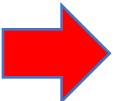




Making Deep Learning Interpretable for Analyzing Sequential Data about Gene Regulation

Dr. Yanjun Qi
Department of Computer Science
University of Virginia

Today

- 
- Machine Learning: a quick review
 - Deep Learning: a quick review
 - Background Biology: a quick review
 - Deep Learning for analyzing **Sequential Data** about Regulation:
 - DeepChrome
 - AttentiveChrome
 - DeepMotif

<https://qdata.github.io/deep2Read/>

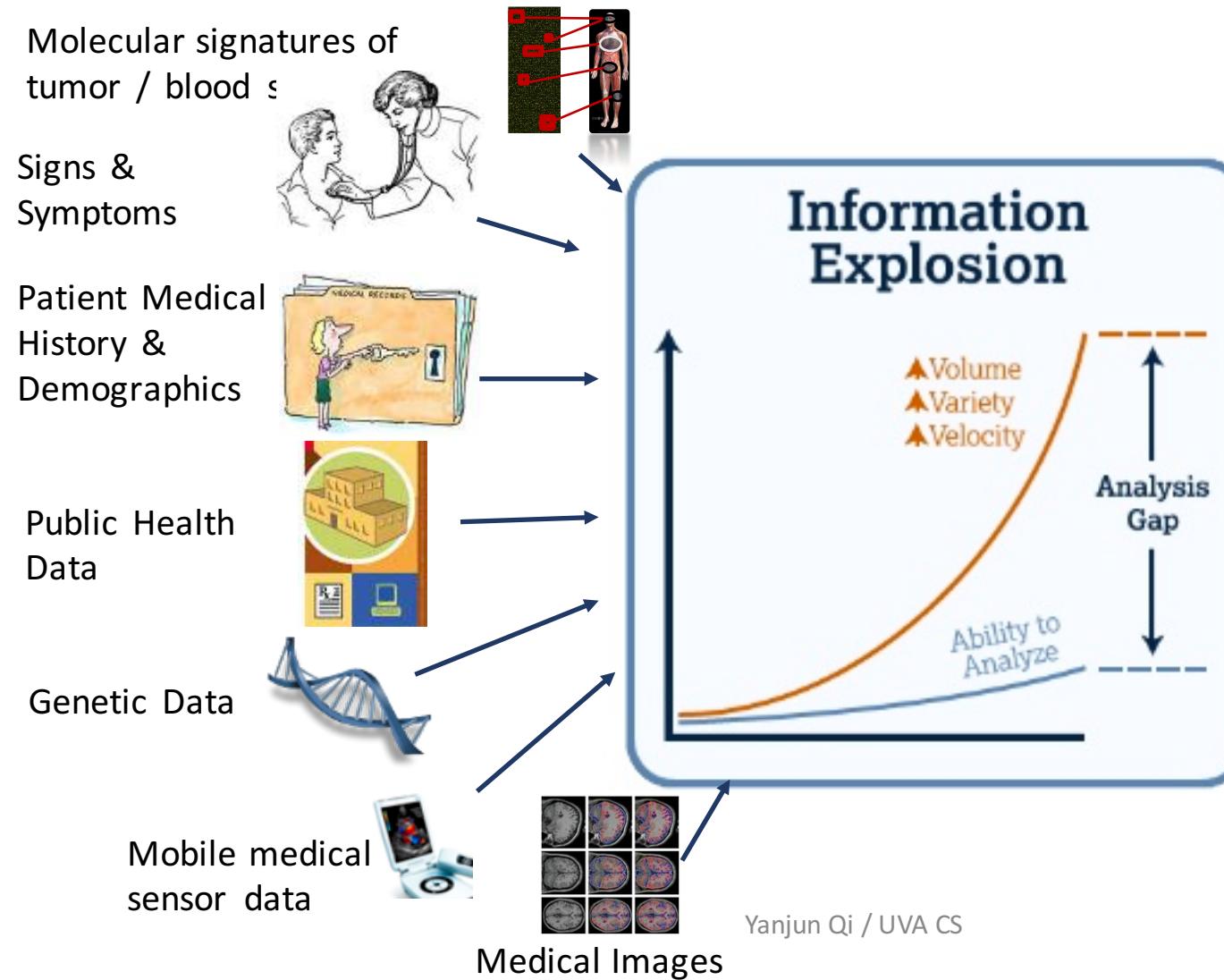
<https://www.deepchrome.org>

OUR DATA-RICH WORLD



- Biomedicine
 - Patient records, brain imaging, MRI & CT scans, ...
 - Genomic sequences, bio-structure, drug effect info, ...
- Science
 - Historical documents, scanned books, databases from astronomy, environmental data, climate records, ...
- Social media
 - Social interactions data, twitter, facebook records, online reviews, ...
- Business
 - Stock market transactions, corporate sales, airline traffic, ...

Challenge of Data Explosion in Biomedicine



Traditional
Approaches

Machine
Learning

BASICS OF MACHINE LEARNING

- “The goal of machine learning is to build computer systems that can **learn and adapt from their experience.**” – Tom Dietterich
- “**Experience**” in the form of available **data examples** (also called as instances, samples)
- Available examples are described with properties (**data points in feature space X**)

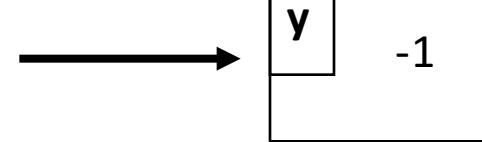
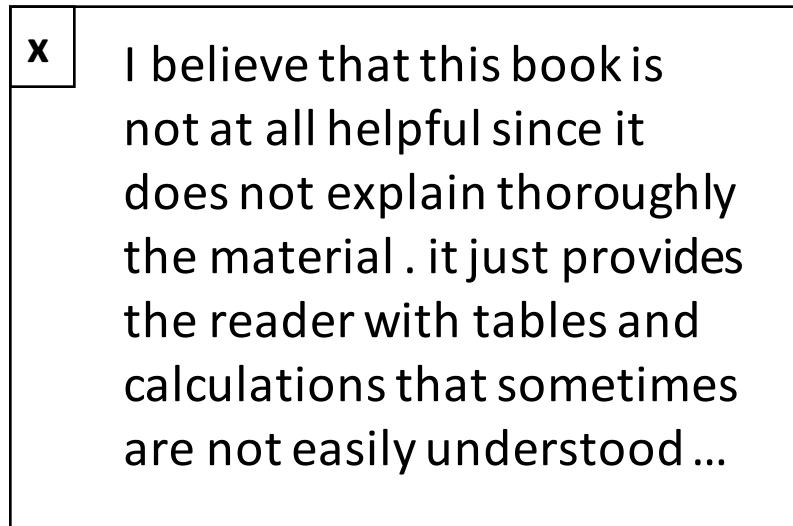
e.g. SUPERVISED LEARNING

- Find function to map **input** space X to **output** space Y

$$f : X \longrightarrow Y$$

- So that the **difference** between y and $f(x)$ of each example x is small.

e.g.



Output Y: {1 / Yes , -1 / No }
e.g. Is this a positive product review ?

Input X : e.g. a piece of English text

SUPERVISED Linear Binary Classifier

- Now let us check out a **VERY SIMPLE** case of

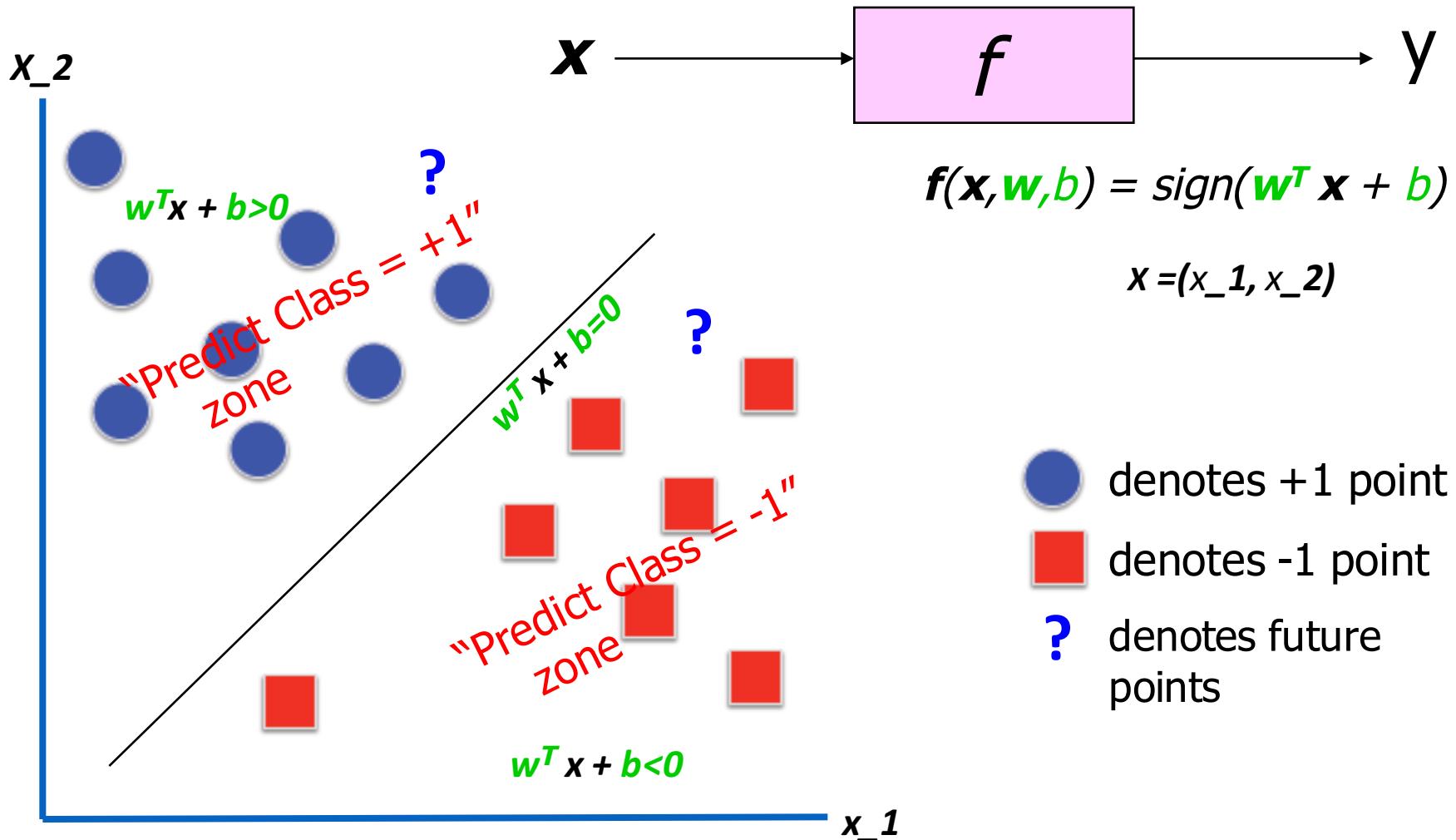


e.g.: Binary y / Linear f / X as \mathbb{R}^2

$$f(x, w, b) = \text{sign}(w^T x + b)$$

$$x = (x_1, x_2)$$

SUPERVISED Linear Binary Classifier



Basic Concepts

- Training (i.e. learning parameters (\mathbf{w}, b))
 - Training set includes
 - available examples' feature representation: $\mathbf{x}_1, \dots, \mathbf{x}_L$
 - available corresponding labels y_1, \dots, y_L
 - Find (\mathbf{w}, b) by minimizing loss (i.e. difference between y and $f(\mathbf{x})$ on available examples *in training set*)

$$(\mathbf{w}, b) = \underset{\mathbf{w}, b}{\operatorname{argmin}} \sum_{i=1}^L \ell(f(\mathbf{x}_i), y_i)$$

Basic Concepts

- Testing (i.e. evaluating performance on “future” points)
 - Difference between true $y_?$ and the predicted $f(\mathbf{x}_?)$ on a set of testing examples (i.e. *testing set*)
 - Key: example $\mathbf{x}_?$ not in the training set
- Generalisation: learn function / hypothesis from past data in order to “explain”, “predict”, “model” or “control” new data examples

Basic Concepts

- Loss function

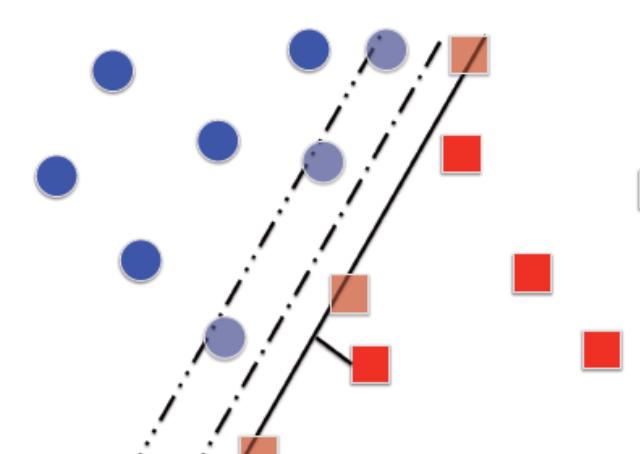
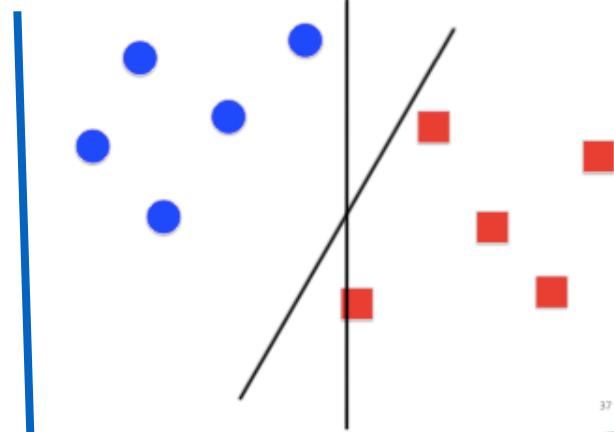
- e.g. hinge loss for binary classification task

$$\sum_{i=1}^L \ell(f(x_i), y_i) = \sum_{i=1}^L \max(0, 1 - y_i f(x_i)).$$

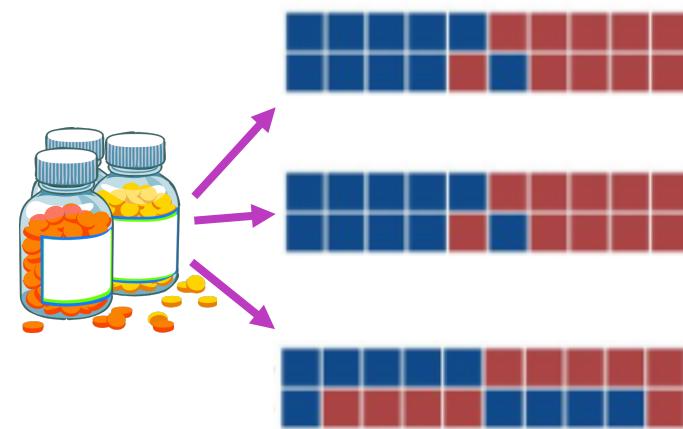
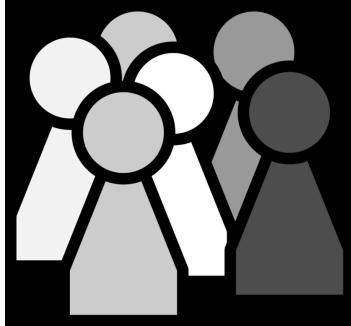
- Regularization

- E.g. additional information added on loss function to control f

Maximize Separation Margin => Minimize $\|w\|^2$

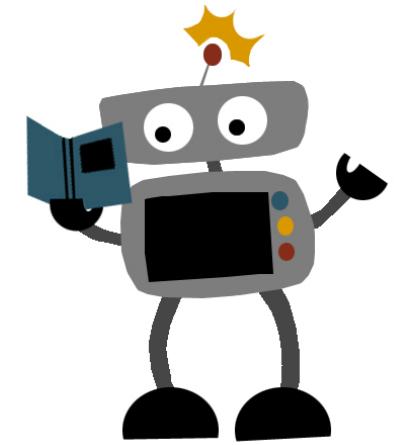


Basics of Machine Learning

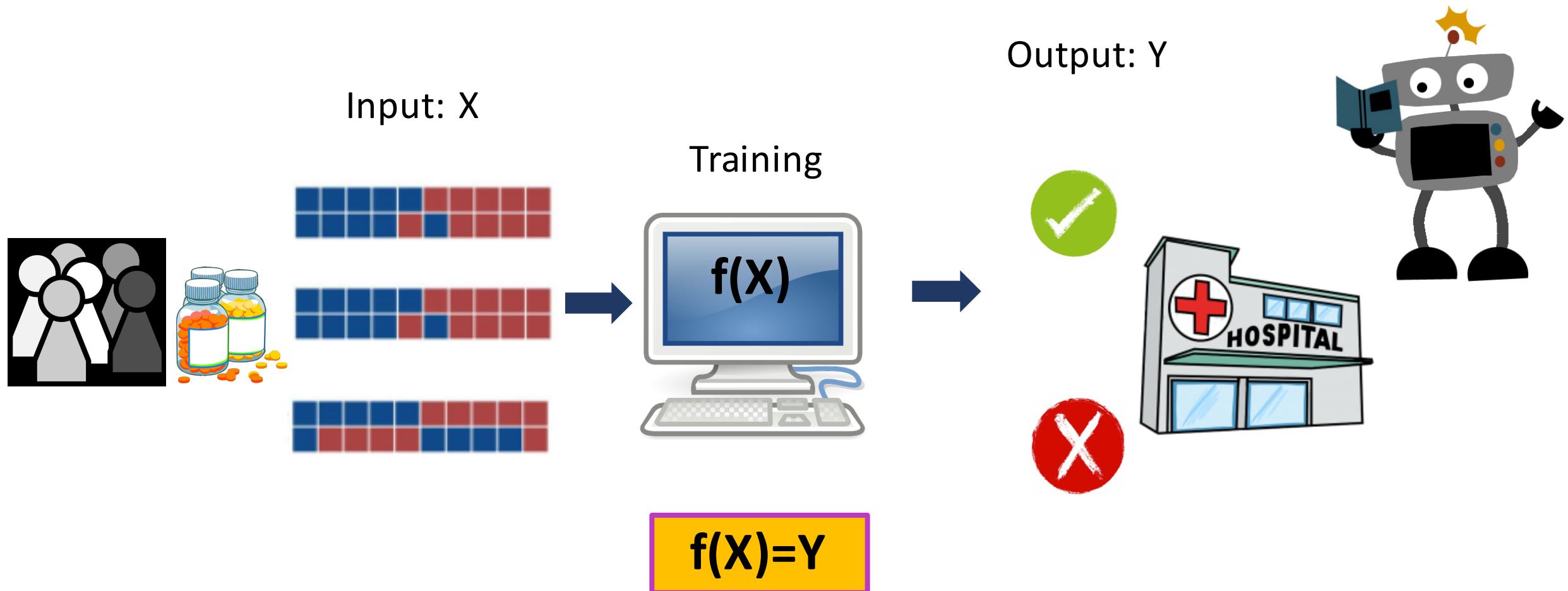


Input: X

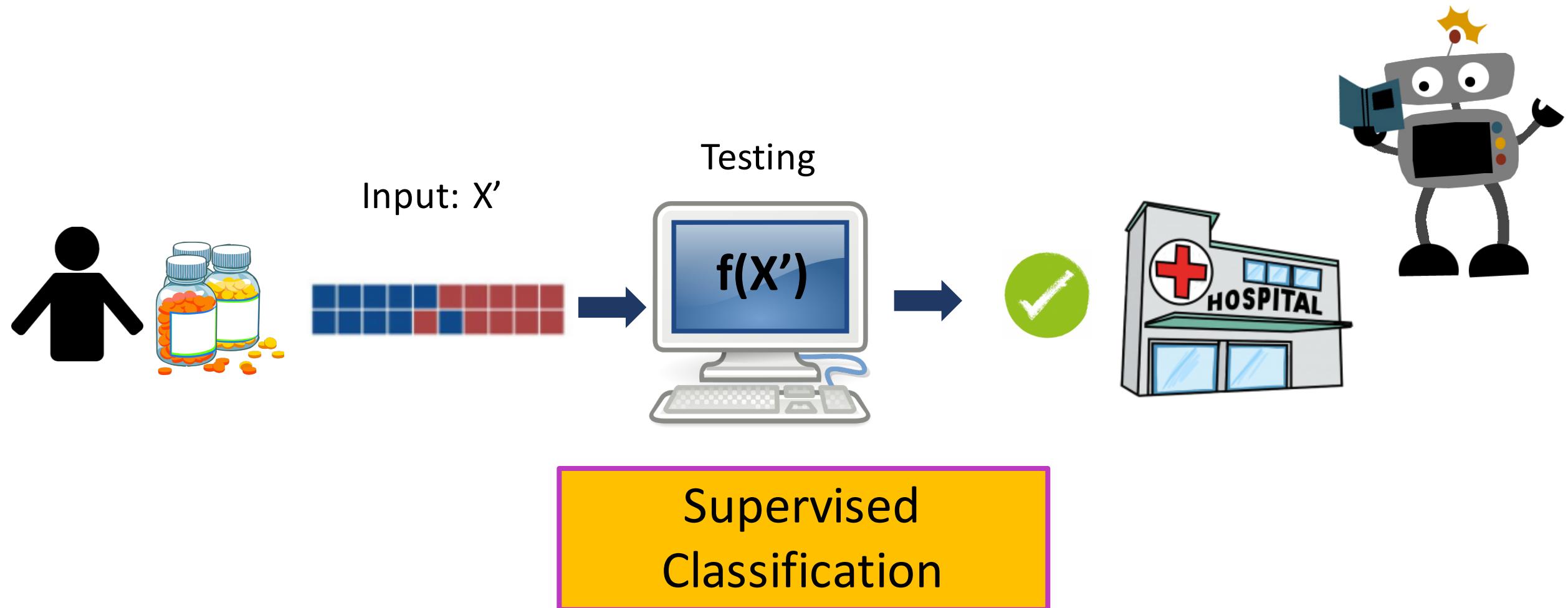
Output: Y



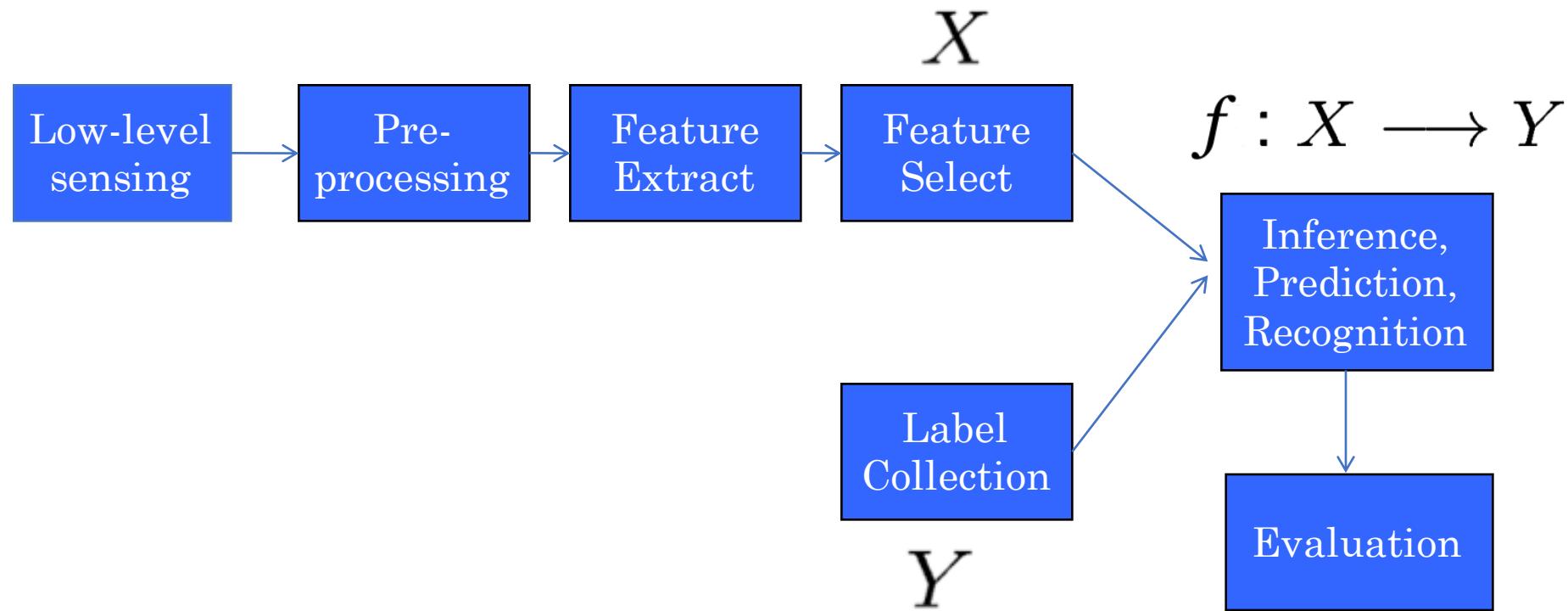
Basics of Machine Learning



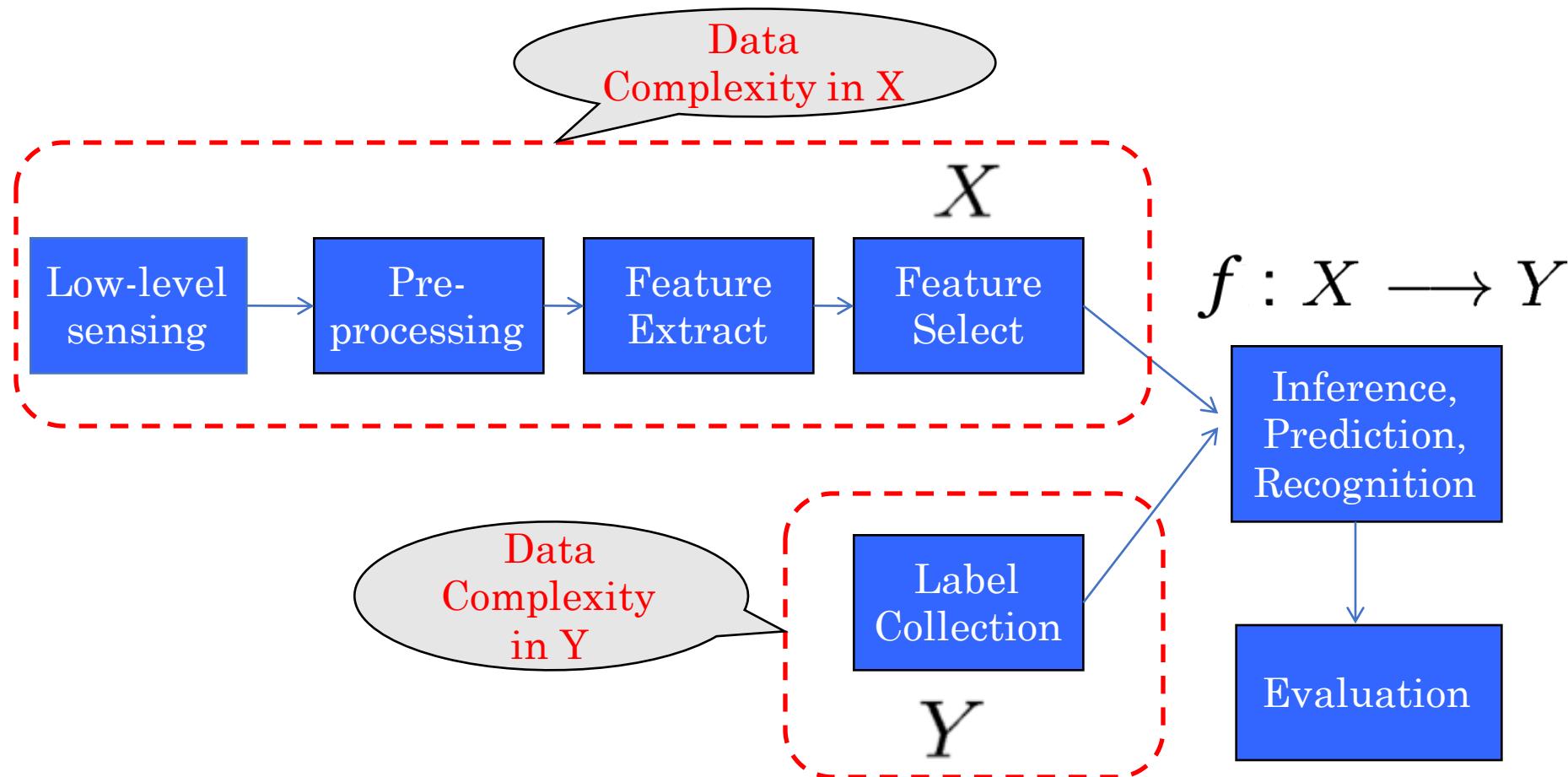
Basics of Machine Learning



TYPICAL MACHINE LEARNING SYSTEM

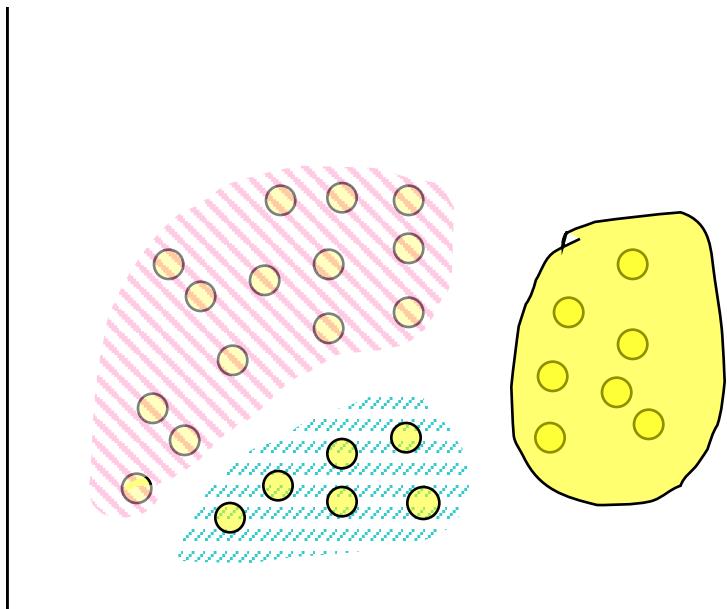


TYPICAL MACHINE LEARNING SYSTEM



UNSUPERVISED LEARNING : [COMPLEXITY OF Y]

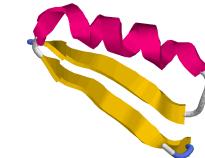
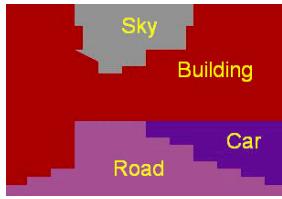
- No labels are provided (e.g. No Y provided)
- Find patterns from unlabeled data, e.g. clustering



e.g. clustering => to find
“natural” grouping of
instances given un-labeled
data

Structured Output Prediction: [COMPLEXITY in Y]

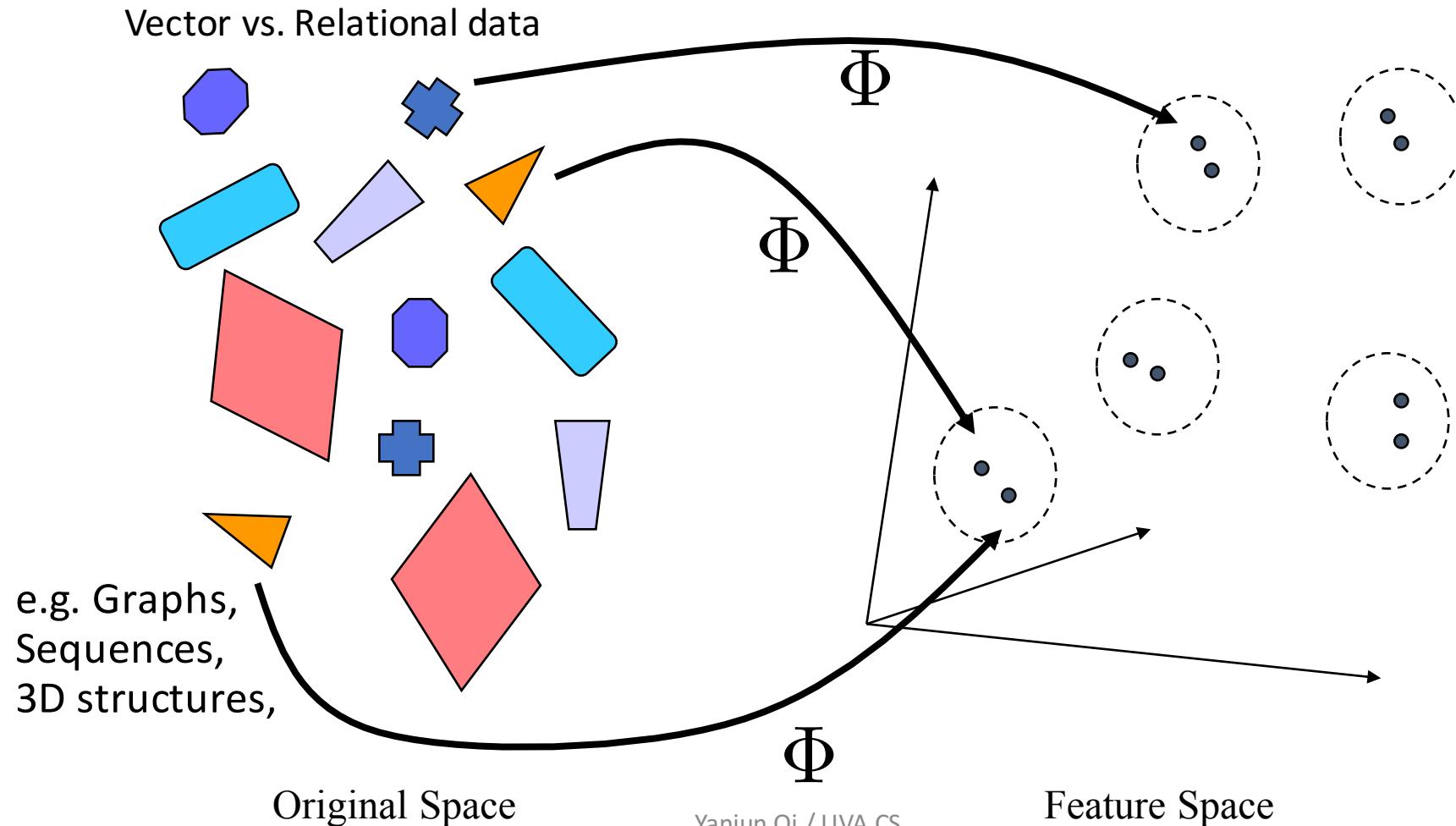
- Many prediction tasks involve **output labels having structured correlations or constraints among instances**

Structured Dependency between Examples' Y	Sequence	Tree	Grid
Input X	APAFSVSPASGACGPECA...	The dog chased the cat	
Output Y	 CCEEEEEECCCCCCCIIIICCC...	<pre>graph TD; S --> NP1[NP]; S --> VP; NP1 --> Det1[Det]; NP1 --> N1[N]; VP --> V[V]; VP --> NP2[NP]; NP2 --> Det2[Det]; NP2 --> N2[N];</pre>	

Many more possible structures between y_i , e.g. **spatial** , **temporal**, **relational** ...

Structured Input: Kernel Methods

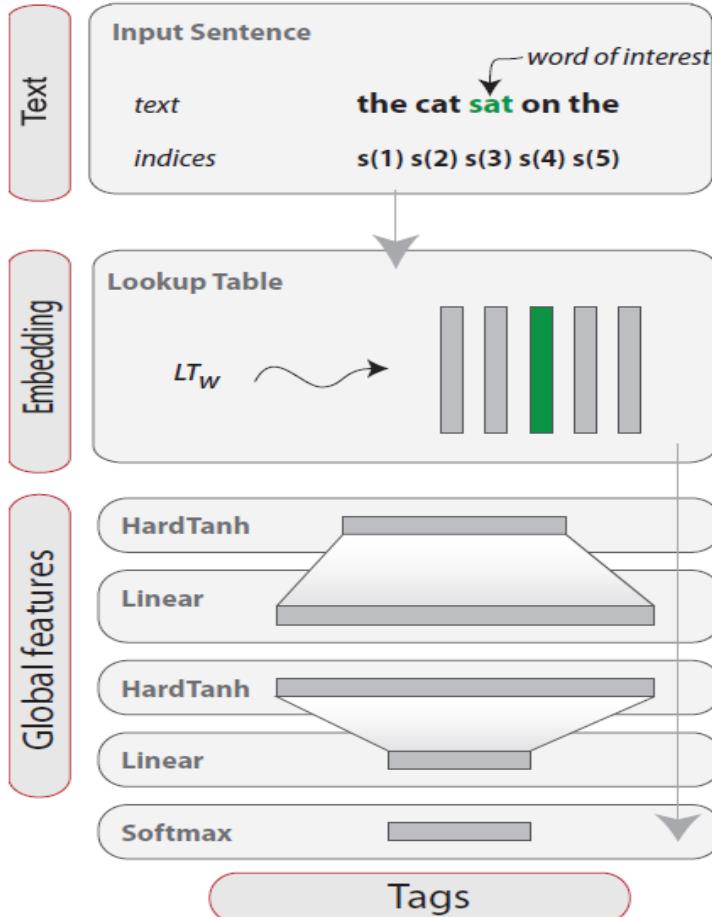
[COMPLEXITY OF X]



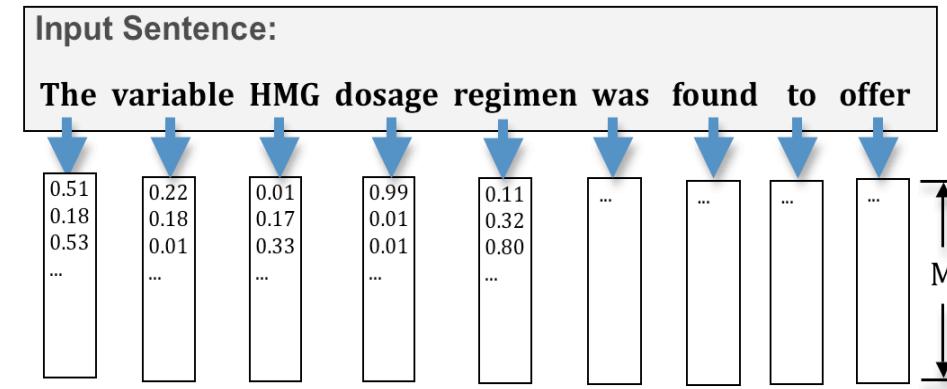
More Recent: Representation Learning

[COMPLEXITY OF X]

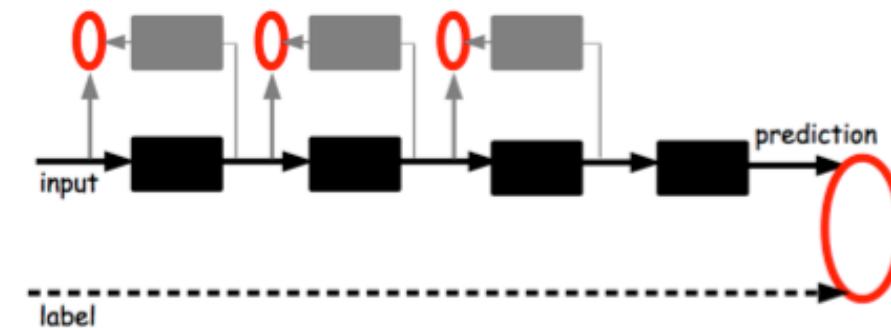
Deep Learning

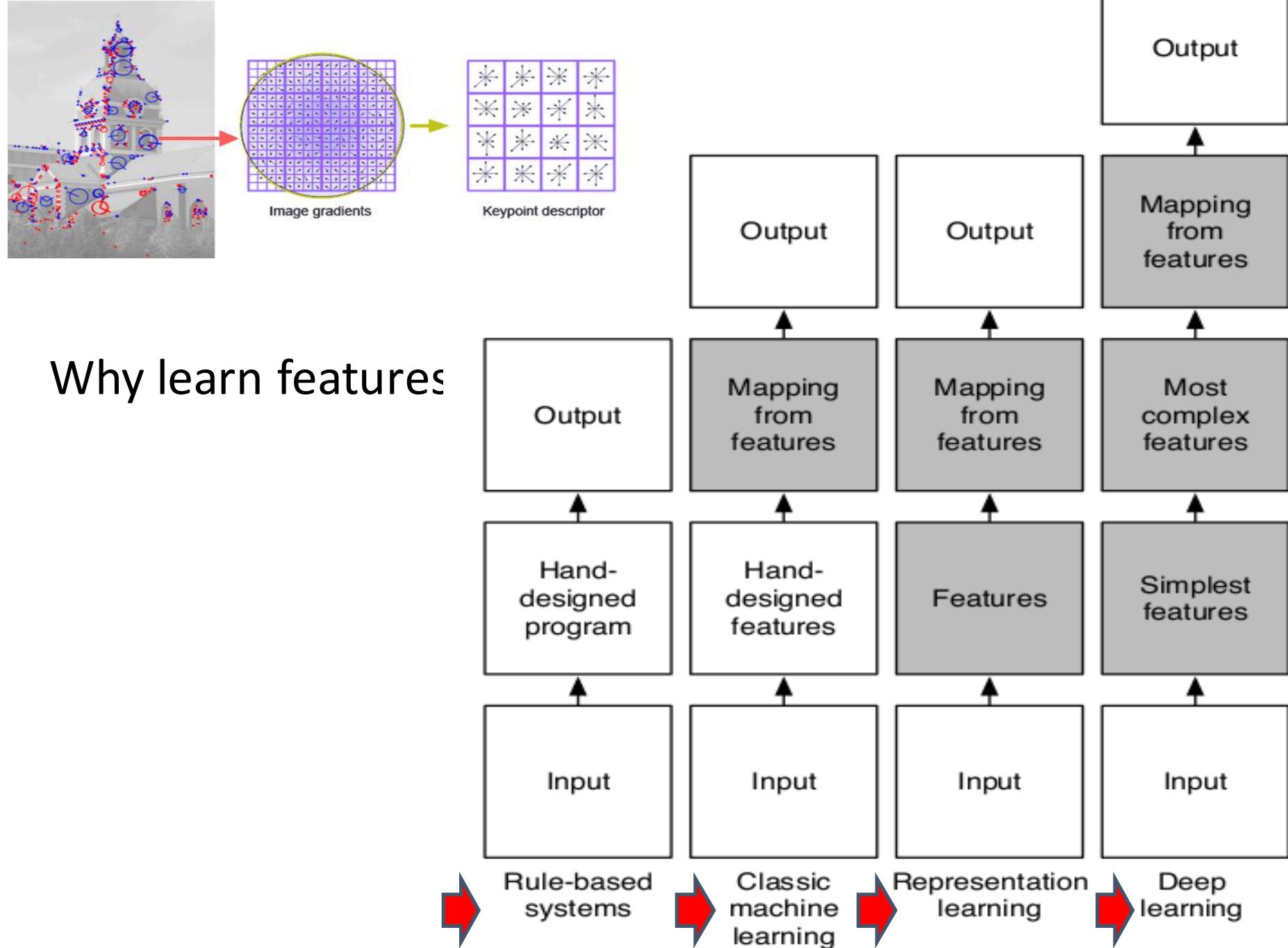


Supervised Embedding



Layer-wise Pretraining





Why learn features

When to use Machine Learning ?

- 1. Extract knowledge from data
 - Relationships and correlations can be hidden within large amounts of data
 - The amount of knowledge available about certain tasks is simply too large for explicit encoding (e.g. rules) by humans
- 2. Learn tasks that are difficult to formalise
 - Hard to define well, except by examples (e.g. face recognition)
- 3. Create software that improves over time
 - New knowledge is constantly being discovered.
 - Rule or human encoding-based system is difficult to continuously re-design “by hand”.

Recap

$$f : X \longrightarrow Y$$

- Goal of Machine Learning:
Generalisation
- Training
- Testing
- Loss

Today

- Machine Learning: a quick review
- Deep Learning: a quick review
- Background Biology: a quick review
- Deep Learning for analyzing **Sequential Data** about Regulation:
 - DeepChrome
 - AttentiveChrome
 - DeepMotif

<https://qdata.github.io/deep2Read/>

<https://www.deepchrome.org>

- Deep Learning
- 
- Why is this a breakthrough ?
 - Basics
 - History
 - A Few Recent trends

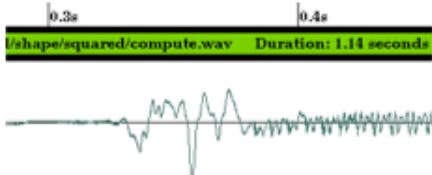
<https://qdata.github.io/deep2Read/>

Deep Learning is Changing the World

How may I help you, human?

Text analysis

Peter H. van Oppen, Chairman of the Board & Chief Executive Officer
Mr. van Oppen has served as chairman of the board and chief executive officer of ADIC since its acquisition by Interpoint in 1994 and a director of ADIC since 1986. Until its acquisition by Crane Co. in October 1996, Mr. van Oppen served as chairman of the board of directors, president and chief executive officer of Interpoint. Prior to 1985, Mr. van Oppen worked as a consulting manager at Price Waterhouse LLP and at Bain & Company in Boston and London. He has additional experience in medical electronics and venture capital. Mr. van Oppen also serves as a director of Seattle FilmWorks Inc. and Spacelabs Medical, Inc.. He holds a B.A. from Whitman College and an M.B.A. from Harvard Business School, where he was a Baker Scholar.



Speech Recognition



Control learning



Object recognition

Many more !

10 Breakthrough Technologies 2013

Think of the most frustrating, intractable, or simply annoying problems you can imagine. Now think about what technology is doing to fix them. That's what we did in coming up with our annual list of 10 Breakthrough Technologies. We're looking for technologies that we believe will expand the scope of human possibilities.

Deep Learning

10 Breakthrough Technologies 2017

These technologies all have staying power. They will affect the economy and our politics, improve medicine, or influence our culture. Some are unfolding now; others will take a decade or more to develop. But you should know about all of them right now.

Deep Reinforcement Learning



Generative
Adversarial
Network (GAN)

Why breakthrough ?

Breakthrough from 2012 Large-Scale Visual Recognition Challenge (ImageNet)

10% improve
with deepCNN



72%, 2010

74%, 2011

85%, 2012

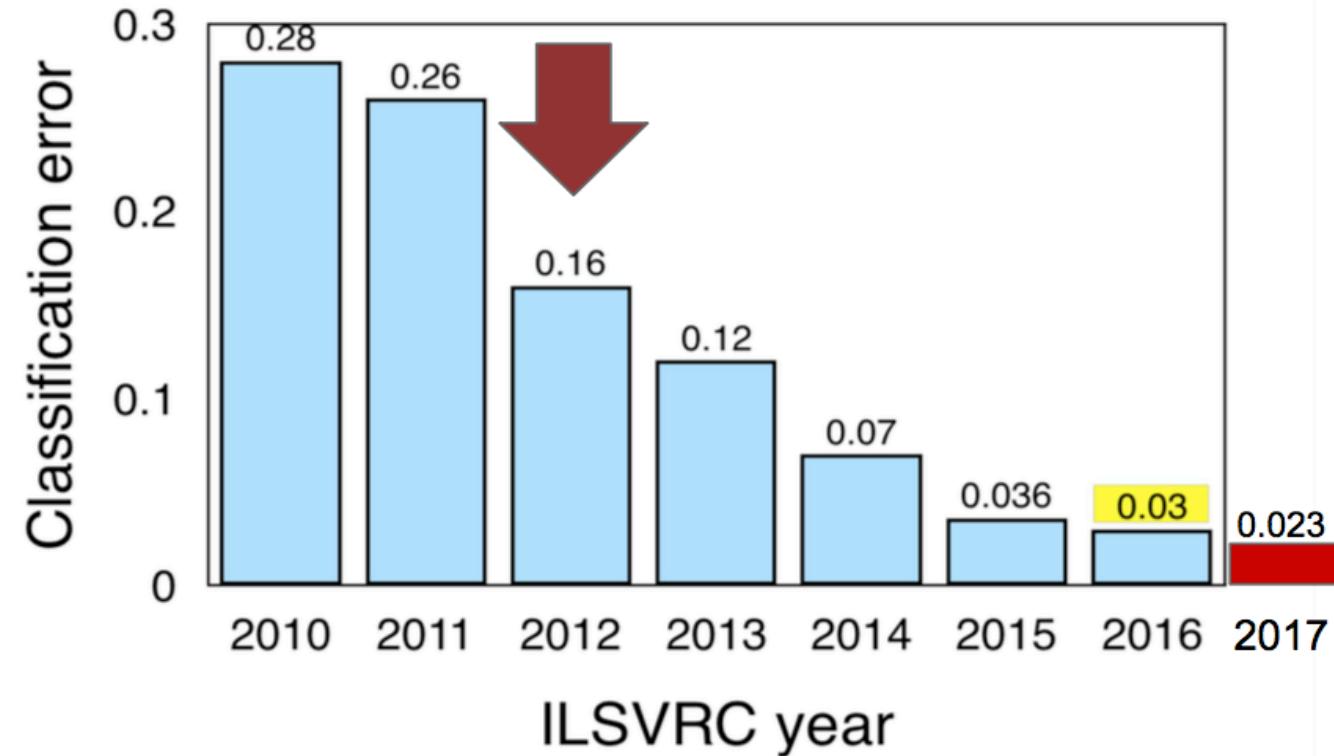
In one “very large-scale” benchmark competition
(1.2 million images [X] vs. 1000 different word labels [Y])

ImageNet Challenge

Arch



- 2010-11: hand-crafted computer vision pipelines
- 2012-2016: ConvNets
 - 2012: AlexNet
 - major deep learning success
 - 2013: ZFNet
 - improvements over AlexNet
 - 2014
 - VGGNet: deeper, simpler
 - InceptionNet: deeper, faster
 - 2015
 - ResNet: even deeper
 - 2016
 - ensembled networks
 - 2017
 - Squeeze and Excitation Network



DNNs help us build more intelligent computers

- Perceive the world,
 - e.g., objective recognition, speech recognition, ...
- Understand the world,
 - e.g., machine translation, text semantic understanding
- Interact with the world,
 - e.g., AlphaGo, AlphaZero, self-driving cars, ...
- Being able to think / reason,
 - e.g., learn to code programs, learn to search deepNN, ...
- Being able to imagine / to make analogy,
 - e.g., learn to draw with styles,

Deep Learning Way: Learning Representation from data



Feature Engineering

- ✓ Most critical for accuracy
- ✓ Account for most of the computation
- ✓ Most time-consuming in development cycle
- ✓ Often hand-craft and task dependent in practice

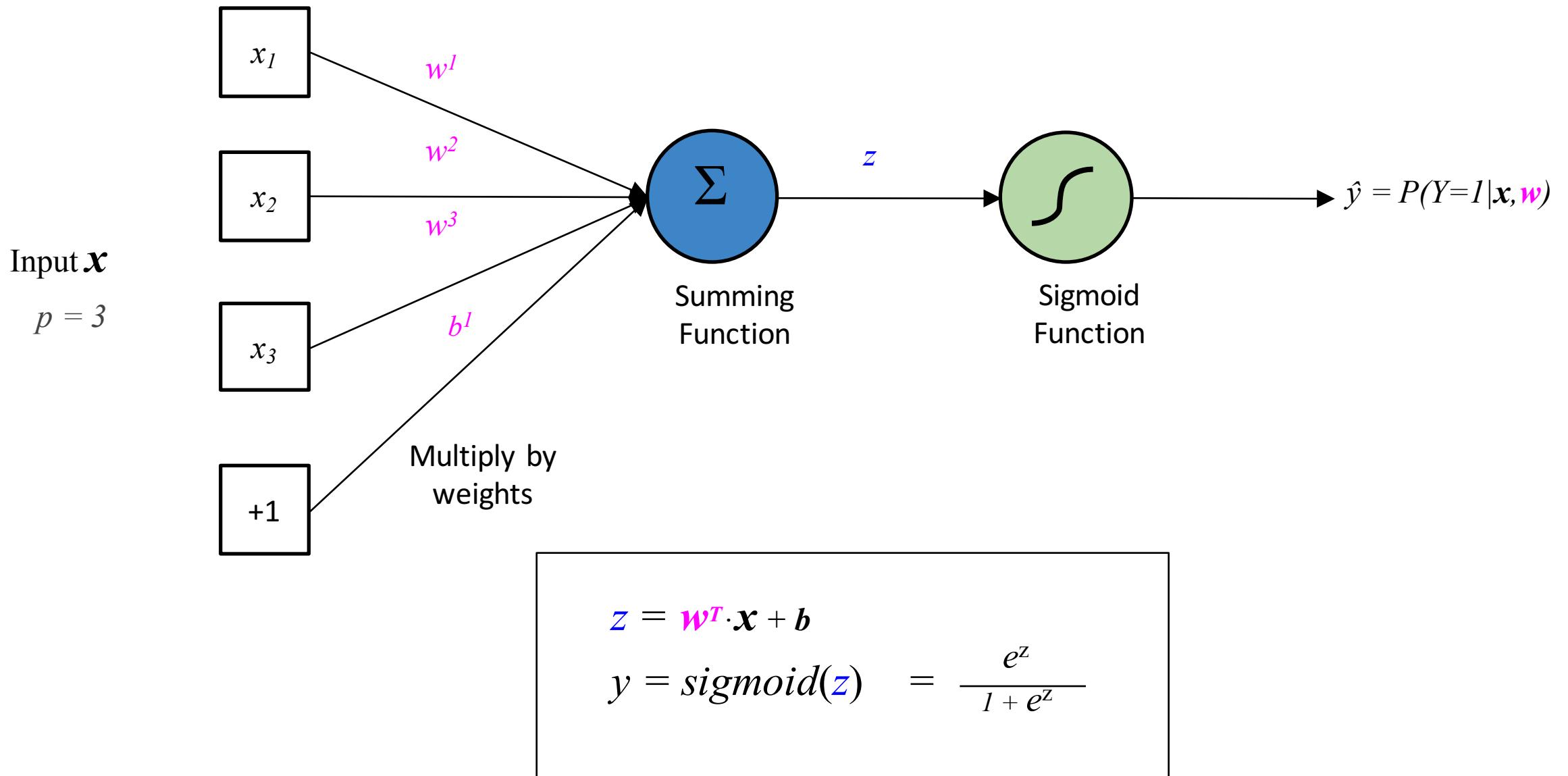
Feature Learning

- ✓ Easily adaptable to new similar tasks
- ✓ Learn layerwise representation from data

Basics

- Basic Neural Network (NN)
 - single neuron, e.g. logistic regression unit
 - multilayer perceptron (MLP)
 - various loss function
 - E.g., when for multi-class classification, softmax layer
 - training NN with backprop algorithm

One “Neuron”: Expanded Logistic Regression



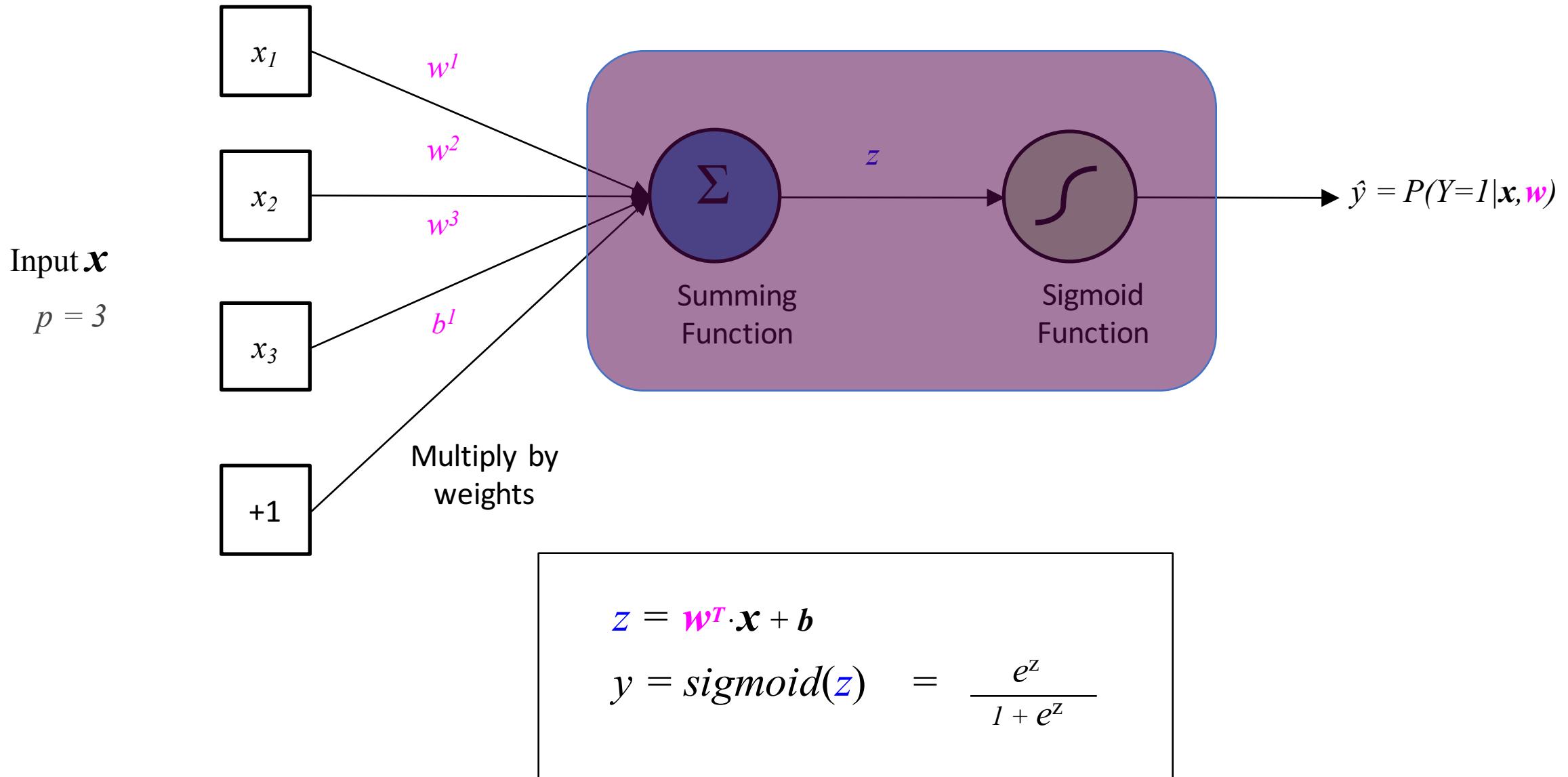
E.g., Many Possible Nonlinearity Functions

(aka transfer or activation functions)

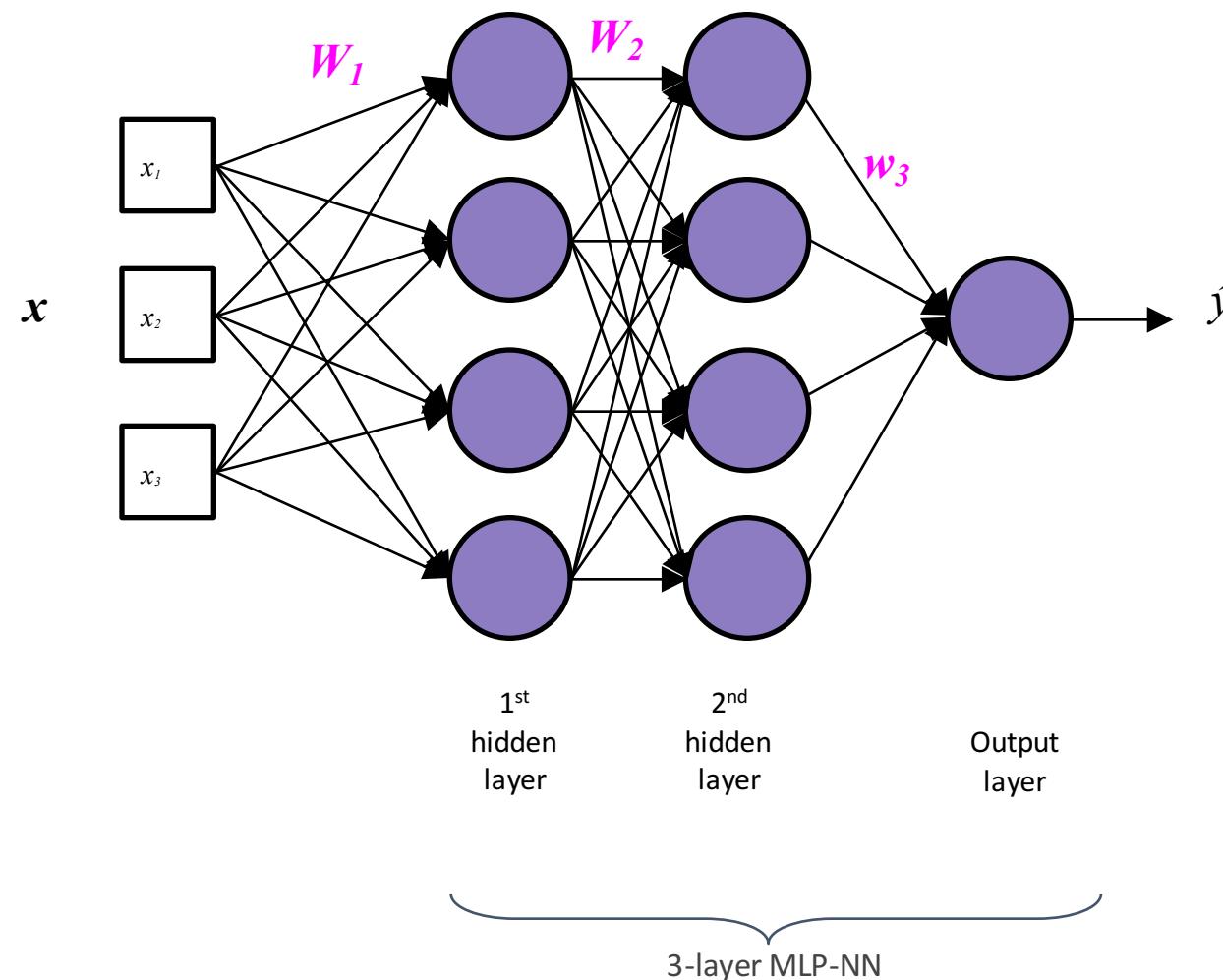
Name	Plot	Equation	Derivative (w.r.t x)
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
Rectifier (ReLU) ^[9]		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$

usually works best in practice

One “Neuron”: Expanded Logistic Regression => “Neuron View”

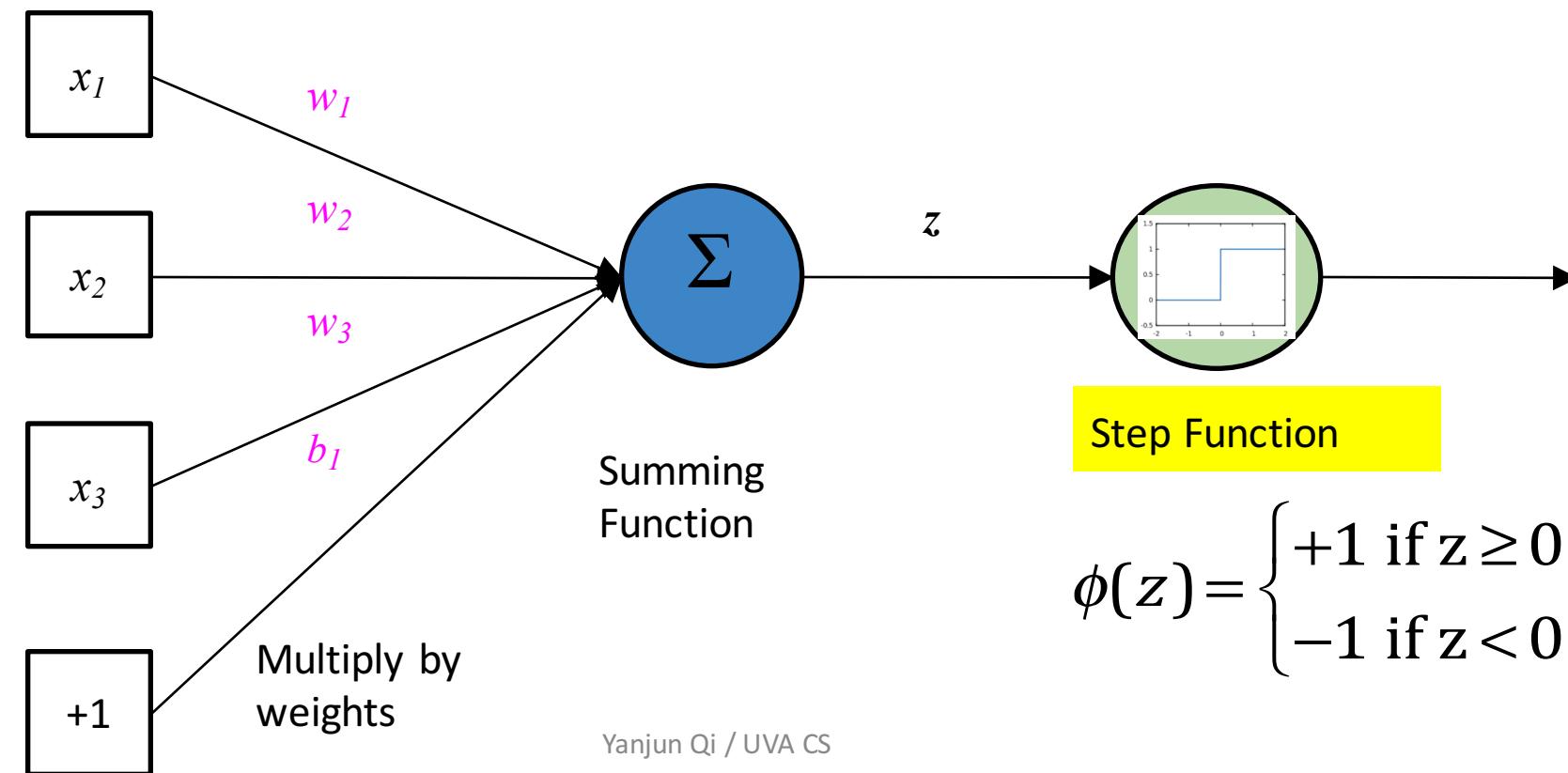


Multi-Layer Perceptron (MLP)- (Feed-Forward NN)

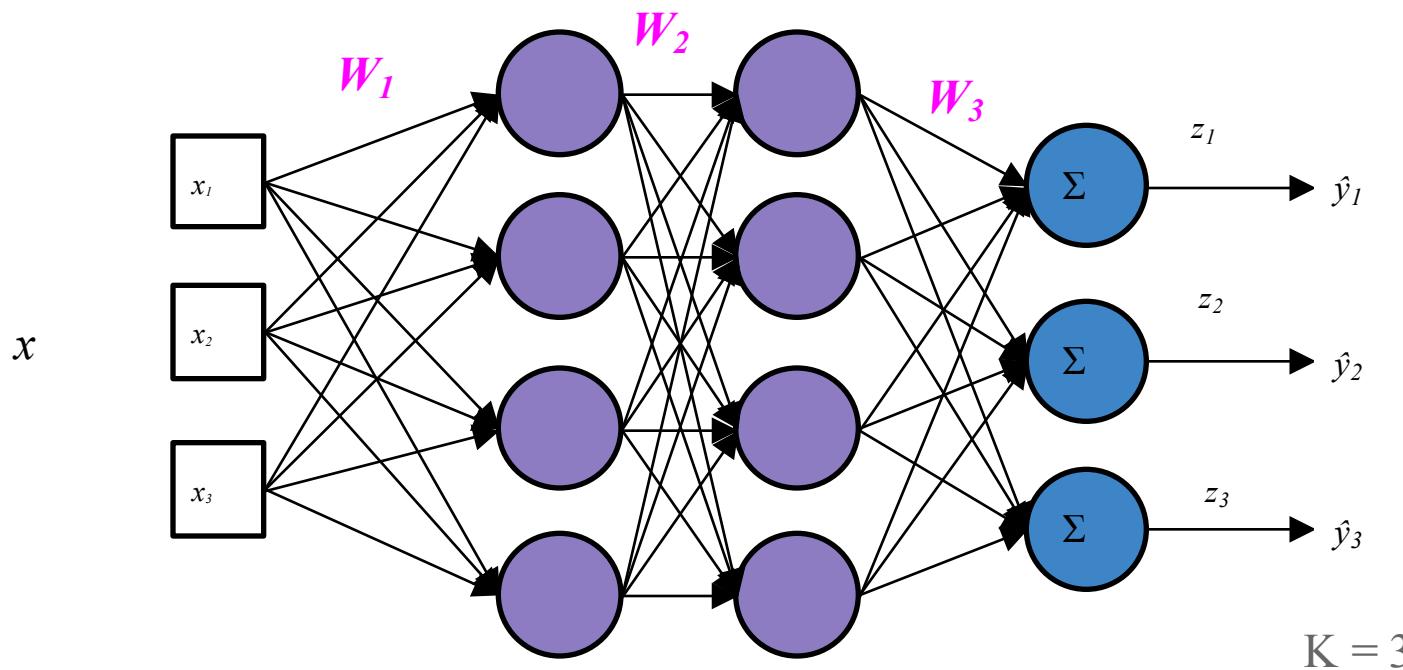


History → Perceptron: 1-Neuron Unit with Step

- First proposed by Rosenblatt (1958)
- A simple neuron that is used to classify its input into one of two categories.
- A perceptron uses a **step function**

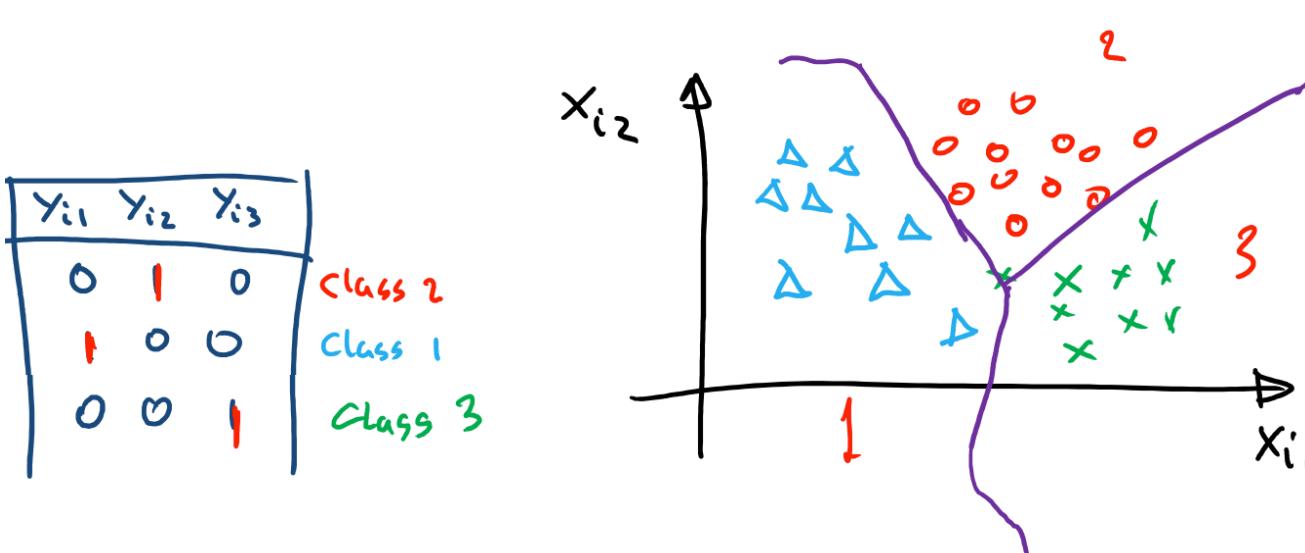


E.g., Cross-Entropy Loss for Multi-Class Classification



$$\hat{y}_i = \frac{e^{z_i}}{\sum_j e^{z_j}} = P(\hat{y}_i = 1 | \mathbf{x})$$

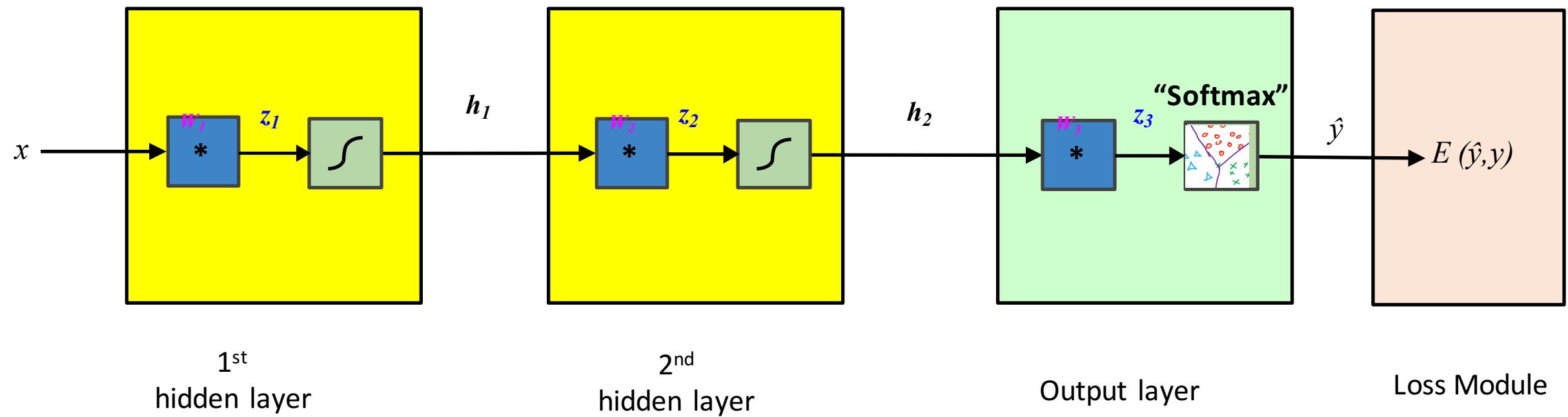
“Softmax” function. Normalizing function which converts each class output to a probability.



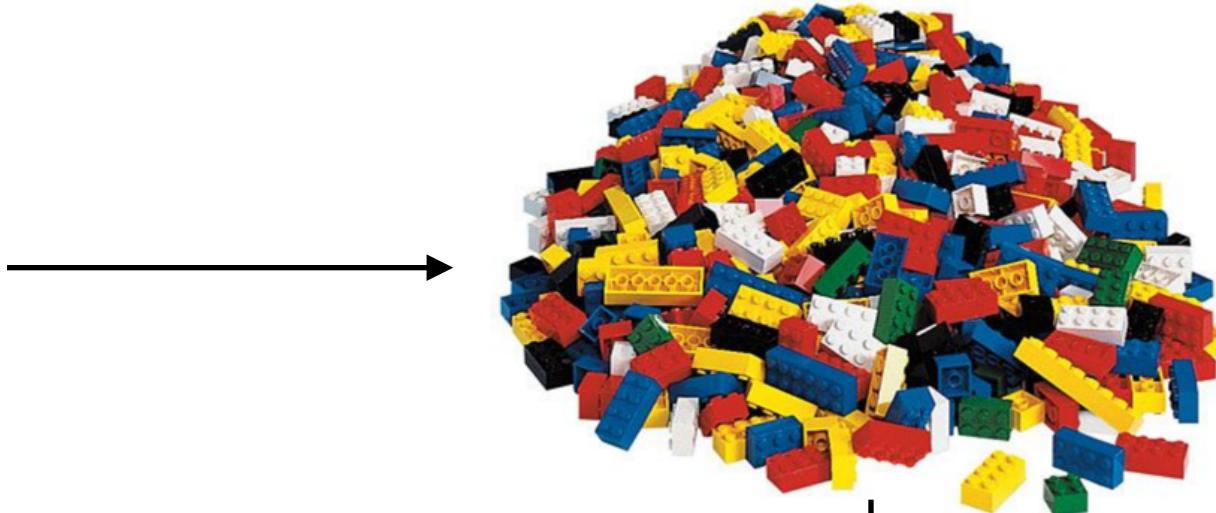
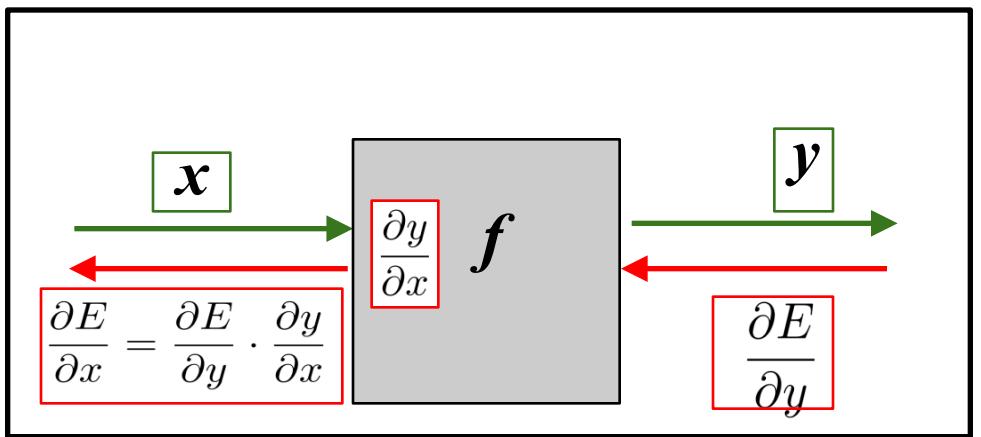
$$E_W(\hat{y}, y) = \text{loss} = \sum_{j=1 \dots K} y_j \ln \hat{y}_j$$

Cross-entropy loss

“Block View”



Building Deep Neural Nets



Training Neural Networks

How do we learn the optimal weights $\textcolor{magenta}{W}_L$ for our task??

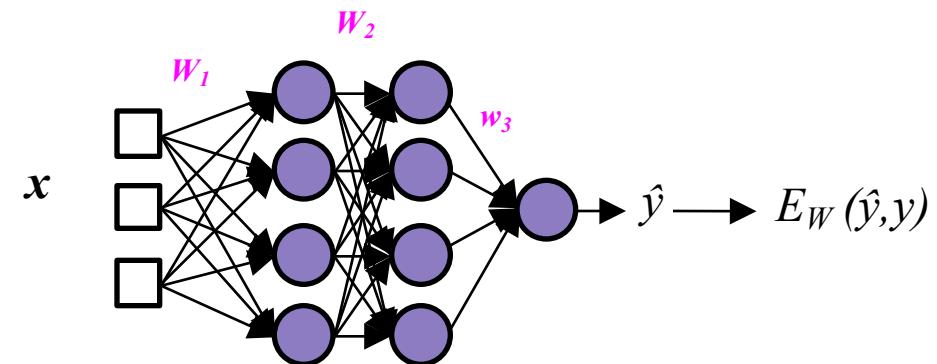
- **Stochastic Gradient descent:**

$$W_L^t = W_L^{t-1} - \eta \frac{\partial E}{\partial W_L}$$

But how do we get gradients of lower layers?

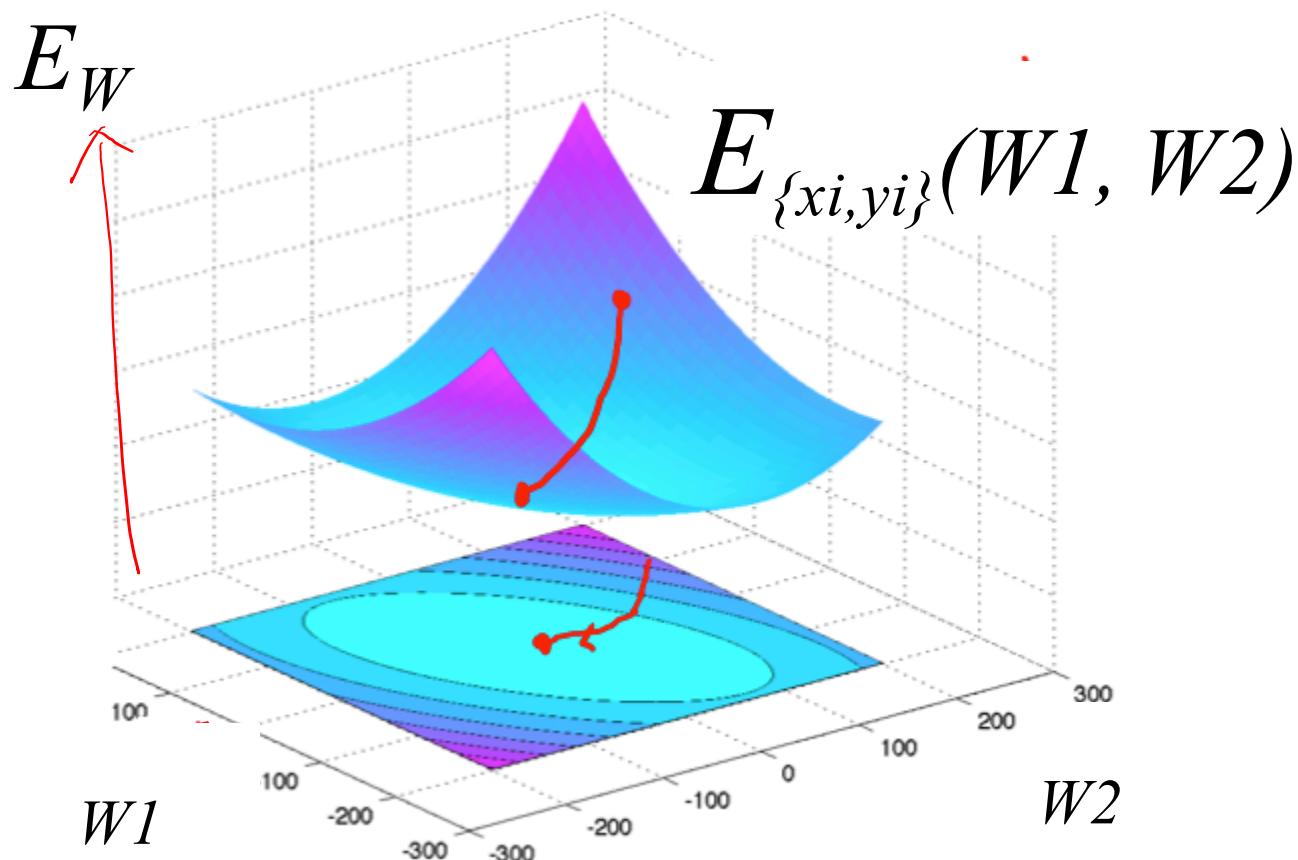
- **Backpropagation!**

- Repeated application of chain rule of calculus
- Locally minimize the objective
- Requires all “blocks” of the network to be differentiable



– Main Idea: error in hidden layers

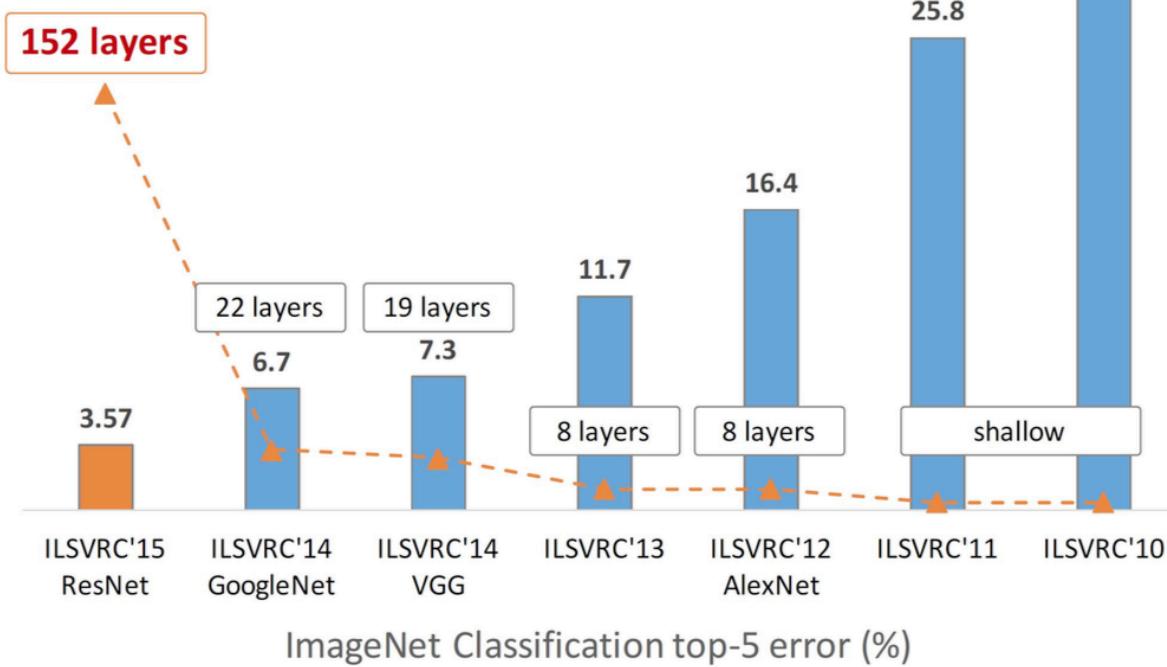
Illustrating Objective Loss Function (extremely simplified) and Gradient Descent (2D case)



The gradient points in the direction (in the variable space) of the greatest rate of increase of the function and its magnitude is the slope of the surface graph in that direction

Revolution of Depth

Arch



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Important **Block**: Convolutional Neural Networks (CNN)

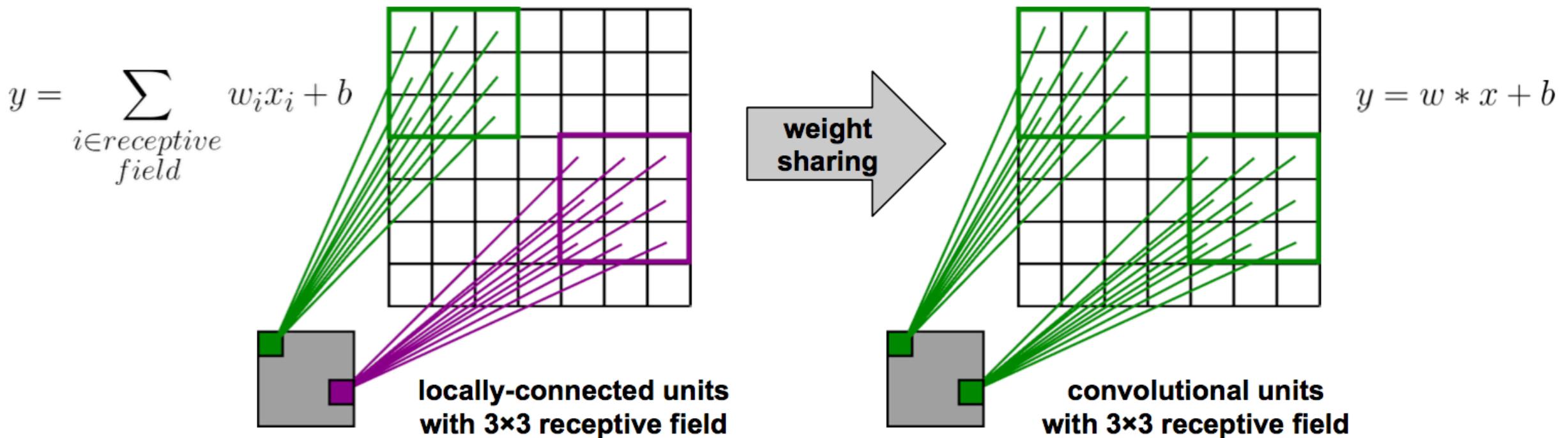
- Prof. Yann LeCun invented **CNN** in 1998
- First NN successfully trained with many layers



The bird occupies a local area and looks the same in different parts of an image.
We should construct neural nets which exploit these properties!

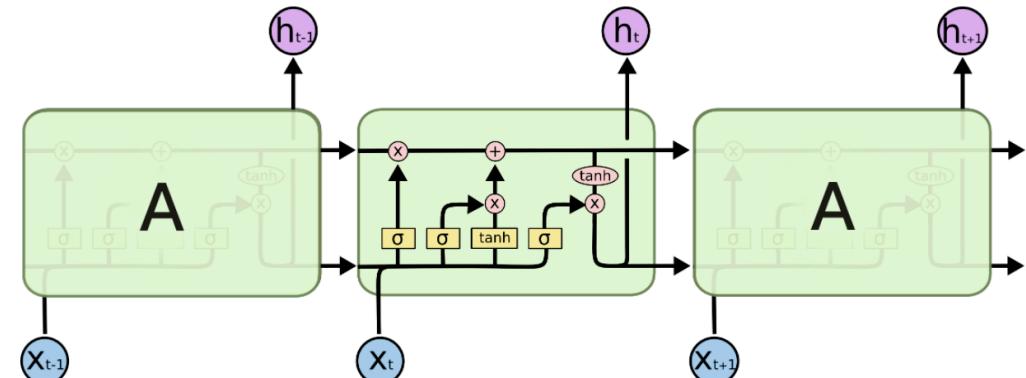
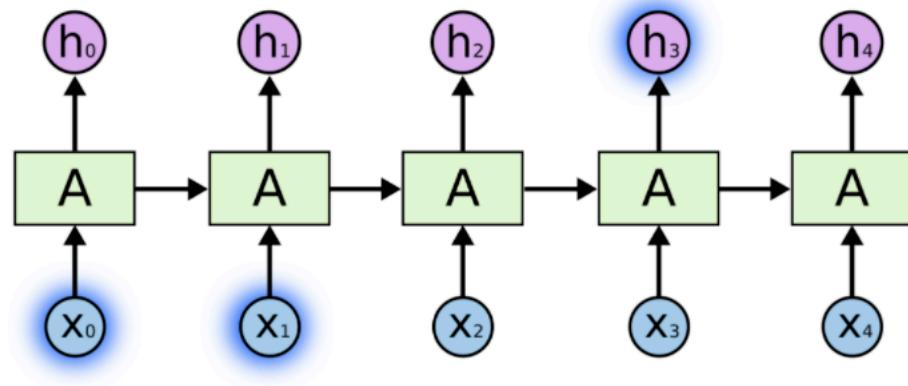
CNN models Locality and Translation Invariance

Make **fully-connected layer** **locally-connected** and **sharing weight**



Important Block: Recurrent Neural Networks (RNN)

- Prof. Schmidhuber invented "Long short-term memory" – Recurrent NN (LSTM-RNN) model in 1997

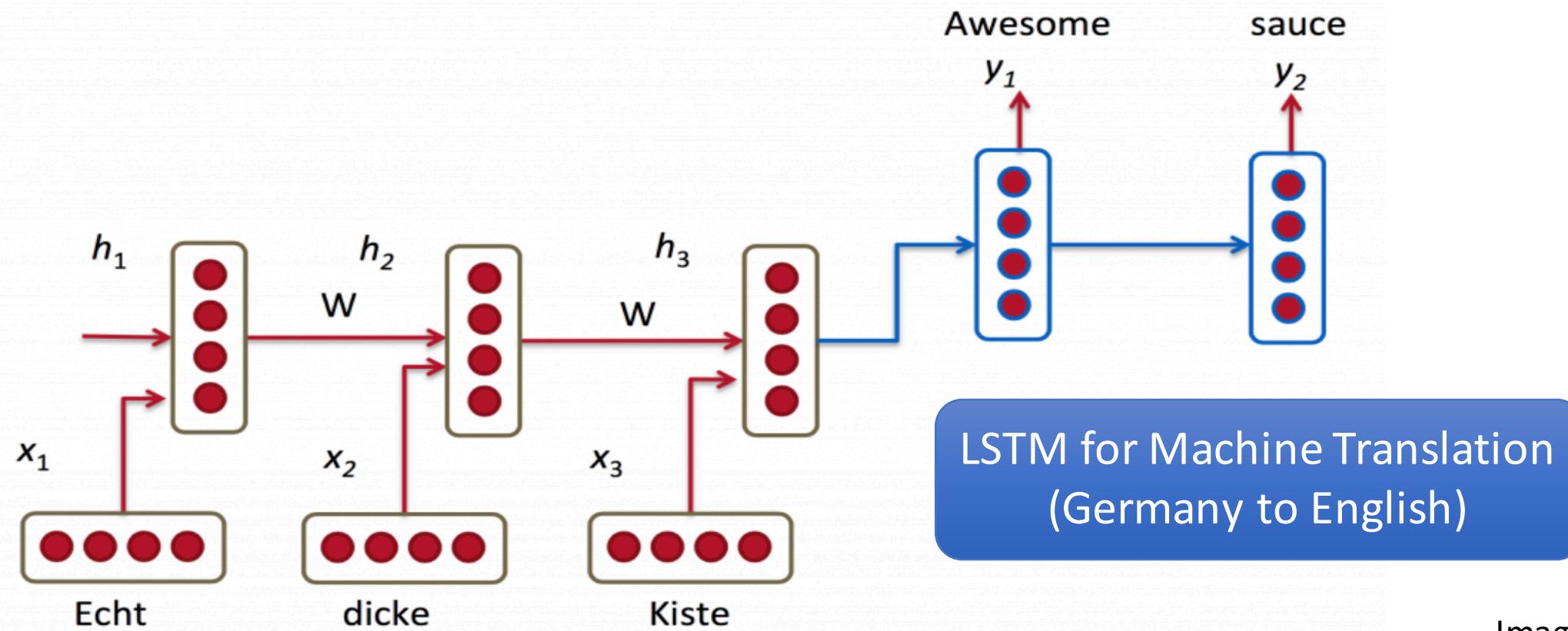


The repeating module in an LSTM contains four interacting layers.

Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". *Neural Computation*. 9 (8): 1735–1780.

RNN models dynamic temporal dependency

- Make **fully-connected** layer model **each unit recurrently**
- Units form a **directed chain graph** along a sequence
- Each unit uses **recent history** and current input in modeling





- Deep Learning
 - Why is this a breakthrough ?
 - Basics
 - History
 - A Few Recent trends

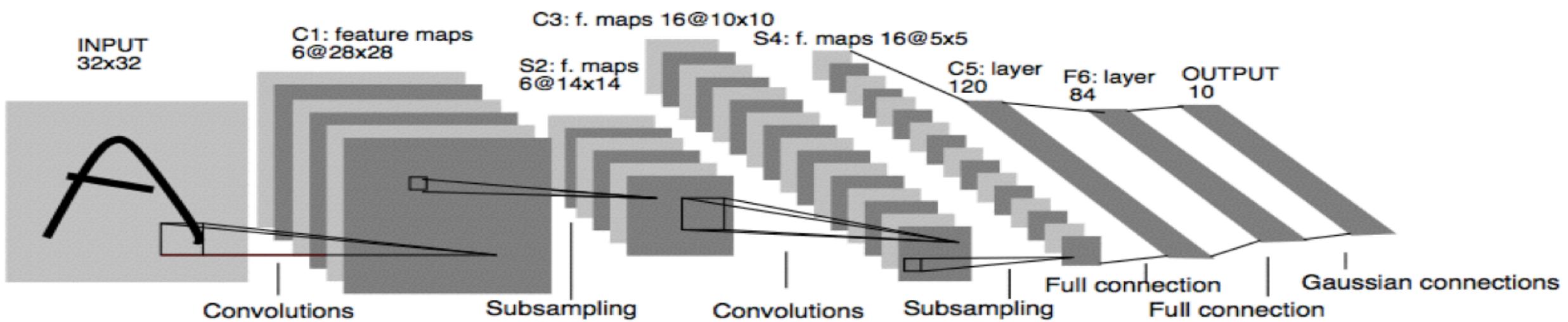
<https://qdata.github.io/deep2Read/>

Many classification models invented since late 80's

- Neural networks
- Boosting
- Support Vector Machine
- Maximum Entropy
- Random Forest
-

Deep Learning (CNN) in the 90's

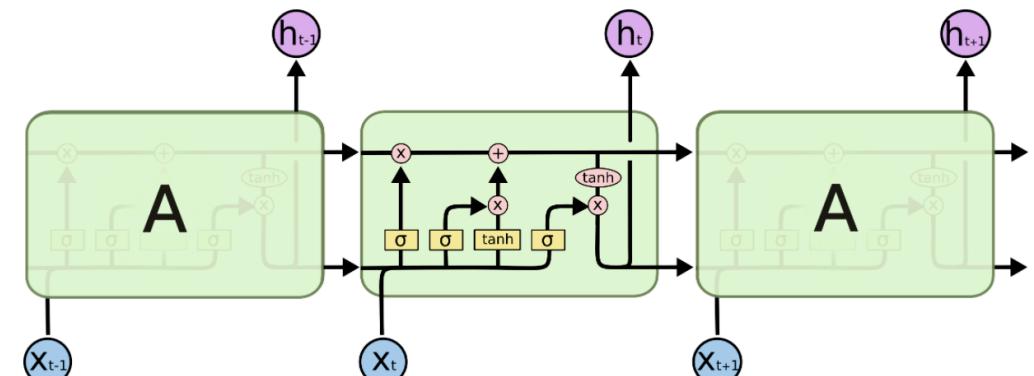
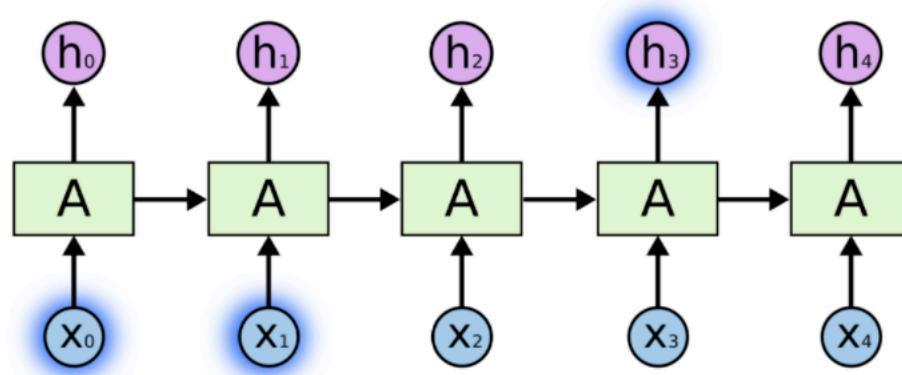
- Prof. Yann LeCun invented **Convolutional Neural Networks (CNN)** in 1998
- First NN successfully trained with many layers



Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

Deep Learning (RNN) in the 90's

- Prof. Schmidhuber invented "Long short-term memory" – Recurrent NN (LSTM-RNN) model in 1997



The repeating module in an LSTM contains four interacting layers.

Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780.

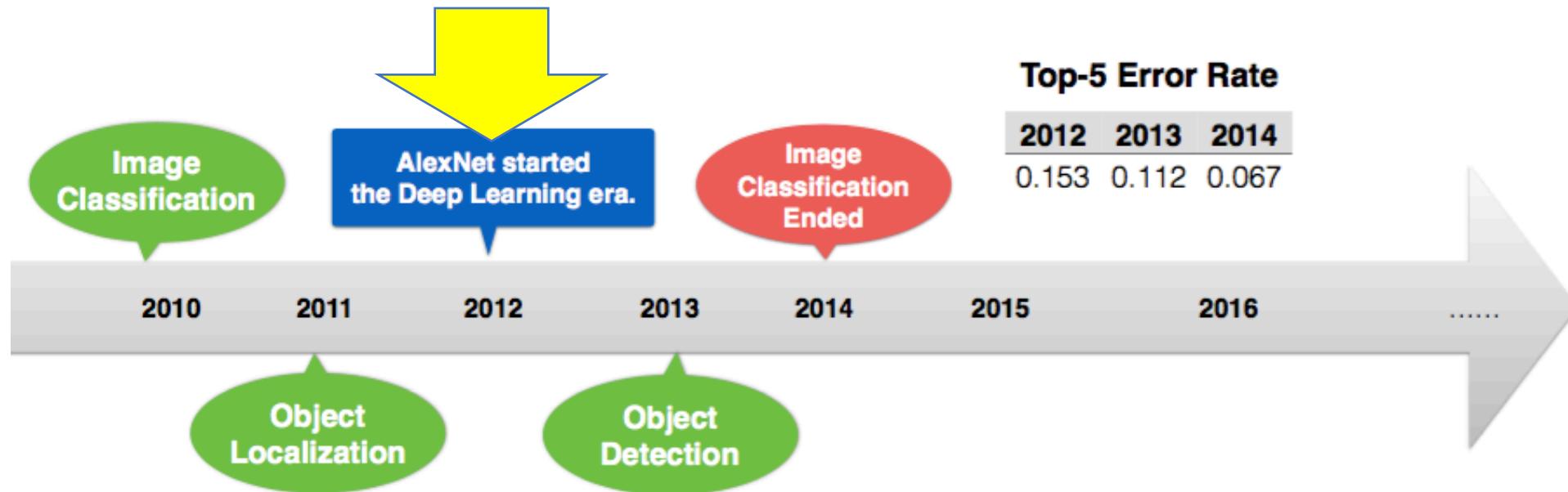
Between ~2000 to ~2011 Machine Learning Field Interest

- Learning with Structures ! + Convex Formulation!
 - Kernel learning
 - Manifold Learning
 - Sparse Learning
 - Structured input-output learning ...
 - Graphical model
 - Transfer Learning
 - Semi-supervised
 - Matrix factorization
 -

“Winter of Neural Networks” Since 90’s to ~2011

- Non-convex
- Need a lot of tricks to play with
 - How many layers ?
 - How many hidden units per layer ?
 - What topology among layers ?
- Hard to perform theoretical analysis

Breakthrough in 2012 Large-Scale Visual Recognition Challenge (ImageNet) : Milestones in Recent Vision/AI Fields

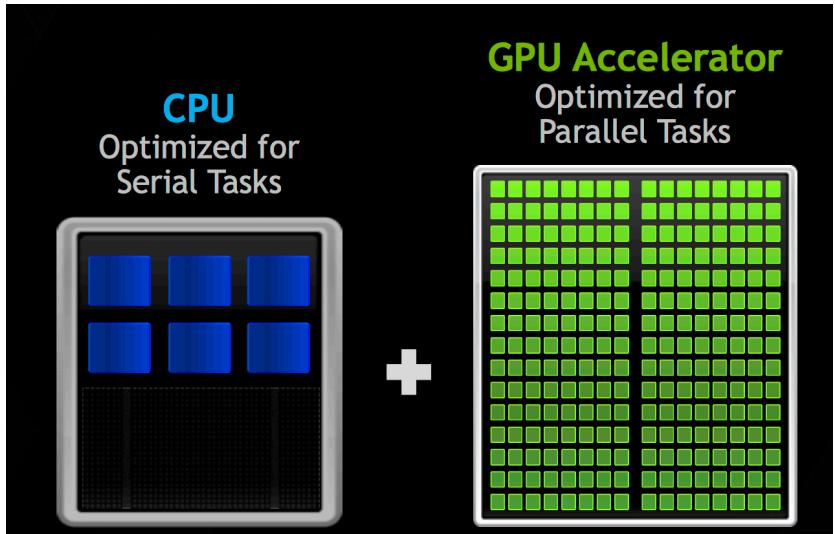


- 2013, Google Acquired Deep Neural Networks Company headed by Utoronto “Deep Learning” Professor Hinton
- 2013, Facebook Built New Artificial Intelligence Lab headed by NYU “Deep Learning” Professor LeCun
- 2016, Google's DeepMind defeats legendary Go player Lee Se-dol in historic victory / 2017 Alpha Zero

Reason: Plenty of (Labeled) Data

- **Text:** trillions of words of English + other languages
- **Visual:** billions of images and videos
- **Audio:** thousands of hours of speech per day
- **User activity:** queries, user page clicks, map requests, etc,
- **Knowledge graph:** billions of labeled relational triplets
-

Reason: Advanced Computer Architecture that fits DNNs



http://www.nvidia.com/content/events/geoInt2015/LBrown_DL.pdf

	Neural Networks	GPUs
Inherently Parallel	✓	✓
Matrix Operations	✓	✓
FLOPS	✓	✓

GPUs deliver --

- *same or better prediction accuracy*
- *faster results*
- *smaller footprint*
- *lower power*

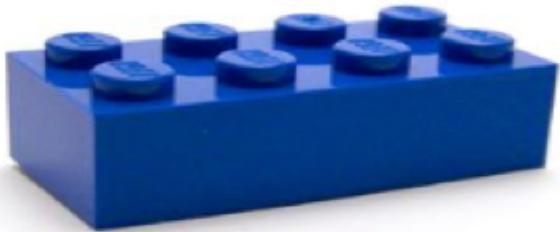
Some Recent Trends

<https://qdata.github.io/deep2Read/>

- 1. Autoencoder / layer-wise training
- 2. CNN / Residual / Dynamic parameter
- 3. RNN / Attention / Seq2Seq, ...
- 4. Neural Architecture with explicit Memory
- 5. NTM 4program induction / sequential decisions
- 6. Learning to optimize / Learning DNN architectures
- 7. Learning to learn / meta-learning/ few-shots
- 8. DNN on graphs / trees / sets
- 9. Deep Generative models, e.g., autoregressive
- 10. Generative Adversarial Networks (GAN)
- 11. Deep reinforcement learning
- 12. Validate / Evade / Test / Understand / Verify DNNs

Recap

<https://qdata.github.io/deep2Read/>



Inputs and Outputs



Losses



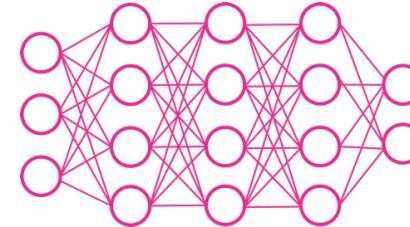
Architectures:



Learned Models

BREAK -> 5mins

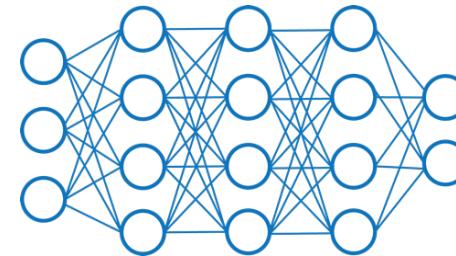
State-of-the-art: Deep Neural Networks (DNNs)



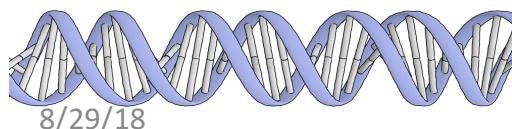
“Dog”

Can get overly sentimental at times, but
Gus Van Sant's sensitive direction... and
his excellent use of the city make it a
hugely entertaining and effective film.

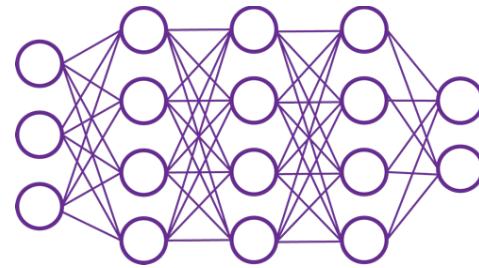
[Full Review...](#) | May 25, 2006



ATGCGATCAAGTCTG



8/29/18

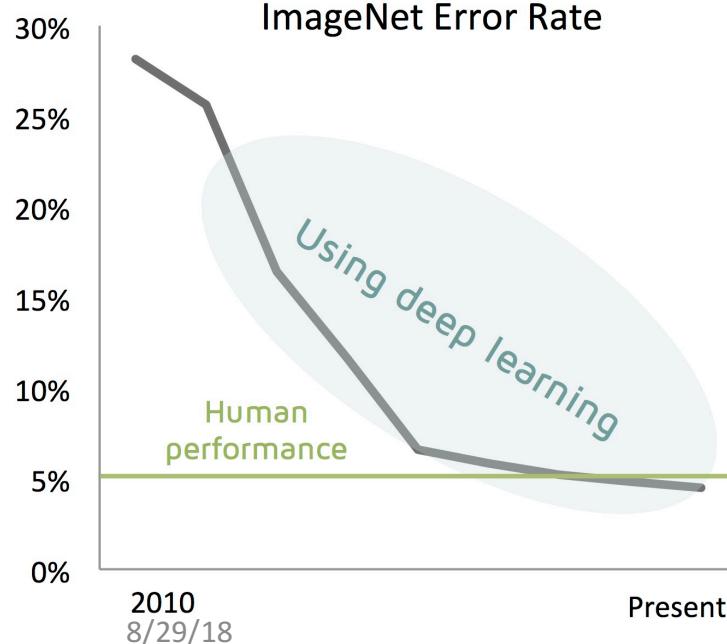
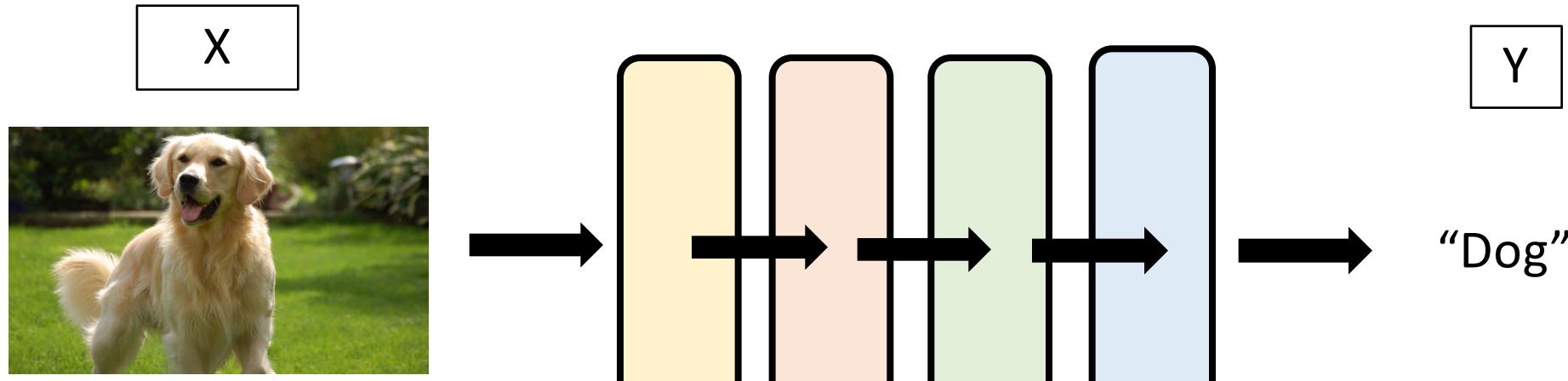


Yanjun Qi / UVA CS



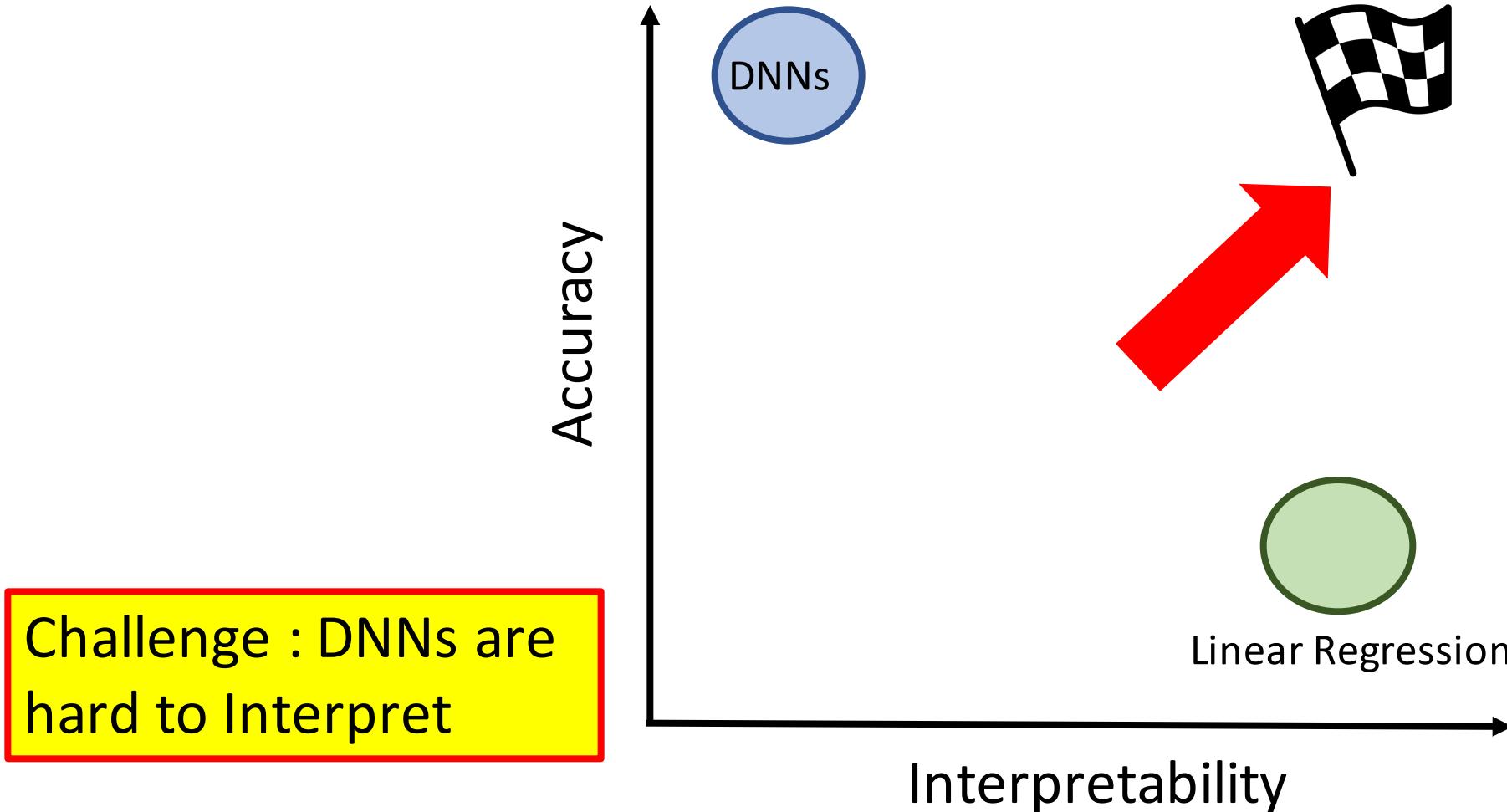
“Protein-binding Site”

Challenge : DNNs are hard to Interpret



$$Y = f_4(f_3(f_2(f_1(X))))$$

Our Goal: Interpretable DNNs

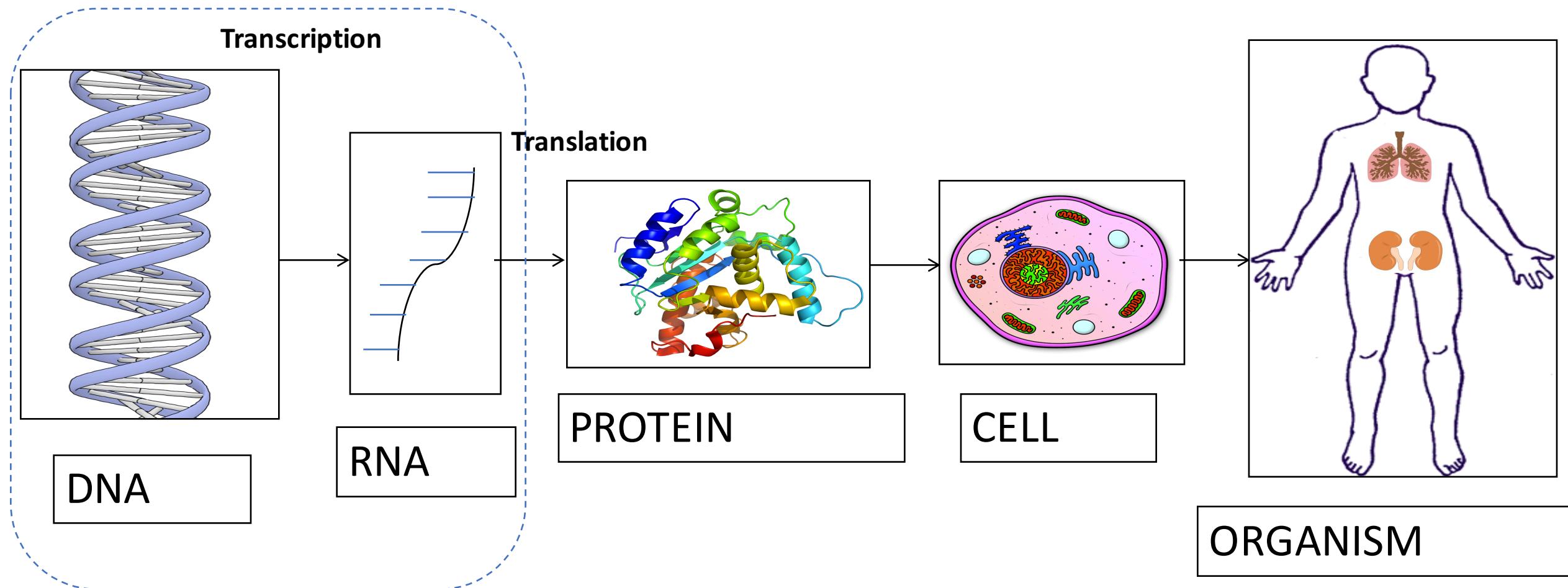


Challenge : DNNs are hard to Interpret

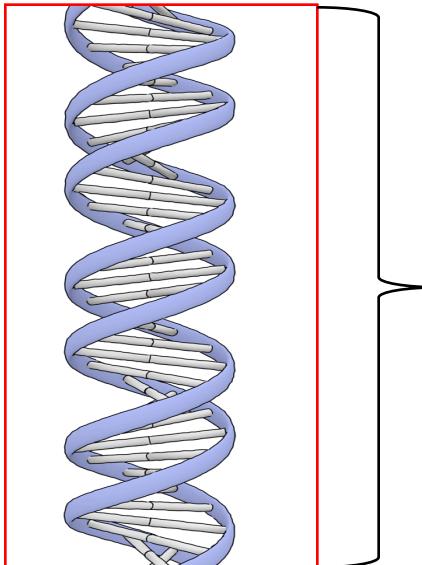
Today

- Machine Learning: a quick review
 - Deep Learning: a quick review
 - Background Biology: a quick review
 - Deep Learning for analyzing **Sequential Data** about Regulation:
 - DeepChrome
 - AttentiveChrome
 - DeepMotif
- <https://qdata.github.io/deep2Read/>
- <https://www.deepchrome.org>

Biology in a Slide



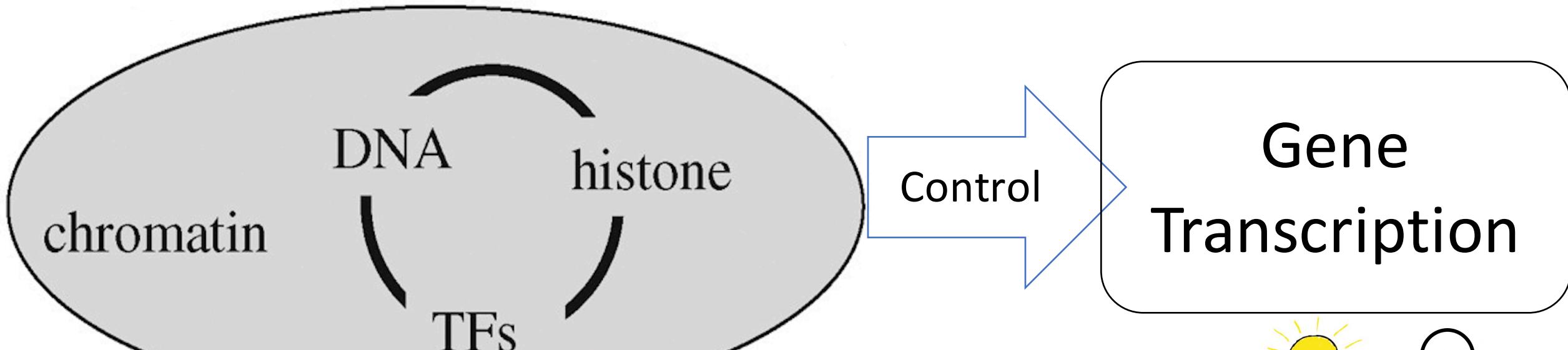
DNA and Diseases



DNA

- Down Syndrome
 - Parkinson's Disease
 - Autism
 - Muscular Atrophy
 - Sickle Cell Disease
-
-

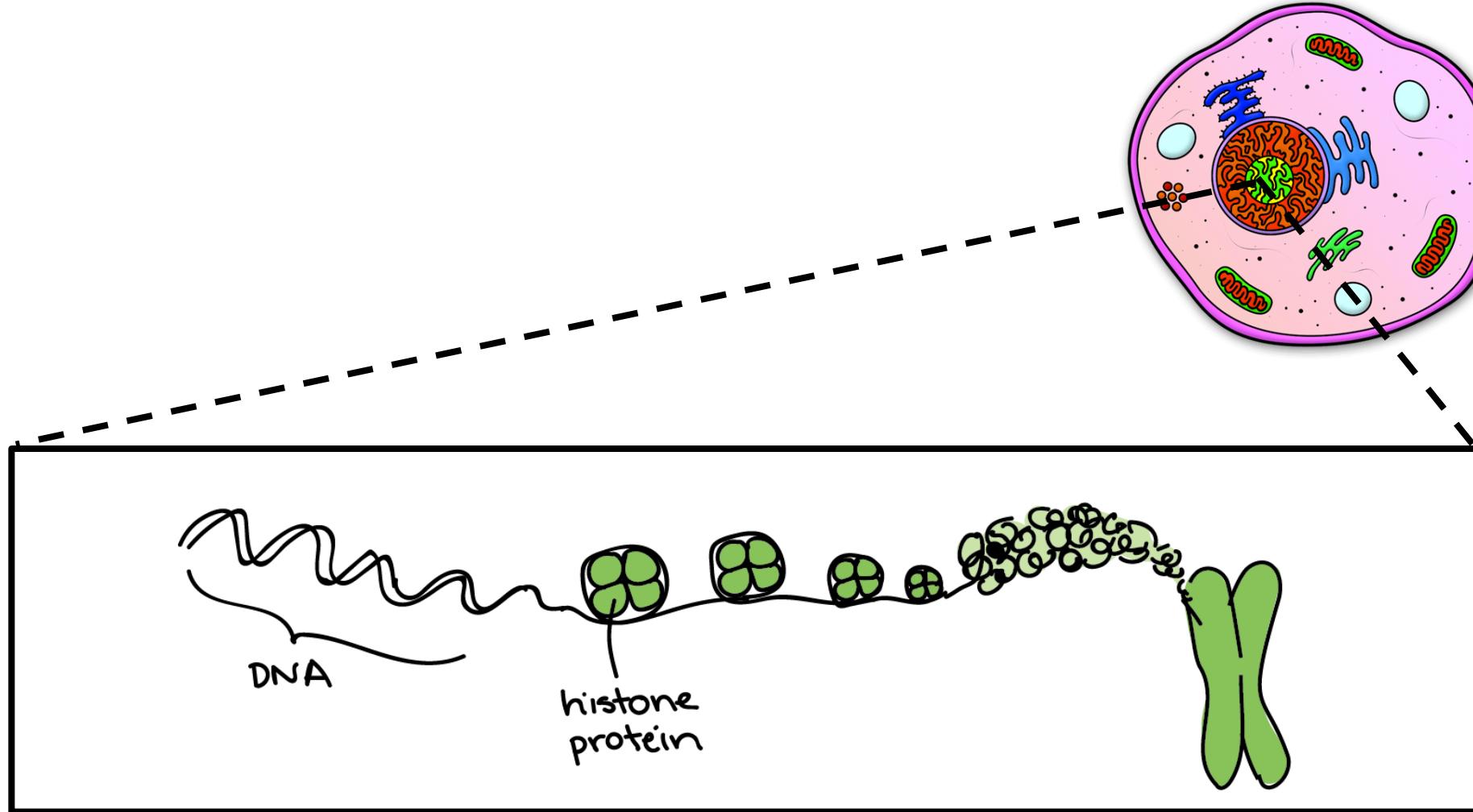
Chromatin



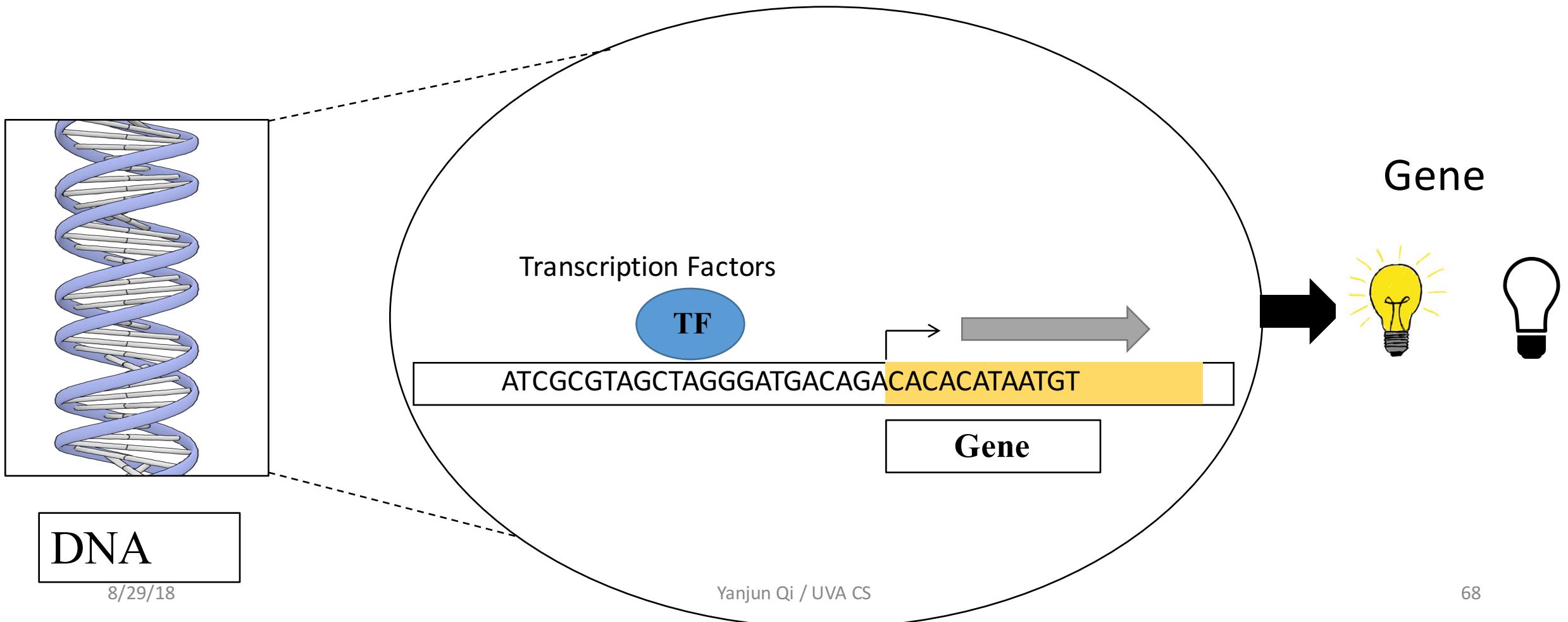
Epigenetics
“Environment
of the DNA”

Histone Proteins

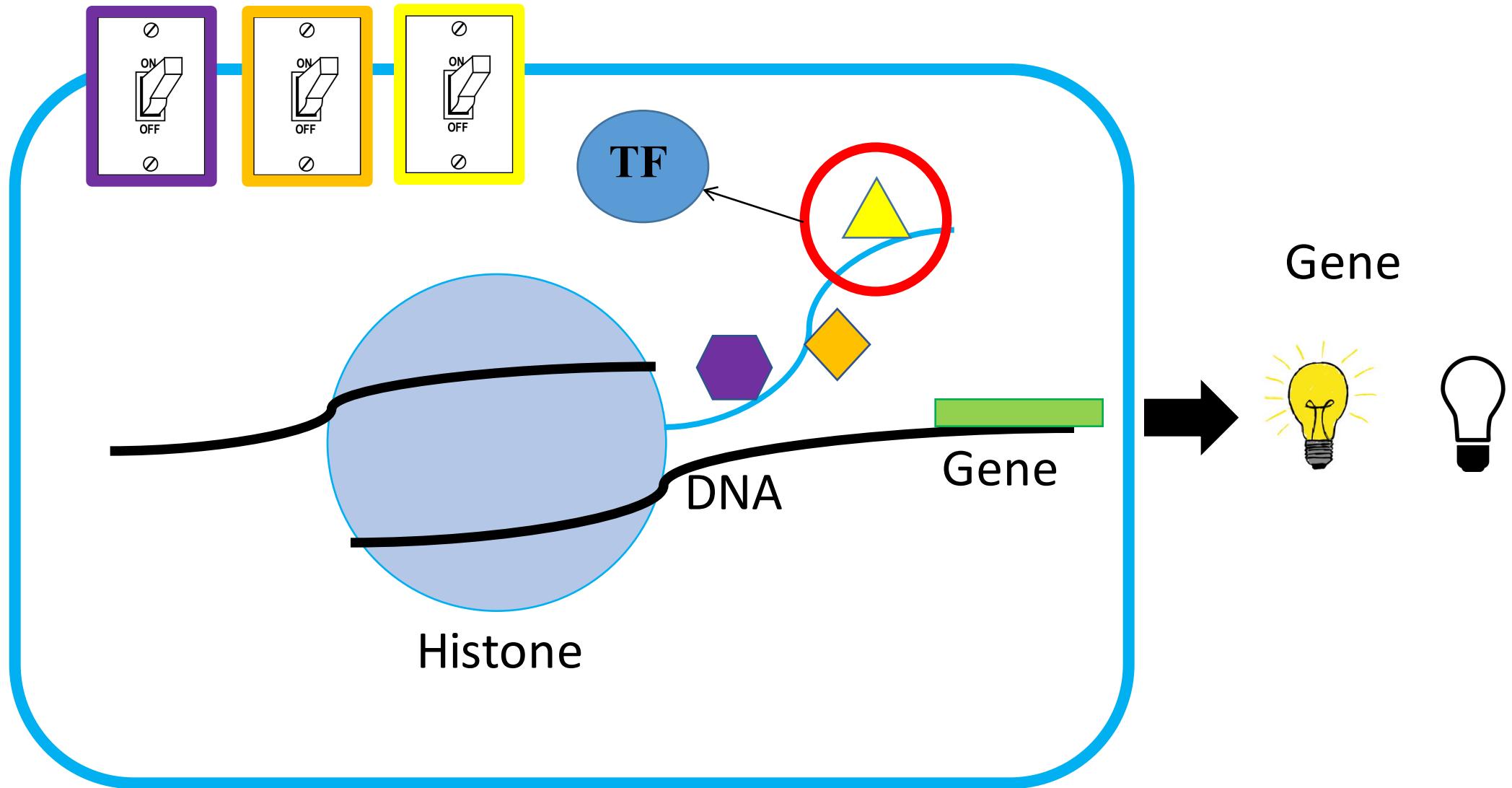
CELL



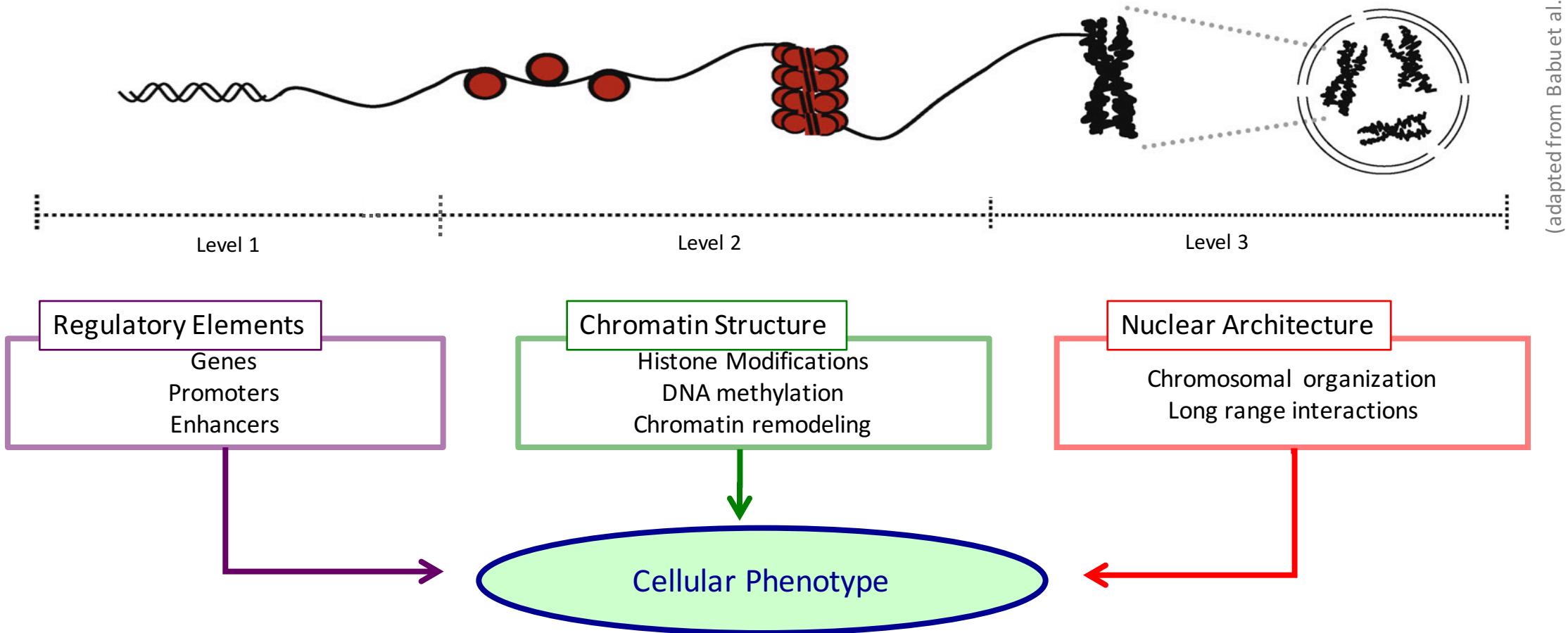
Transcription Factor Binding => Gene Transcription

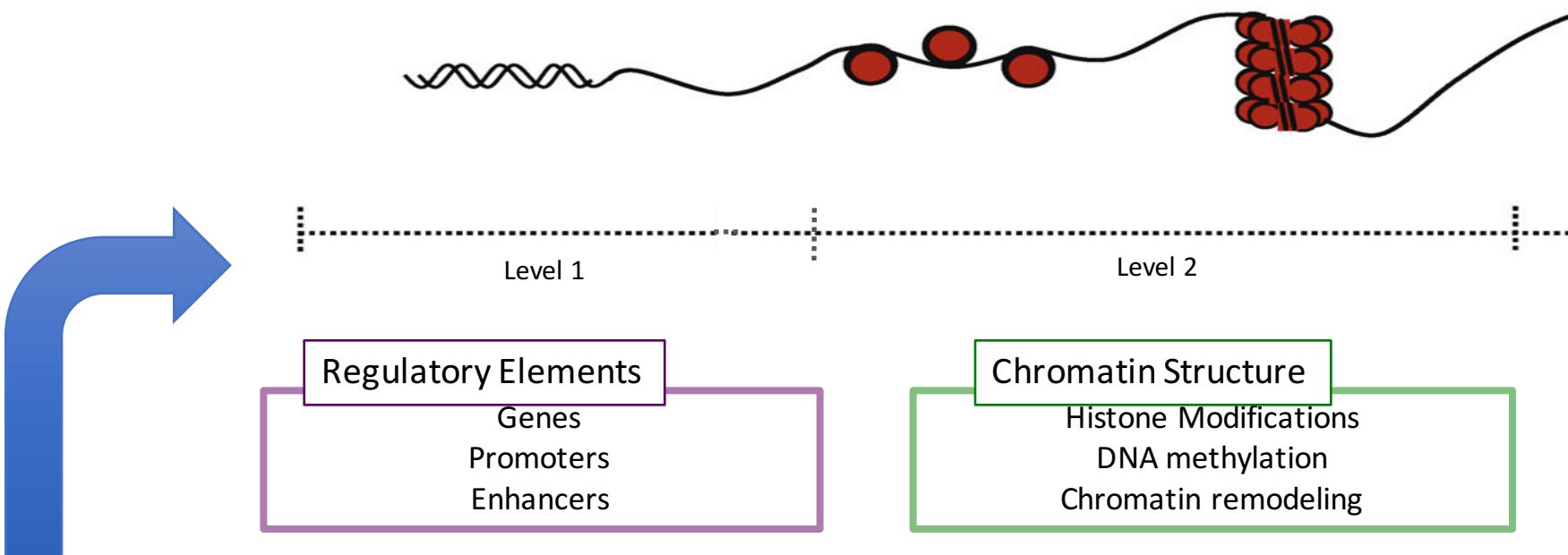


Histone Modifications (HM)



Genome Organization and Gene Regulation



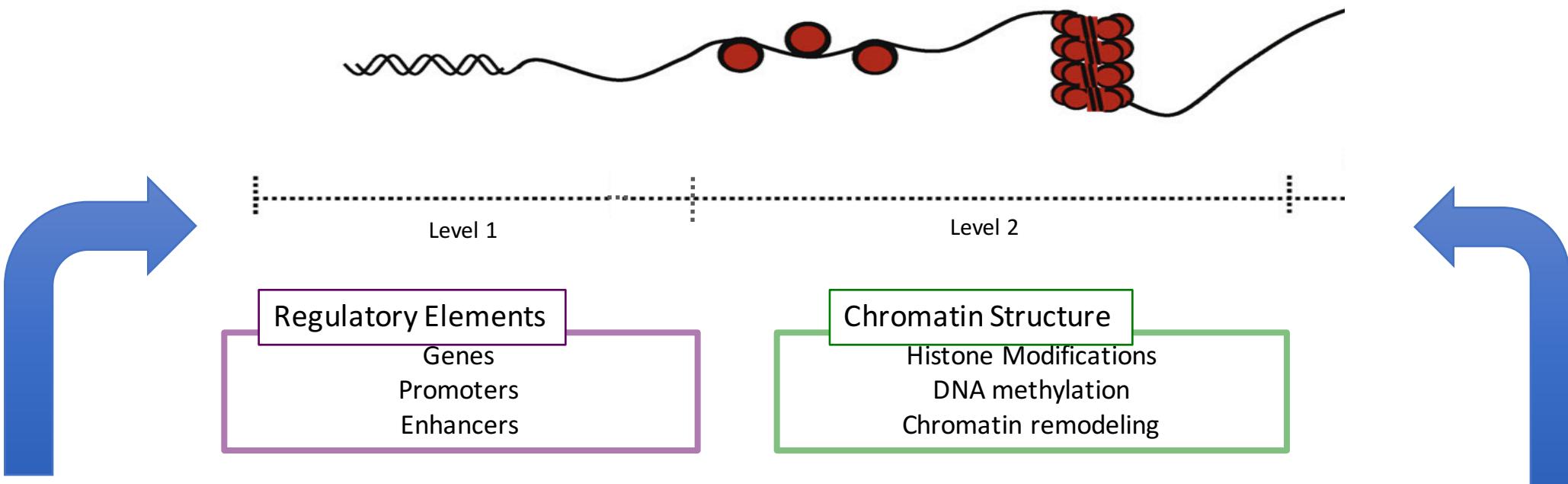


ENCODE Project (2003-Present)

Describe the functional elements encoded in human DNA

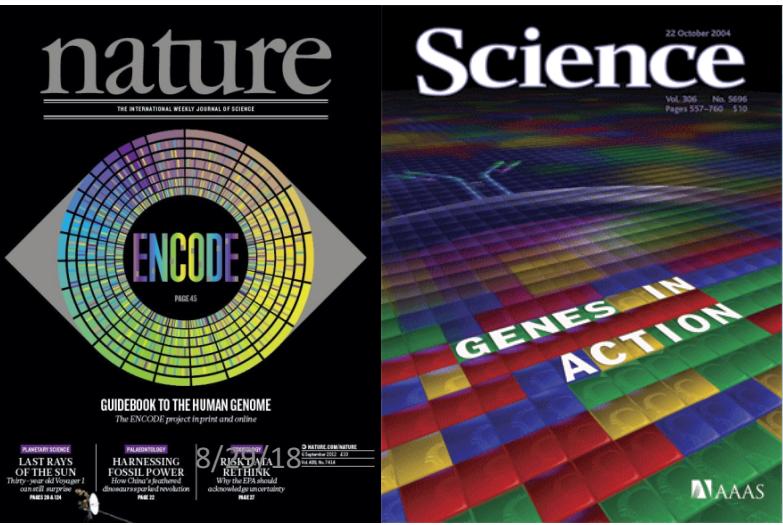


Yanjun Qi / UVA CS



ENCODE Project (2003-)

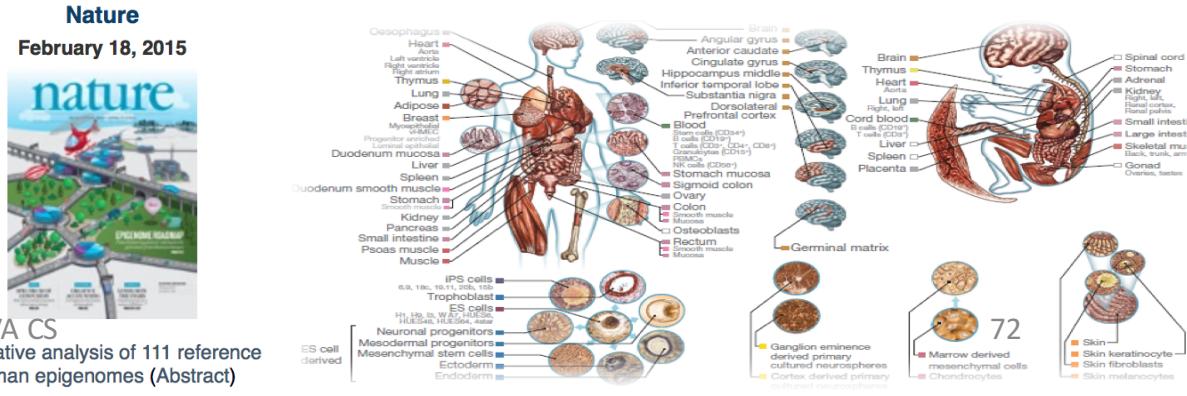
Describe the functional elements encoded in human DNA



Yanjun Qi / UVA CS
Integrative analysis of 111 reference human epigenomes (Abstract)

Roadmap Epigenetics Project (REMC, 2008-)

To produce a public resource of epigenomic maps for stem cells and primary ex vivo tissues selected to represent the normal counterparts of tissues and organ systems frequently involved in human disease.



72

Many Important Data-Driven Computational Tasks

DNA
Segments
on
Genomes

ATGCGATCAAGTCTG

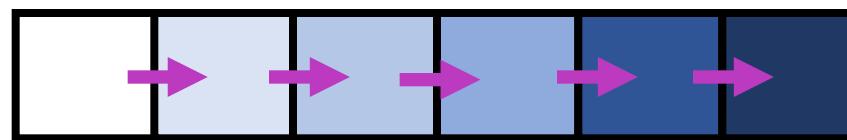
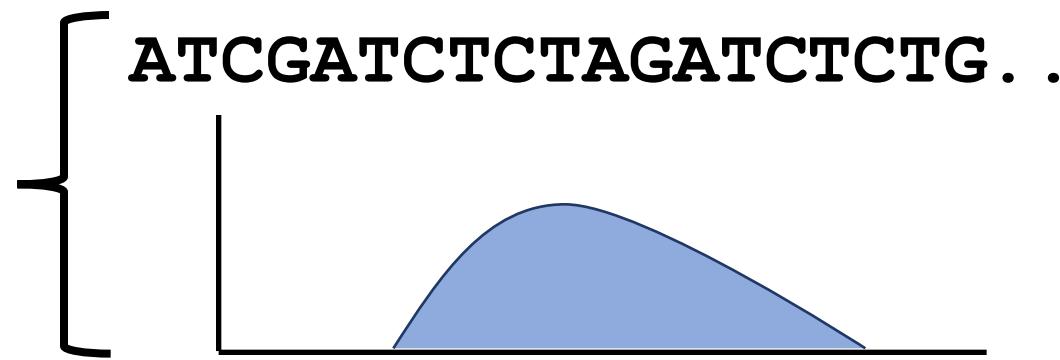
TF Binding
Signals

Gene
Expression

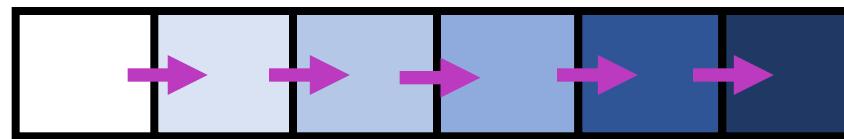
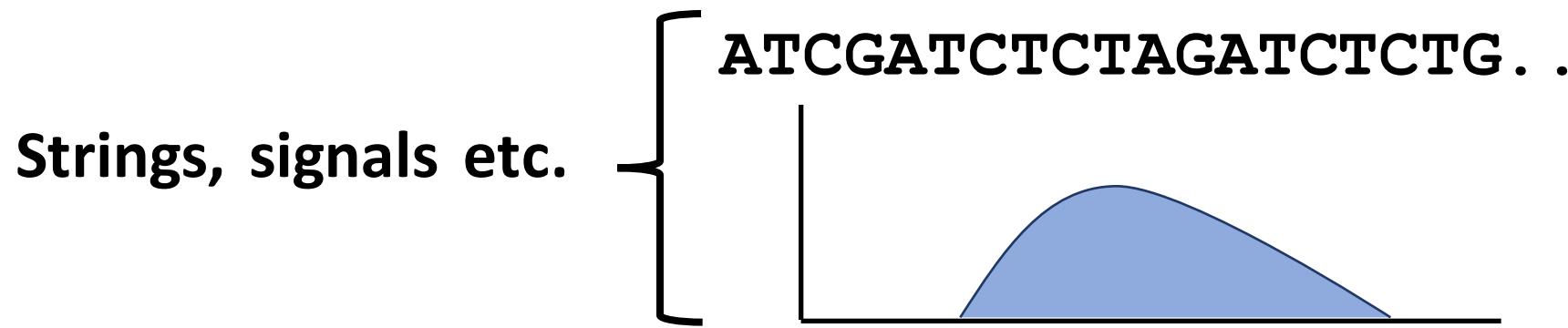
Histone
Modification
Signals

Sequential Input (X)

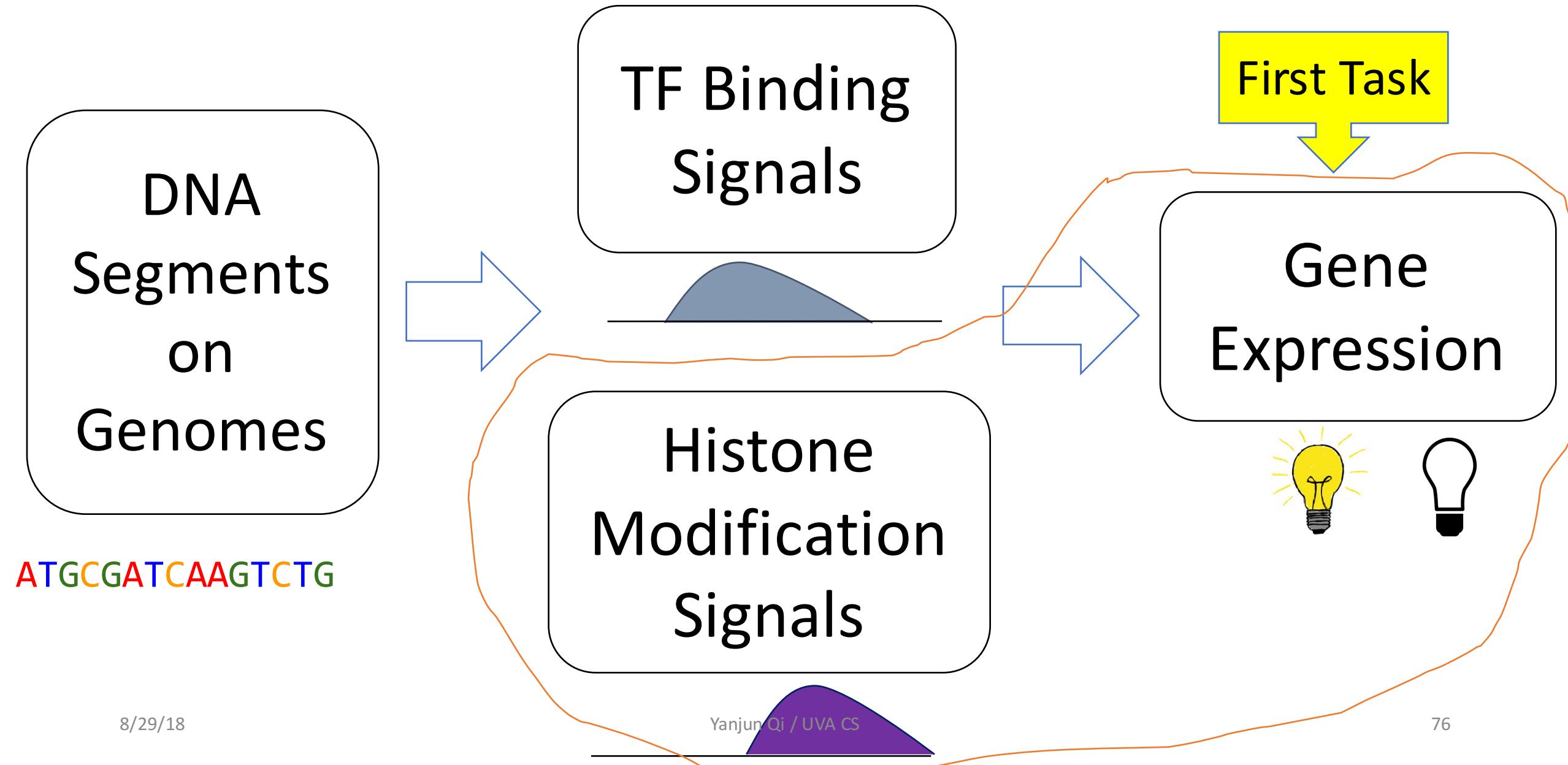
Strings, signals etc.



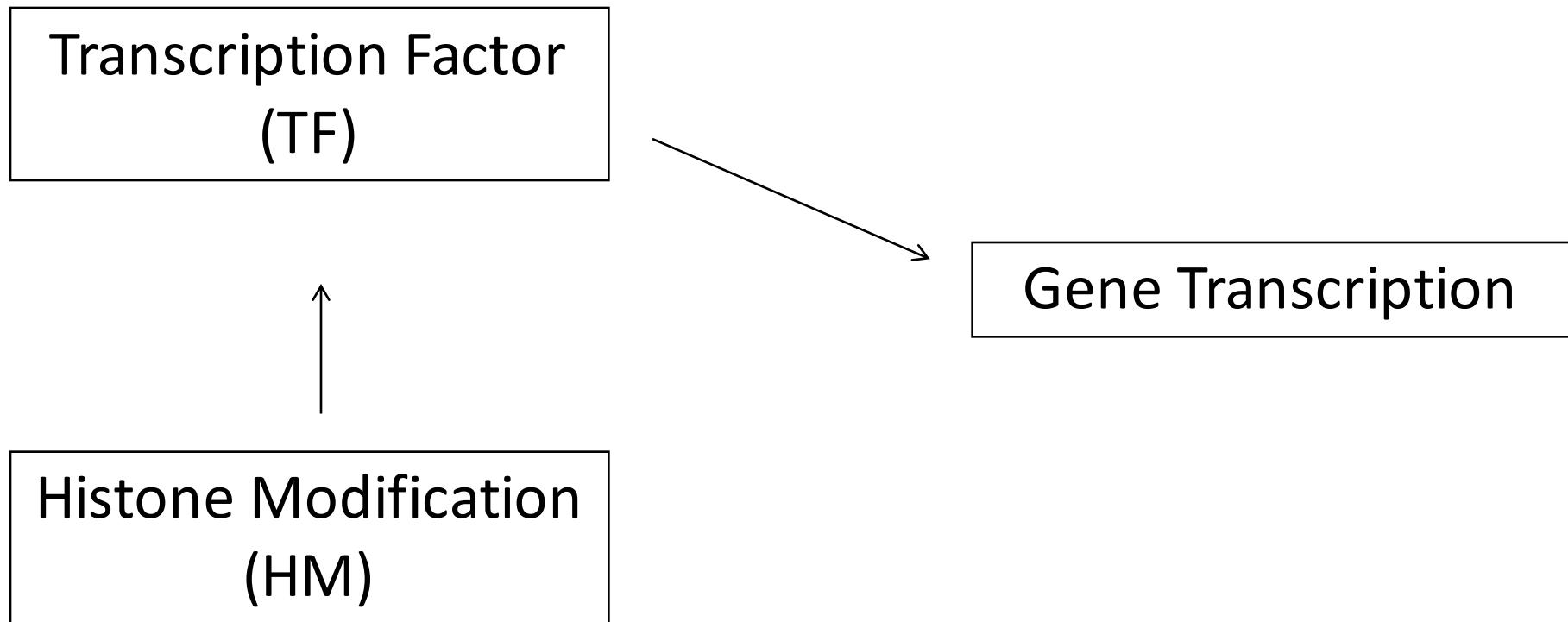
Sequential Input (X)



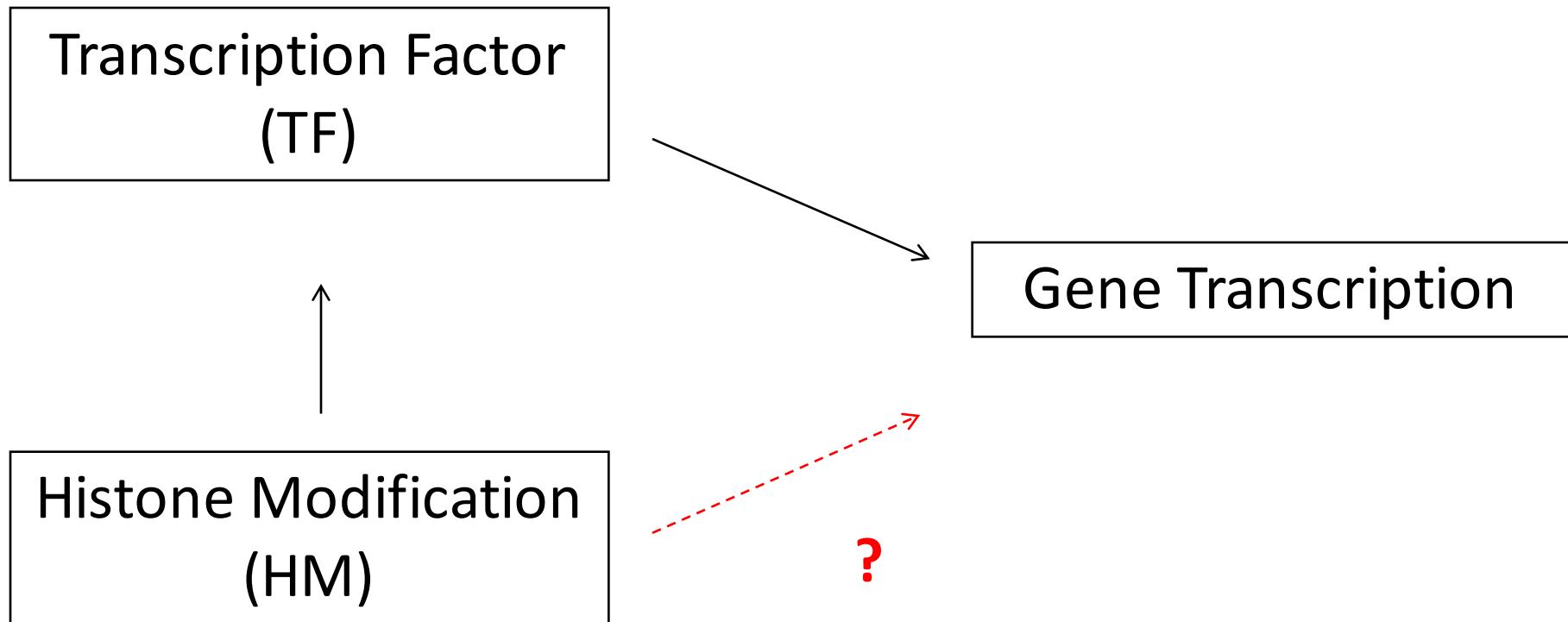
Many Important Data-Driven Computational Tasks



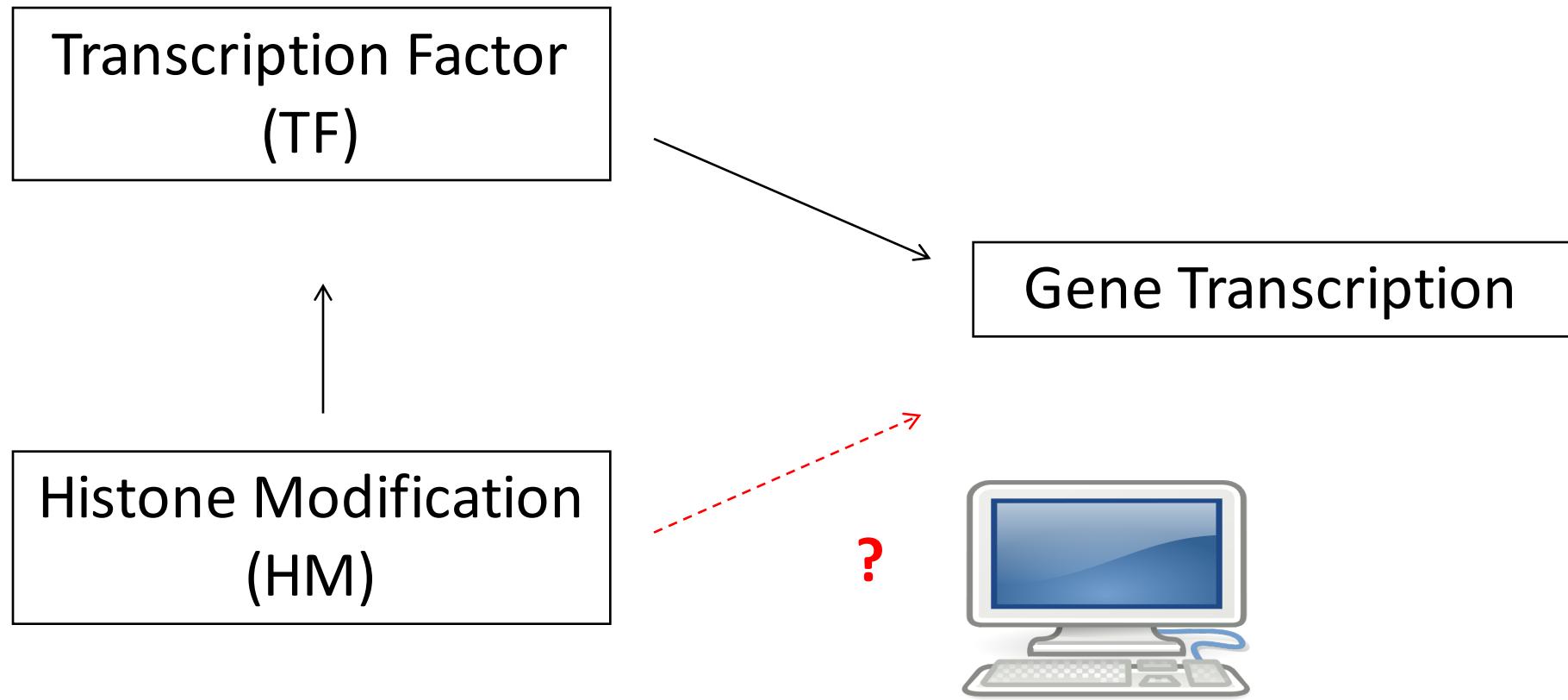
Histone Modification and Gene Transcription



Histone Modification and Gene Transcription



Histone Modification and Gene Transcription

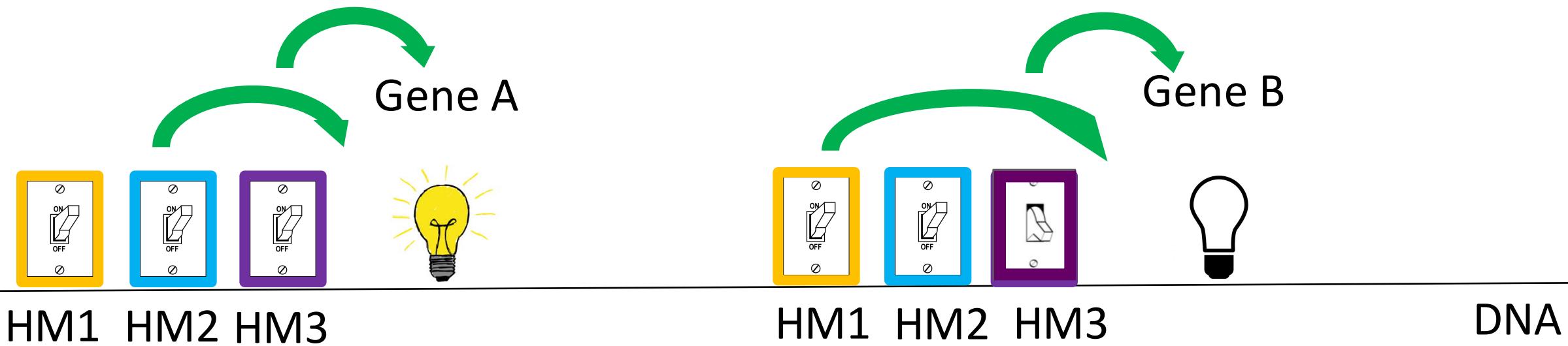


Why Studying [HM => Gene Expression] ?

- Epigenomics:
 - Study of chemical changes in DNA and histones (without altering DNA sequence)
 - Inheritable and involved in regulating gene expression, development, tissue differentiation and suppression ...
- Modification in DNA/histones (changes in chromatin structure and function)
 - => influence how easily DNA can be accessed by TF
- Epigenome is dynamic
 - Can be altered by environmental conditions
 - Unlike genetic mutations, chromatin changes such as histone modifications are potentially reversible => **Epigenome Drug** for Cancer Cells?

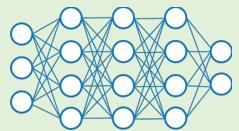
Study how HMs influence genes?

~56 Cell Types



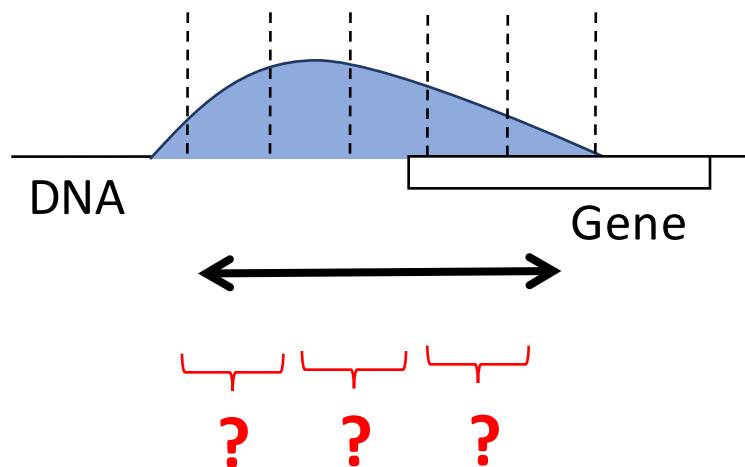
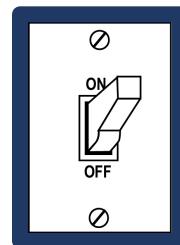
K~20 HMs
G~30,000 Genes

81



Task Formulation

Input:



HM1

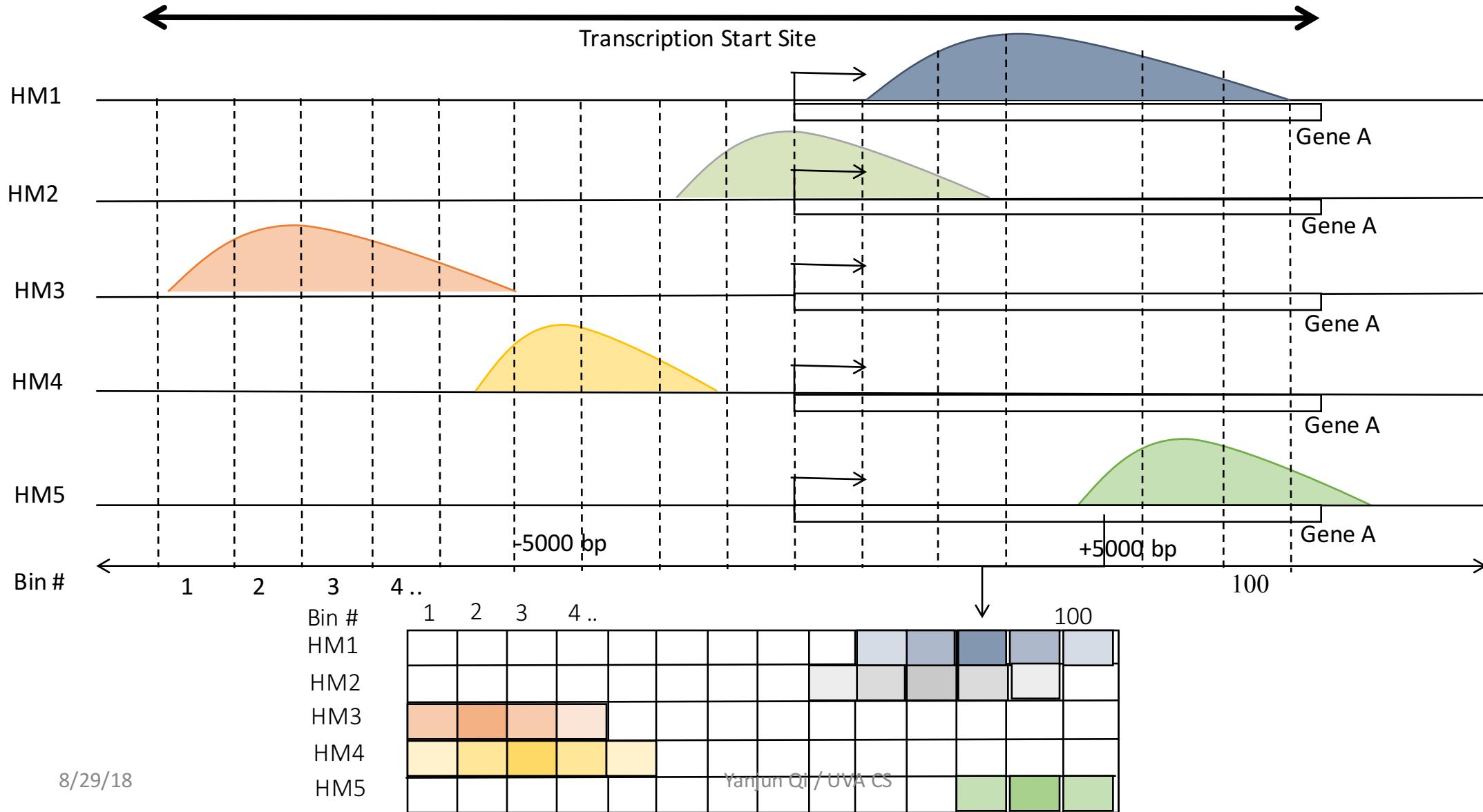


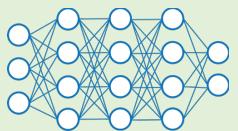
Output:



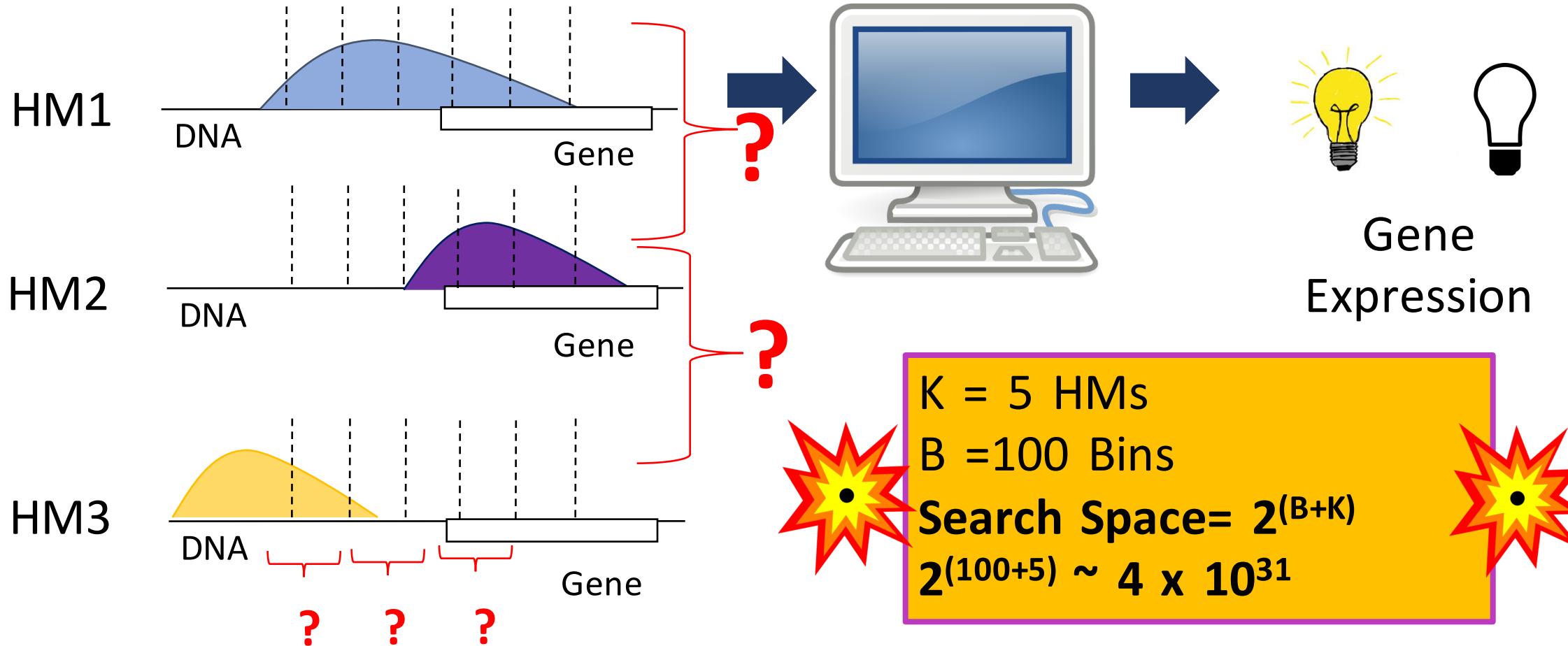
Gene
Expression

Input

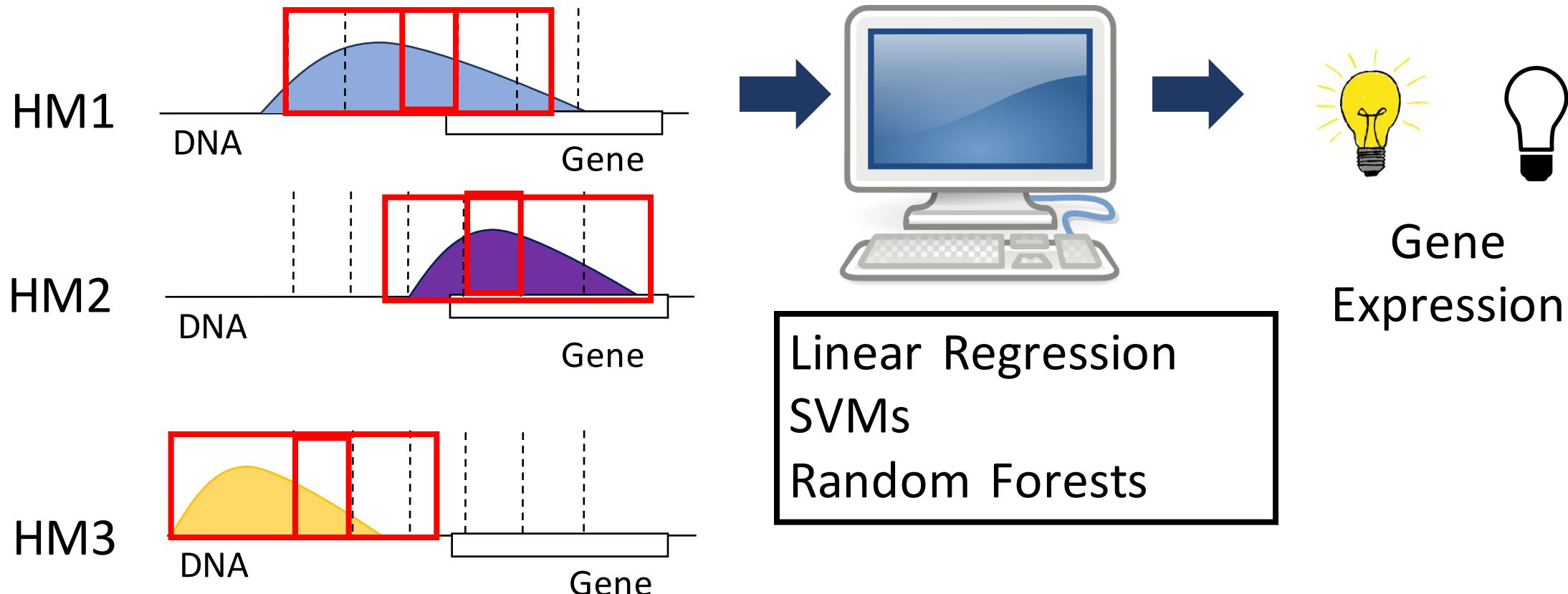




Computational Challenge



Related Work

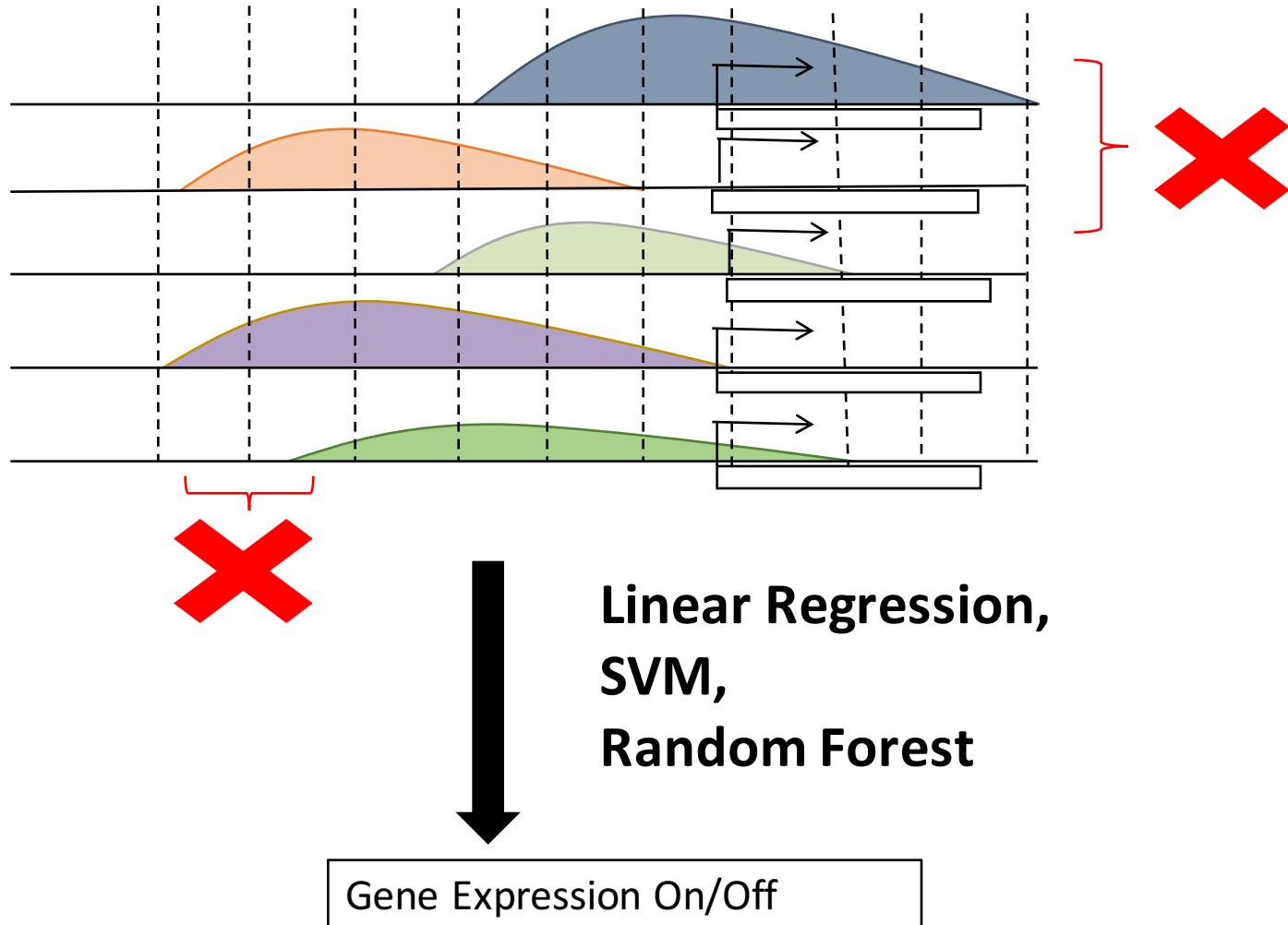


[1] Karlić, R. et al, Histone modification levels are predictive for gene expression. Proceedings of the National Academy of Sciences (2010)

[2] Cheng, C. et al, A statistical framework for modeling gene expression using chromatin features and application to modENCODE datasets. Genome Biology (2011)

[3] Dong, X. et al, Modeling gene expression using chromatin features in various cellular contexts. Genome Biology (2012)

Drawback of Related Works



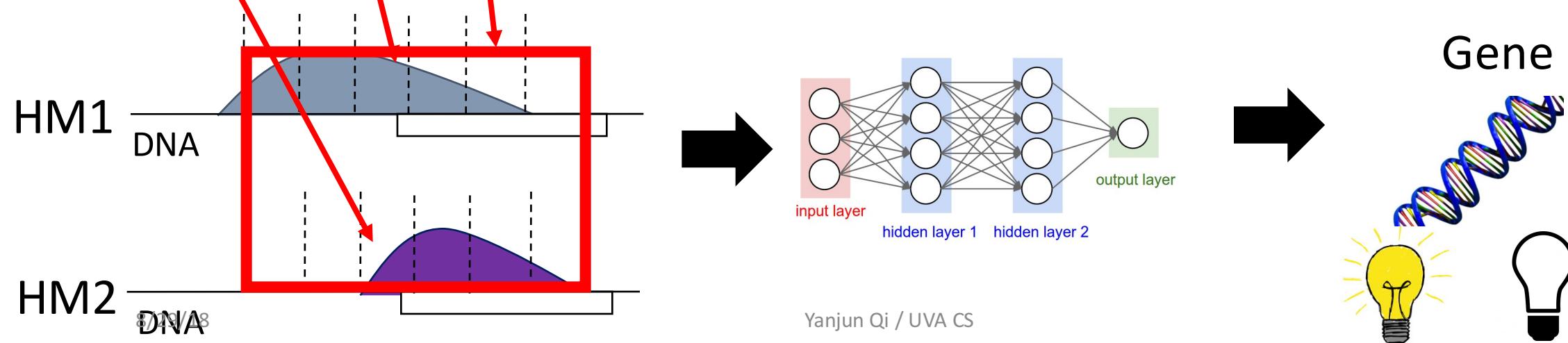
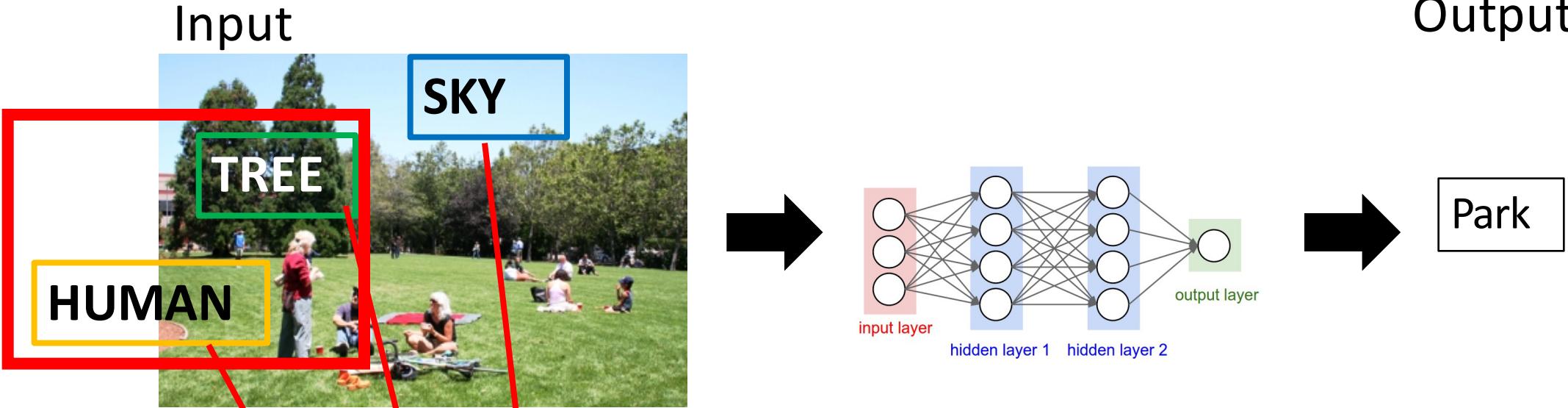
[1] Karlić, R. et al. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences* (2010)

[2] Cheng, C. et al. A statistical framework for modeling gene expression using chromatin features and application to modENCODE datasets. *Genome Biology* (2011)

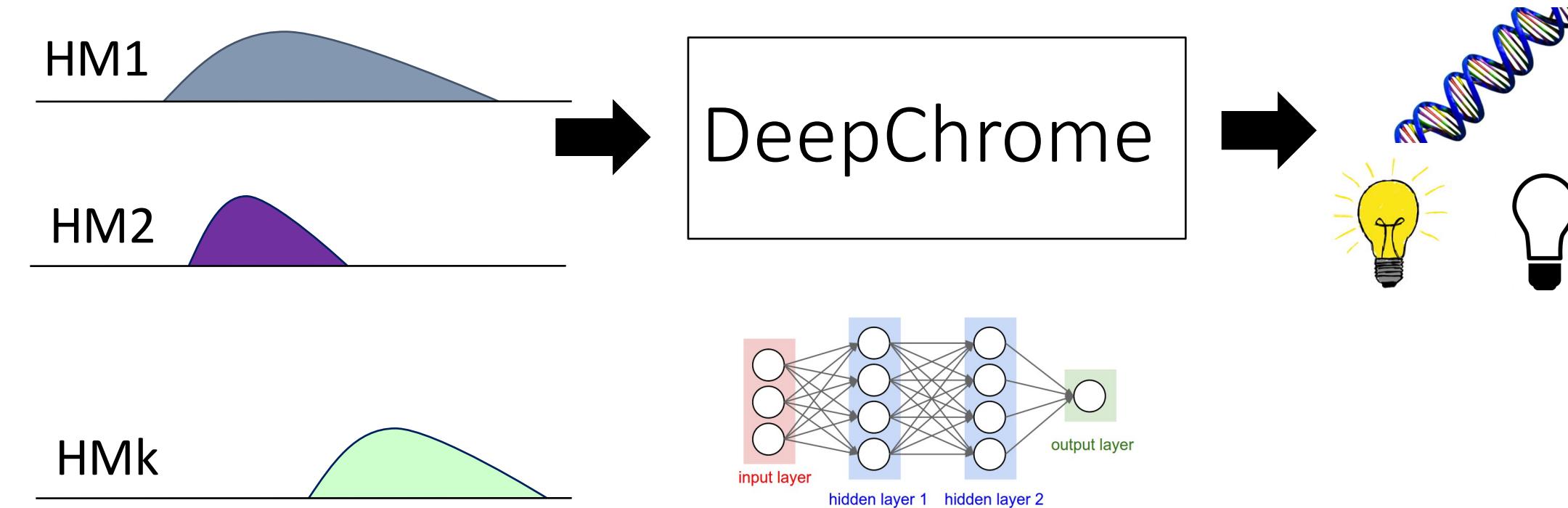
[3] Dong, X. et al. Modeling gene expression using chromatin features in various cellular contexts. *Genome Biology* (2012)

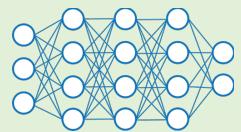
First Solution : CNN

HM signals occupy a local region
and look similar in different parts?

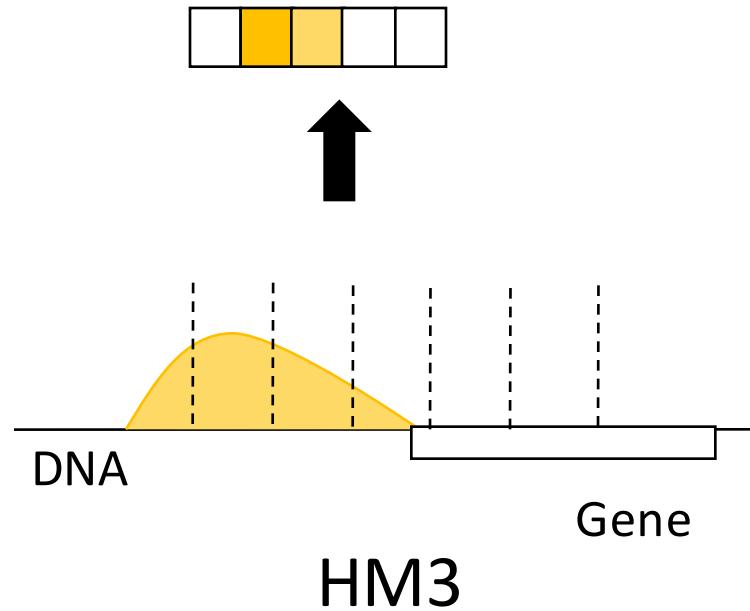
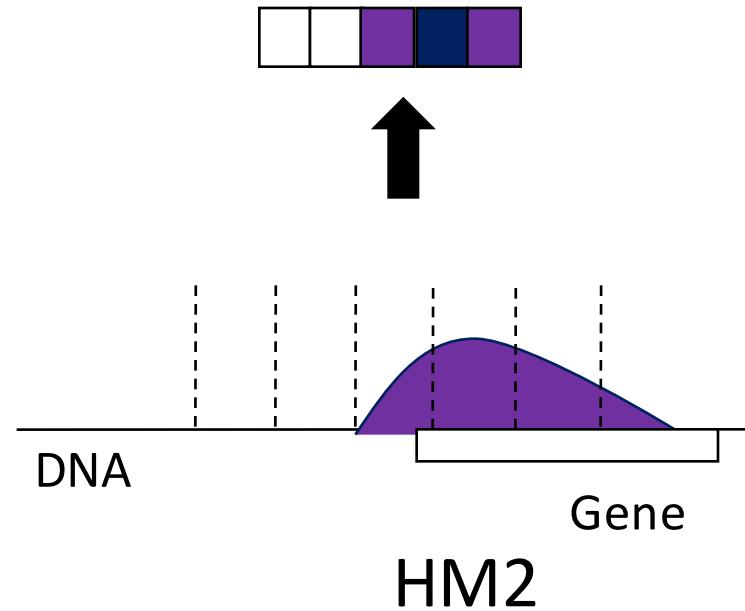
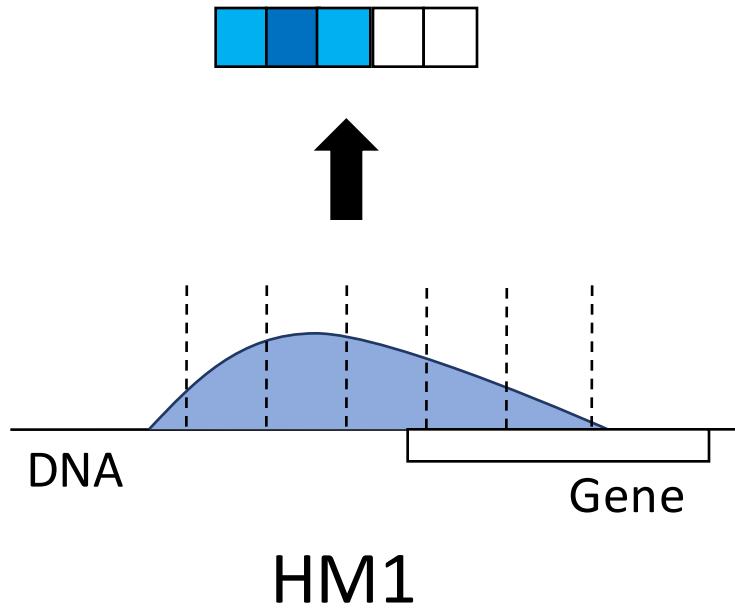


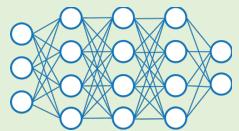
First Solution: DeepChrome : CNN



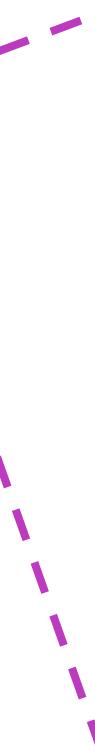
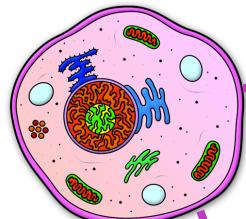


Input (X)

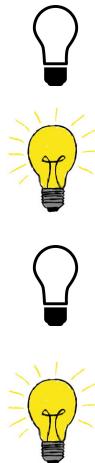




Output (Y) Labels

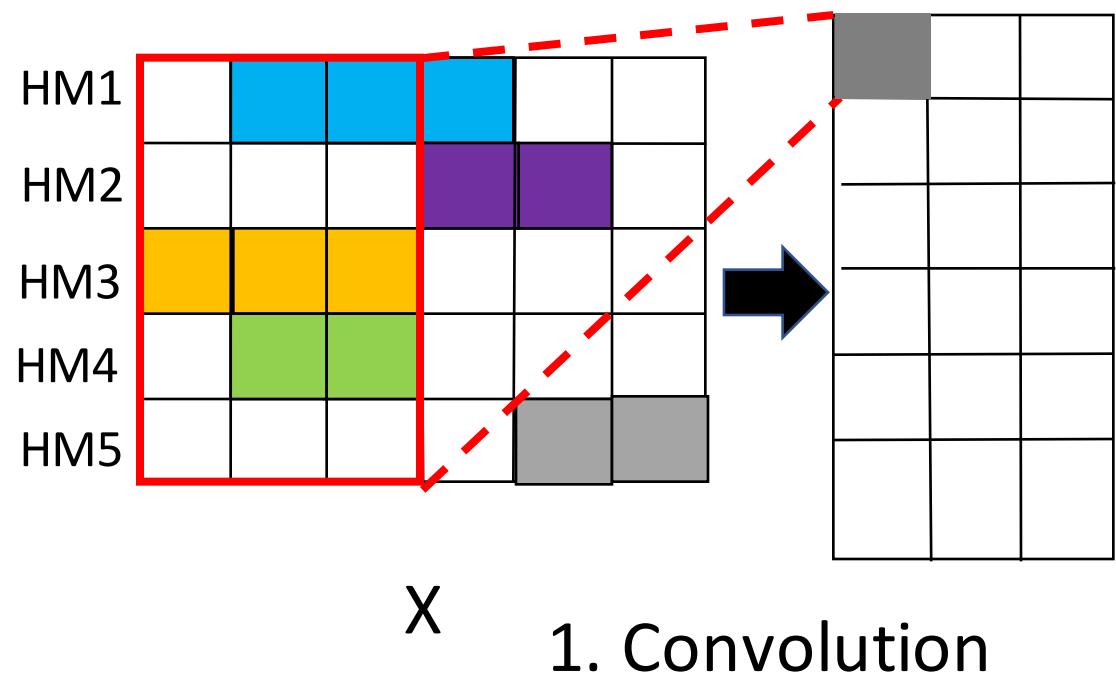


Genes	Gene Expression (RPKM)	Y Labels
RUNX1	1.296	0
SMAD2	14.902	1
MYC	3.805	0
PAX5	15.066	1
.....

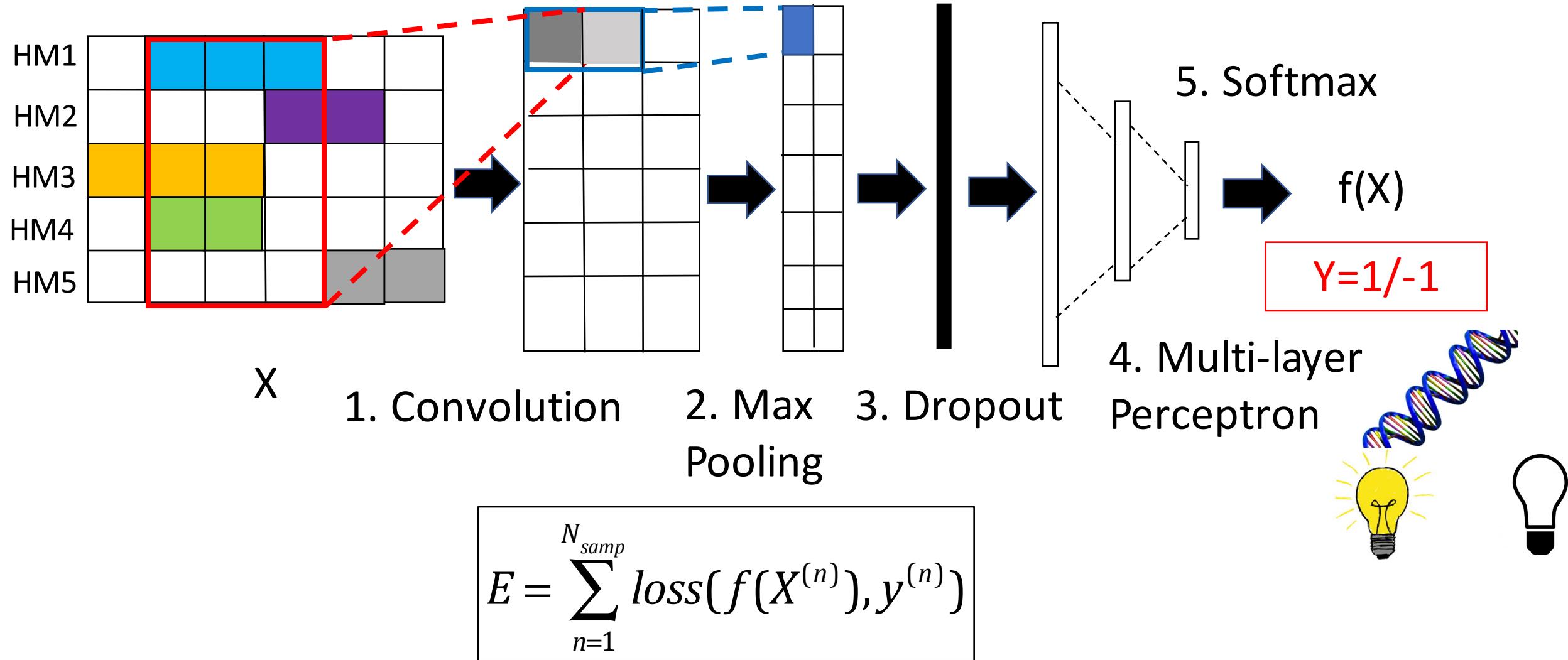


Threshold = 10.245 (Median)

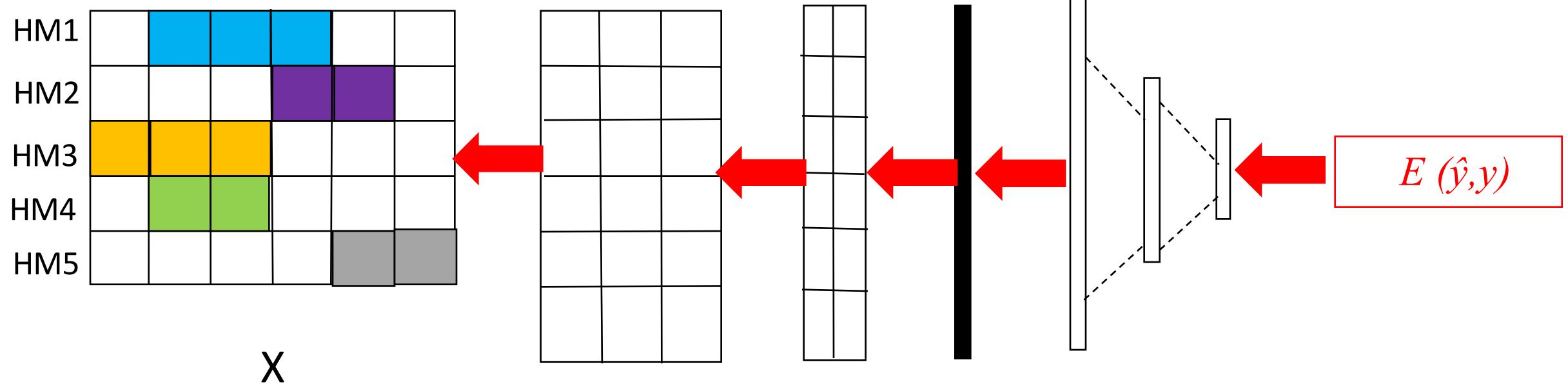
DeepChrome: Convolutional Neural Network (CNN)



DeepChrome: Convolutional Neural Network (CNN)



DeepChrome: Convolutional Neural Network (CNN)



Back-propagation: $\Theta \leftarrow \Theta - \eta \frac{\partial E}{\partial \Theta}$

Experimental Setup

- Roadmap Epigenetics Project (REMC)
- **Cell-types:** 56
- **Input (HM):** ChIP-Seq Maps / 5 Tier-1 HMs

Histone Mark	Functional Category
H3K27me3	Repressor
H3K36me3	Structural Promoter
H3K4me1	Distal Promoter
H3K4me3	Promoter
H3K9me3	Repressor

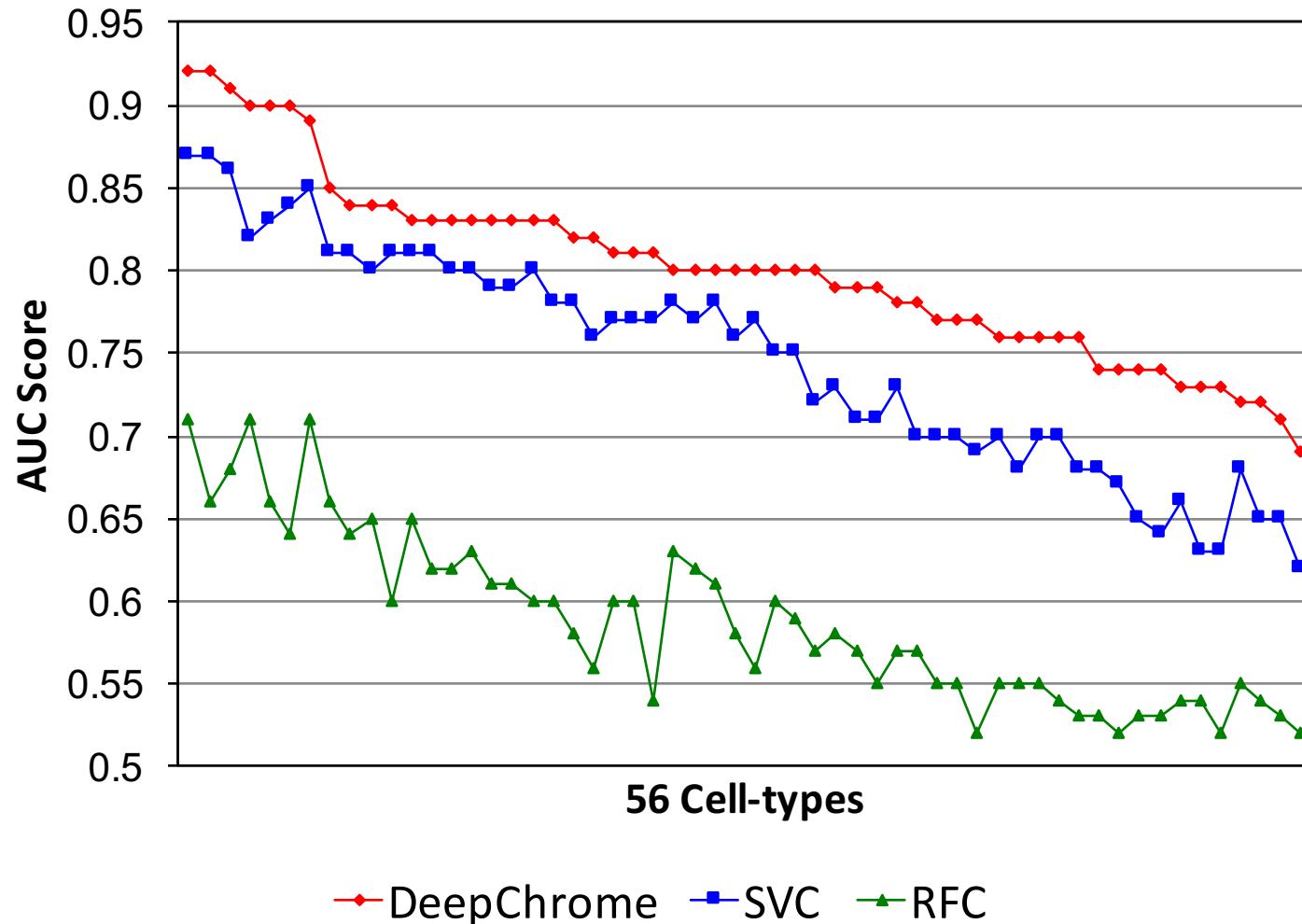
- **Output (Gene Expression):** Discretized RNA-Seq
- **Baselines:** Support Vector Classifier (SVC) and Random Forest Classifier (RFC)

Training Set
6601 Genes

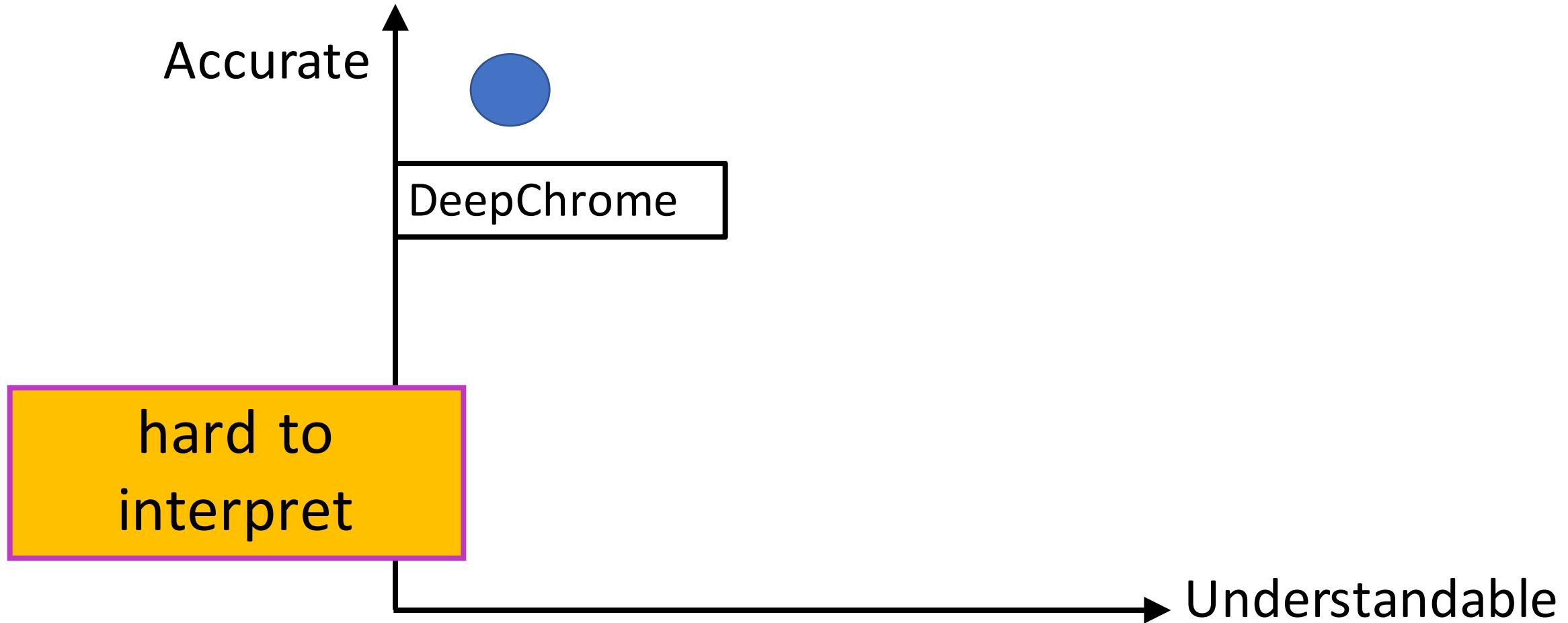
Validation Set
6601 Genes
Tianjun Qu / UMA CS

Test Set
6600 Genes

Results: Accuracy



Summary of tools



Solution: Interpretability by Hierarchical Attention

Input

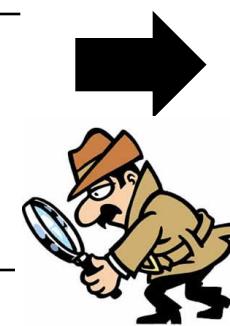
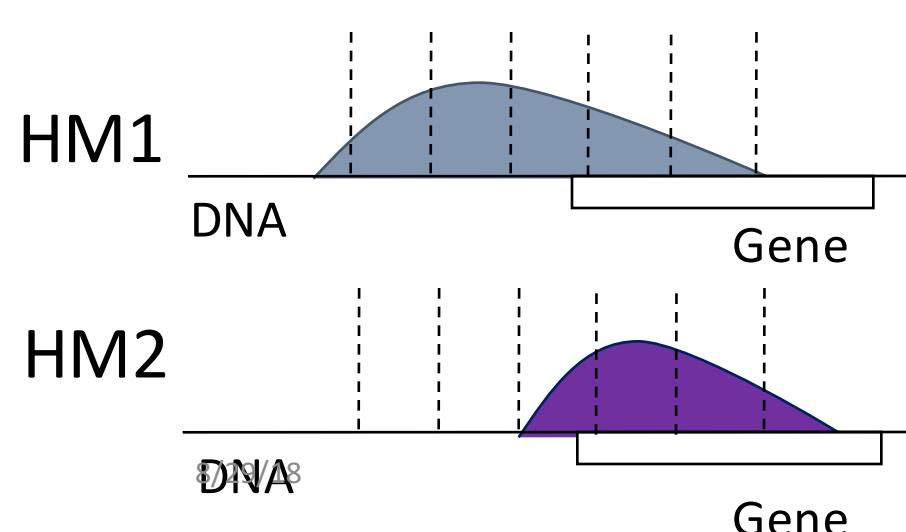


Output

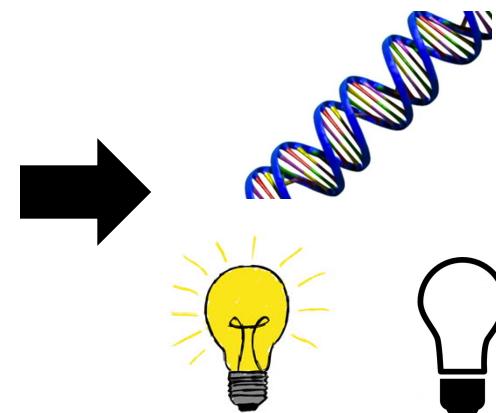
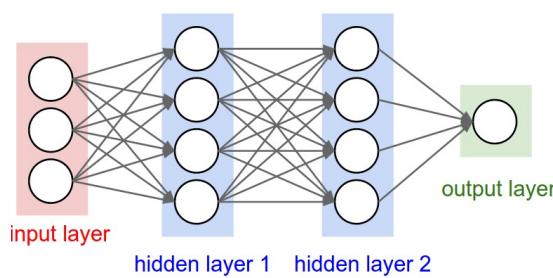
Park

Gene

Attention
Mechanism



Yanjun Qi / UVA CS



Solution: Interpretability by Hierarchical Attention

Input



Output

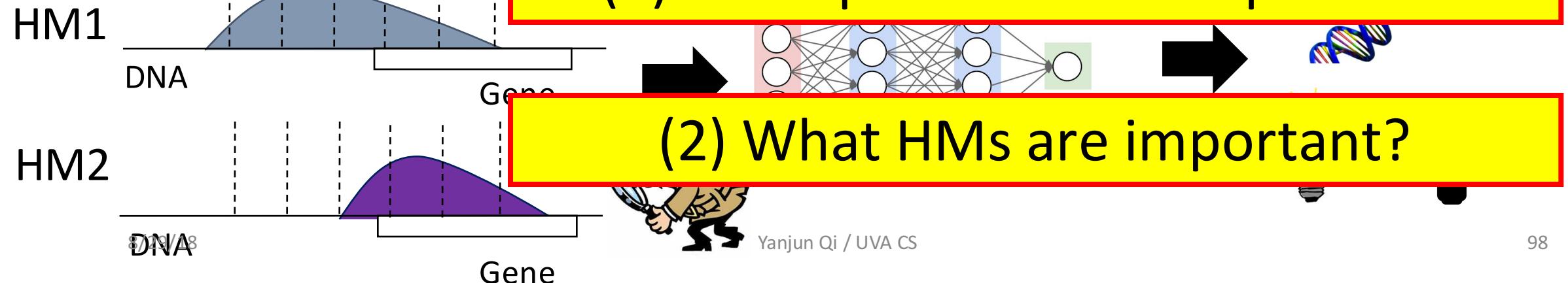
Park

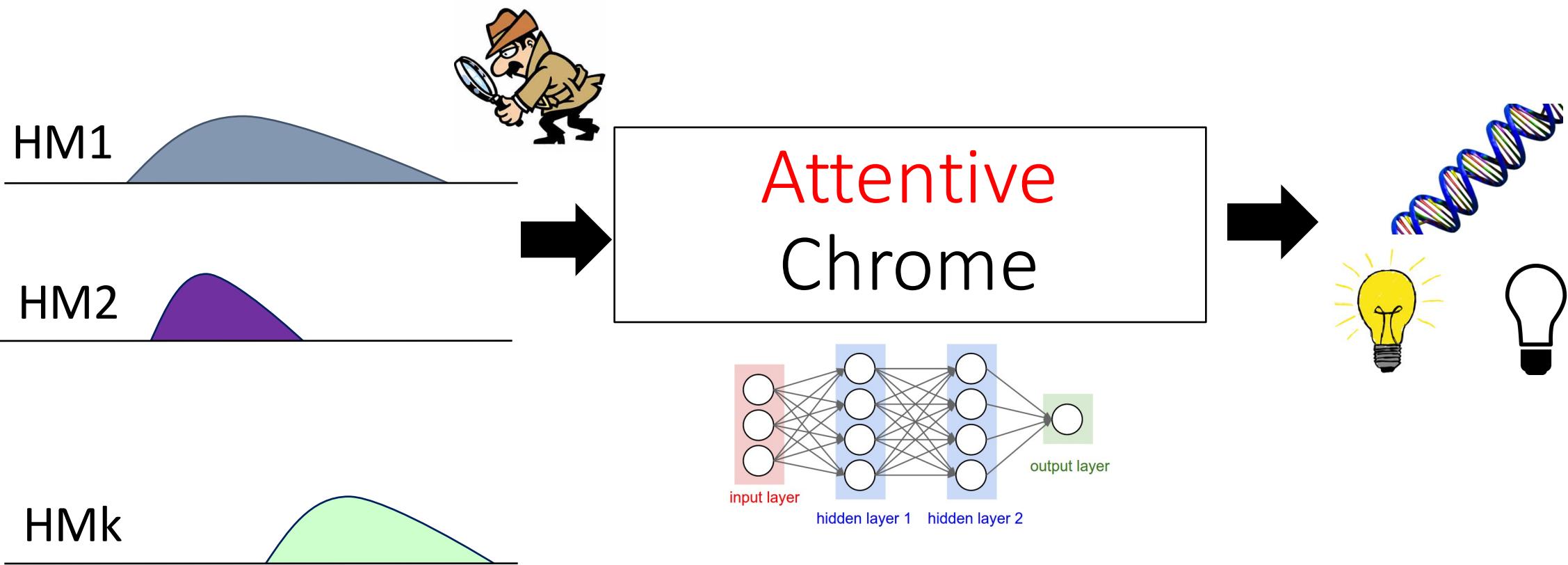
Attention
Mechanism

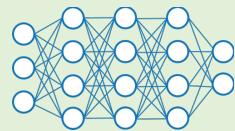


Gene

(1) What positions are important?

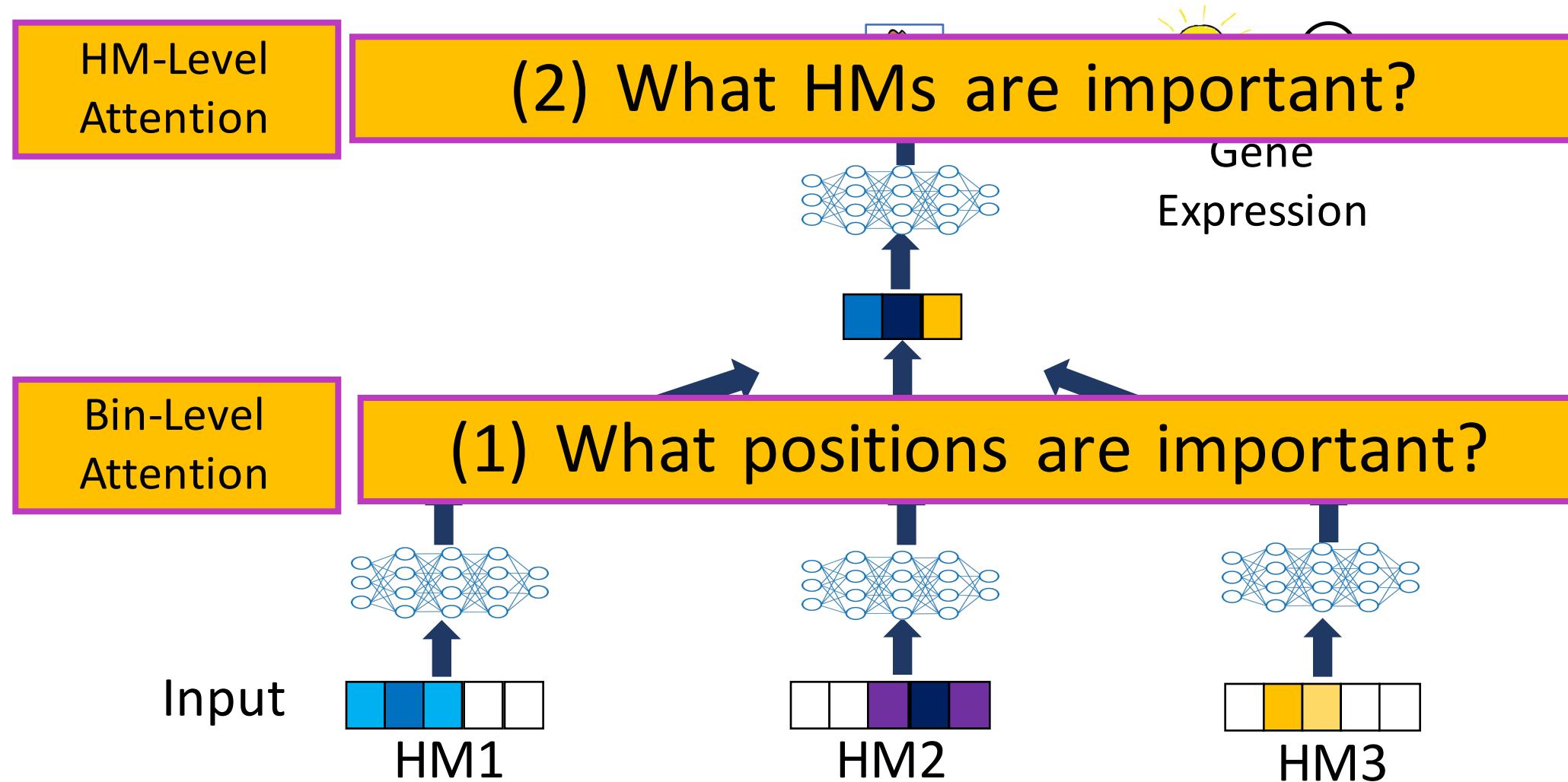






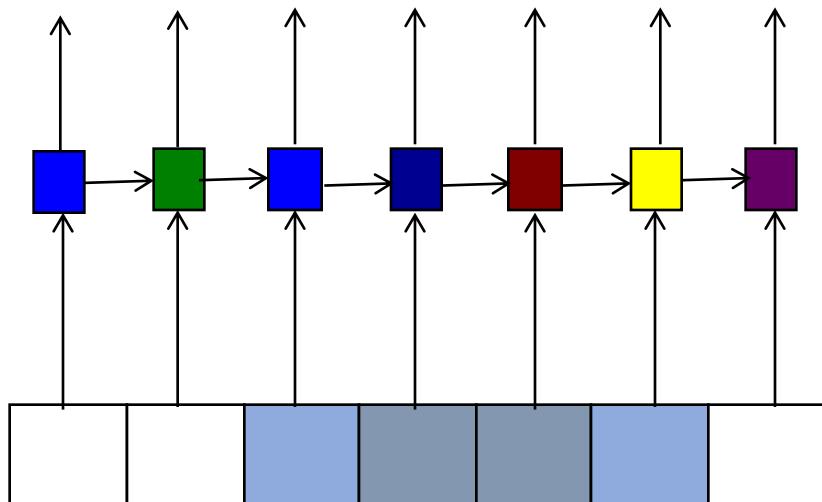
AttentiveChrome

[NIPS 2017]

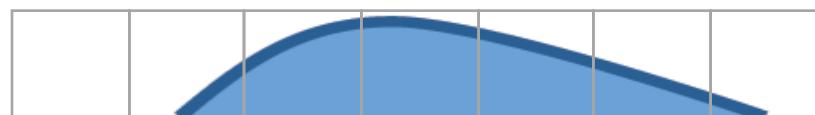
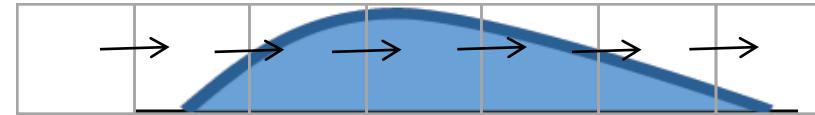


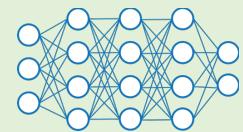
Multiple Recurrent Neural Networks (Hierarchical RNNs)

to model each HM and the Combination of all HMs : for example on HM1

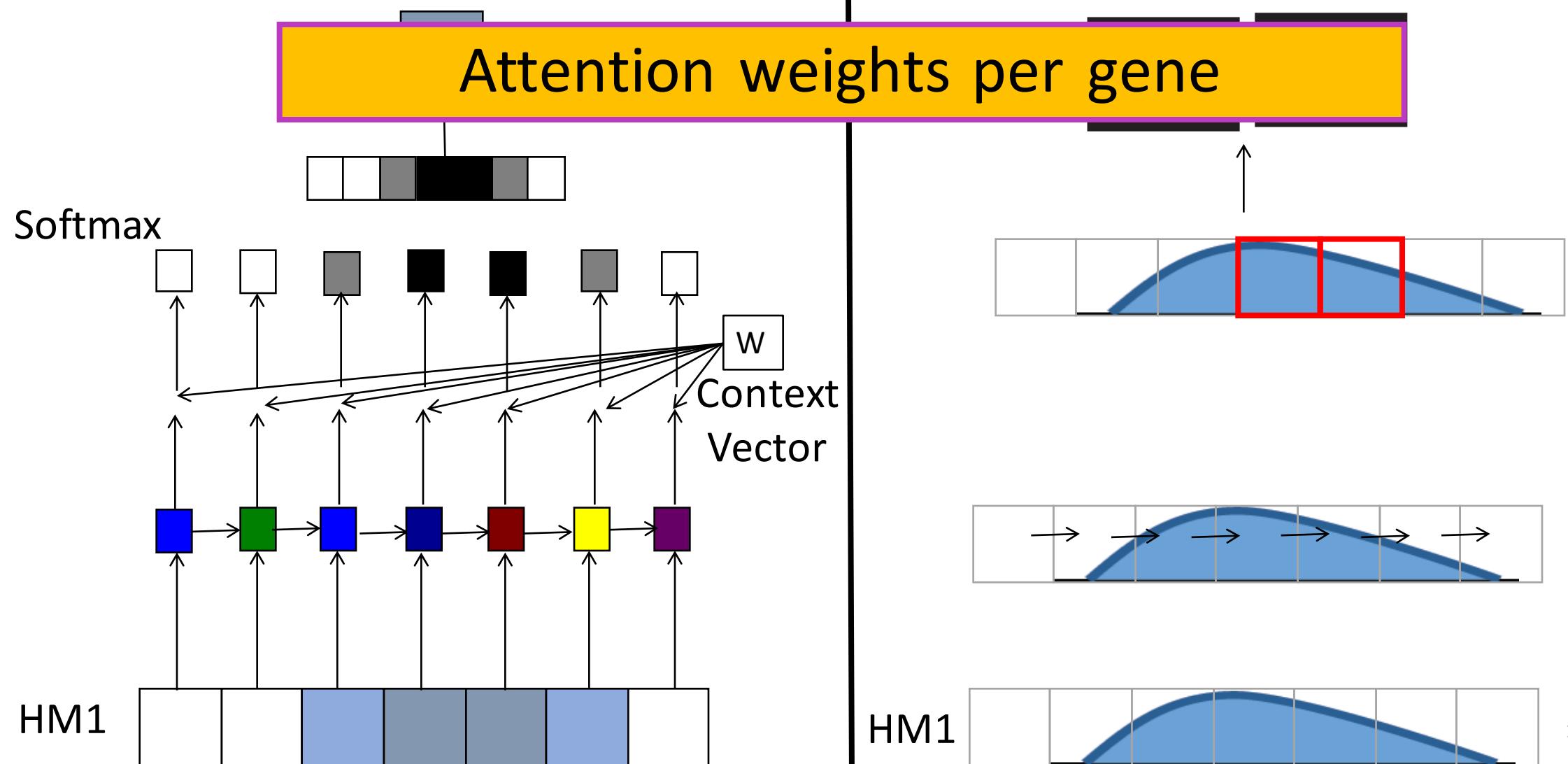


HM1

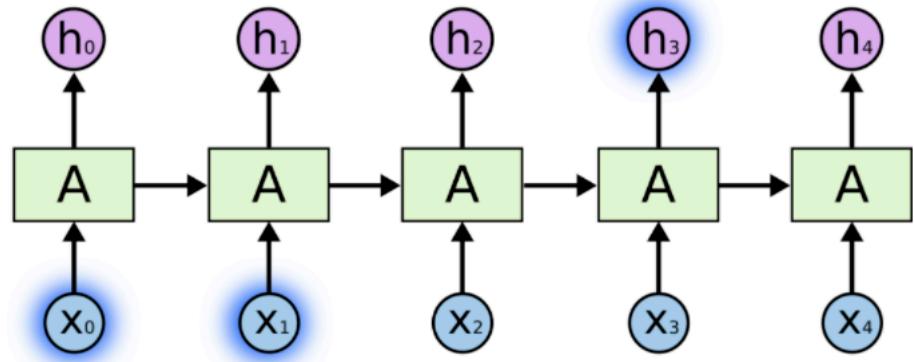




Attention Mechanism



Using Attention to Select RNN per-unit outputs

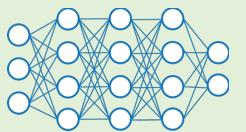


$$h_t = f_W(h_{t-1}, x_t)$$

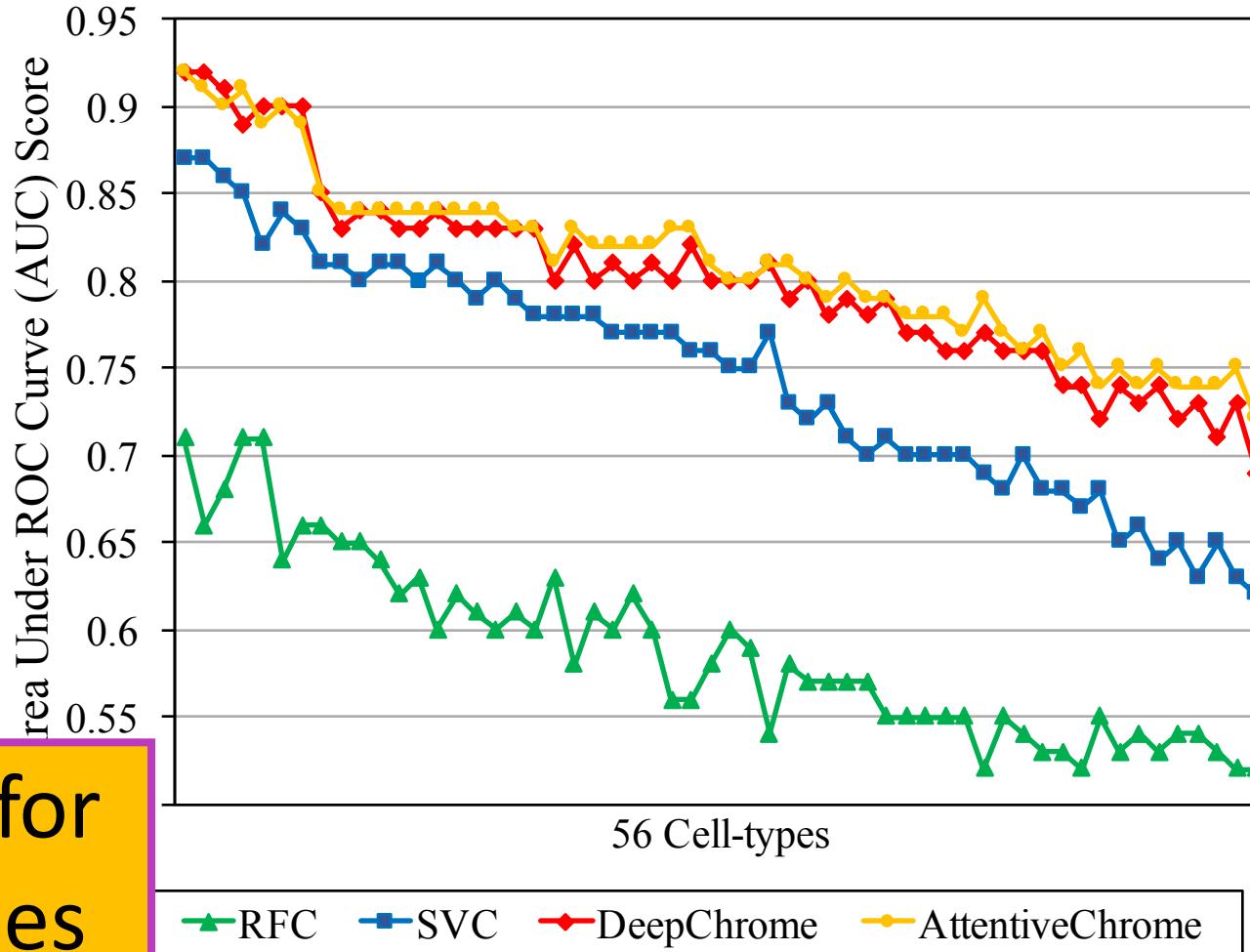
new state / old state input vector at
 \ some time step
 some function
 with parameters W

$$\alpha_t^j = \frac{\exp(\mathbf{W}_b \mathbf{h}_t^j)}{\sum_{i=1}^T \exp(\mathbf{W}_b \mathbf{h}_i^j)}$$

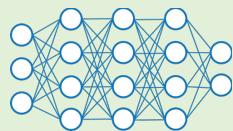
\mathbf{W}_b is learned



Prediction



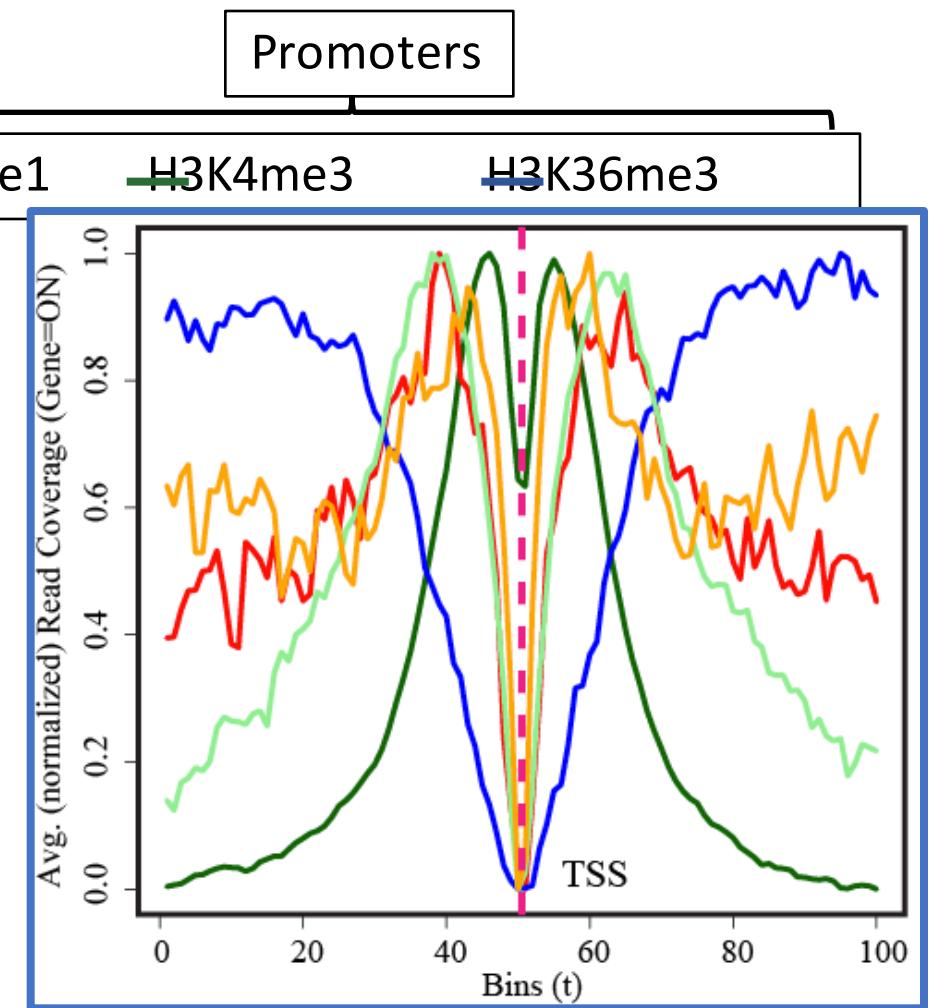
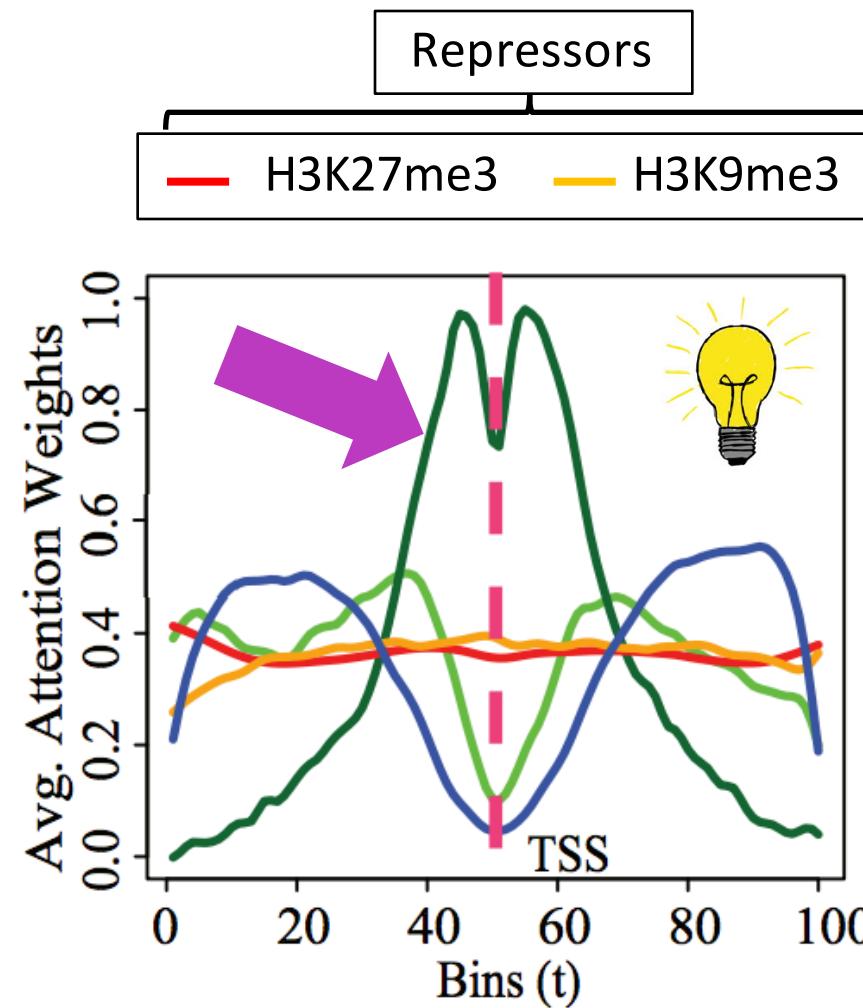
Improvement for
49/56 Cell-types

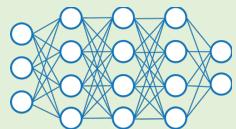


Bin-Level Visualization

CELL TYPE: GM1

(1) What positions are important?





HM-Level Visualization

(2) What HMs are important?

Cell Types:

(Stem Cell)

(Blood Cell)

(Leukemia)

Color Scale

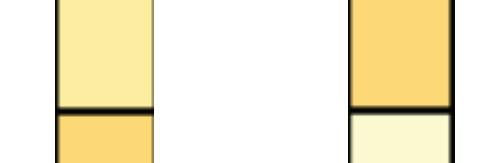


0 1

H3K27me3



H3K36me3



H3K4me1



H3K4me3



H3K9me3

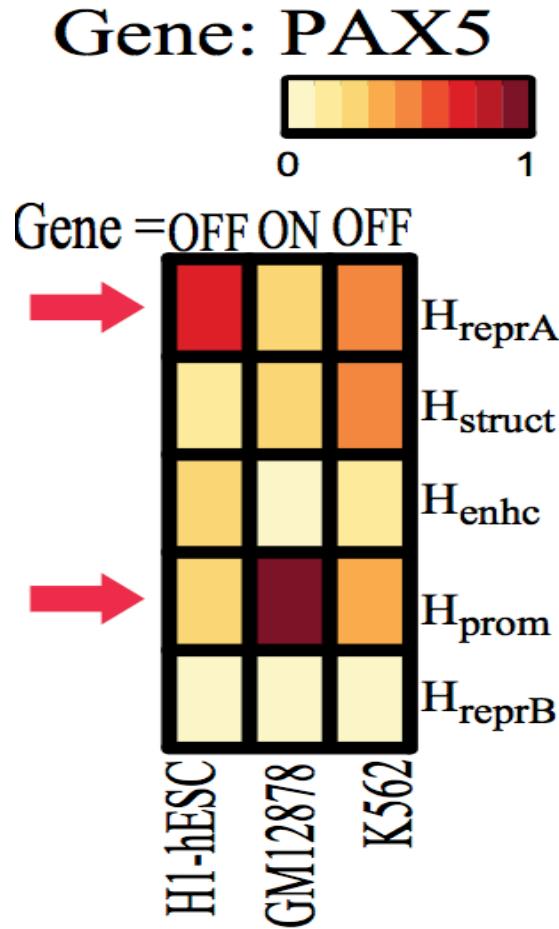


PROMOTER
DISTAL PROMOTER
REPRESSOR

Gene: PAX5



Results: HM level attention



- An important differentially regulated gene (PAX5) across three blood lineage cell types:
 - H1-hESC (stem cell),
 - GM12878 (blood cell),
 - K562 (leukemia cell).
- Trend of its global weights (beta) Verified through the literature.

β Maps

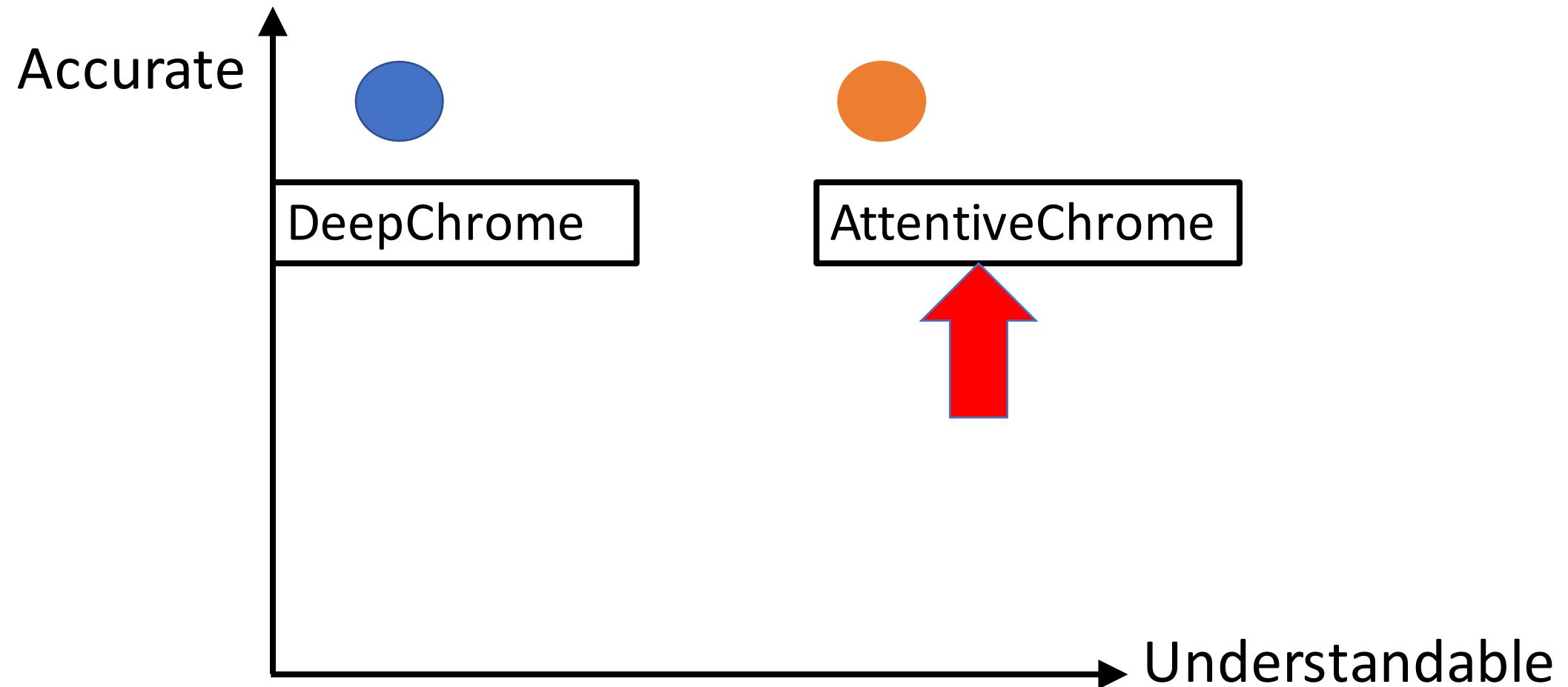
Validation of Attention Weights (using one extra HM signals)

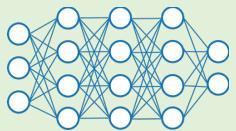
Table 3: Pearson Correlation values between weights assigned for H_{prom} (active HM) by different visualization techniques and H_{active} read coverage (indicating actual activity near "ON" genes) for predicted "ON" genes across three major cell types.

Viz. Methods	H1-hESC	GM12878	K562
α Map (LSTM- α)	0.8523	0.8827	0.9147
α Map (LSTM- α, β)	0.8995	0.8456	0.9027
Class-based Optimization (CNN)	0.0562	0.1741	0.1116
Saliency Map (CNN)	0.1822	-0.1421	0.2238

- Additional signal - H3K27ac (H-Active) from REMC
- Average local attention weights of gene=ON correspond well with H-active
- Indicating AttentiveChrome is focusing on the correct bin positions

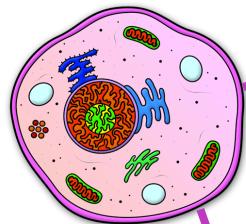
Summary of tools





Where are we heading?

Changing Task : Classification → Regression

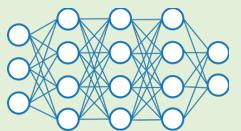


1.770
Gene
Expression

Genes	Gene Expression (RPKM)	Y $\log(\text{RPKM})$
RUNX1	1.296	01126
SMAD2	14.902	1.1737
MYC	3.805	0.5803
PAX5	15.066	1.779
.....

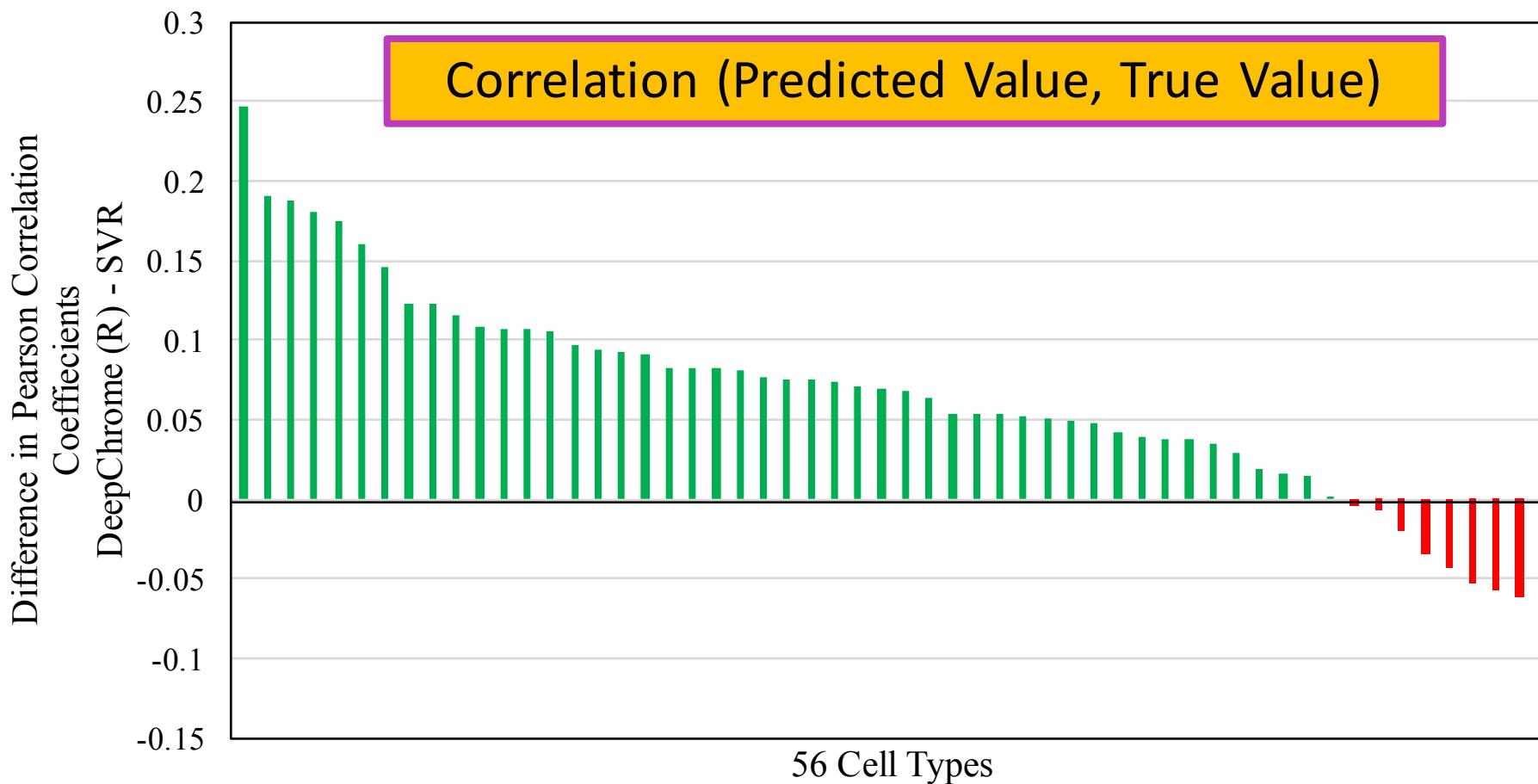
Mean Square Error
Loss

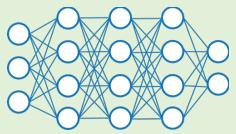
$$(Y - f(X))^2$$



Where are we heading?

Changing Task : Classification → Regression

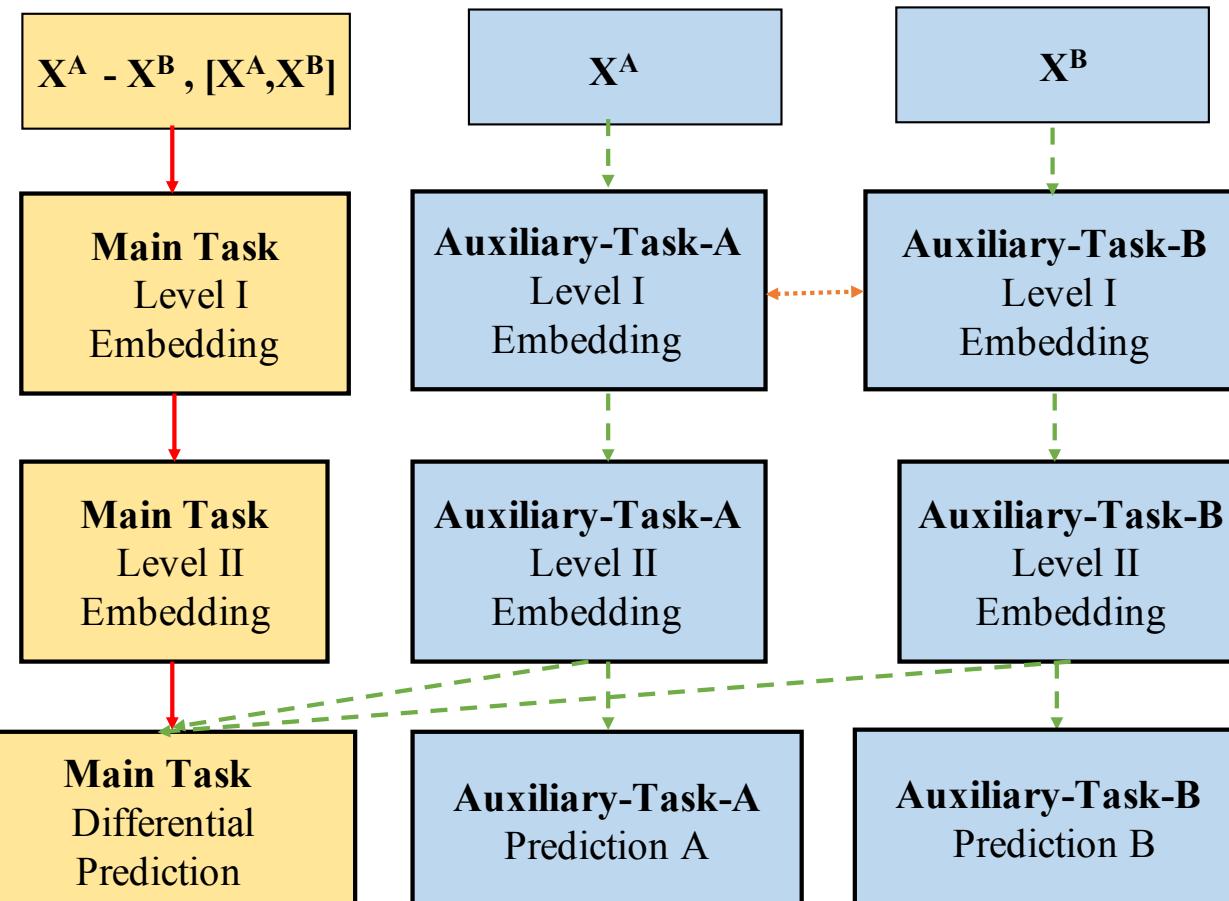




Where are we heading?

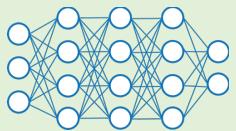
DeepDiff: Deep-learning for predicting Differential gene expression from histone modifications

Changing Task : Cell-Specific → Cross Cell



- 1 Main Task: Differential gene expression prediction
- 2 Cell-Specific Auxiliary: Auxiliary-Task-A and Auxiliary-Task-B cell type specific prediction
- 3 Siamese Auxiliary: Siamese contrastive loss

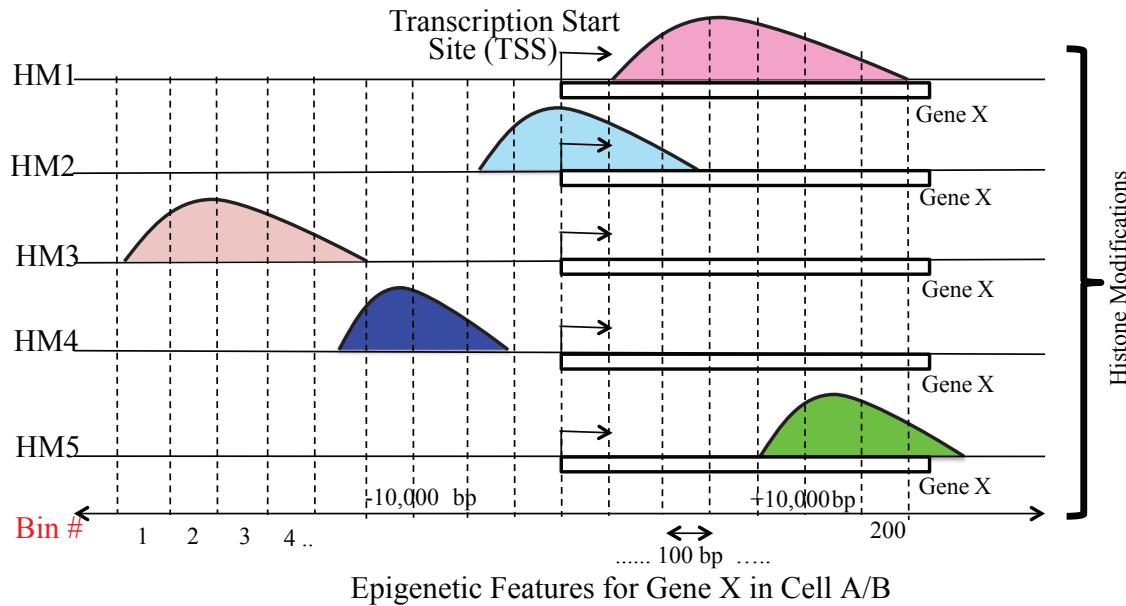
DeepDiff Variations	Objective Loss
1 Raw:d, Raw:c, Raw	ℓ_{Diff}
2 Aux	$\ell_{\text{Diff}} + \ell_{\text{CellAux}}$
1 + 2 Raw+Aux	$\ell_{\text{Diff}} + \ell_{\text{CellAux}}$
2 + 3 Aux+Siamese	$\ell_{\text{Diff}} + \ell_{\text{CellAux}} + \ell_{\text{Siamese}}$
1 + 2 + 3 Raw+Aux+Siamese	$\ell_{\text{Diff}} + \ell_{\text{CellAux}} + \ell_{\text{Siamese}}$



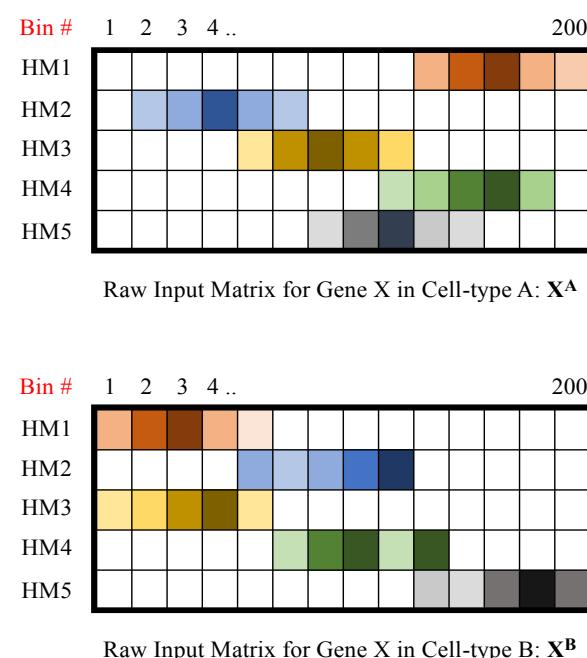
Where are we heading?

DeepDiff: Deep-learning for predicting Differential gene expression from histone modifications

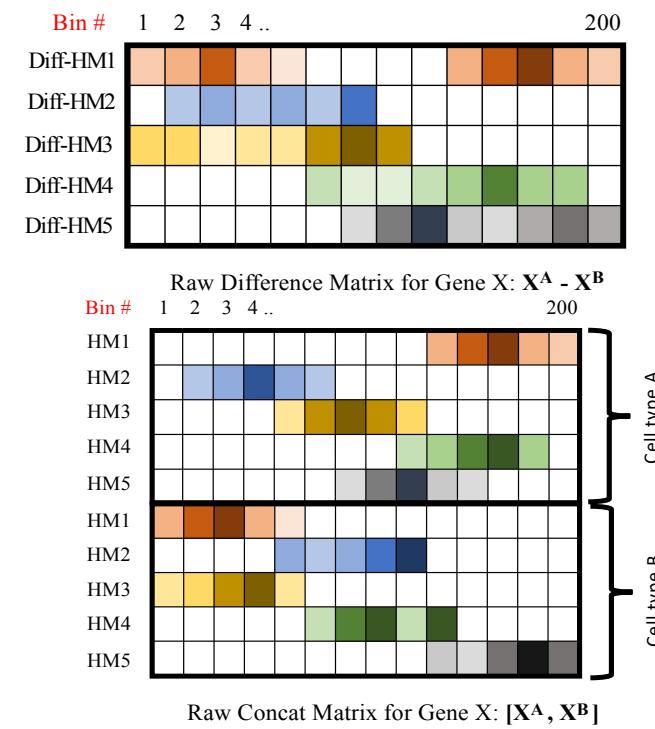
Changing Task : Cell-Specific → Cross Cell



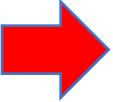
(a)



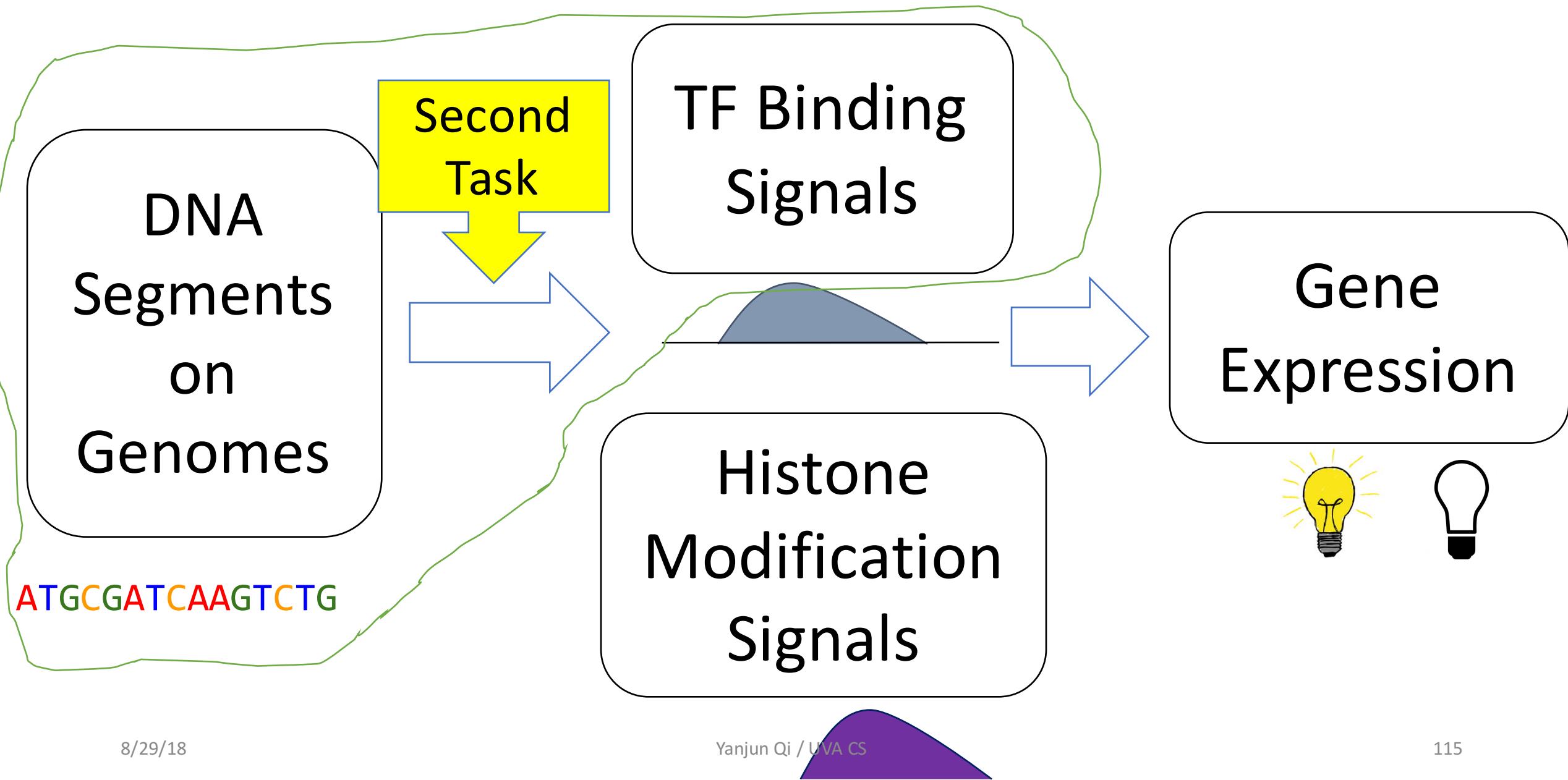
(b)



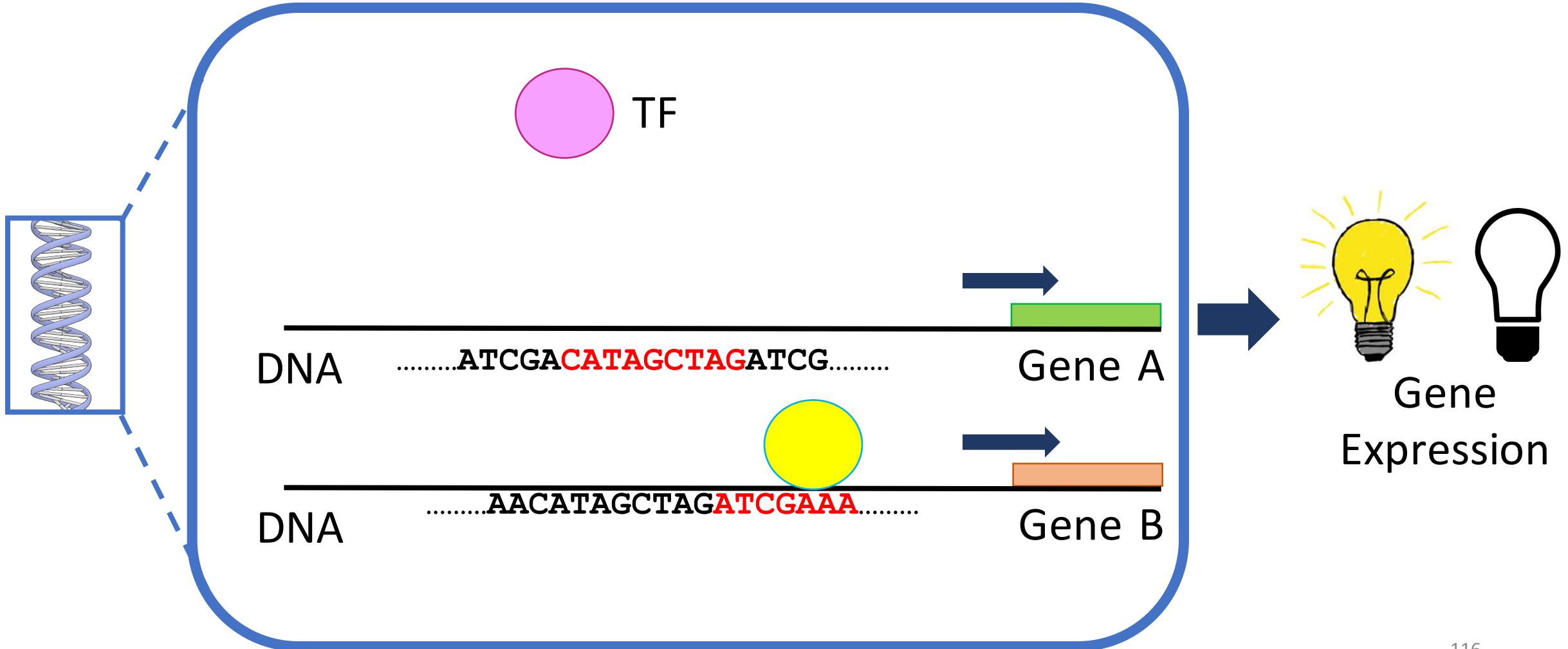
Today

- Machine Learning: a quick review
 - Deep Learning: a quick review
 - Background Biology: a quick review
 - Deep Learning for analyzing **Sequential Data** about Regulation:
 - DeepChrome
 - AttentiveChrome
 - DeepMotif
- <https://qdata.github.io/deep2Read/>
-  <https://www.deepchrome.org>

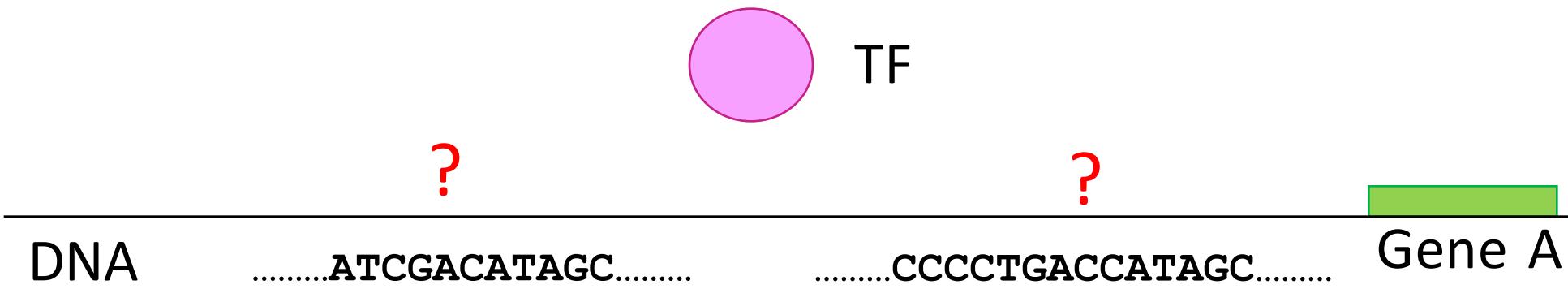
Many Important Data-Driven Computational Tasks



Transcription Factors



TF-Binding Site?



“TF-Binding Site?”

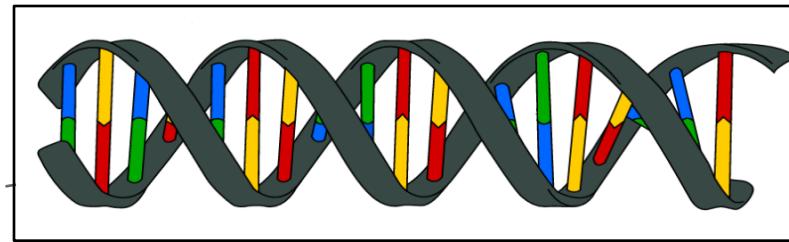
ATCGAATCCG



CCCTCTATCG



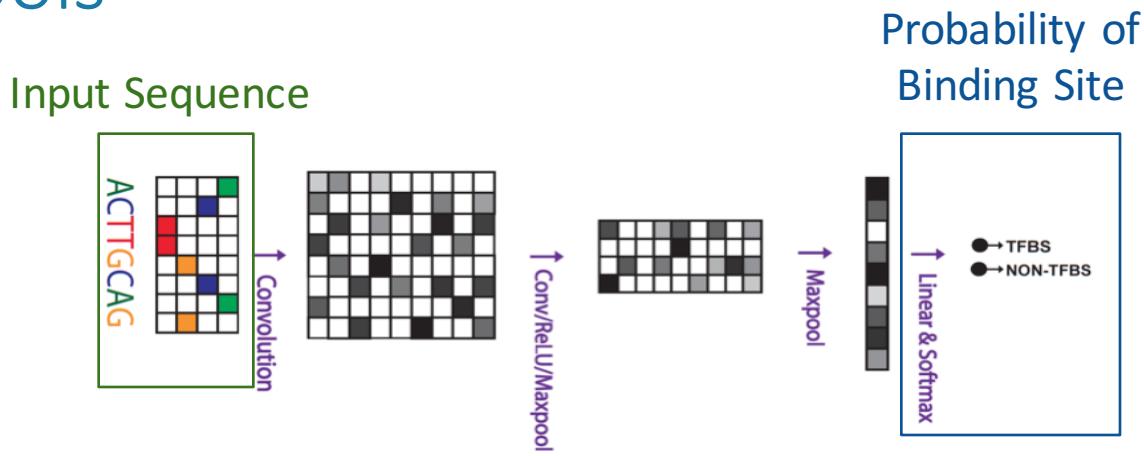
Task: Sequence Based Functional Annotation Tasks



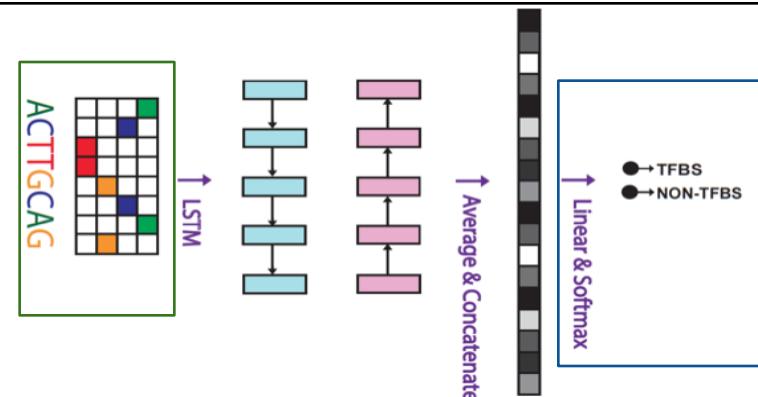
TF1 TF2 TF3 TF1 TF2 TF1 TF3 TF2
...**GCGACGAATCG...** **AACGATATGCT...** **CATATCATTTC...** **TGTCAAG...** **CTCGAGTC...** **TATCAAAGCTG...**

Literature: Various DNN Tools

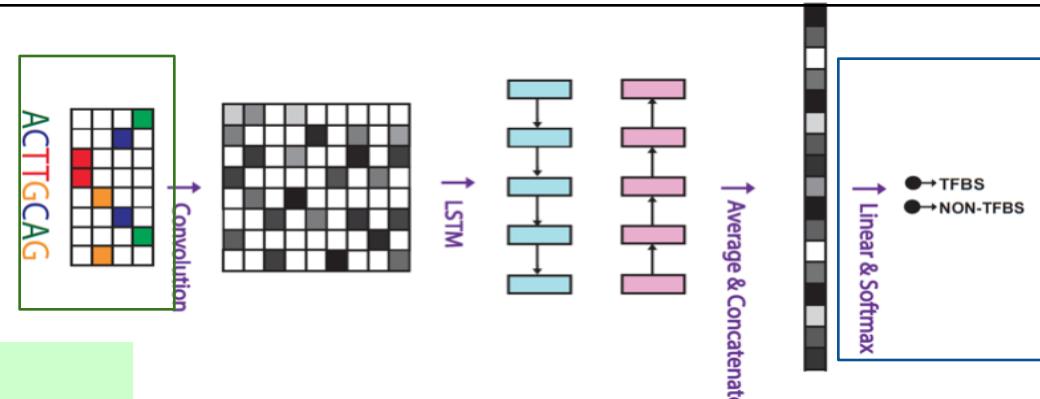
1. Convolutional (CNN)



2. Recurrent (RNN)



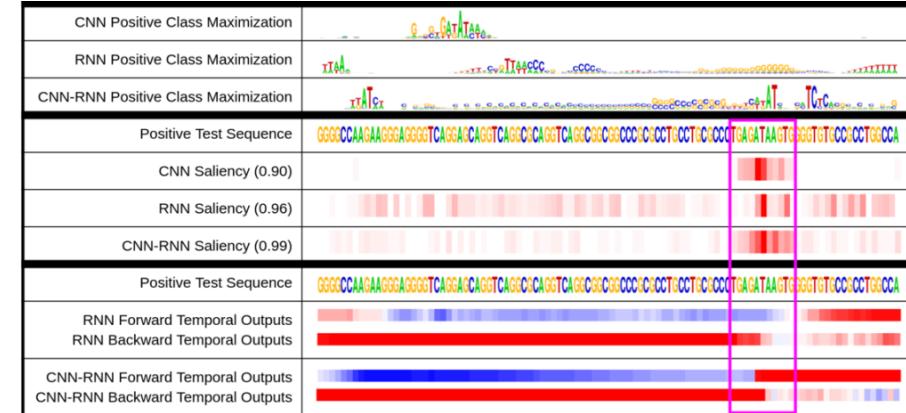
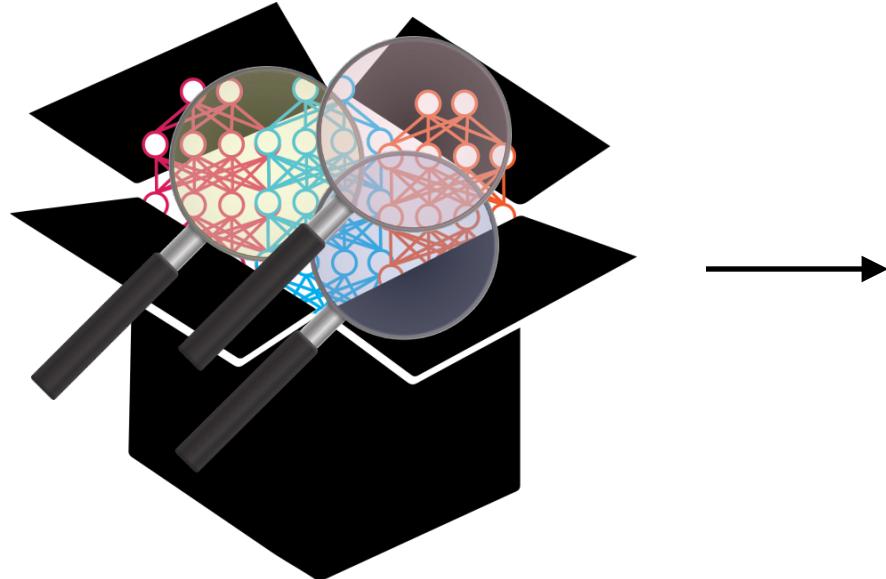
3. Convolutional-Recurrent (CNN-RNN)



DeepSEA, DeepBind, BASSET, DanQ,

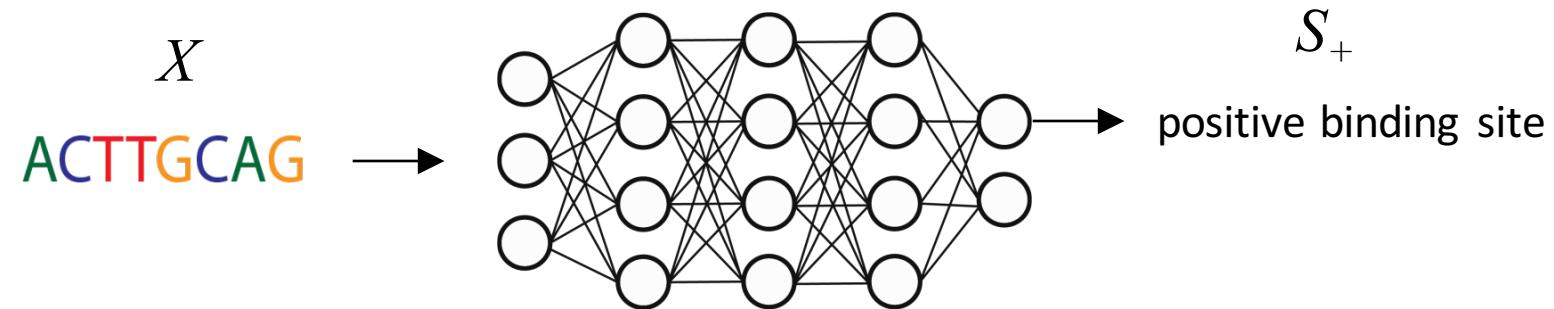
Deep Motif Dashboard: Understand DNNs by Post Analysis

Lanchantin, Singh, Wang & Qi - Pacific Symposium on Biocomputing, 2017



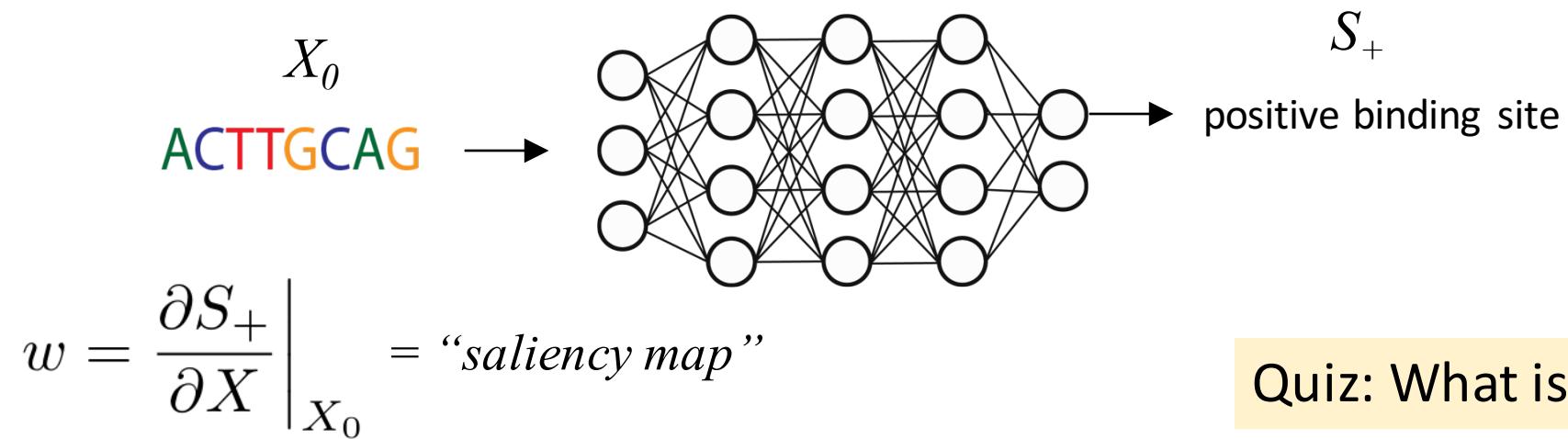
1. Saliency Maps - recommending on CNN kind
2. Temporal Output Values - recommending on RNN kind
3. Class Optimization - recommending on CNN kind

1. Saliency Map



Which nucleotides are most important for my current-sample classification?

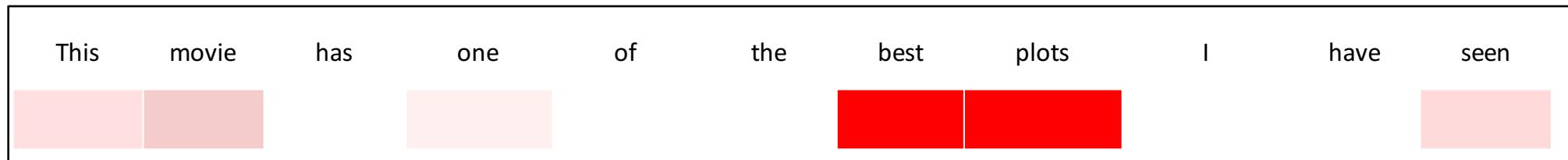
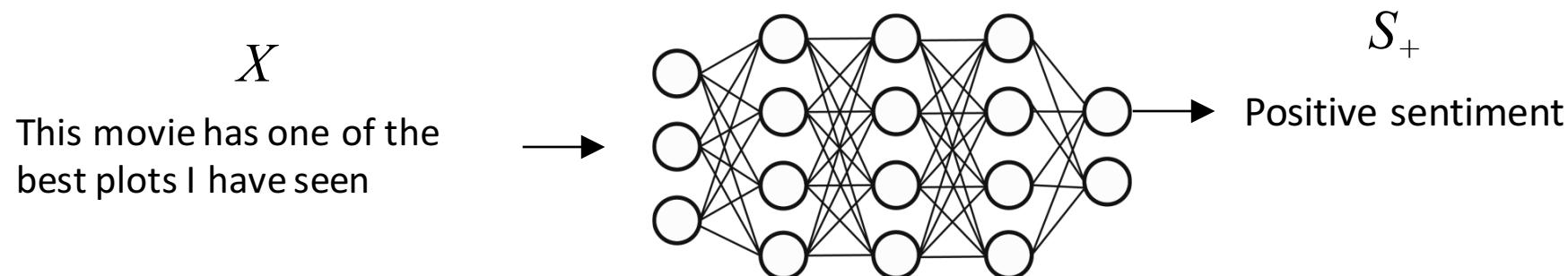
1. Saliency Map



Quiz: What is gradient?

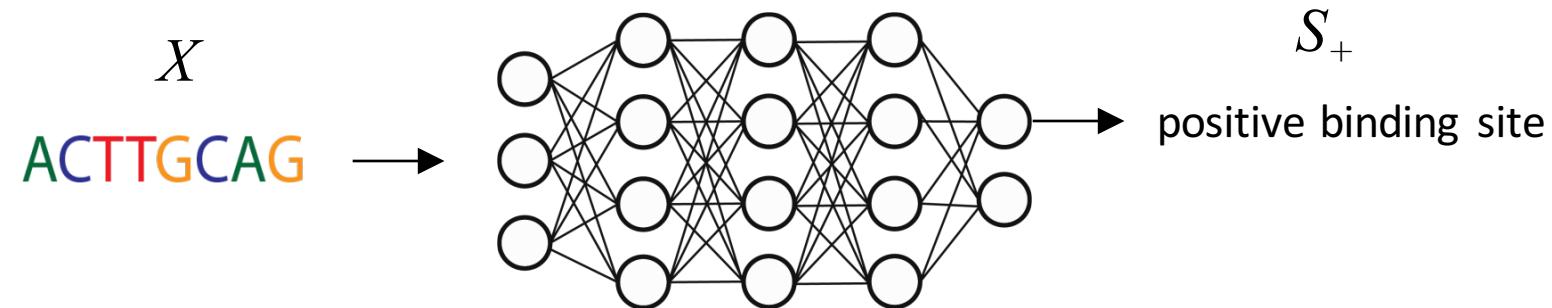
[Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR 2013](#)

1. Saliency Map



= important for classification

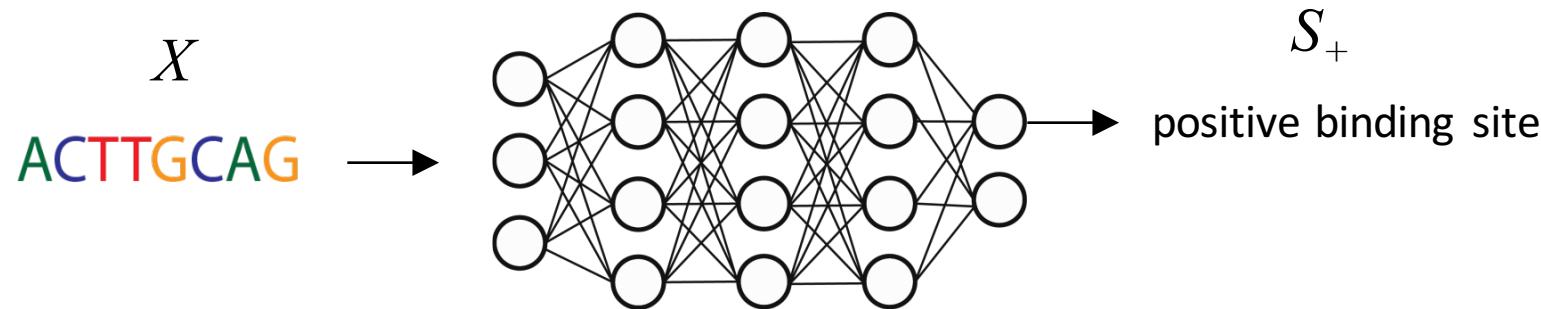
1. Saliency Map



Positive Test Sequence	TGCTCGCATCCTATTGGCCACGTTAGTCACATGGCCCCACCTGGCTGCAAAGCACGCTGGAAACGTAGTCCTTCTT
Saliency Map	A horizontal bar composed of vertical segments of varying heights. The segments are color-coded: most are light red, but there is a prominent dark red segment centered around the 14th position from the start of the sequence.

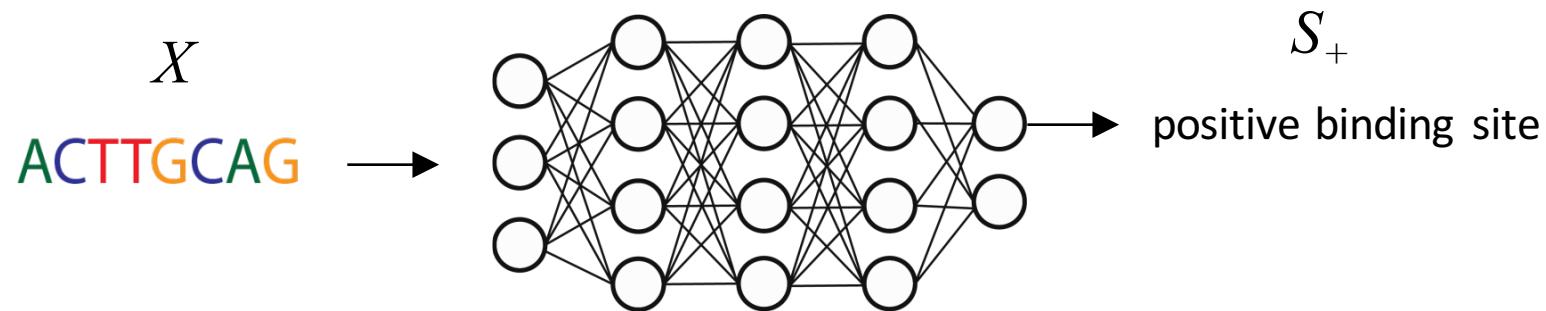
█ = important nucleotide for prediction

2. Temporal Output Values



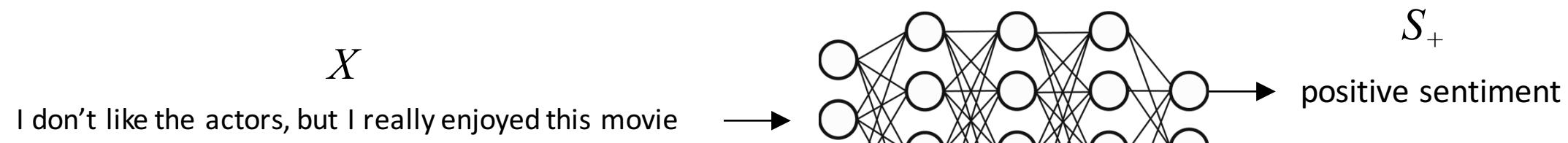
What are the model's predictions at each timestep of the DNA sequence?

2. Temporal Output Values



Check the RNN's prediction scores when we vary the input of the RNN starting from the beginning to the end of a sequence.

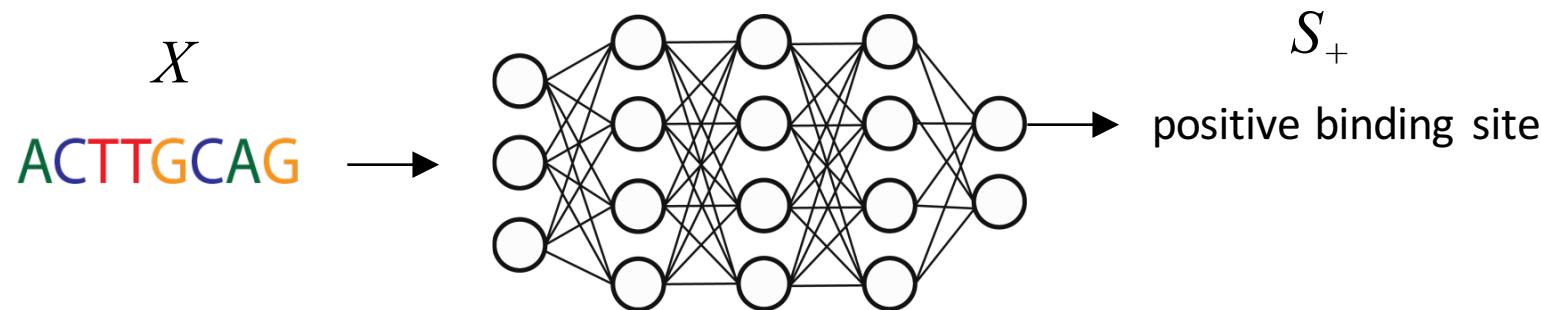
2. Temporal Output Values



 = negative sentiment

 = positive sentiment

2. Temporal Output Values

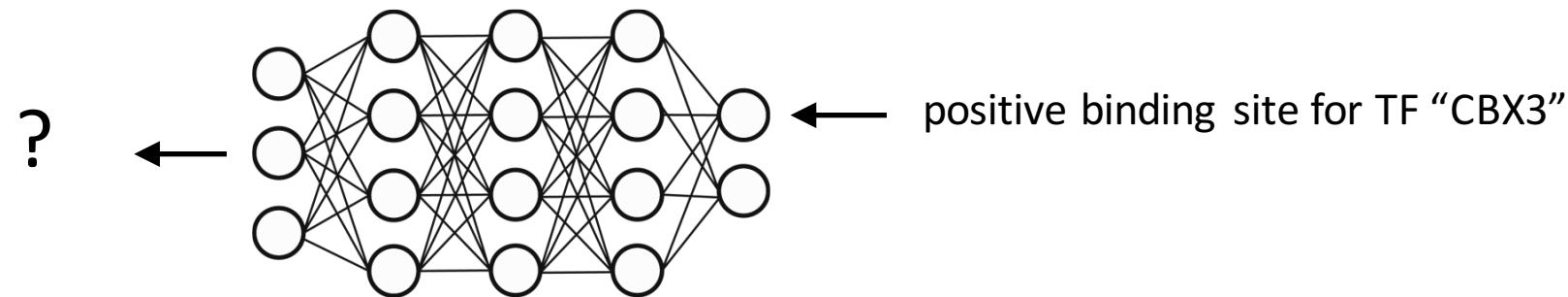


Positive Test Sequence	CTT TGCTCGCATCCTATTGCCACGTTAGTCACATGGCCCCACCTGGCTGCAAAGCACGCTGGAAACGTAGTC TTTCTT
RNN Forward Output	
RNN Backward Output	

= negative binding site prediction

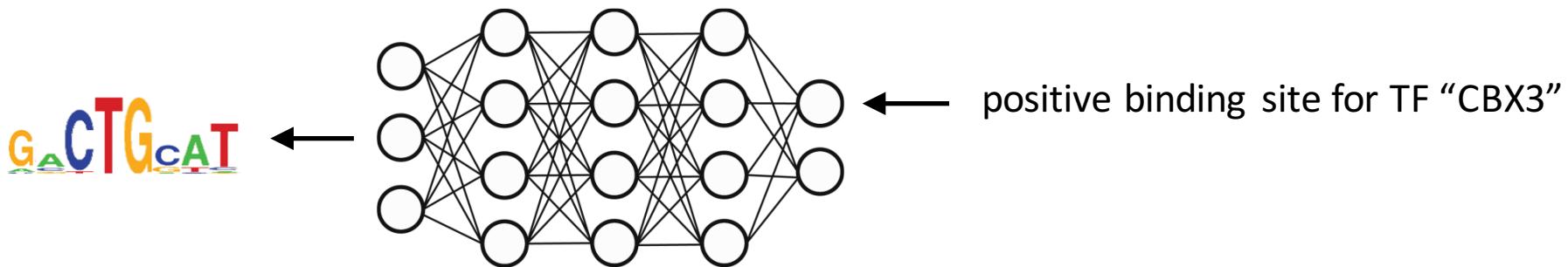
= positive binding site prediction

3. Class Optimization



For a particular TF, what does the optimal binding site sequence look like?

3. Class Optimization

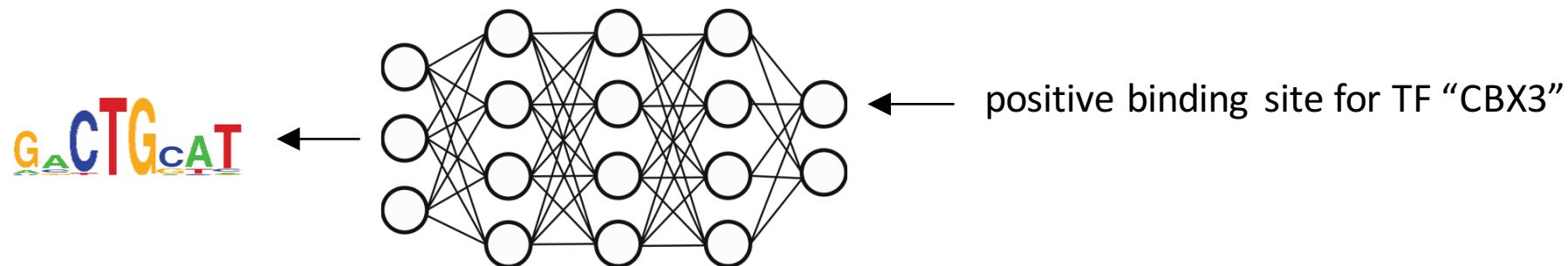


$$\arg \max_X S_+(X) + \lambda \|X\|_2^2$$

Where X is the input sequence and the score S_+ is probability of sequence X being a positive binding site

[Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR 2013](#)

3. Class Optimization



Optimal binding
site for TF “CBX3”



Visualization Methods

Sequence
Specific

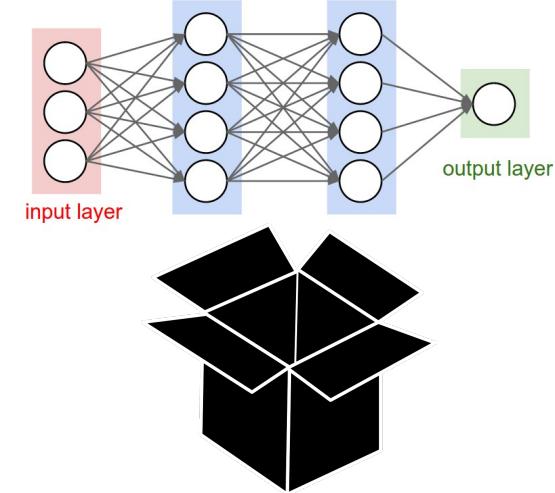
- {
 1. Saliency Maps – (CNN kind)
 2. Temporal Output Values – (RNN kind)

TF Specific

- {
 3. Class Optimization – (CNN kind)

code available at: deepmotif.org

Related Work to Post-Understand DNN



- Deconvolution
- Perturbation-based
- Backpropagation-based
- Difference to Reference
- Influence based

Temporal Output Values

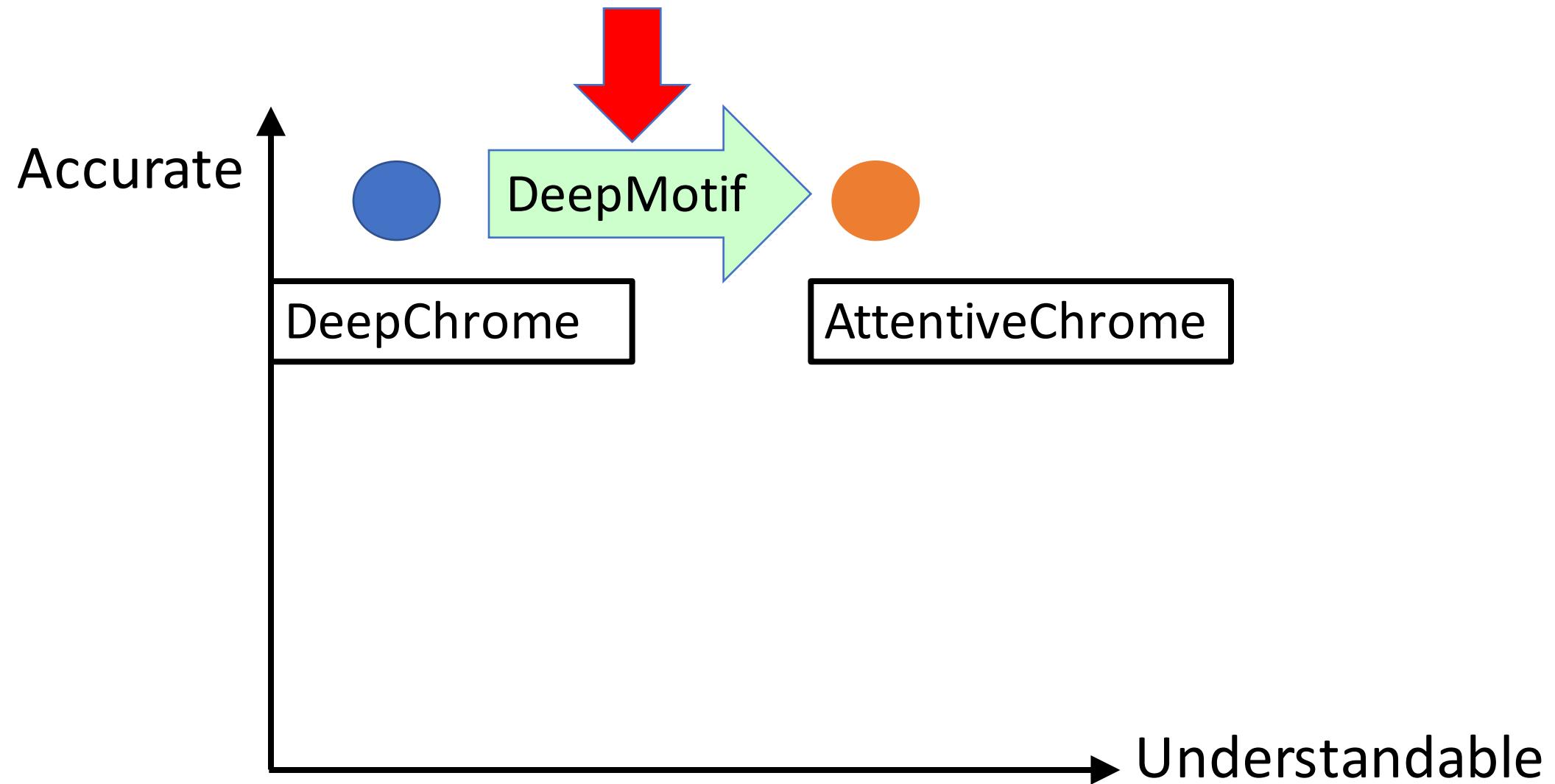
Saliency Map

Class Optimization

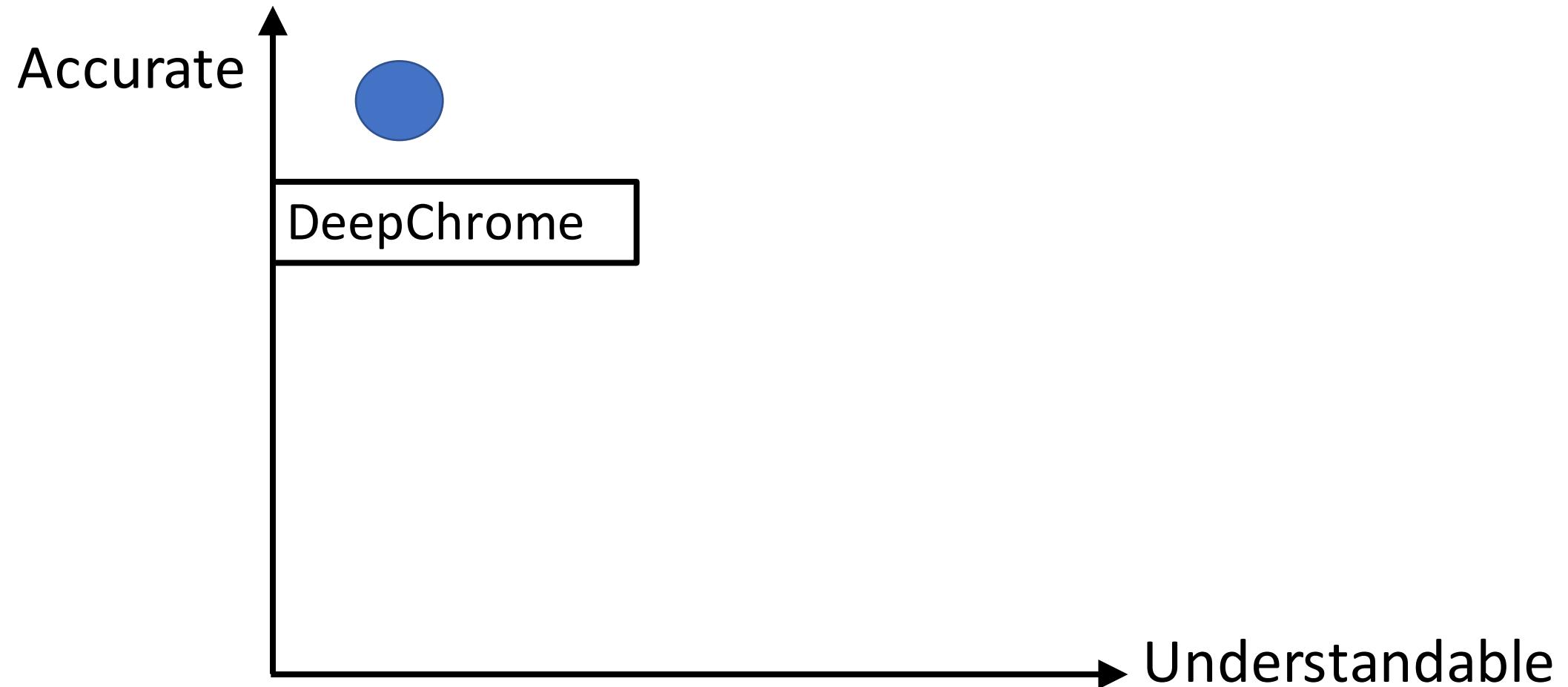
DeepLift

Influential Function / ICML27 Best Paper

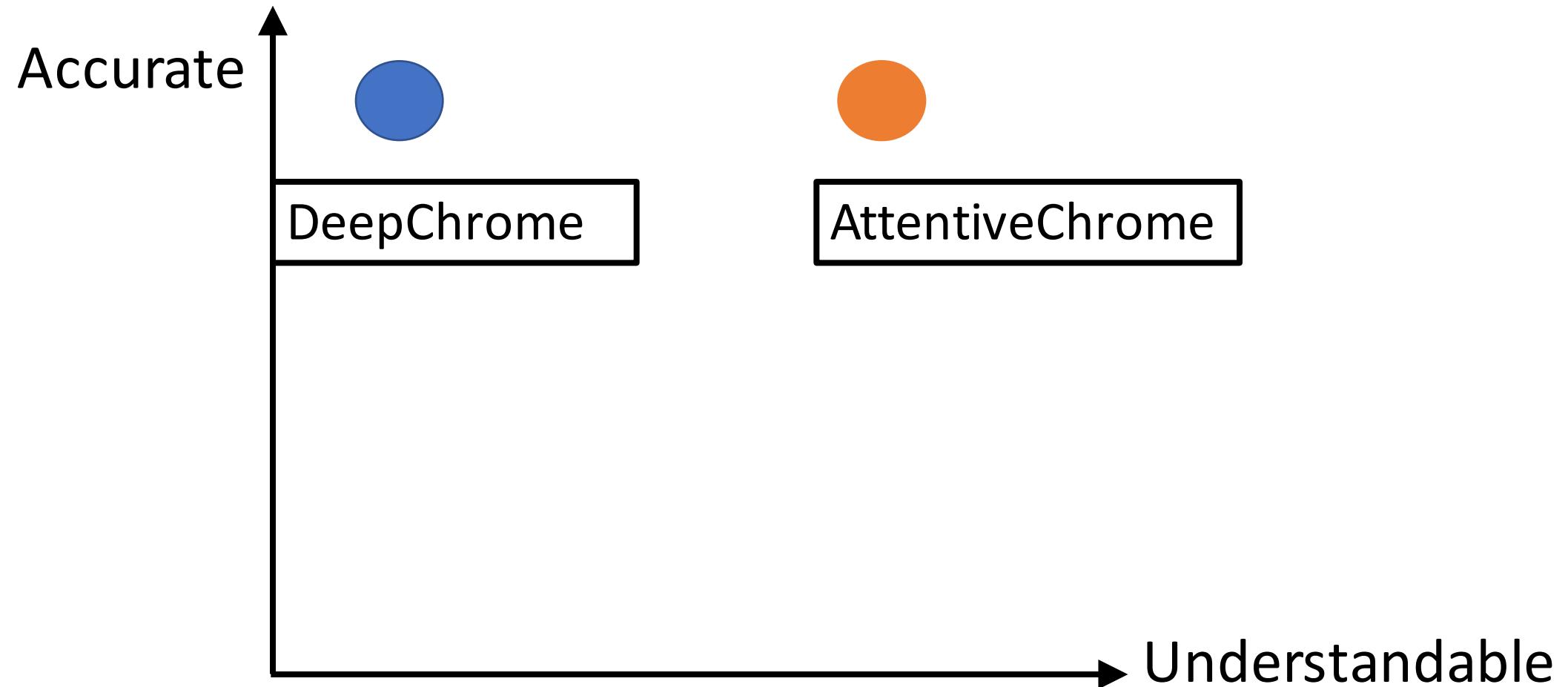
Summary of tools



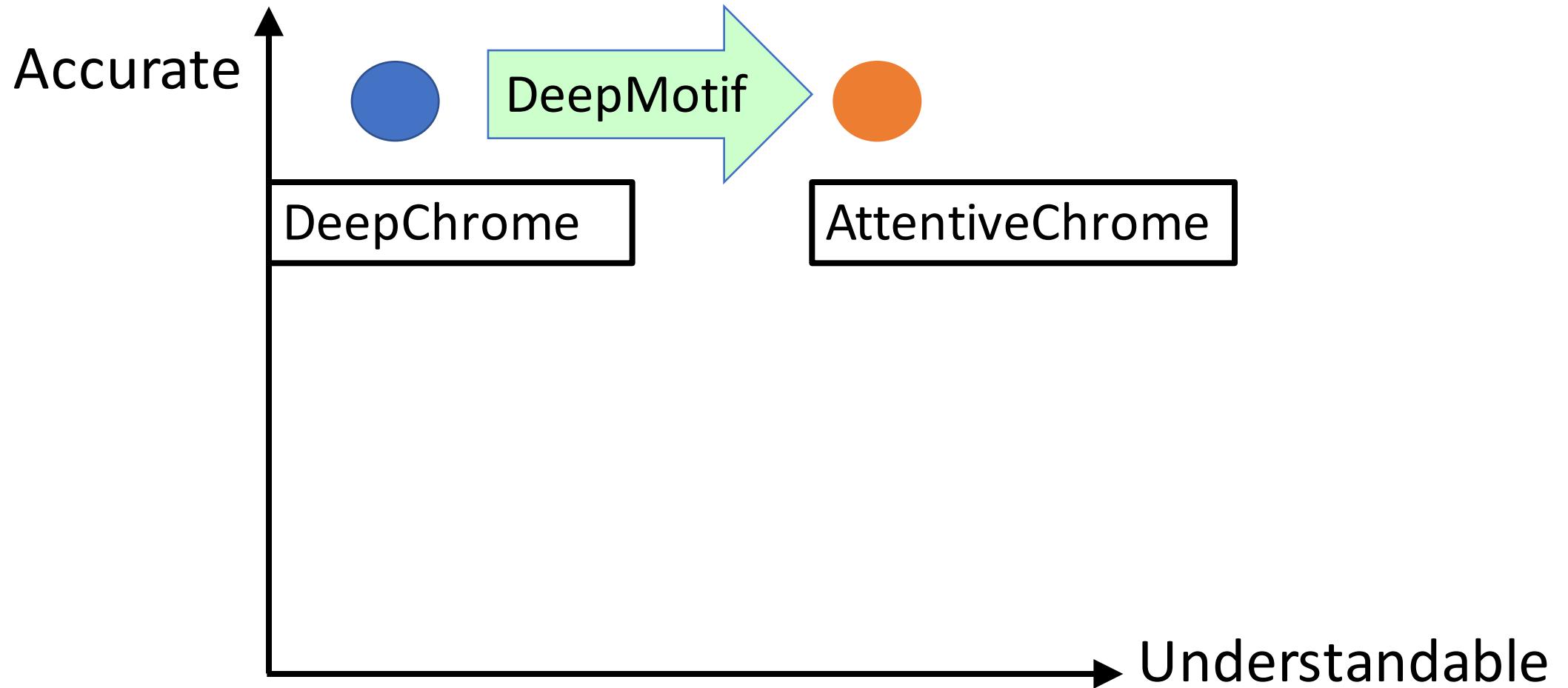
Recap



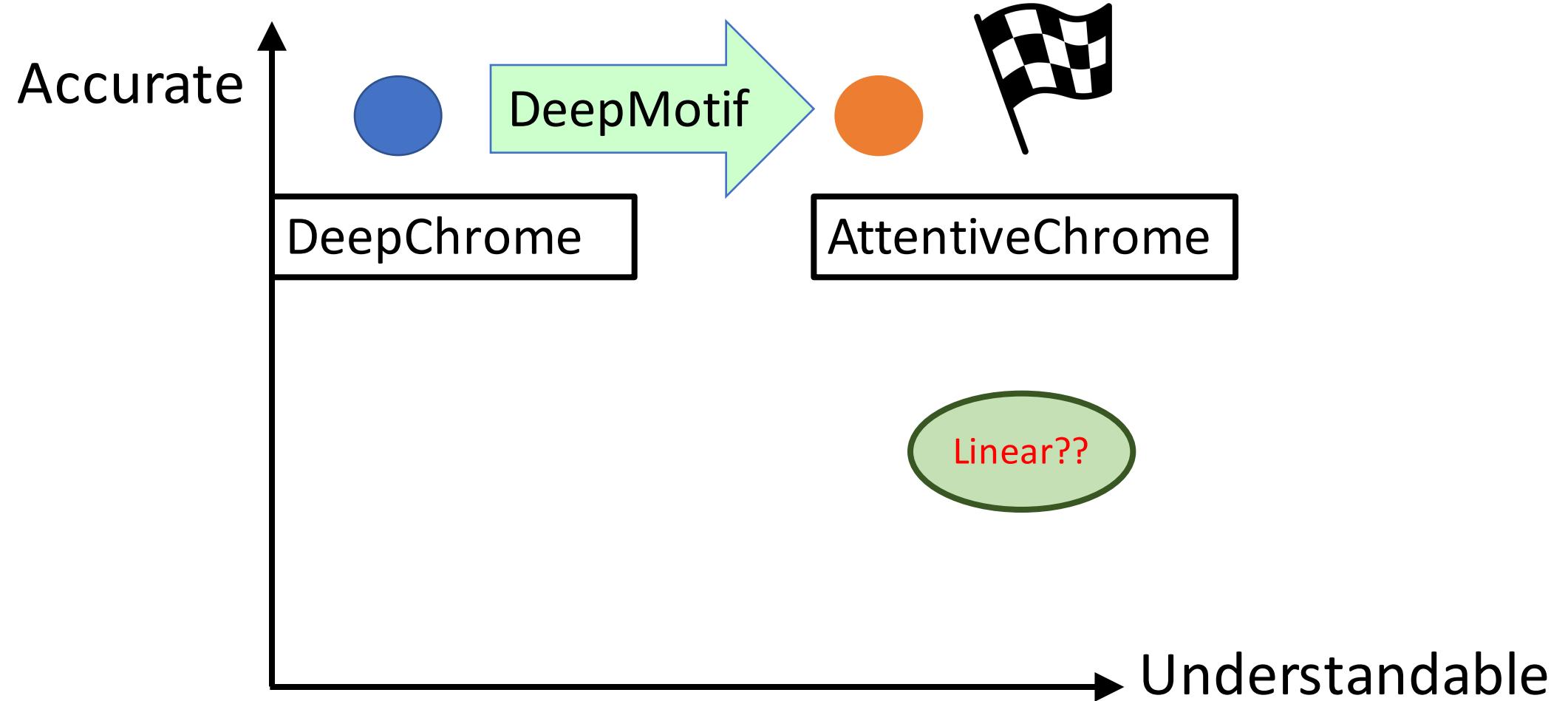
Recap



Recap



Recap



Acknowledgements



Ritambhara Singh



Jack Lanchantin



Arshdeep Sekhon



Beilun Wang

UVA Department of Biochemistry and Molecular Genetics: Dr. Mazhar Adli

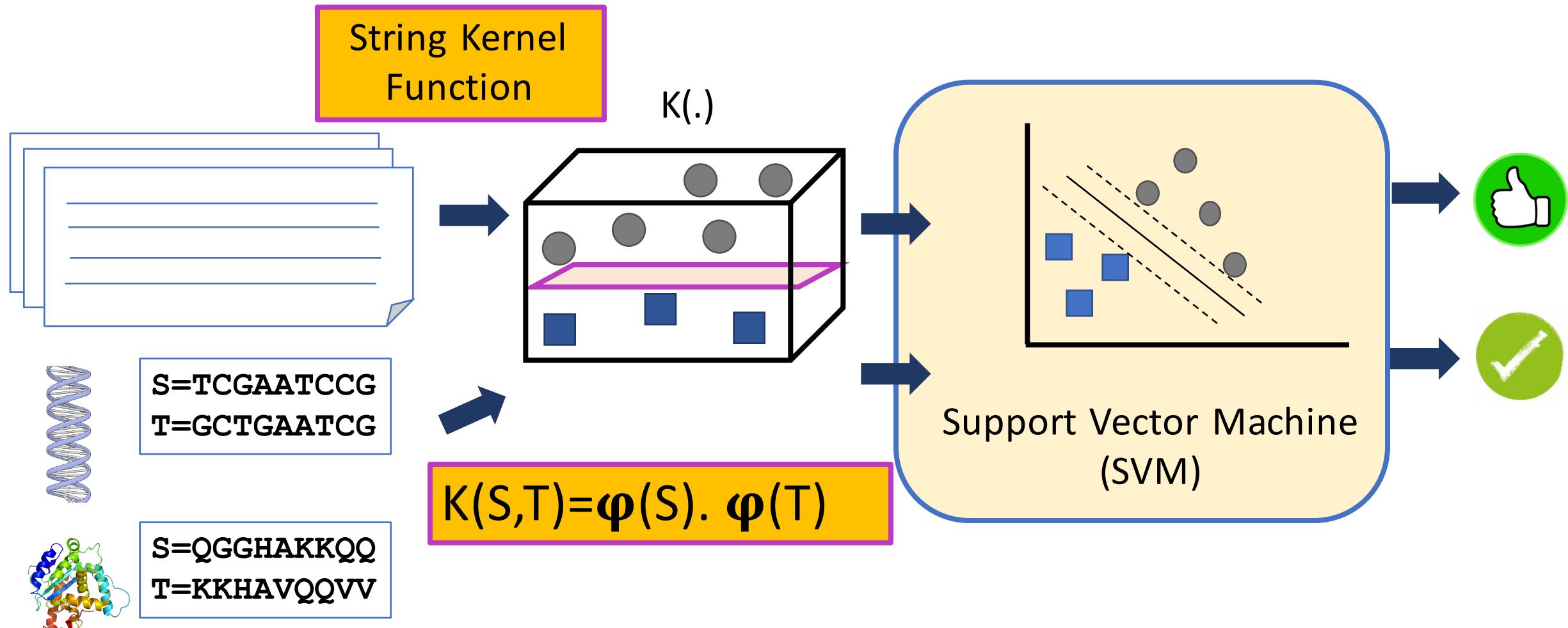
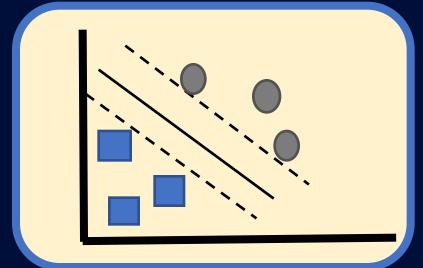


Thank you

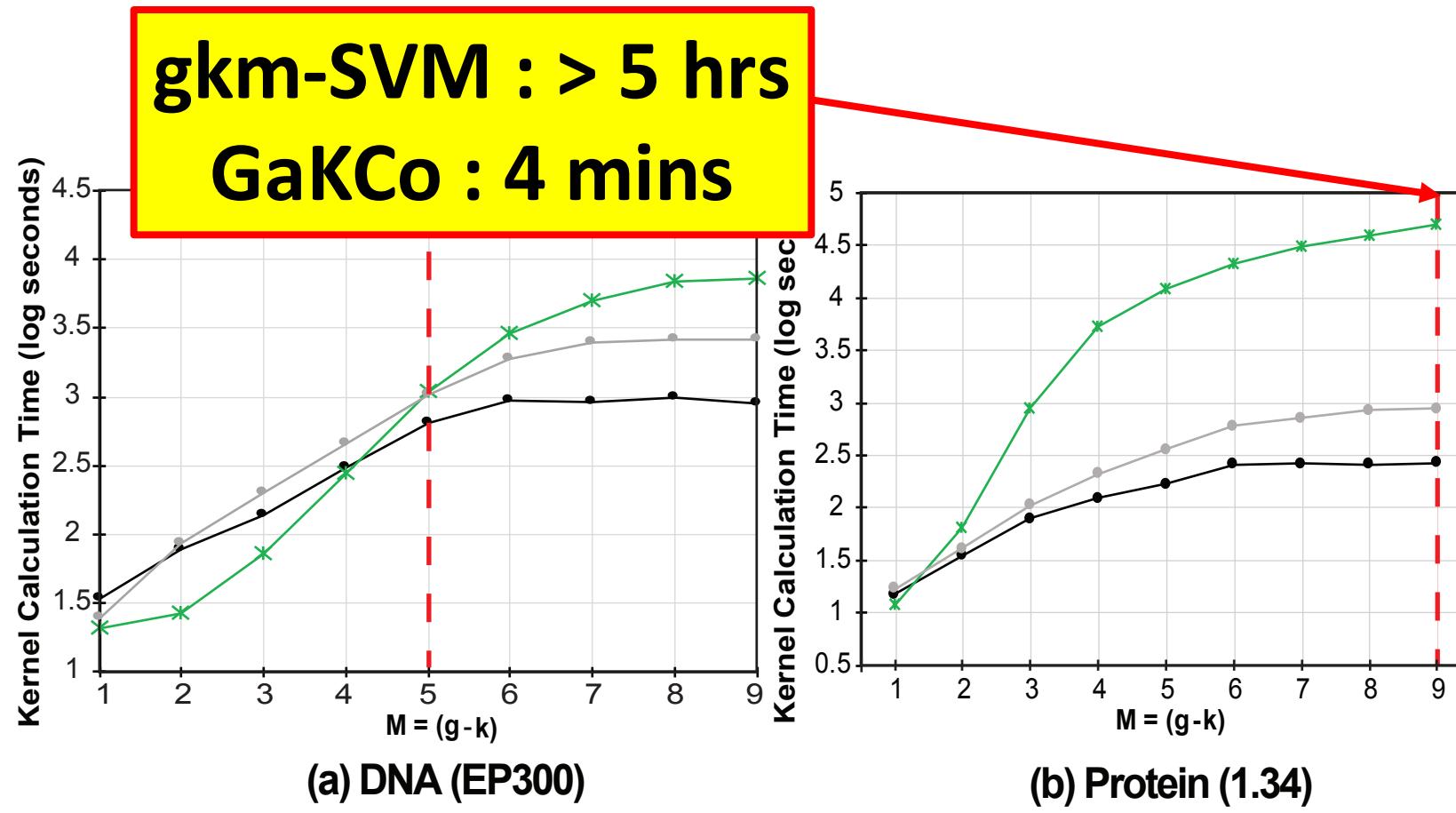
More Tools: A Fast and Scalable Tool to Classify Biological Sequences

<https://github.com/QData/iGakco-SVM>

String Kernel + SVM



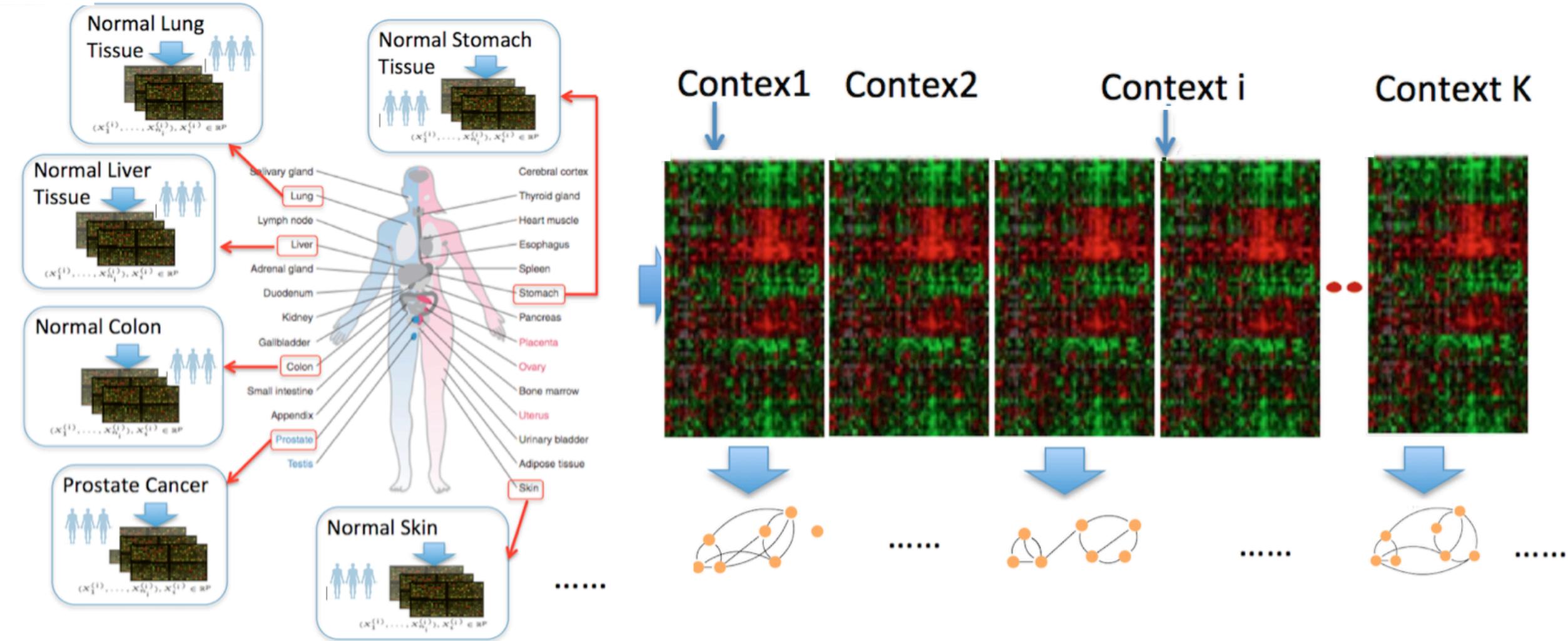
Our Tool Scales well with increasing Σ and m



More Tools: Extracting graphs from data

<https://www.jointggm.org>

Motivation: Graphs vary across contexts



Limitation of Previous Methods : Storage

e.g., calculate the gradient

$$\Sigma = \text{Cov}(X) =$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

$$\Sigma = \text{Cov}(X) =$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

$$\Sigma = \text{Cov}(X) =$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

When K contexts=91, p nodes= 30K

$O(Kp^2)$ in memory

Double type: 65 TB

Limitation of Previous Methods: Speed

Suppose they have the same iteration number T

$$K = 91, p = 30K$$

Traditional Optimization Method

---- Block Coordinate Descent :

$$O(K^3 p^4) / \text{Itera}$$

more than **2 billion years**

Current Optimization: ADMM based

---- Still needs SVD for each covariance matrix

SVD for the matrices needs

$$O(K p^3) \rightarrow 3.5 \text{ days} \\ / \text{Itera}$$

Our Tools

- Fast and scalable estimators for joint graph discovery from heterogeneous samples
- Parallelizable algorithms
- Sharp convergence rate (sharp error bounds)

More details at: <http://www.jointggm.org/>