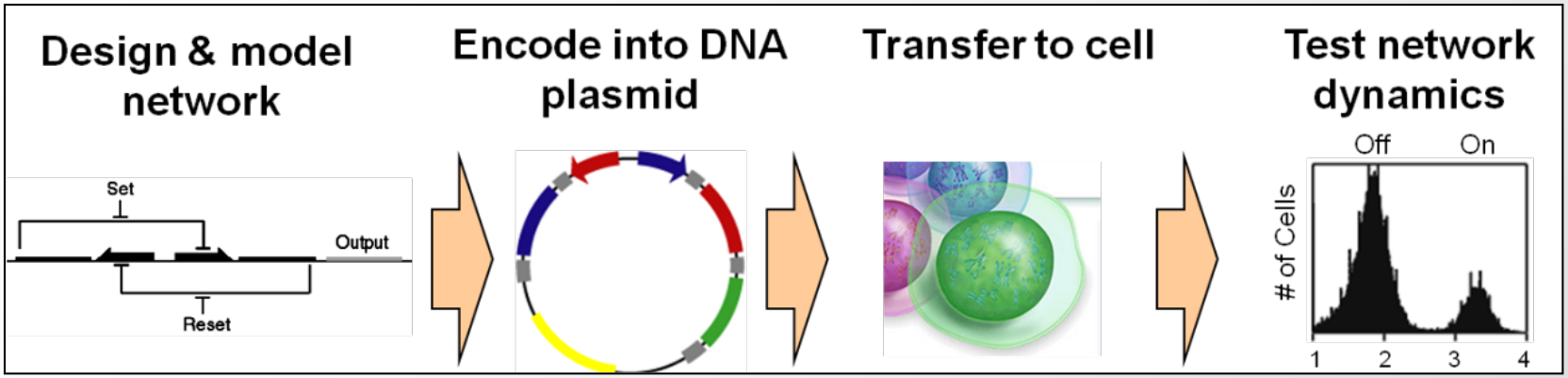




- Hierarchical levels of regulatory control
  - Small number of backward-pointing edges
- Specific / distinct feedback by microRNAs at each level
  - Two classes of TFs: miRNA regulators and miR-regulated

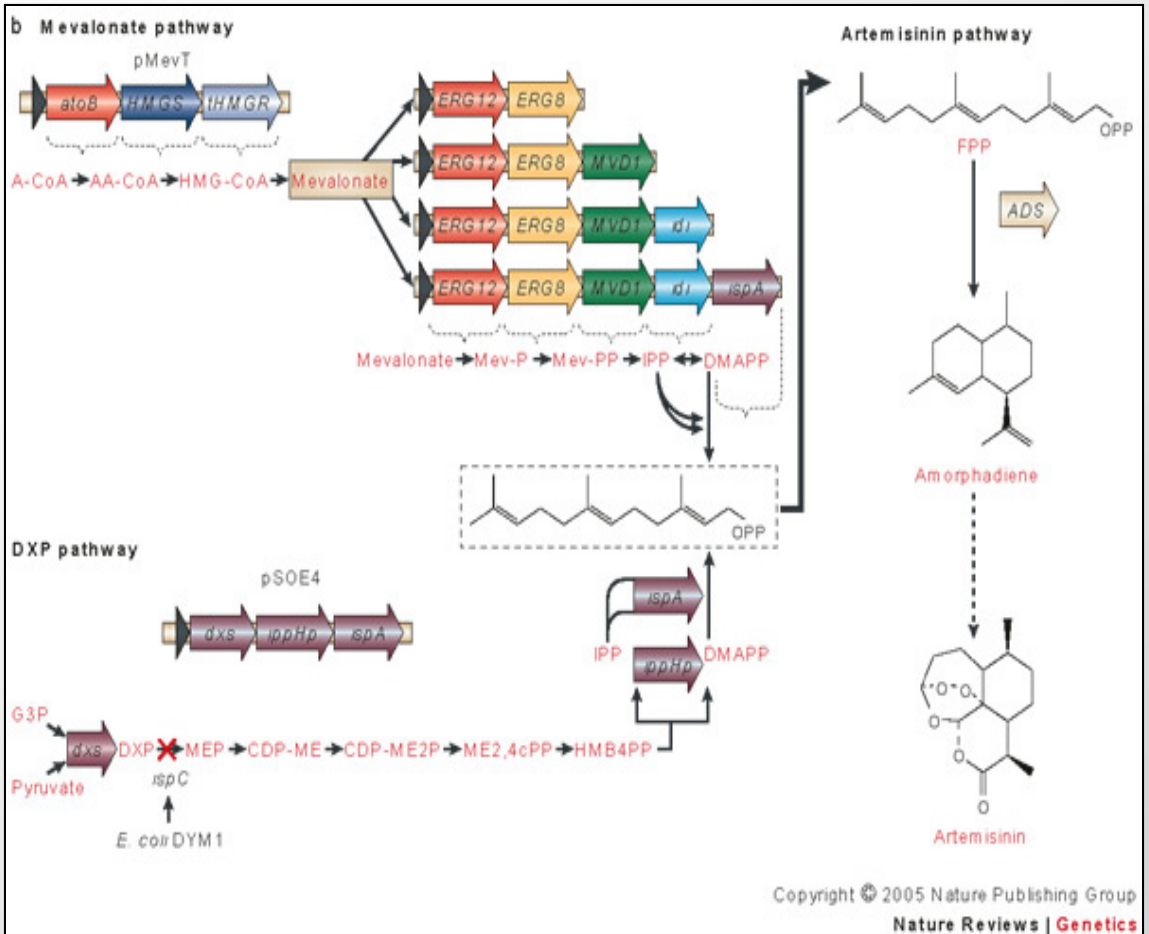
From Systems Biology to Synthetic Biology

Synthetic Regulatory Networks



Jim Collins

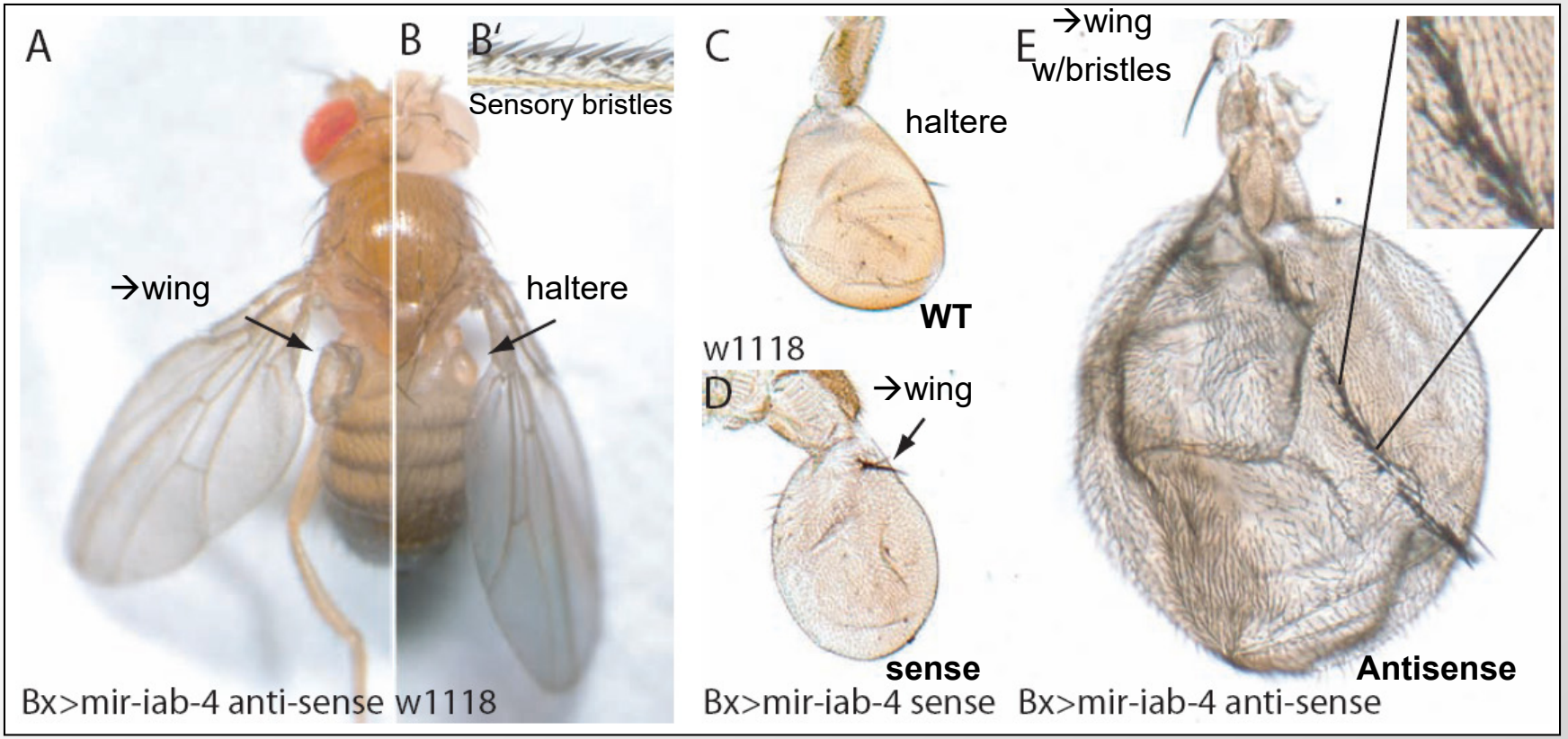
Synthetic Metabolic Pathways



- Components with known properties
- Assemble based on engineering goals / principles
- Implement within engineered cells and organisms
- Study behavior & adjust as needed

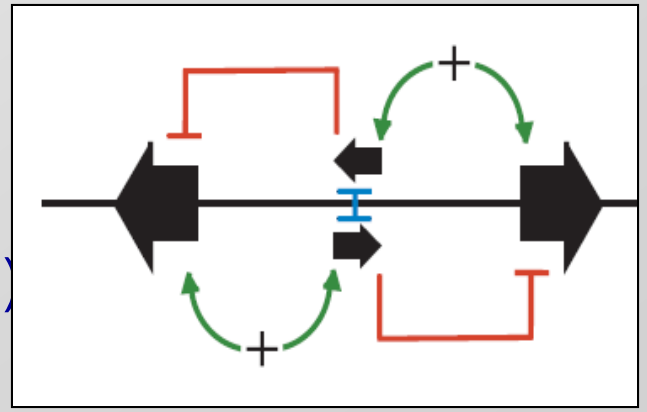
Jay Keasling

Over-express a single microRNA leads to new wing



Note: C,D,E same magnification

- Discovery of sense/anti-sense miRNAs
- Regulatory switch selects between two developmental programs
- By over-expressing one strand (miRNAas) the balance is tilted
- Wing program launched vs. haltere



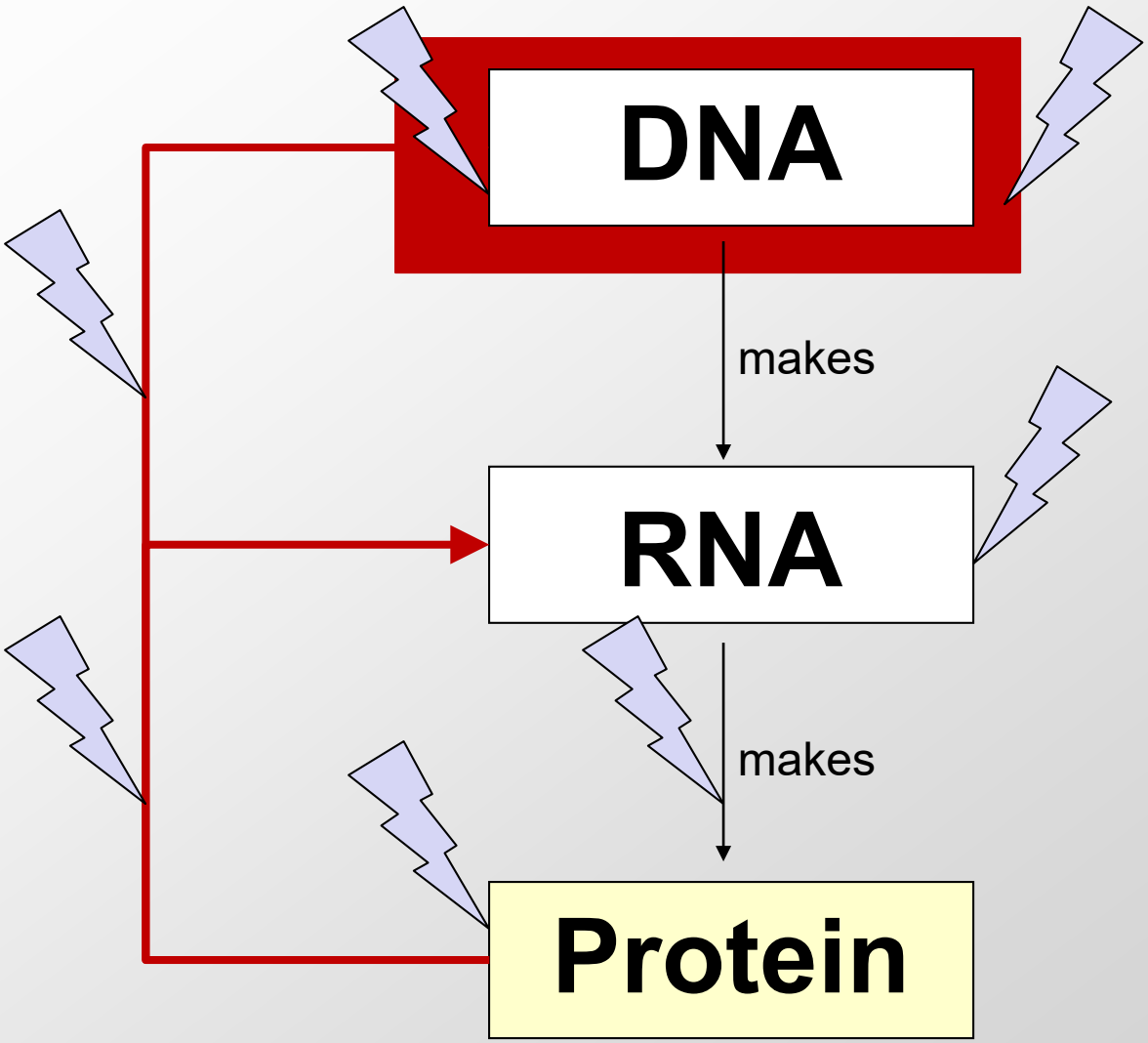
Stark *et al*, Genes&Development 2007

Goals for today: Course Introduction

1. **Course overview:**
   - Staff, students, responses to student survey
   - Foundations, frontiers, textbook, homework, quiz
   - Final project: teams, mentorship, challenge, relevance, originality, achievement, presentation
2. **Why Computational Biology;**
   - What makes our field unique
3. **Overview of the main modules**
   - Genomes, Expression, Epigenomics, Networks, Genetics, Evolution, Frontiers
4. **Biology primer (in the context of this course)**
   - Central Dogma of Molecular Biology
   - DNA, Epigenomics, RNA, Protein, Networks
   - Human genetics, evolution

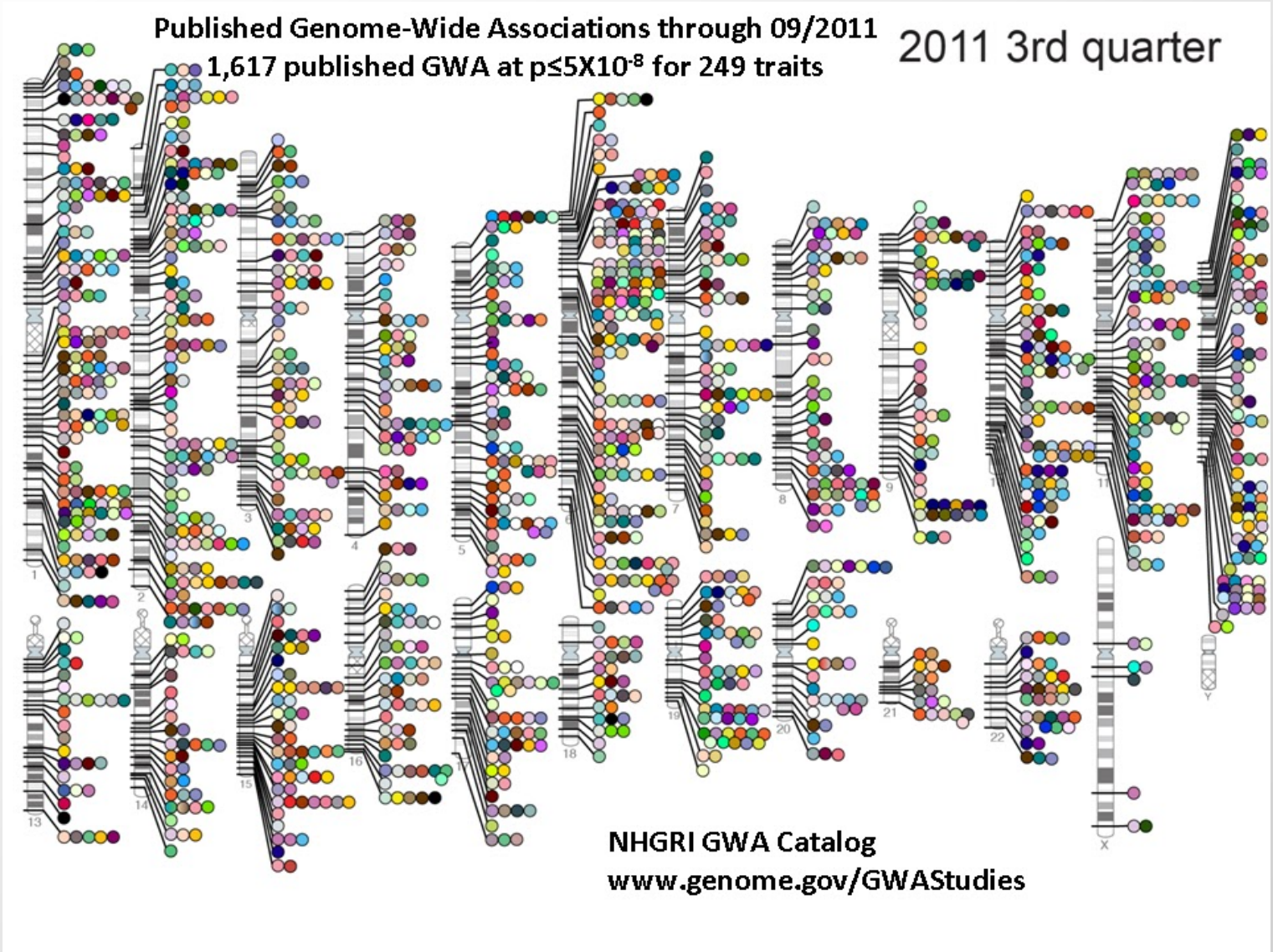
Brief intro to Human Genetics

The role of genetic alterations

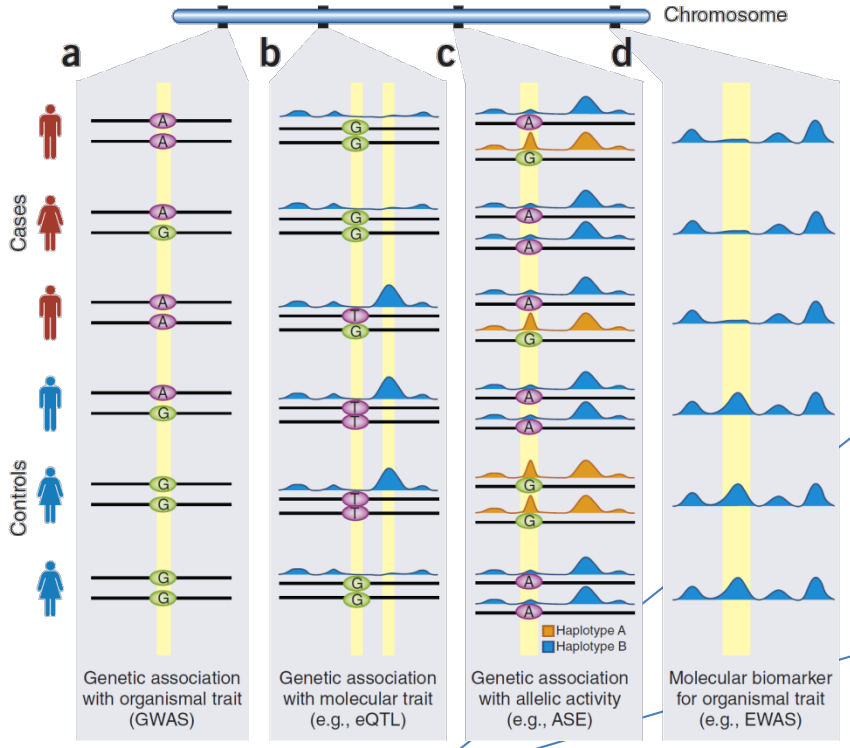


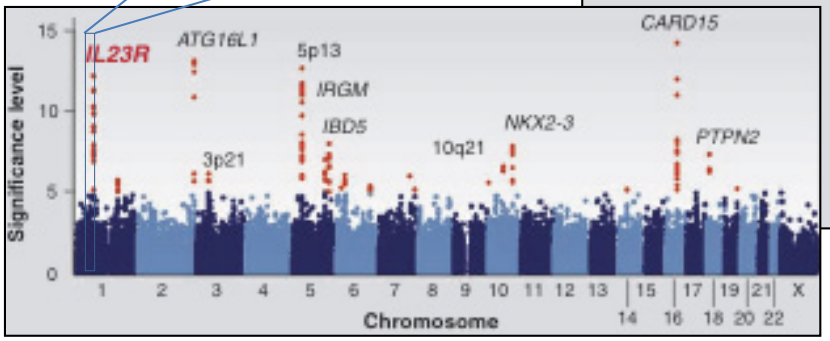
Brief intro to human genetics

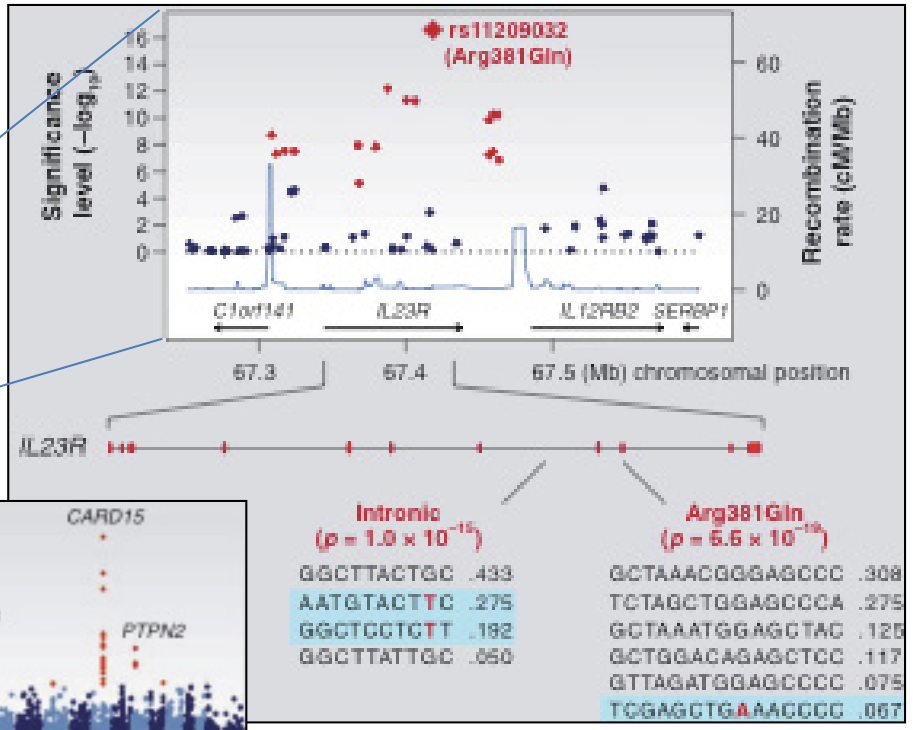
- **Human genome**: 3.2B letters, 2 copies, 23 chromosomes, 20k genes, ~3M common SNPs, ~500k haplotype blocks



The power and challenge of disease-association studies





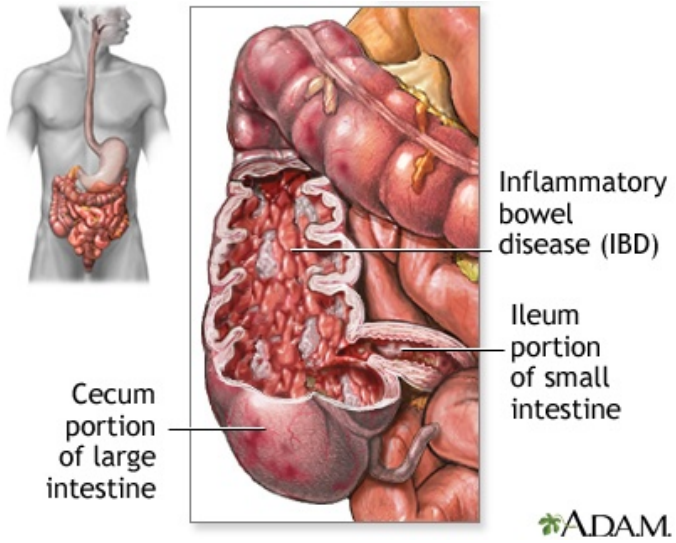


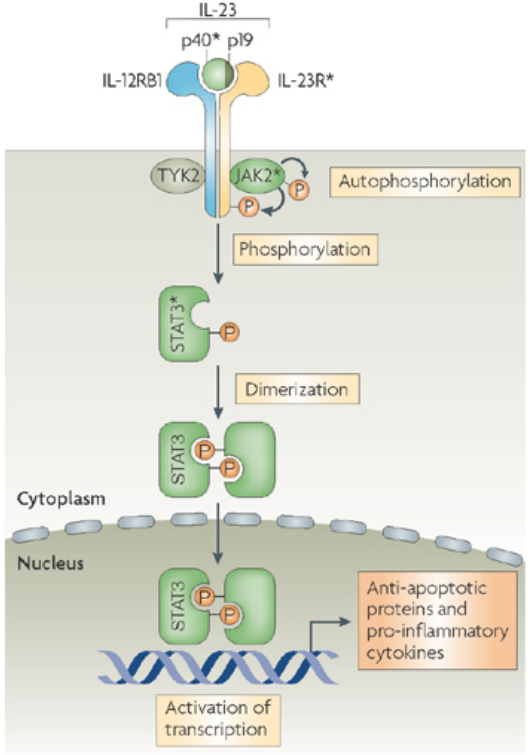
Slide credit: Luke Ward, Mark Daly

- **Large associated blocks with many variants: Fine-mapping challenge**
- **No information on cell type/mechanism, most variants non-coding**

➔ **Epigenomic annotations help find relevant cell types / nucleotides**

The power of GWAS: reveal new disease genes





rs11209026	A	G
Cases	22	976
Controls	68	932

Chi-sq = 24.5, p=7.3 x 10^{-7}

IL23R cytokine receptor on a subset of effector T-cells

Genomewide association in schizophrenia with 40,000 cases

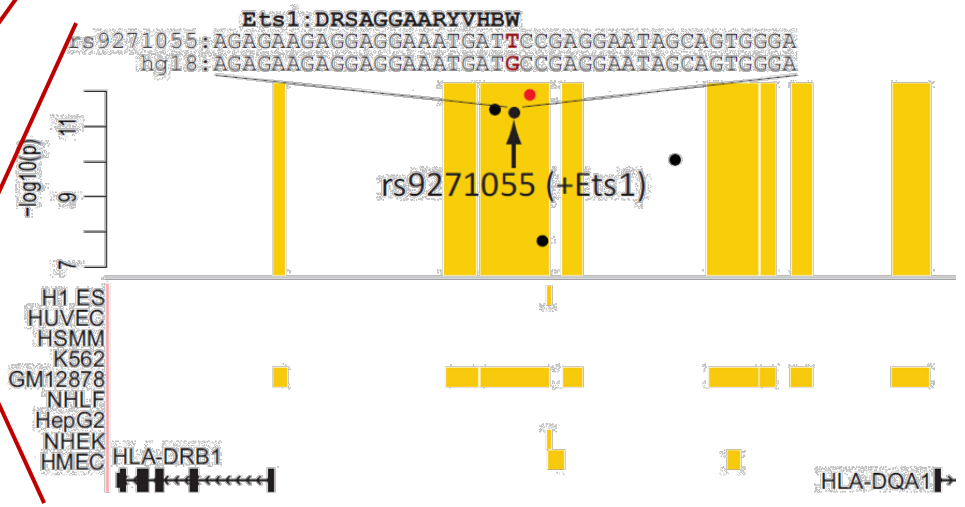
Stephan Ripke

More than 100 distinct regions of the genome associated to schizophrenia!!!

Interpreting non-coding variants

Phenotype	Top Cell Type	Total #SNPs from Study	#SNPs in enh. States 4 and 5	p-value	FDR	H1 ES	K562	GM12878	HepG2	HUVEC	HSMM	NHLF	NHEK	HMEC
Erythrocyte phenotypes (Ref. 38)	K562	35	9	$<10^{-7}$	0.02	9	17	4	0	0	1	2	1	1
Blood lipids (Ref. 39)	HepG2	101	13	$<10^{-7}$	0.02	3	5	0	11	2	3	3	4	3
Rheumatoid arthritis (Ref. 40)	GM12878	29	7	2.0×10^{-7}	0.03	0	0	15	0	2	0	0	2	3
Primary billary cirrhosis (Ref. 41)	GM12878	6	4	6.0×10^{-7}	0.03	0	11	41	0	0	0	0	8	8
Systemic lupus erythromatosus (Ref. 42)	GM12878	18	6	9.0×10^{-7}	0.03	0	4	21	0	5	8	0	3	5
Lipoprotein cholesterol/triglycerides (Ref. 43)	HepG2	18	5	1.2×10^{-6}	0.03	17	8	0	24	3	6	4	3	3
Hematological traits (Ref. 44)	K562	39	7	1.7×10^{-6}	0.03	0	12	10	2	1	0	0	1	0
Hematological parameters (Ref. 45)	K562	28	6	2.2×10^{-6}	0.03	0	15	7	0	5	7	7	3	2
Colorectal cancer (Ref. 46)	HepG2	4	3	3.8×10^{-6}	0.03	0	0	0	66	0	12	0	12	12
Blood pressure (Ref. 47)	K562	9	4	5.0×10^{-6}	0.04	0	30	14	0	10	6	7	5	11

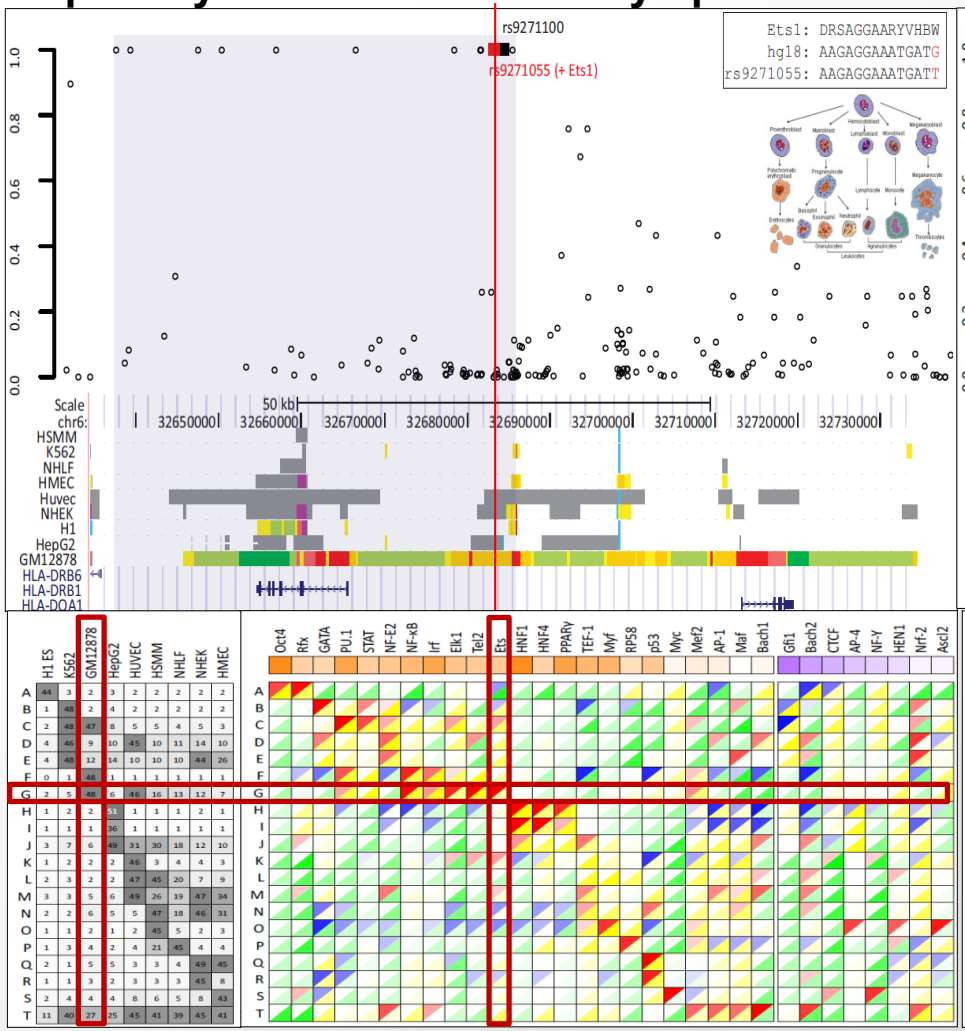
SNP	H1 ES	K562	GM	HepG2	Huvec	HSMM	NHLF	NHEK	HMEC	Chrom. Band	Gene	Link Scc	Distanc
rs13385731										2p22			
rs10036748										5q33			
rs1385374										12q24	MGC16384	-	1
rs2230926										6q23	TNFAIP3	3.7	7
rs4728142										7q32	IRF5	-	4
rs9271100										6p21	HLA-DRB1	4.5	19
rs4917014										7p12	IKZF1	2.2	38
rs7812879										8p23	BLK	2.9	11
rs2205960										1q25			
rs548234										6q21			



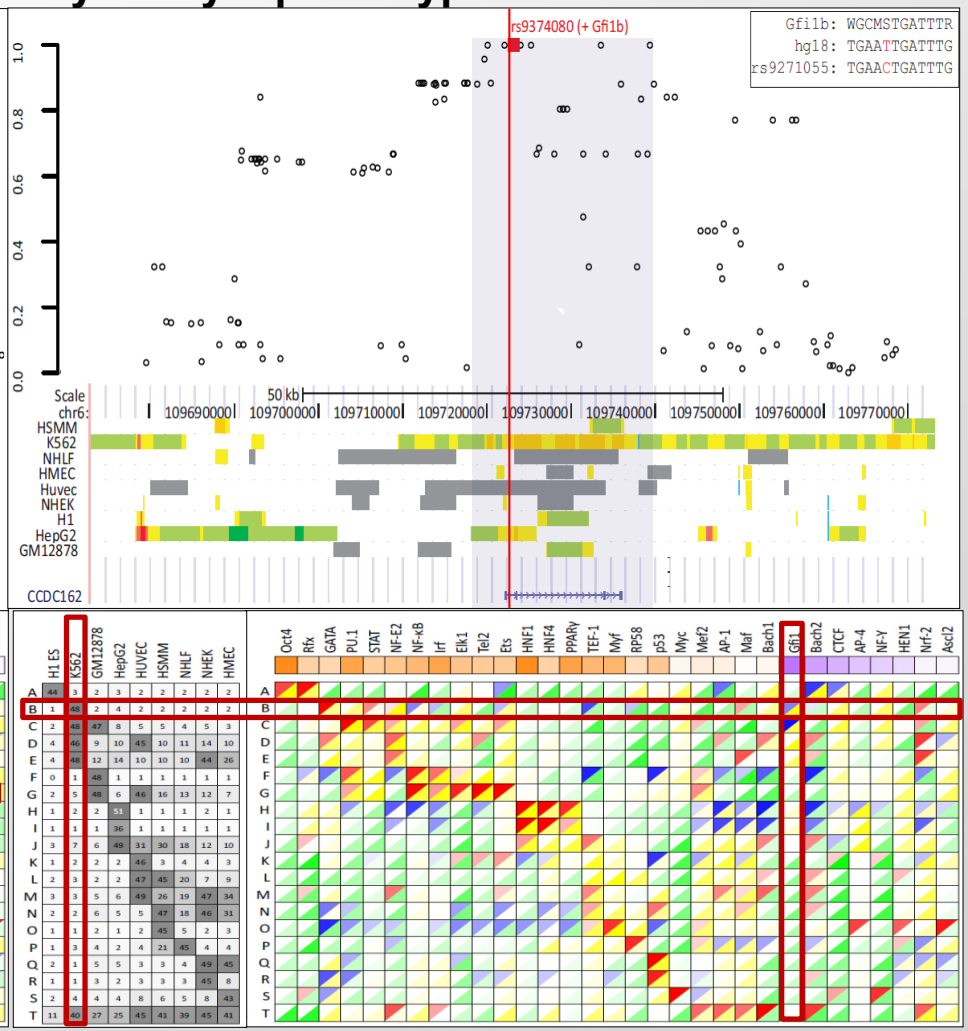
- **Disease-associated SNPs enriched for enhancers in relevant cell types**
- **E.g. lupus SNP in GM enhancer disrupts Ets1 predicted activator**

Mechanistic predictions for top disease-associated SNPs

Lupus erythromatosus in GM lymphoblastoid



Erythrocyte phenotypes in K562 leukemia cells



Disrupt activator Ets-1 motif

➔ **Loss of GM-specific activation**

➔ **Loss of enhancer function**

➔ **Loss of HLA-DRB1 expression**

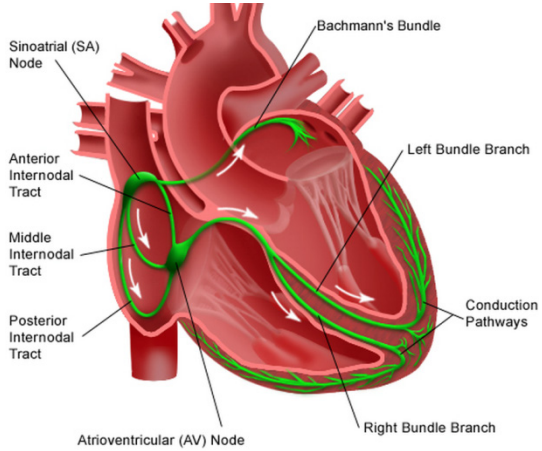
Creation of repressor Gfi1 motif

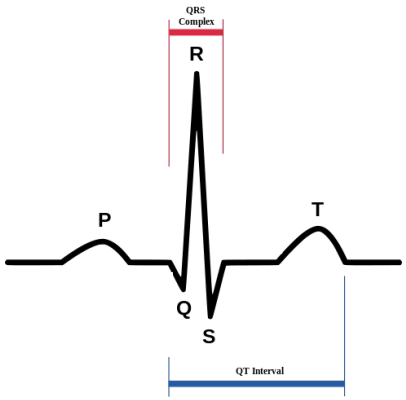
➔ **Gain K562-specific repression**

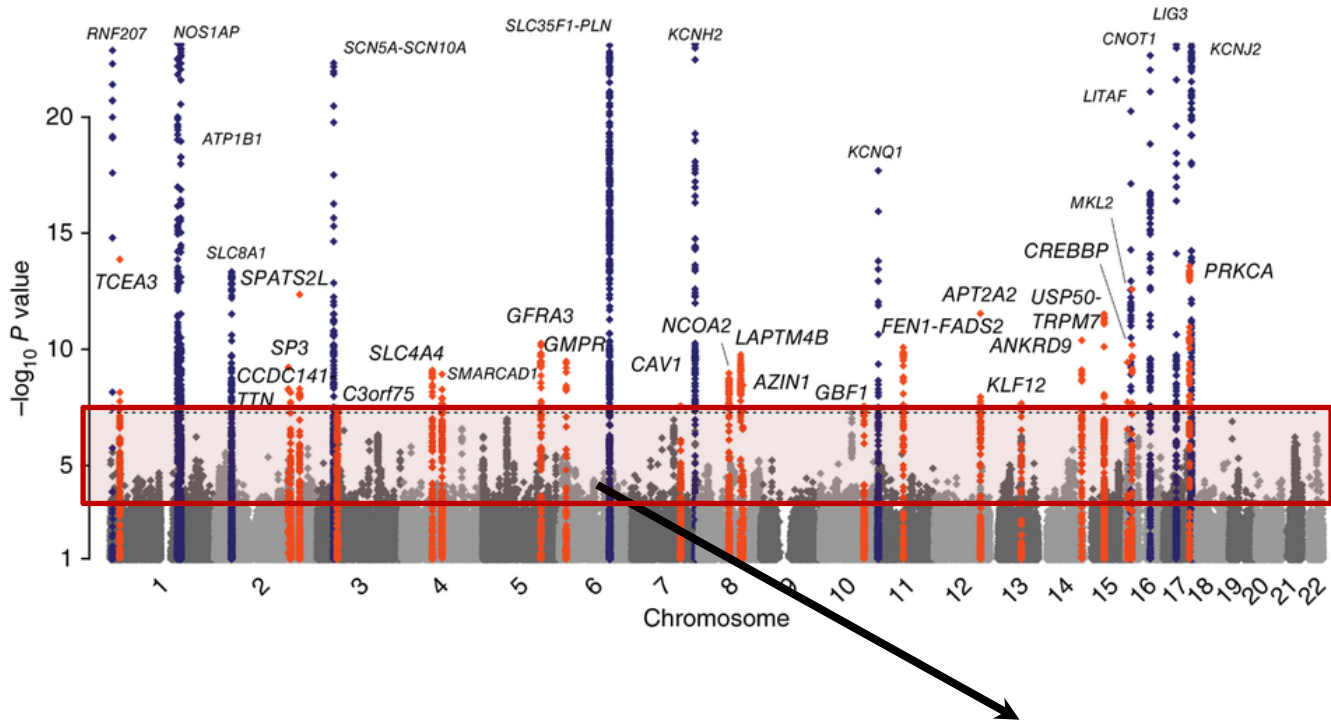
➔ **Loss of enhancer function**

➔ **Loss of CCDC162 expression**

Characterizing sub-threshold variants in heart arrhythmia





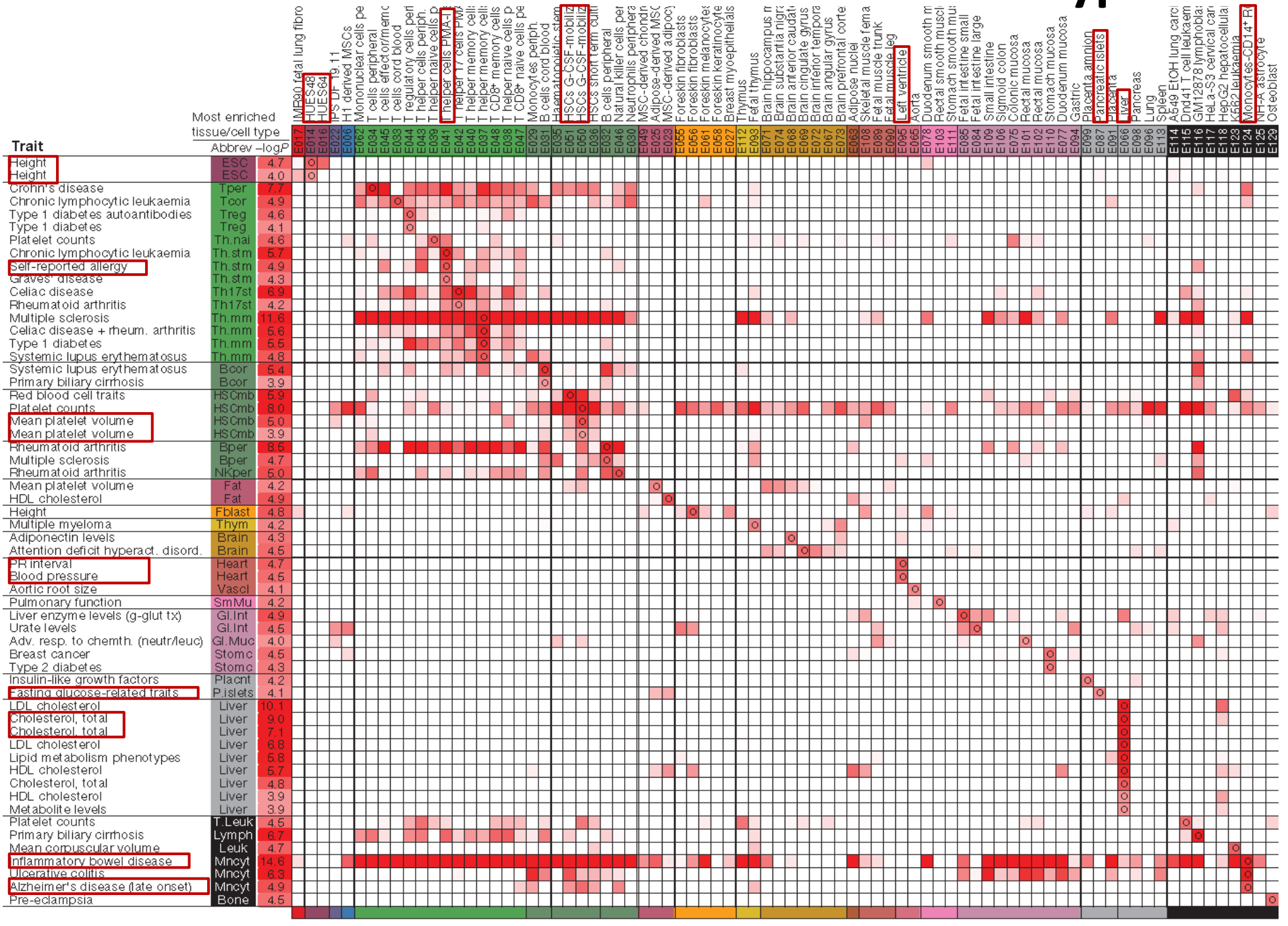


Focus on sub-threshold variants (e.g. rs1743292 $P=10^{-4.2}$)

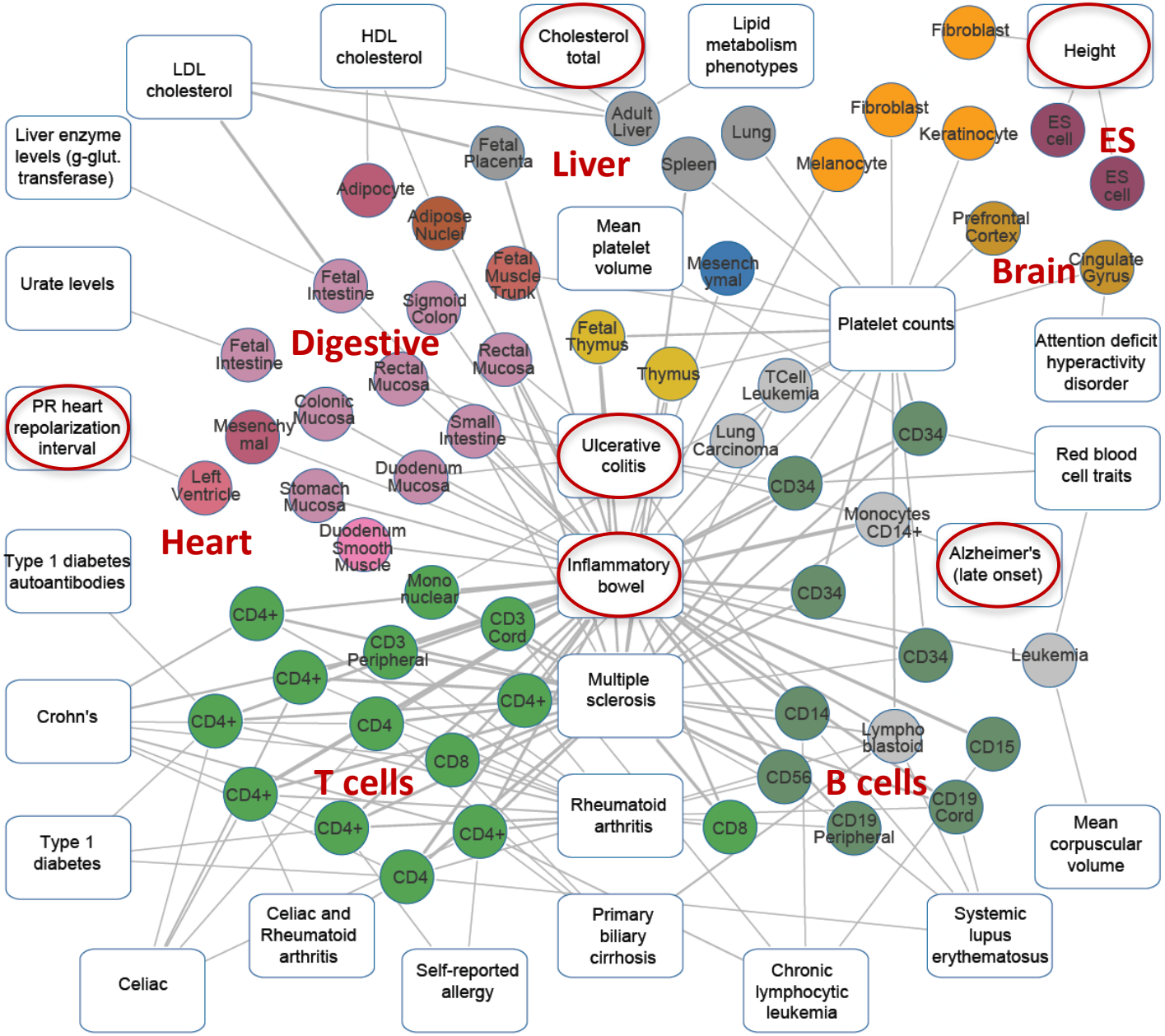
Trait: QRS/QT interval

(1) Large cohorts, (2) many known hits
(3) well-characterized tissue drivers

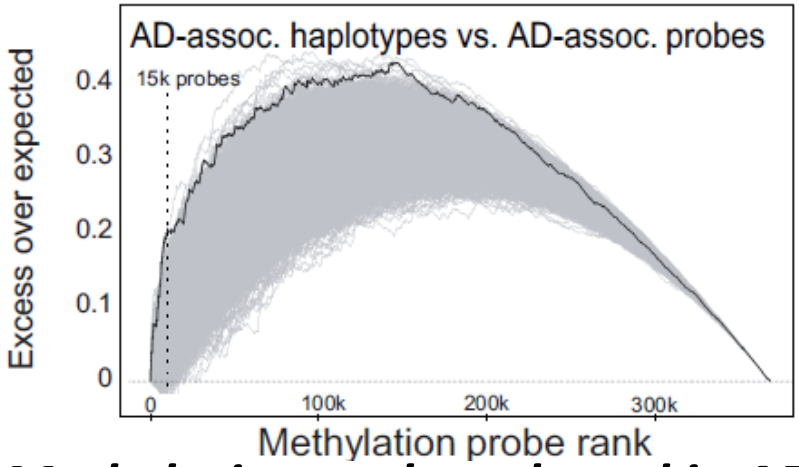
GWAS hits in enhancers of relevant cell types



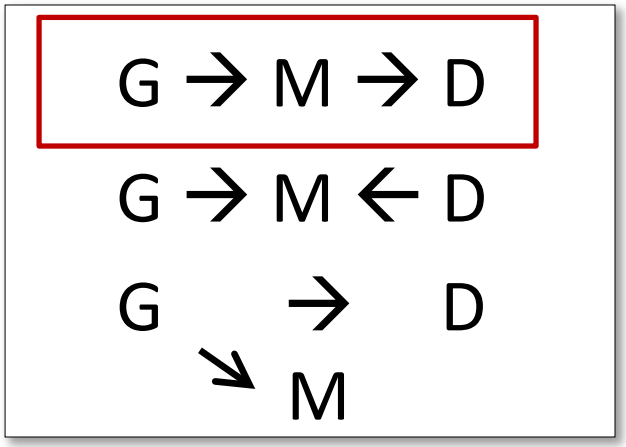
Linking traits to their relevant cell/tissue types



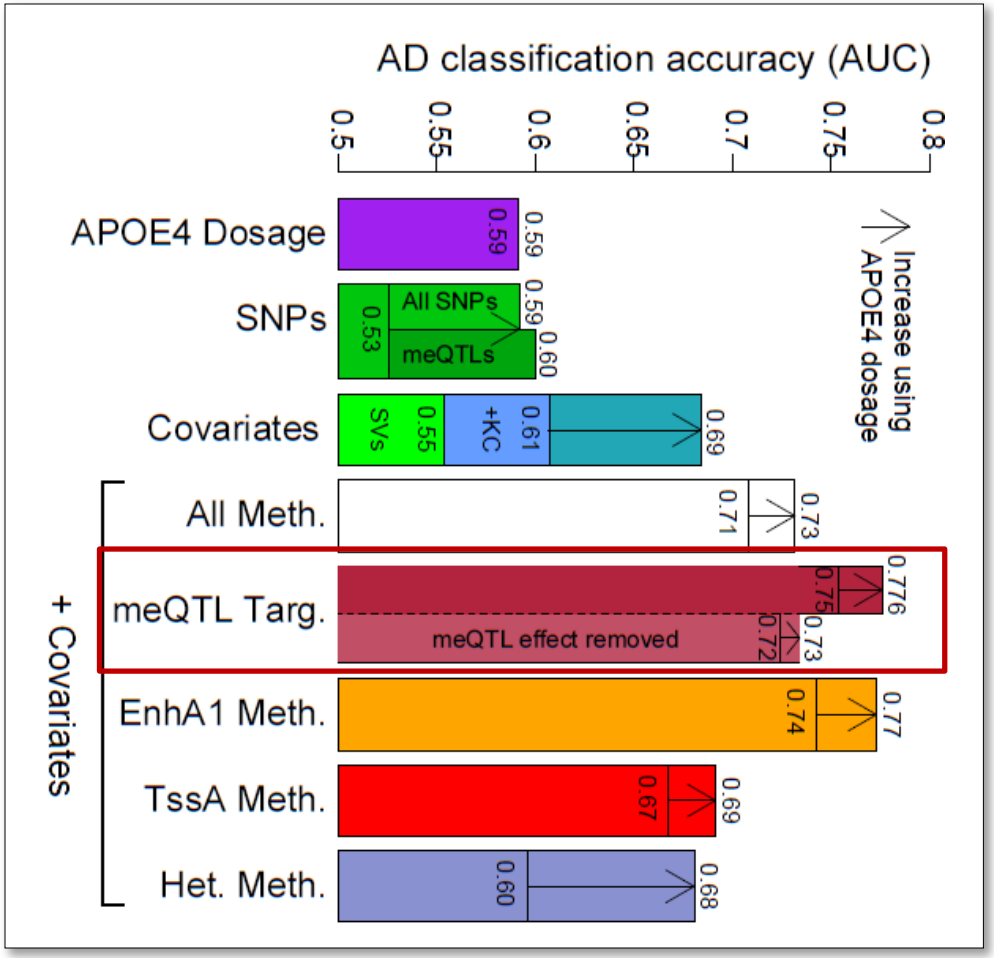
Methylation differences a causal component of AD



Methylation probes altered in AD are enriched in AD-associated SNPs

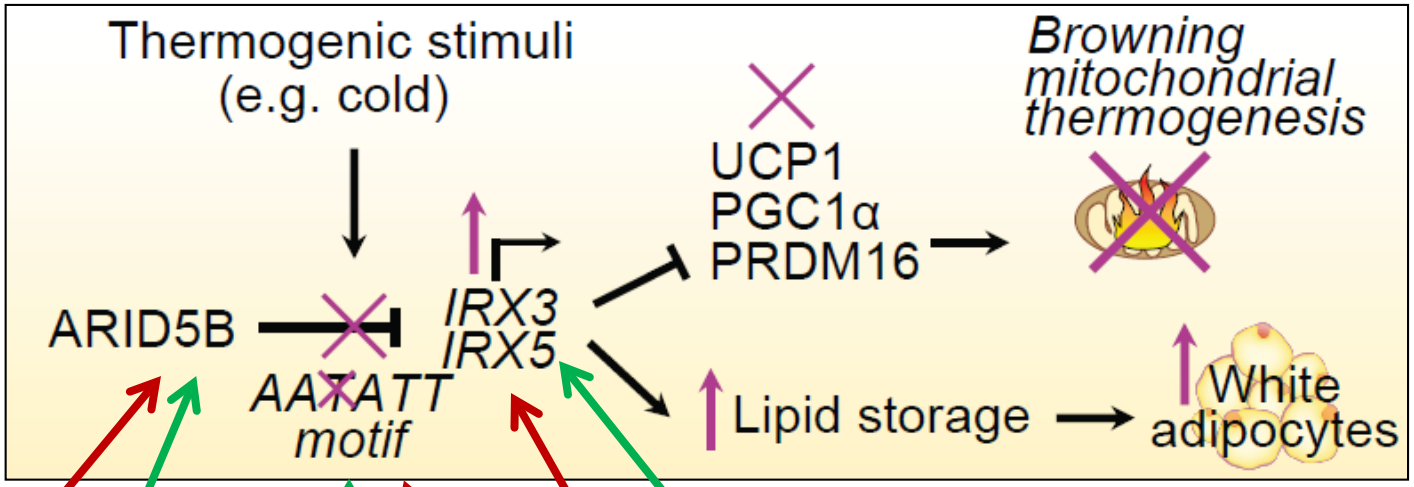


Set-wise causality testing



AD predictive power reduced after removing meQTL effect

Uncovering the molecular basis of top obesity gene



ARID5B KD (obesity)

ARID5B OE (anti-obesity)

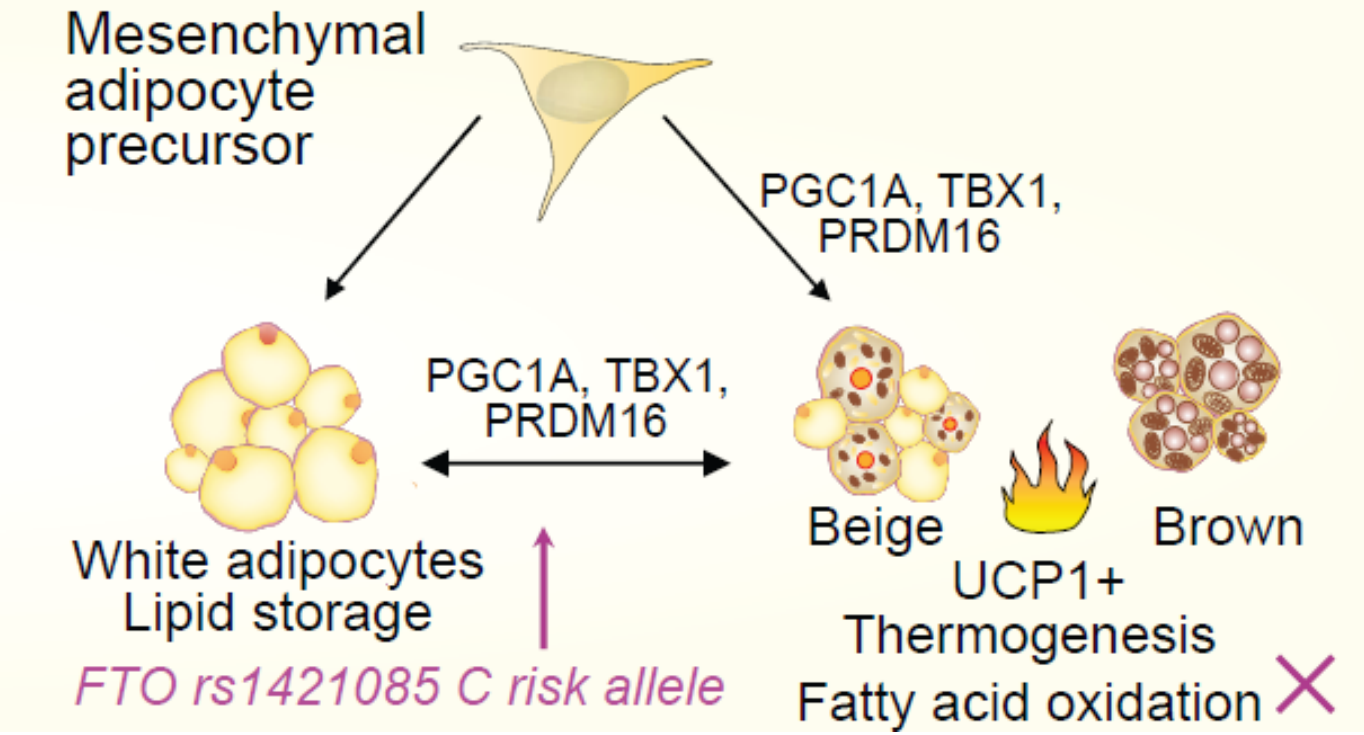
IRX3, IRX5 knock-down ★ (anti-obesity phenotypes)

IRX3, IRX5 overexpression (pro-obesity phenotypes)

★ C-to-T motif rescue (anti-obesity phenotypes)

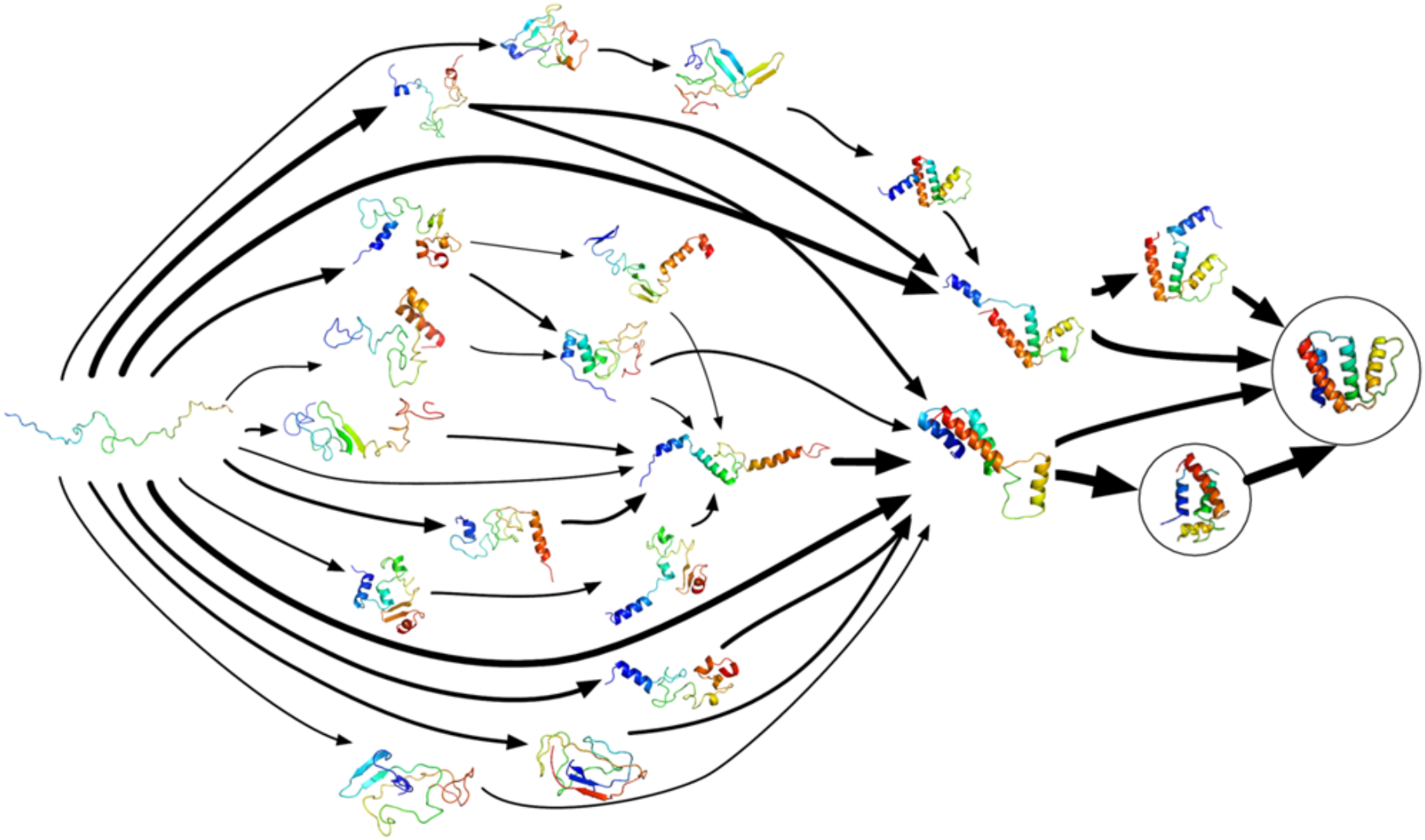
T-to-C motif disruption (pro-obesity phenotypes)

Model: beige ⇔ white adipocyte development

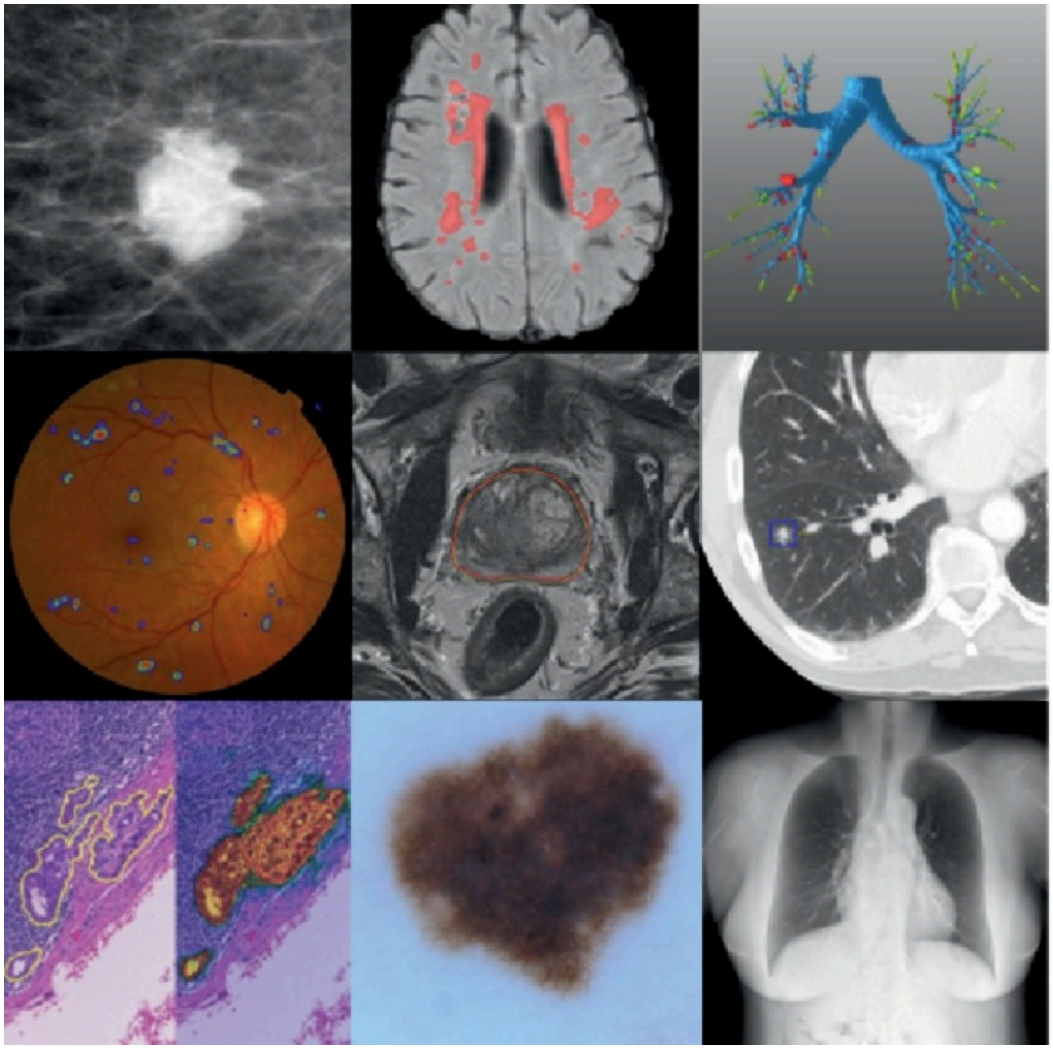


Shift therapeutic focus from brain to adipocytes

Protein Folding + Drug Design



Medical Image Analysis



Collage of some medical imaging applications in which deep learning has achieved state-of-the-art results.

From top-left to bottom-right:

1. mammographic mass classification
2. segmentation of lesions in the brain,
3. leak detection in airway tree segmentation,
4. diabetic retinopathy classification
5. prostate segmentation,
6. nodule classification,
7. breast cancer metastases detection,
8. skin lesion classification
9. bone suppression