



Making Deep Learning Interpretable for Analyzing Sequential Data about Gene Regulation

Dr. Yanjun Qi
Department of Computer Science
University of Virginia

@UCLA 2018 Computational Genomics Winter Institute

Today

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

<https://qdata.github.io/deep2Read/>
- Deep Learning Tools for analyzing **Sequential Data** about Regulation:
 - DeepChrome
 - AttentiveChrome
 - DeepMotif

<https://www.deepchrome.org>
- More Tools: learning graphs from data

<https://www.jointggm.org>

Today

• Deep Learning

- Basics
- History
- Why is this a breakthrough ?
- Recent trends

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

• Deep Learning Tools for analyzing Sequential Data about Regulation:

- DeepChrome
- AttentiveChrome
- DeepMotif

<https://www.deepchrome.org>

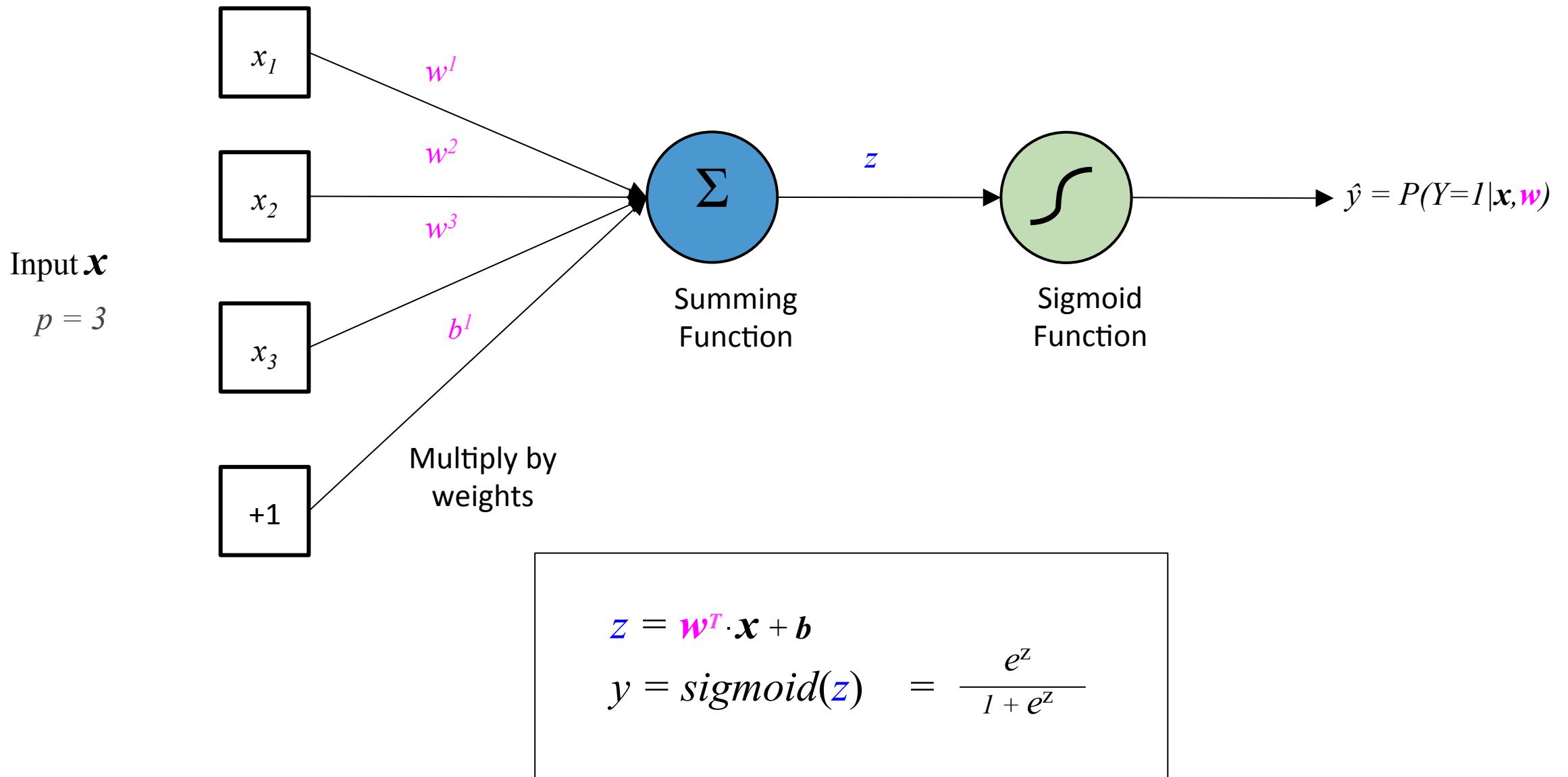
• More Tools: learning graphs from data

<https://www.jointggm.org>

Basics Summary

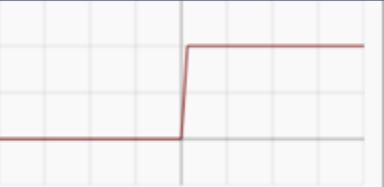
- Basic Neural Network (NN)
 - single neuron, e.g. logistic regression unit
 - multilayer perceptron (MLP)
 - various loss function
 - E.g., when for multi-class classification, softmax layer
 - training NN with backprop algorithm

One “Neuron”: Expanded Logistic Regression



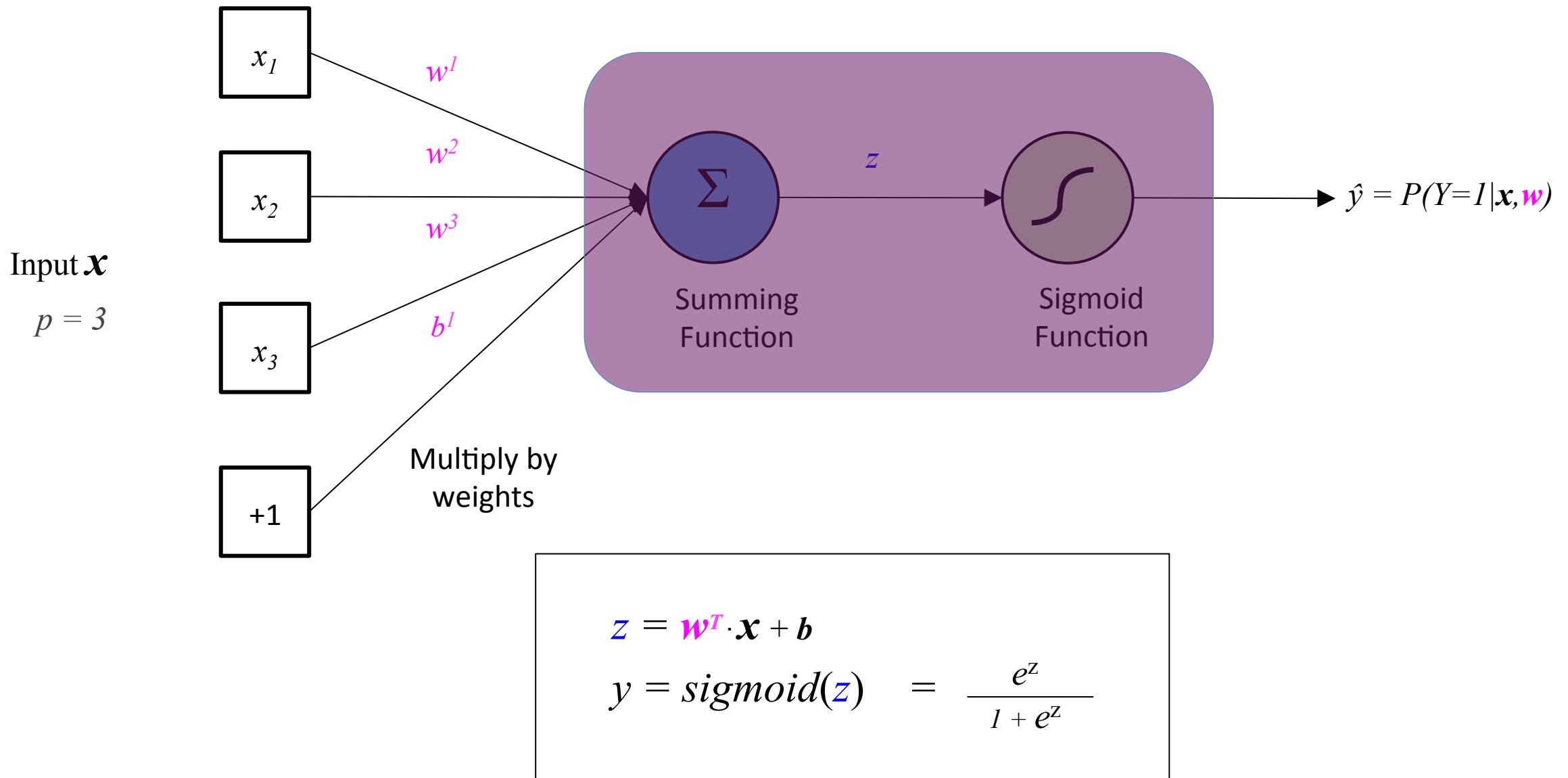
Nonlinearity Functions

(aka transfer or activation functions)

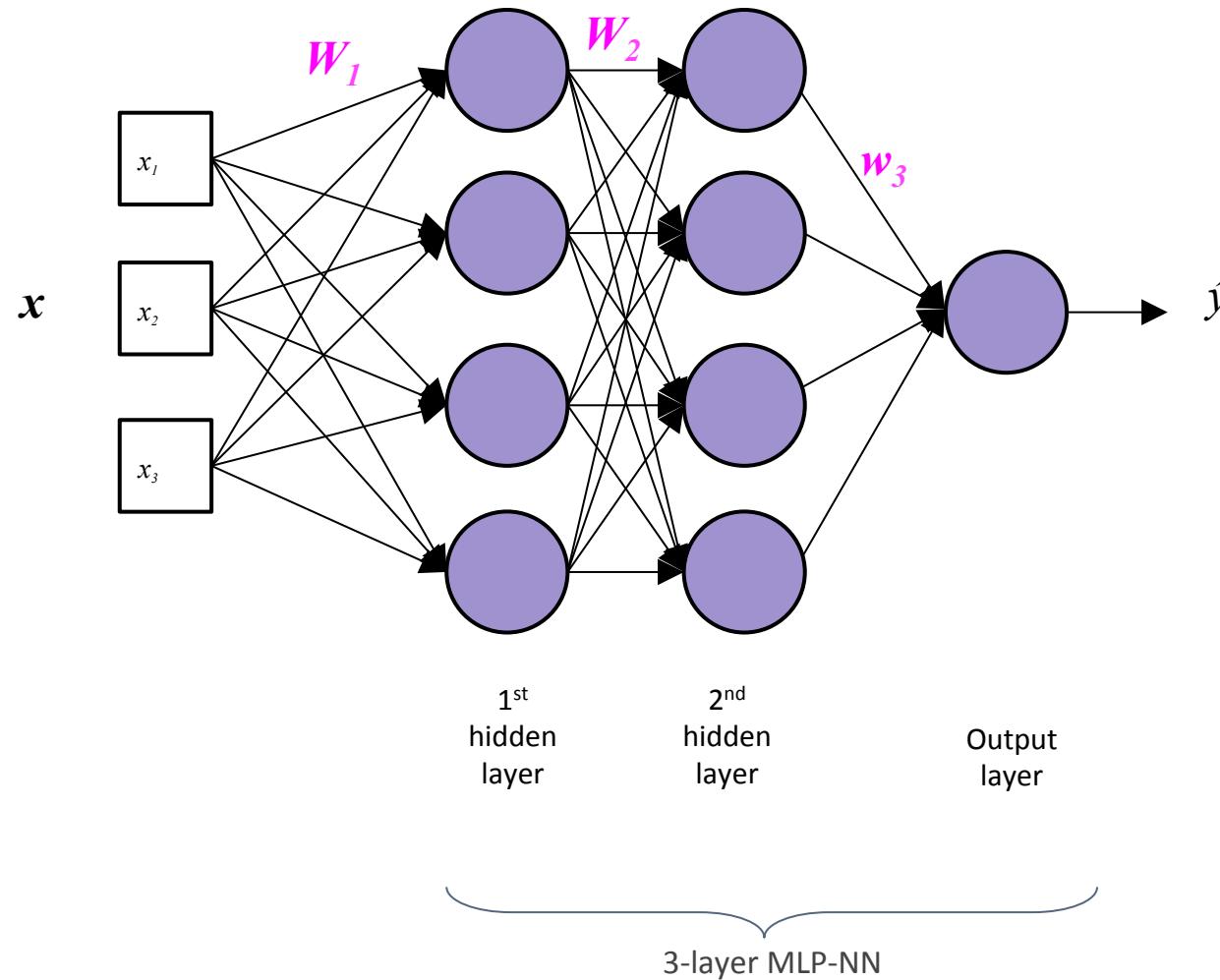
Name	Plot	Equation	Derivative (w.r.t x)
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$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

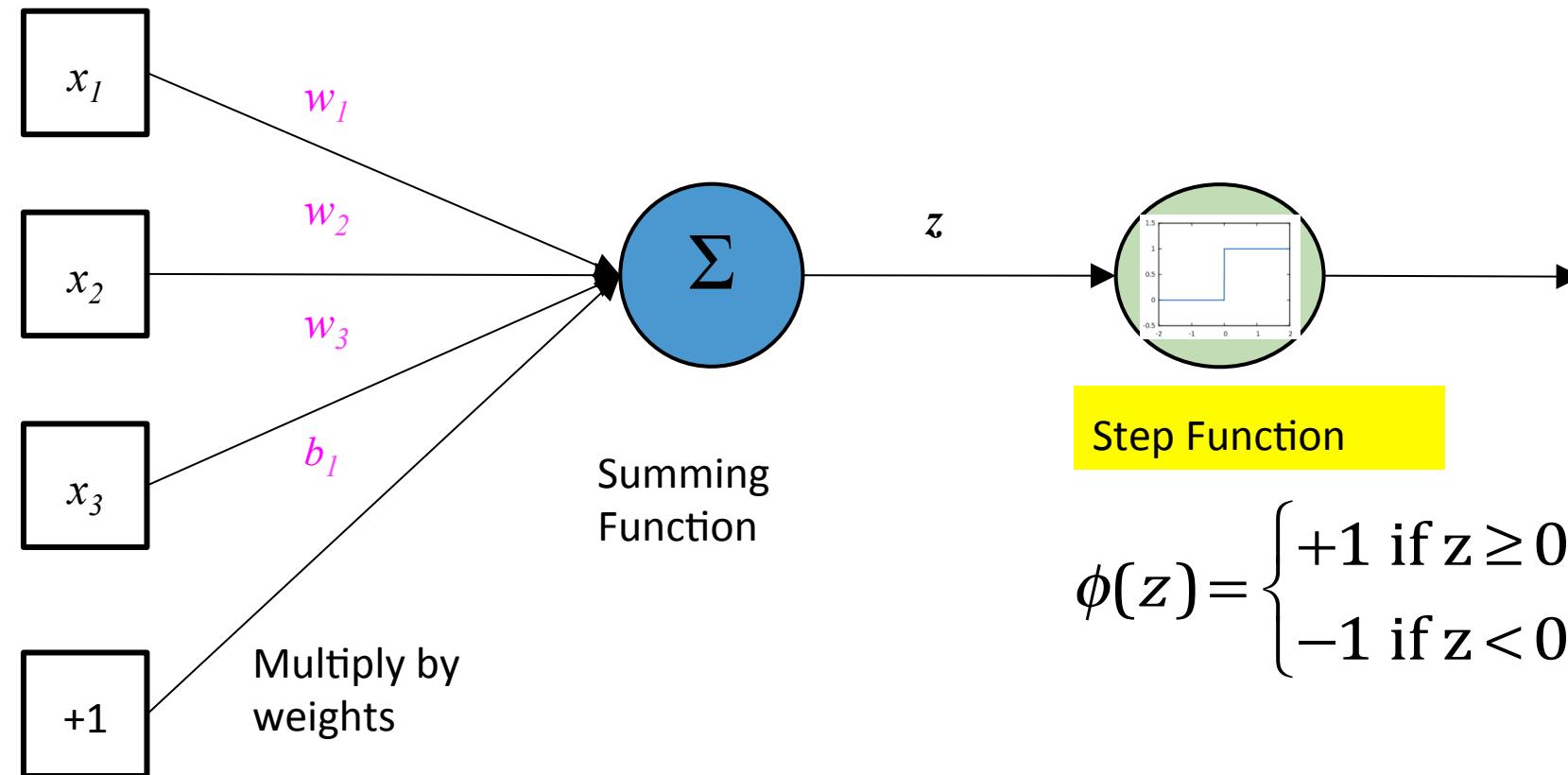


Multi-Layer Neural Network (MLP)

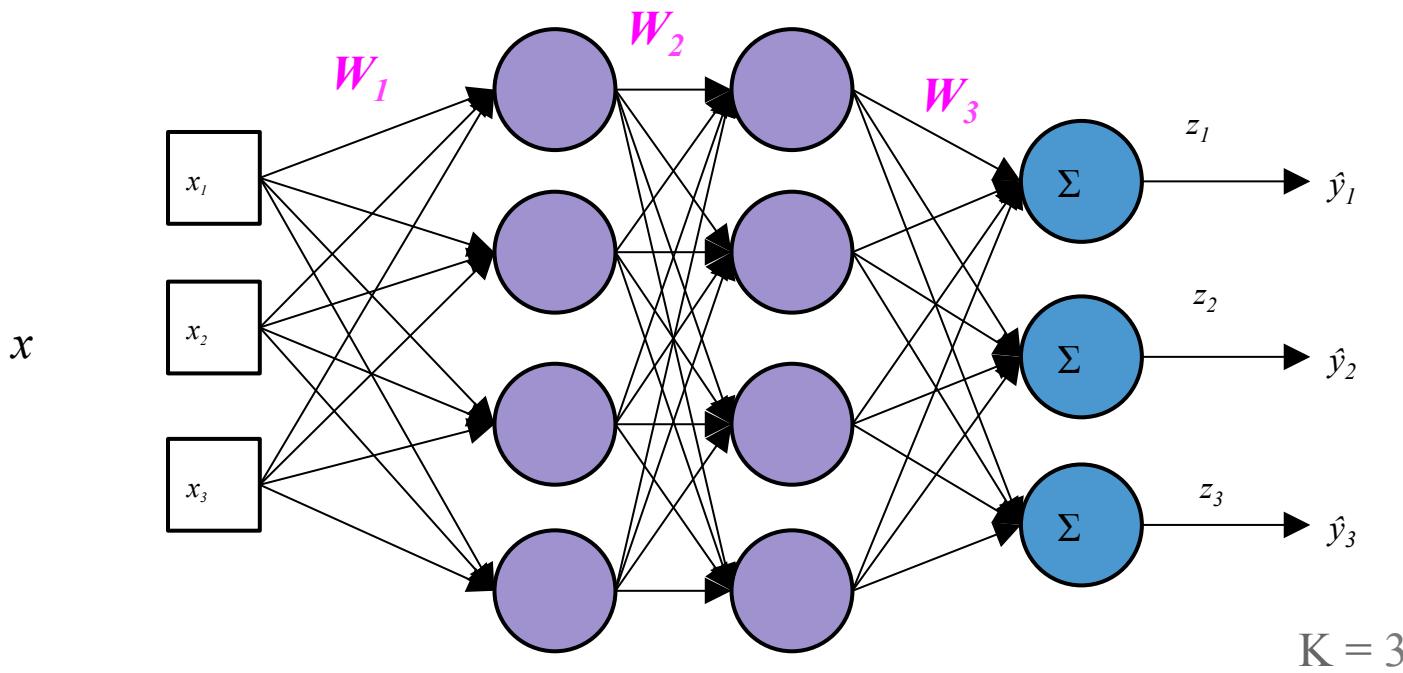


History → Perceptron: 1-Neuron Unit with Step

- 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**

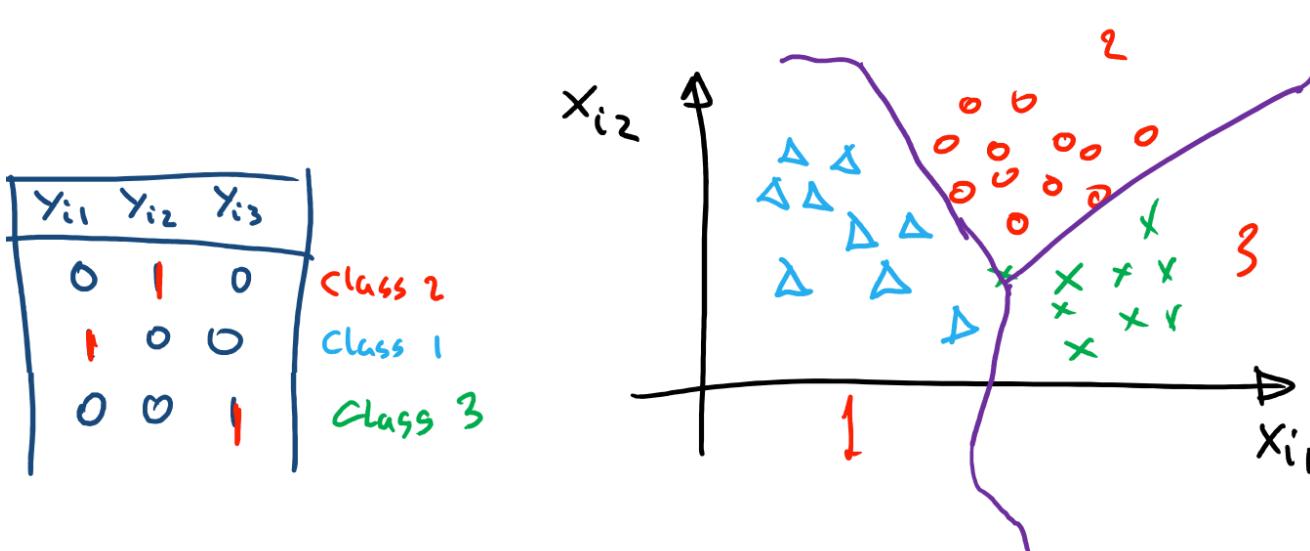


When for Multi-Class Classification



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

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



$$E(\hat{y}, y) = \text{loss} = - \sum_{j=1..K} y_j \ln \hat{y}_j$$

Cross-entropy loss

Training Neural Networks

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

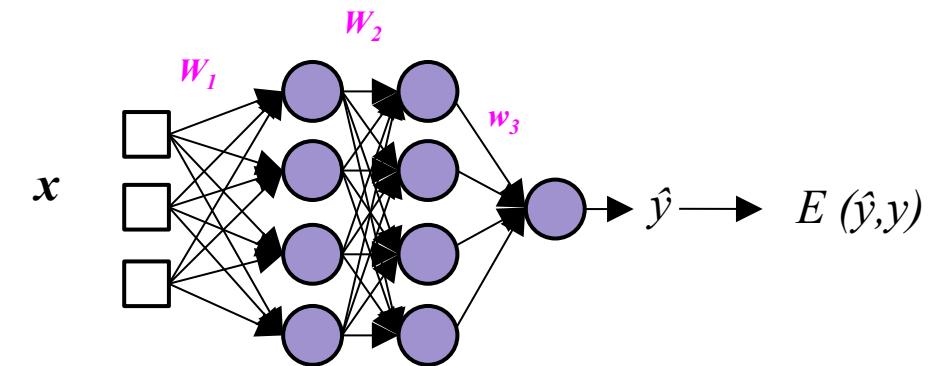
- **Stochastic Gradient descent:**

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

But how do we get gradients of lower layers?

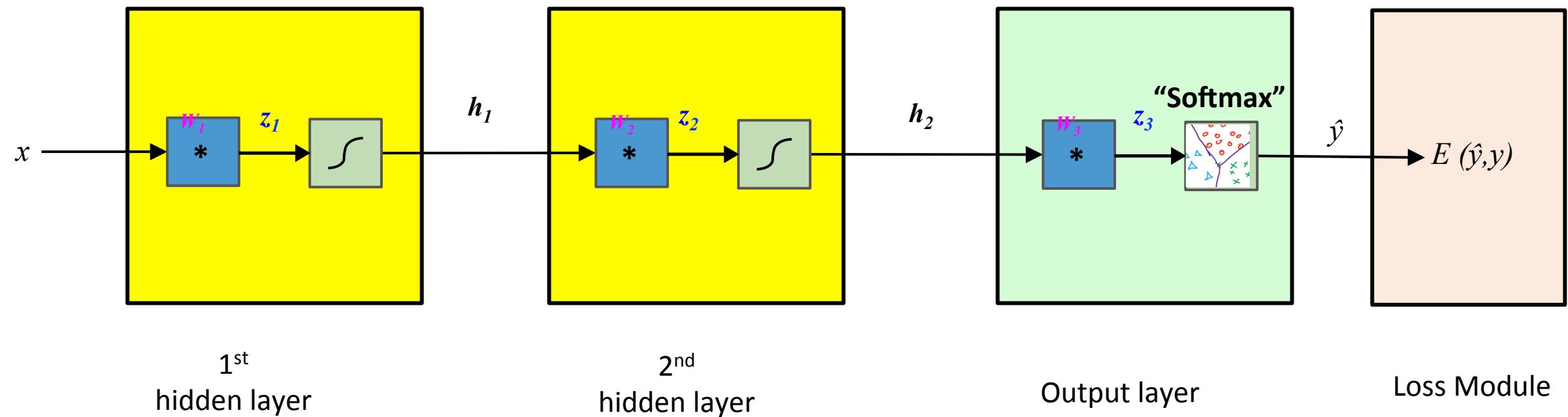
- **Backpropagation!**

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

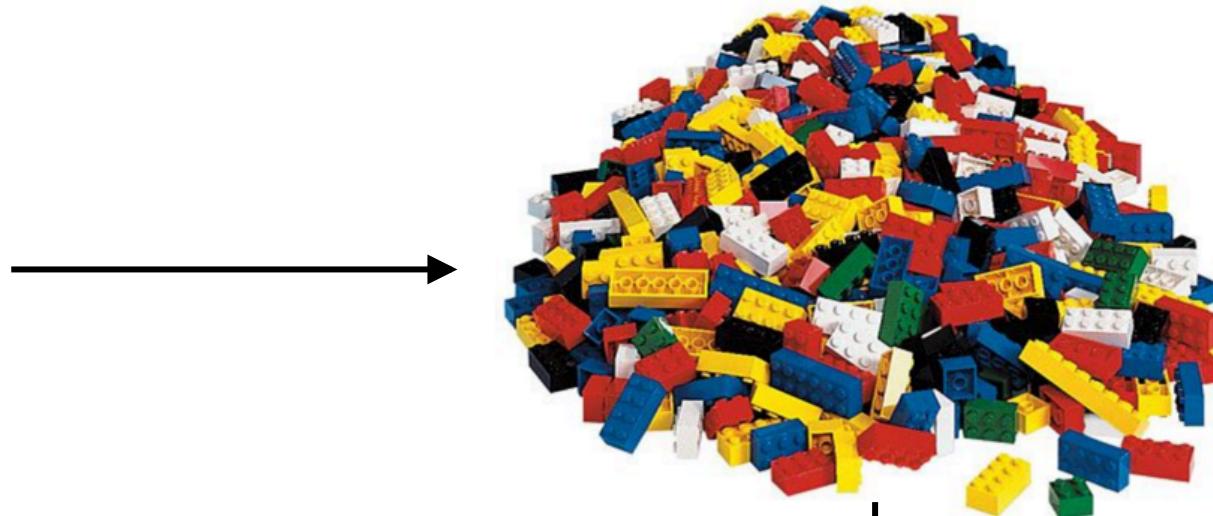
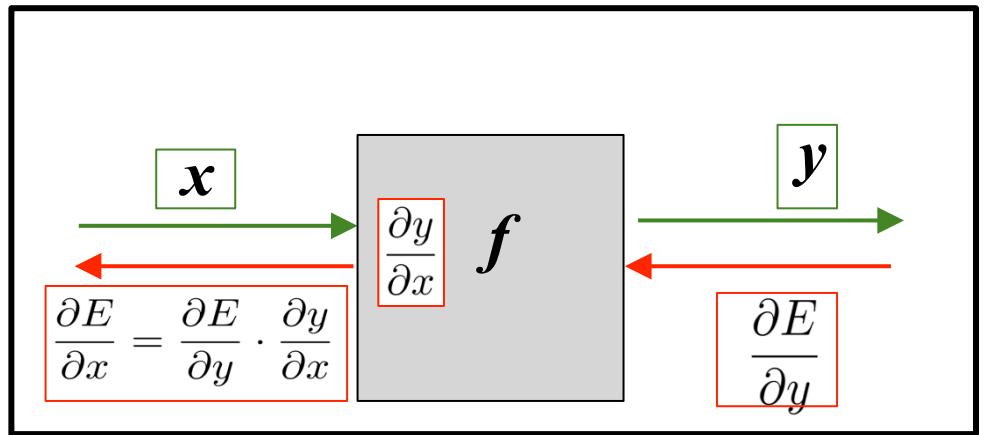


– Main Idea: error in hidden layers

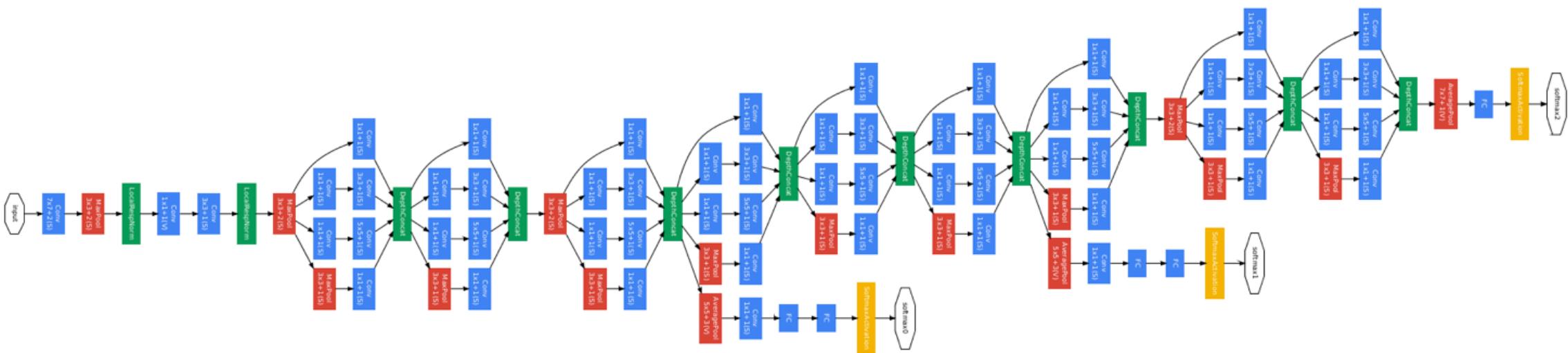
“Block View”



Building Deep Neural Nets



Building Deep Neural Nets



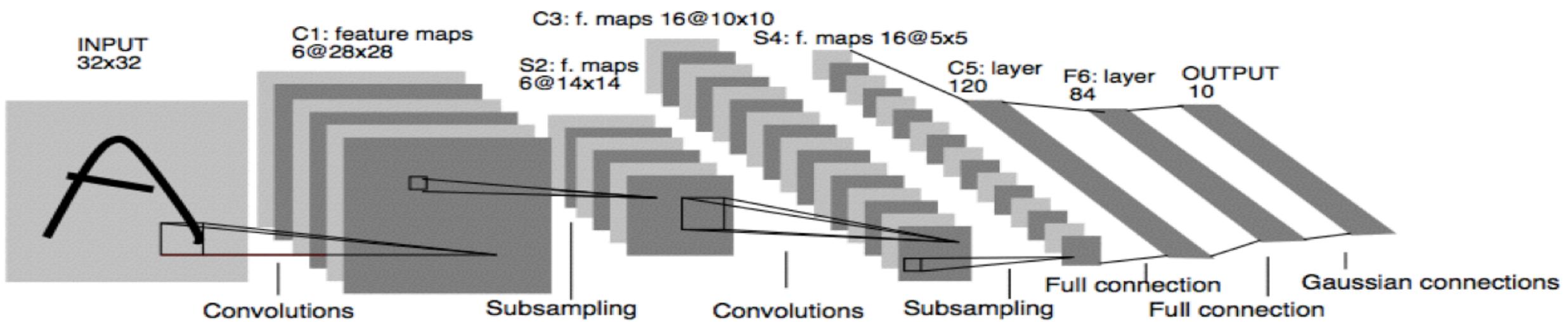
“GoogLeNet” for Object Classification

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

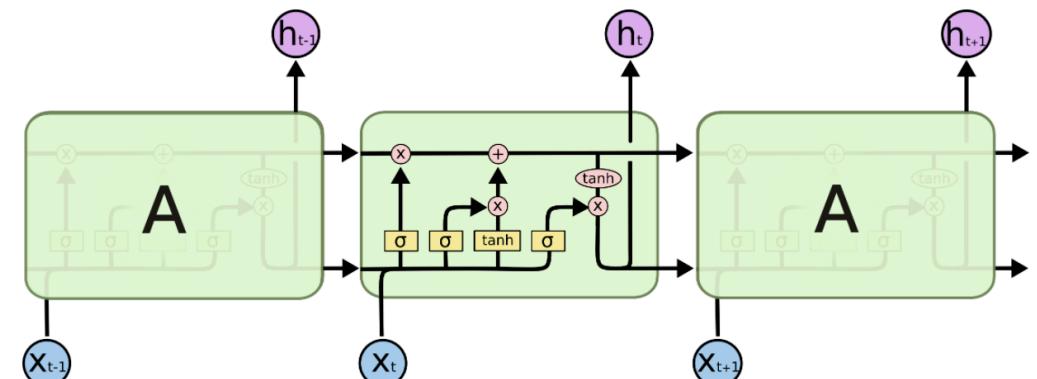
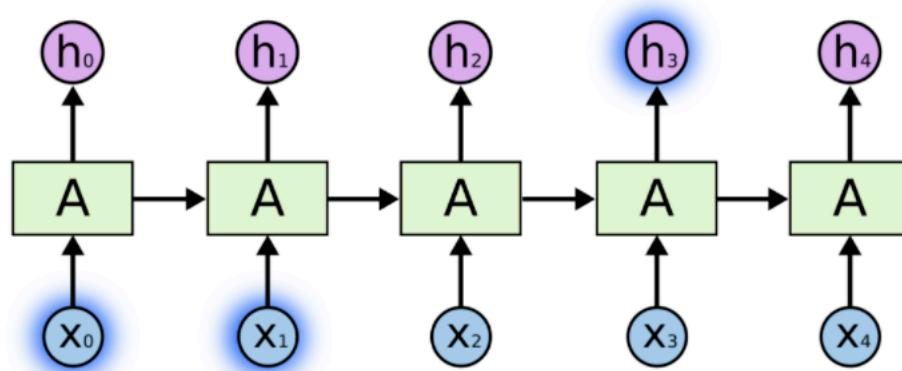
- Prof. Yann LeCun invented Convolutional Neural Networks (CNN) in 1998
- First NN successfully trained with many layers



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)

10% improve
with deepCNN



72%, 2010

74%, 2011

85%, 2012

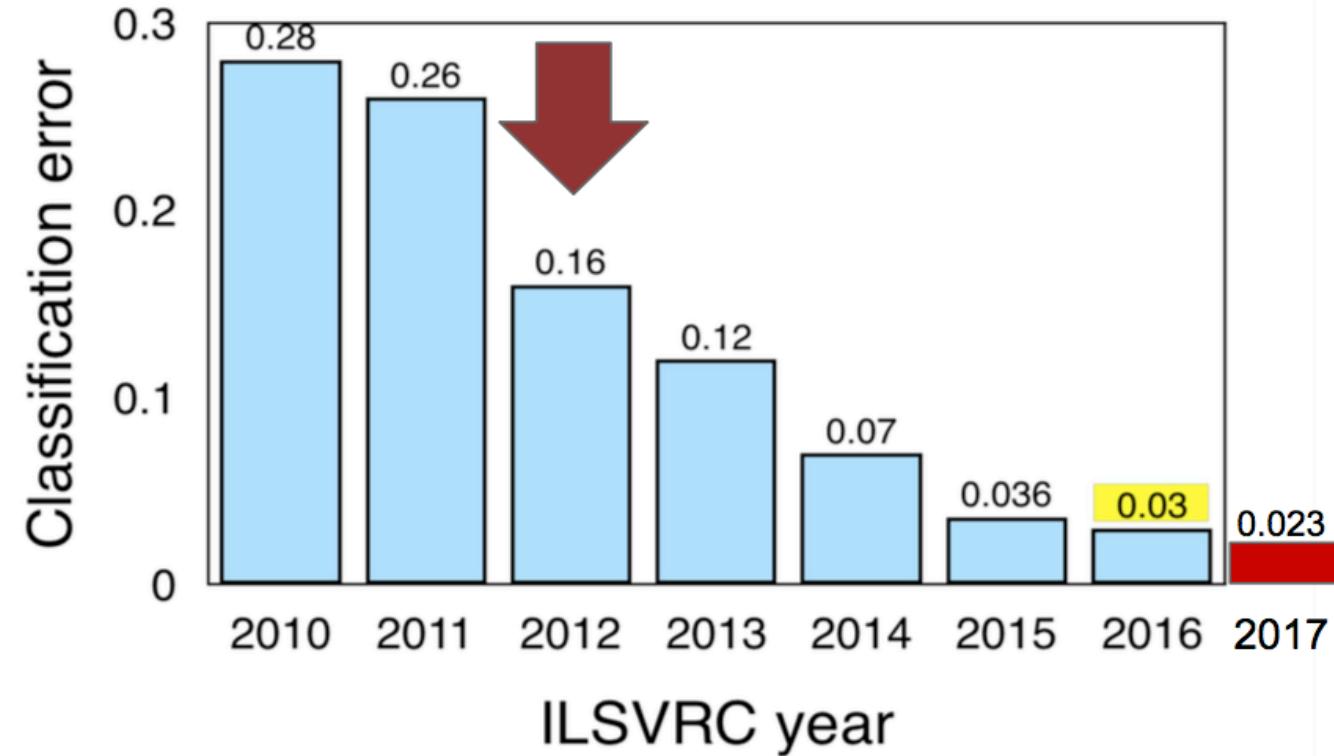
In one “very large-scale” benchmark competition
(1.2 million images [X] vs. 1000 different word labels [Y])

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

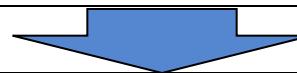


Deep Learning Way: Learning features / Representation from data



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

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)

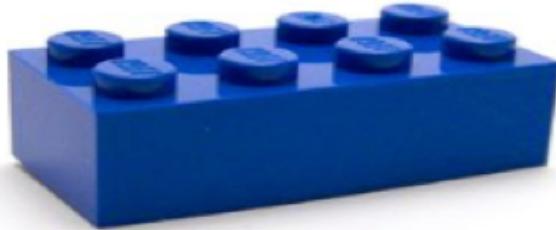
Why breakthrough ?

DNNs help us build more intelligent computers

- Intelligent Machines should be Able to
 - **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, ...
- This needs
 - Basic speech capabilities
 - Basic vision capabilities
 - Language understanding
 - User behavior / emotion understanding
 - Being **able to think / Reason**, e.g., learn to code programs, learn to search deepNN, ...
 - Being **able to imagine / analogy**, e.g., learn to write poem, to draw with styles,

Some Method Trends after ~2012

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



Inputs and Outputs



Losses



Architectures:



Learned Models

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

Today

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

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

- Deep Learning Tools for analyzing Sequential Data about Regulation:

- DeepChrome
- AttentiveChrome
- DeepMotif

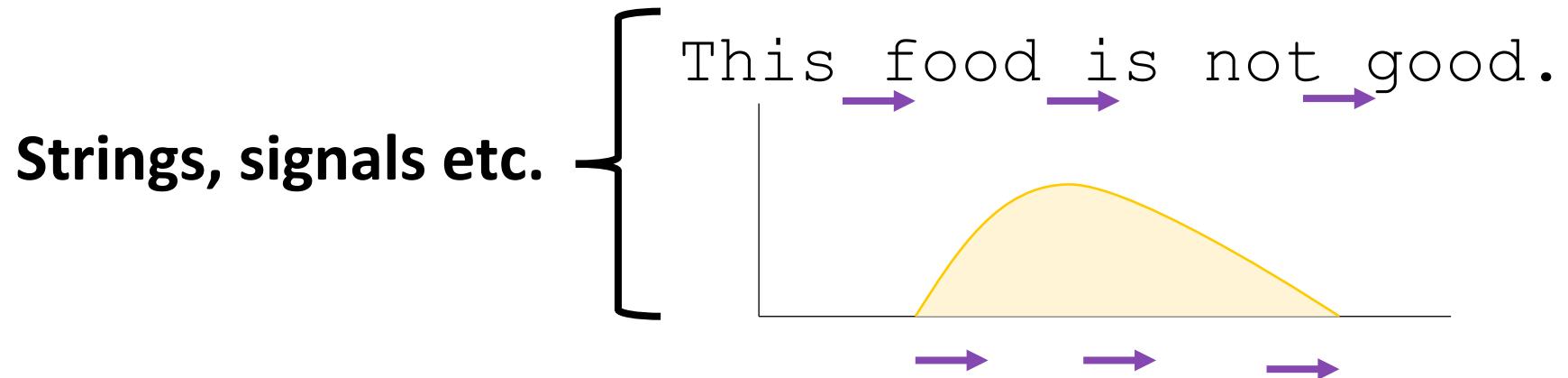
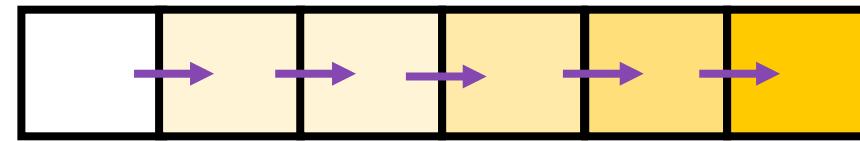
<https://www.deepchrome.org>

- More Tools: learning graphs from data

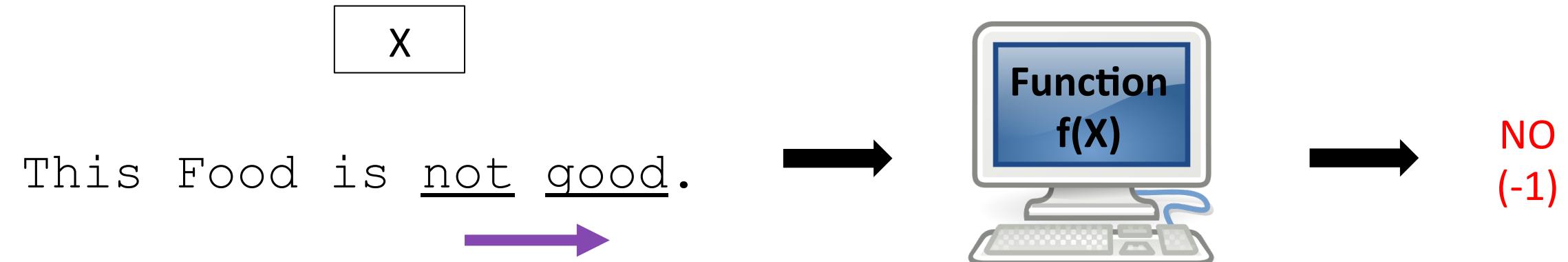
<https://www.jointggm.org>

Sequential Data

**Sequential
Data:**



[Example:] Classification of Sequential Data



Today

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

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

- Deep Learning Tools for analyzing Sequential Data about Regulation:

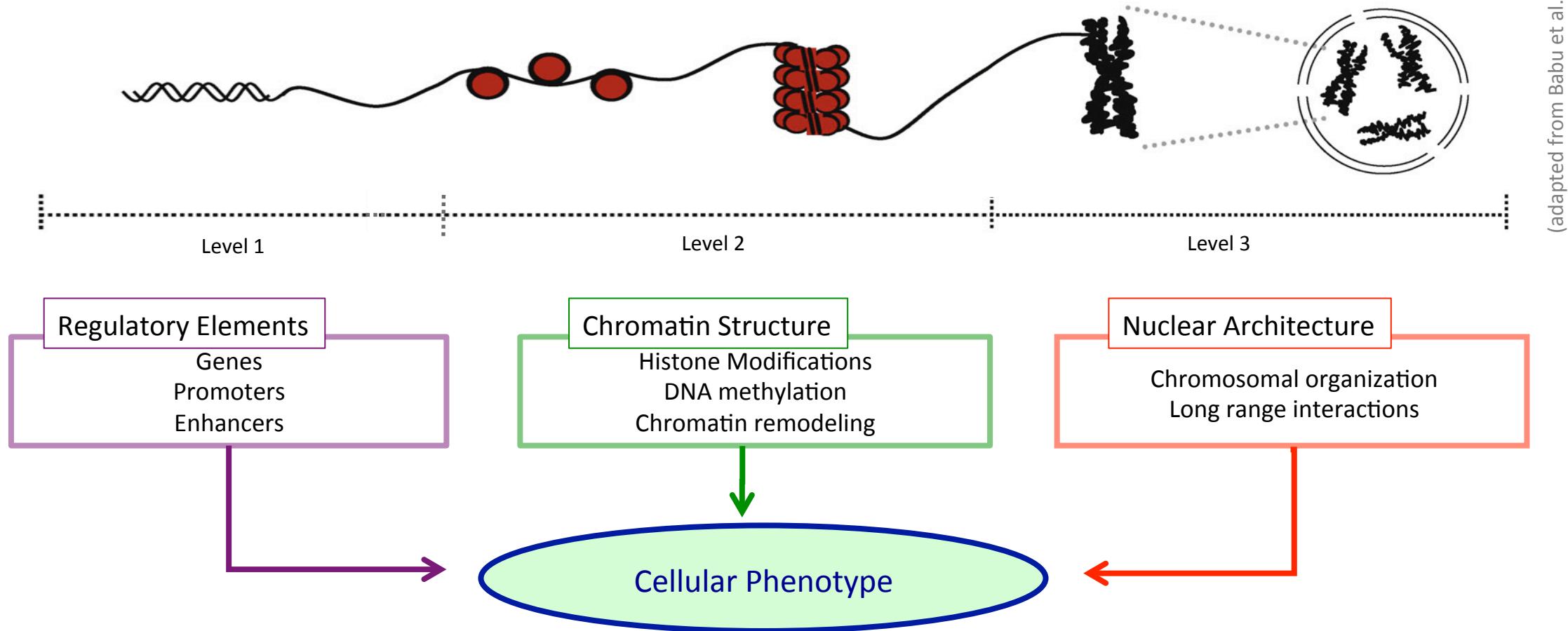
- DeepChrome
- AttentiveChrome
- DeepMotif

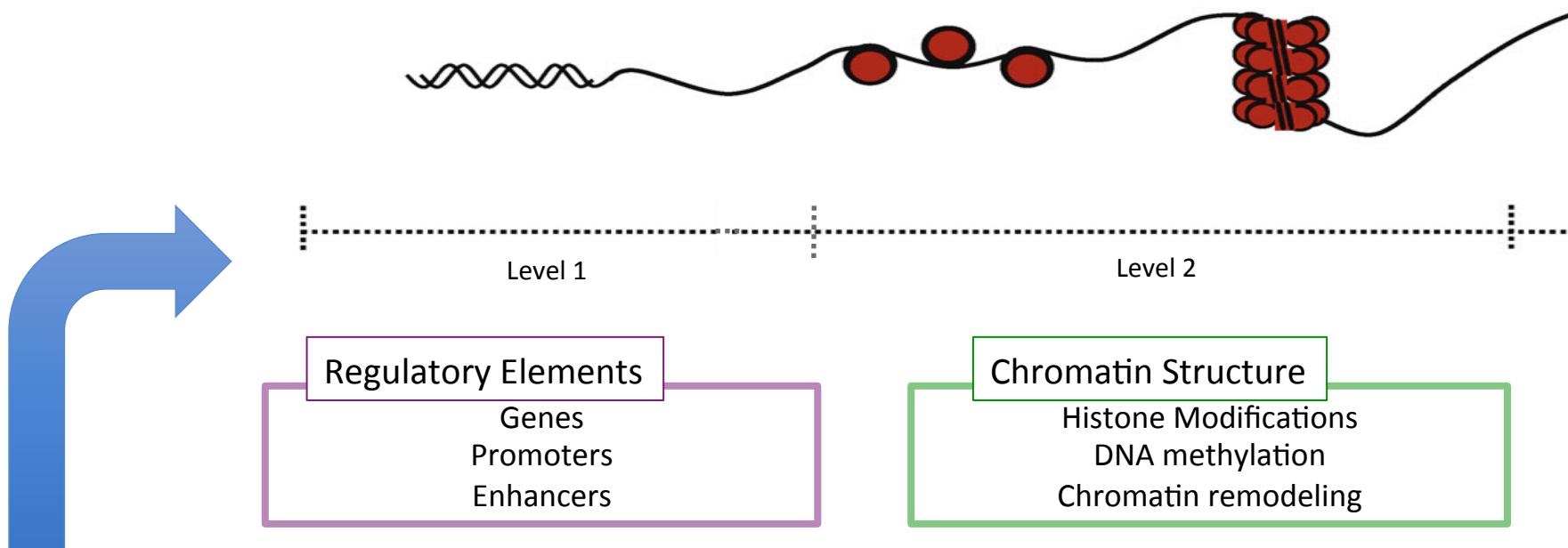
<https://www.deepchrome.org>

- More Tools: learning graphs from data

<https://www.jointggm.org>

Genome Organization and Gene Regulation

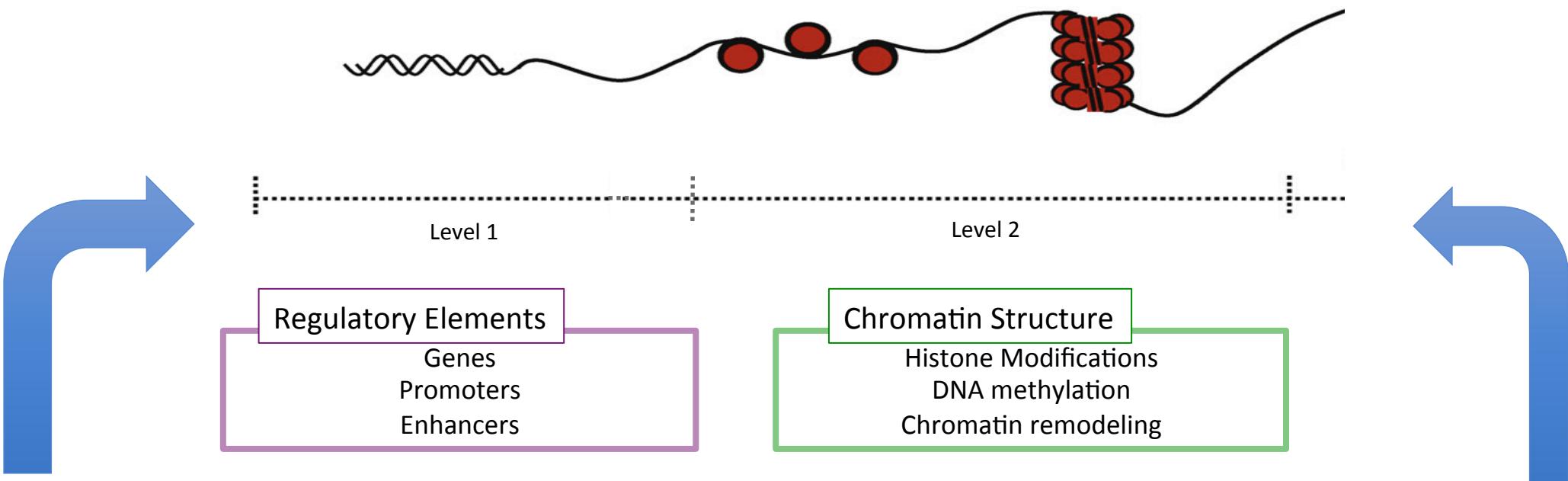




ENCODE Project (2003-Present)

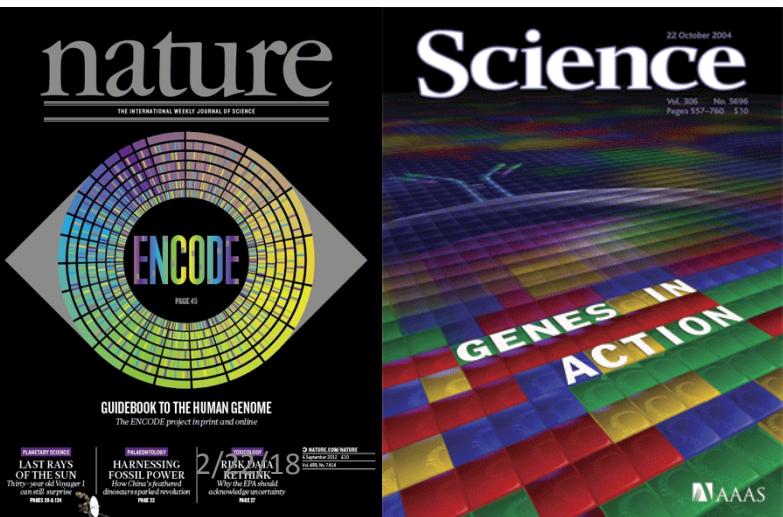
Describe the functional elements encoded in human DNA





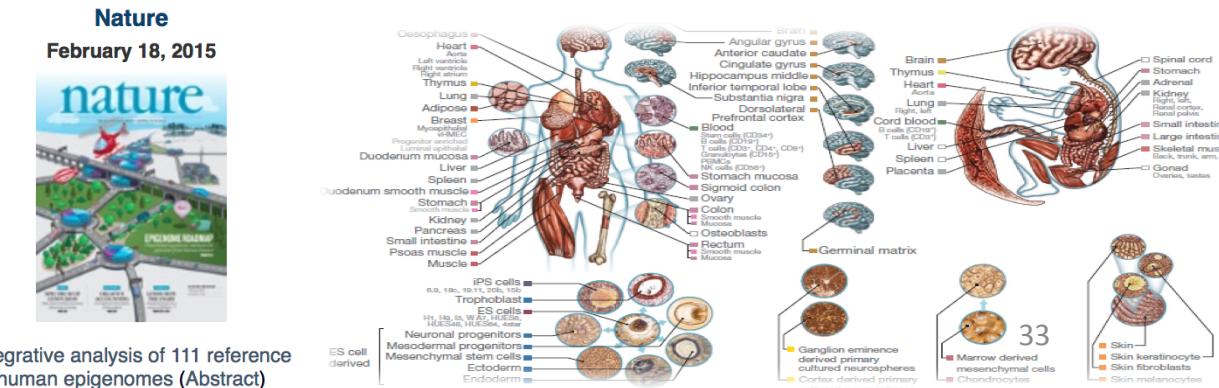
ENCODE Project (2003-)

Describe the functional elements encoded in human DNA



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.



Many Possible Computational Tasks

DNA
Segments
on
Genomes

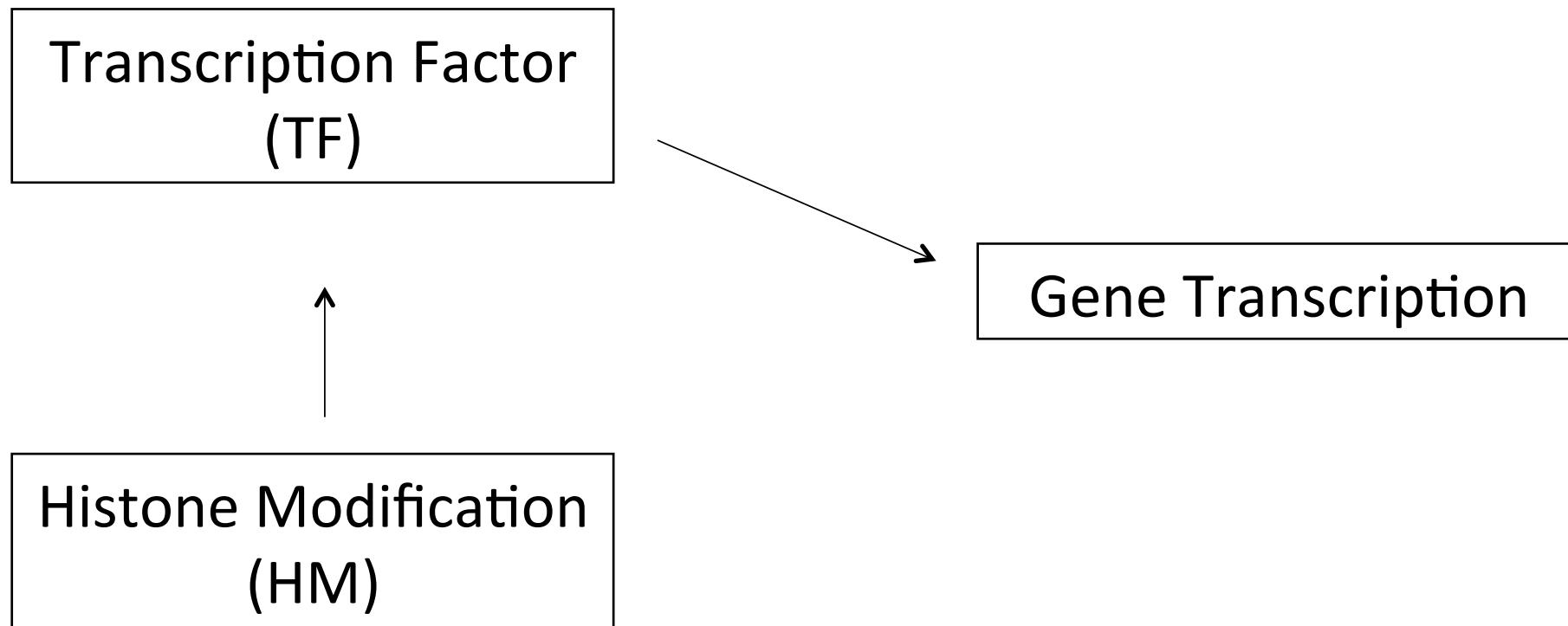
ATGCGATCAAGTCTG

TF Binding
Signals

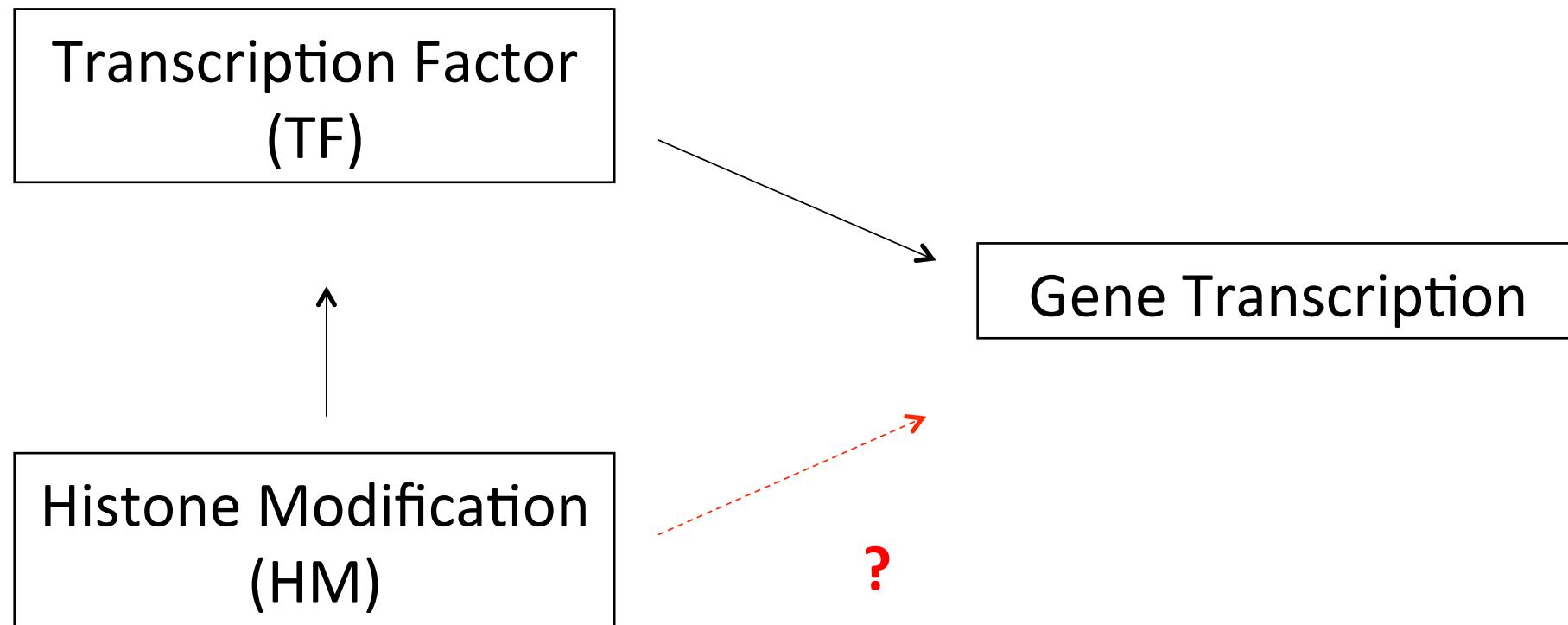
Gene
Expression

Histone
Modification
Signals

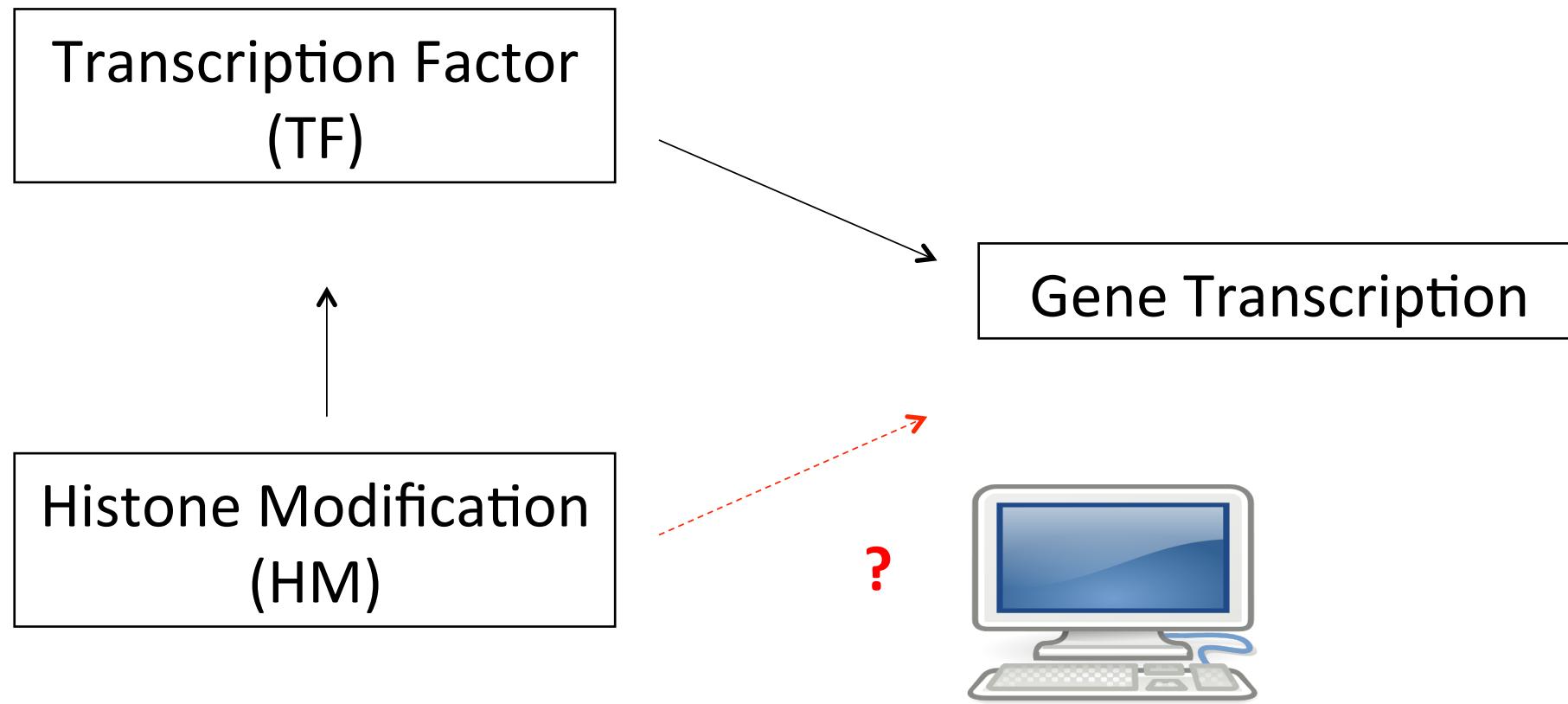
Histone Modification and Gene Transcription



Histone Modification and Gene Transcription



Histone Modification and Gene Transcription

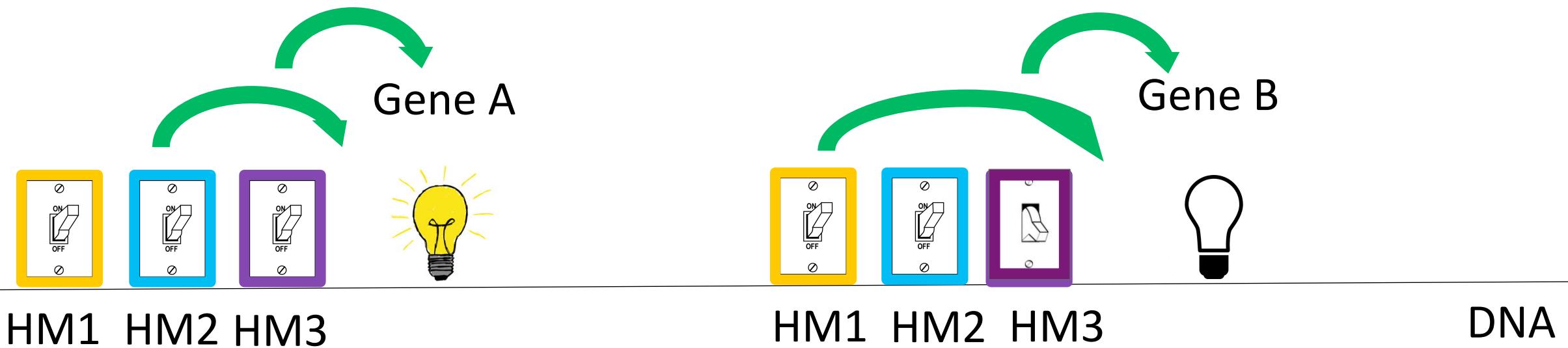


Why Studying [HM => Gene Expression] ?

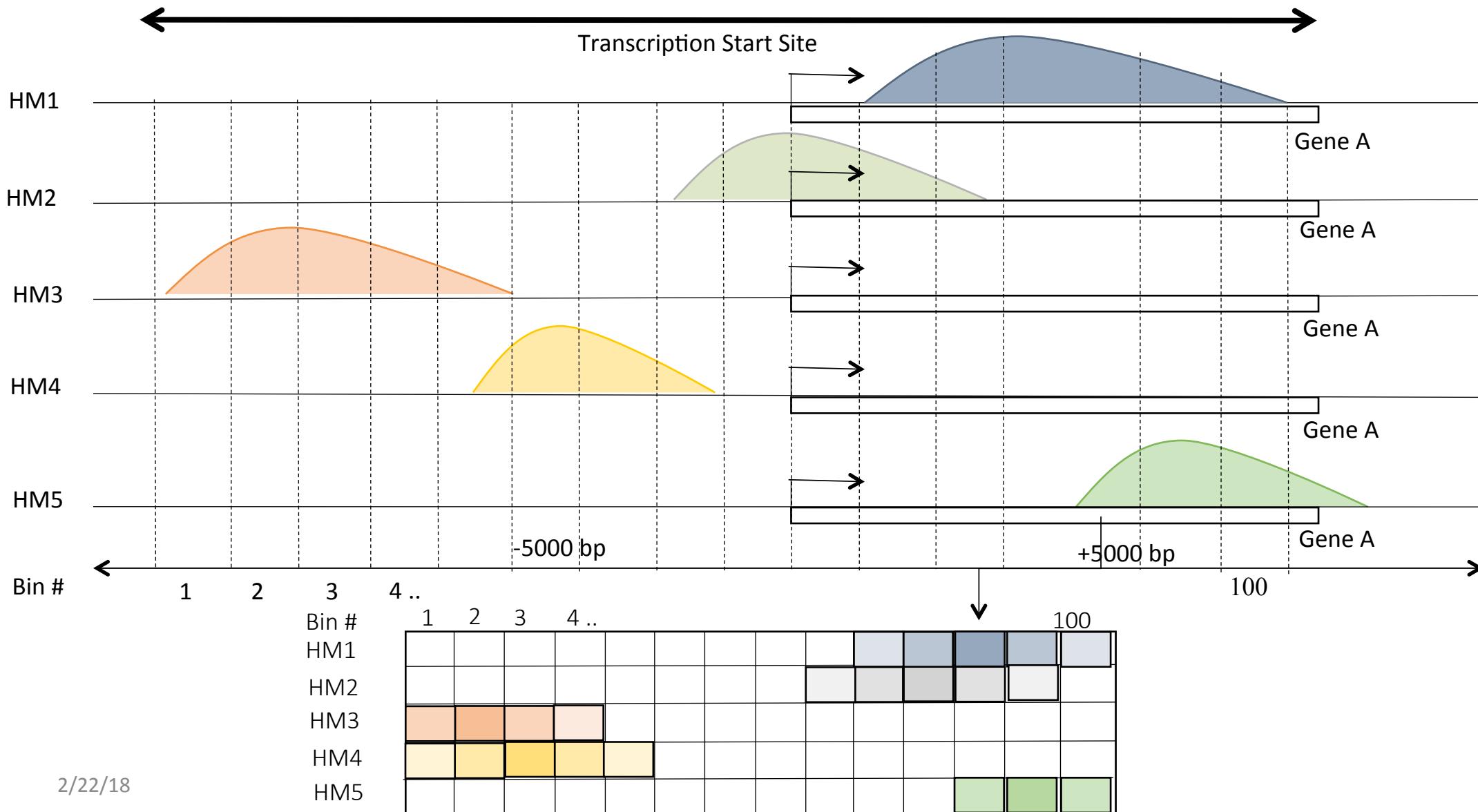
- Epigenomics:
 - Study of chemical changes in DNA and histones (without altering DNA sequence)
 - Inheritable and involved in regulating gene expression, development, tissue differentiation and suppression ...
- Modification in DNA/histones (changes in chromatin structure and function)
 - => influence 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?

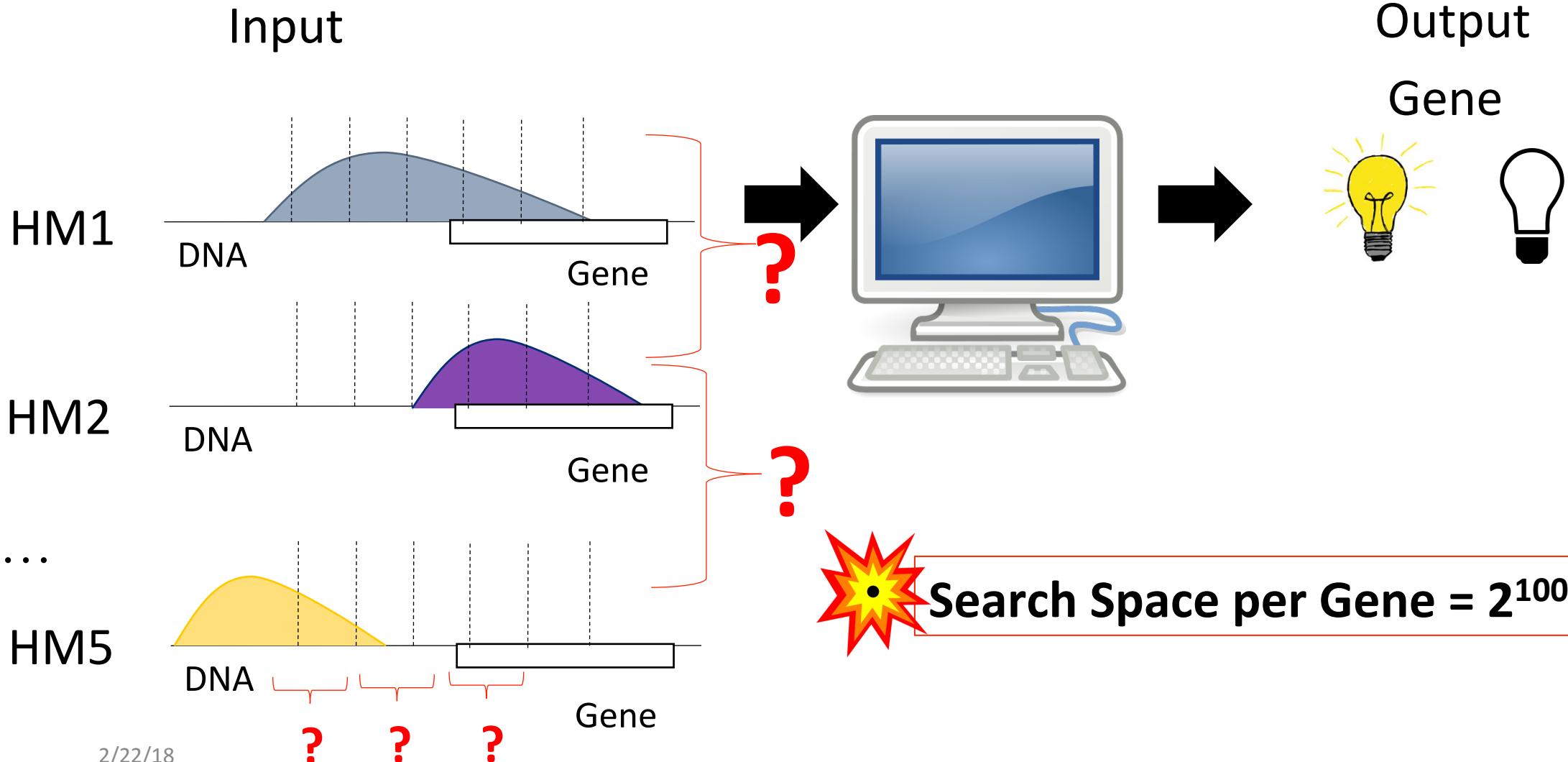
~56 Cell Types



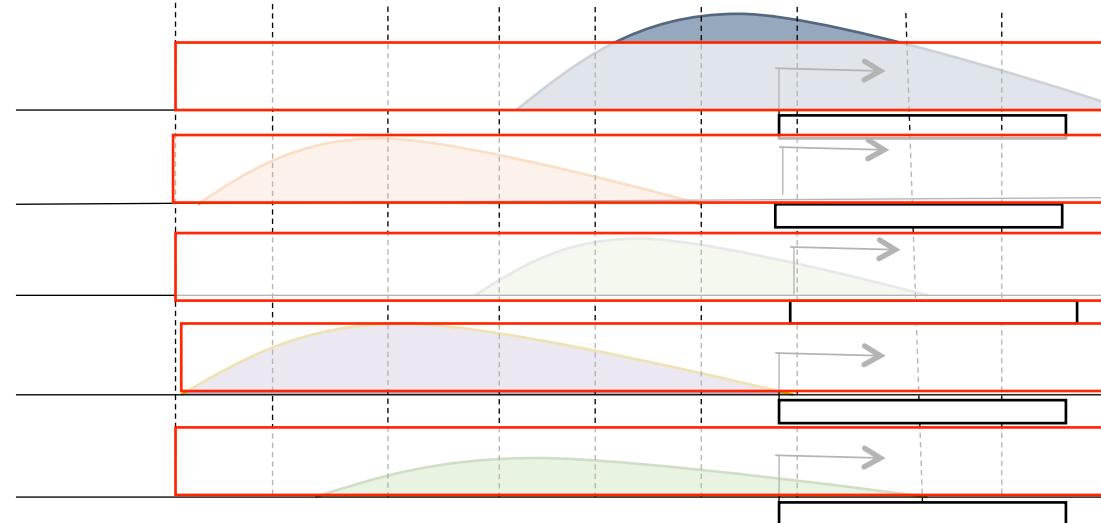
Input



Computational Challenge



Related Work



**Linear Regression,
SVM,
Random Forest**

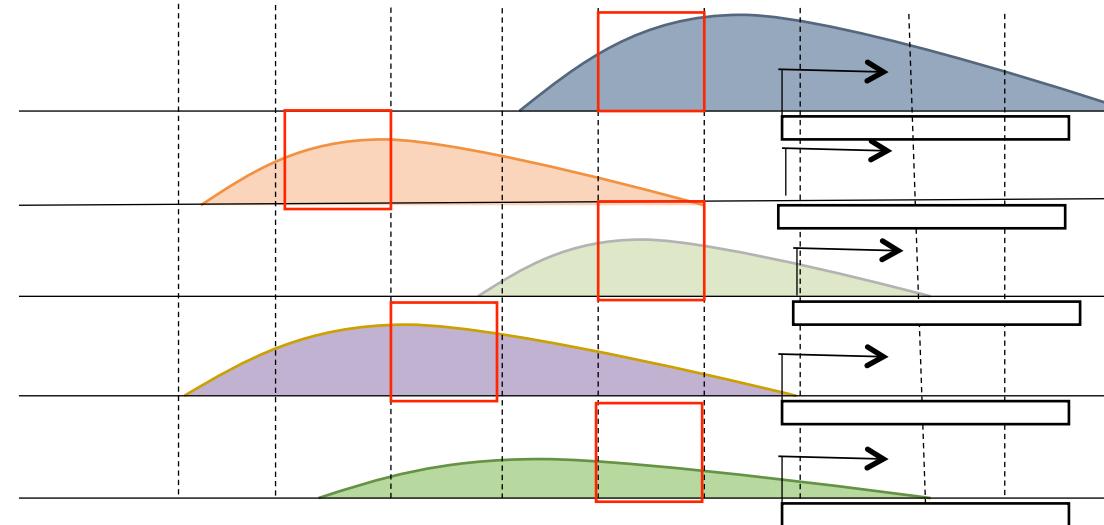
Gene Expression On/Off

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

2/22/18
[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)

Related Work



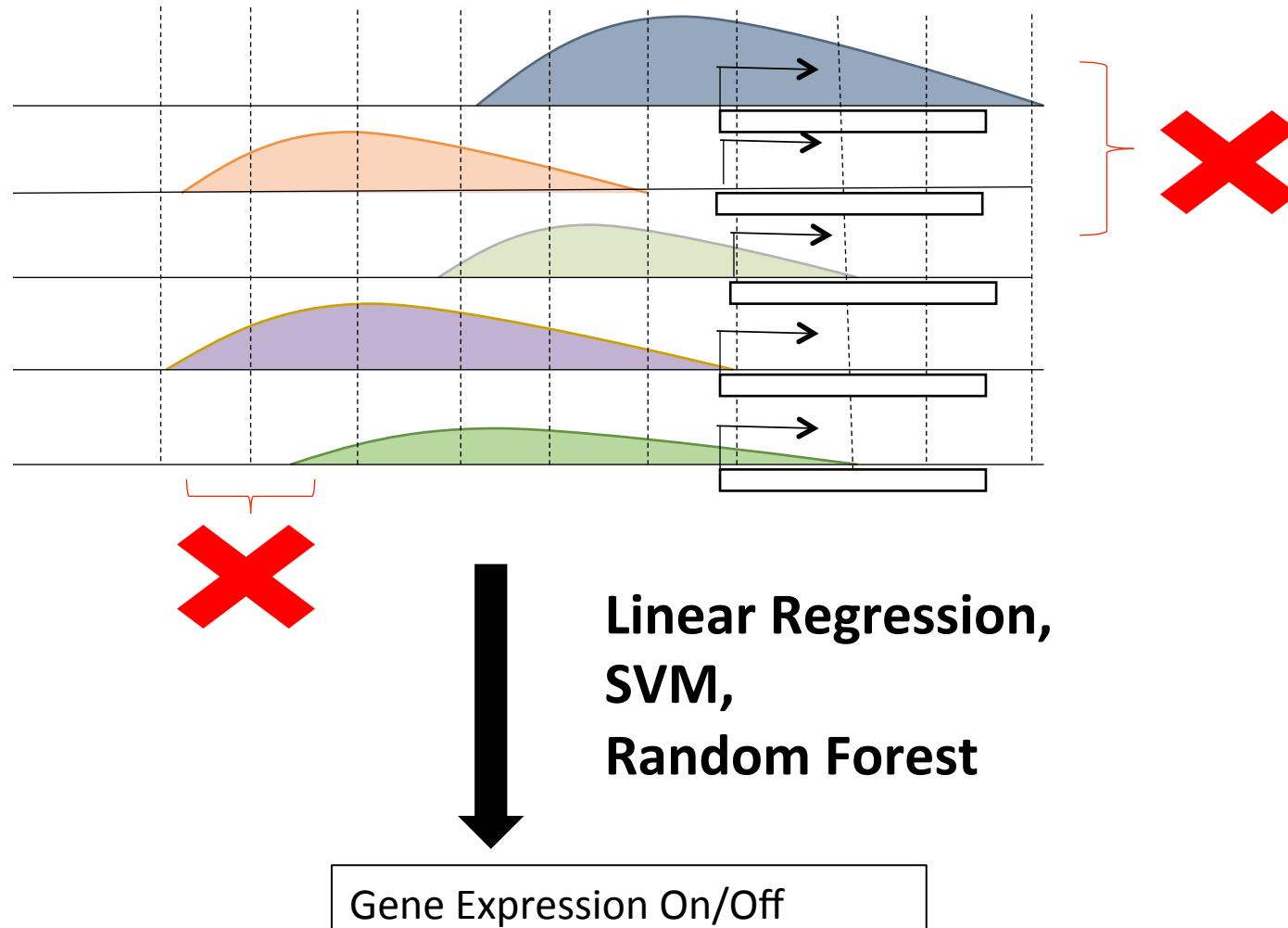
**Linear Regression,
SVM,
Random Forest**

Gene Expression On/Off

- [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)

2/22/18

Drawback of Related Works



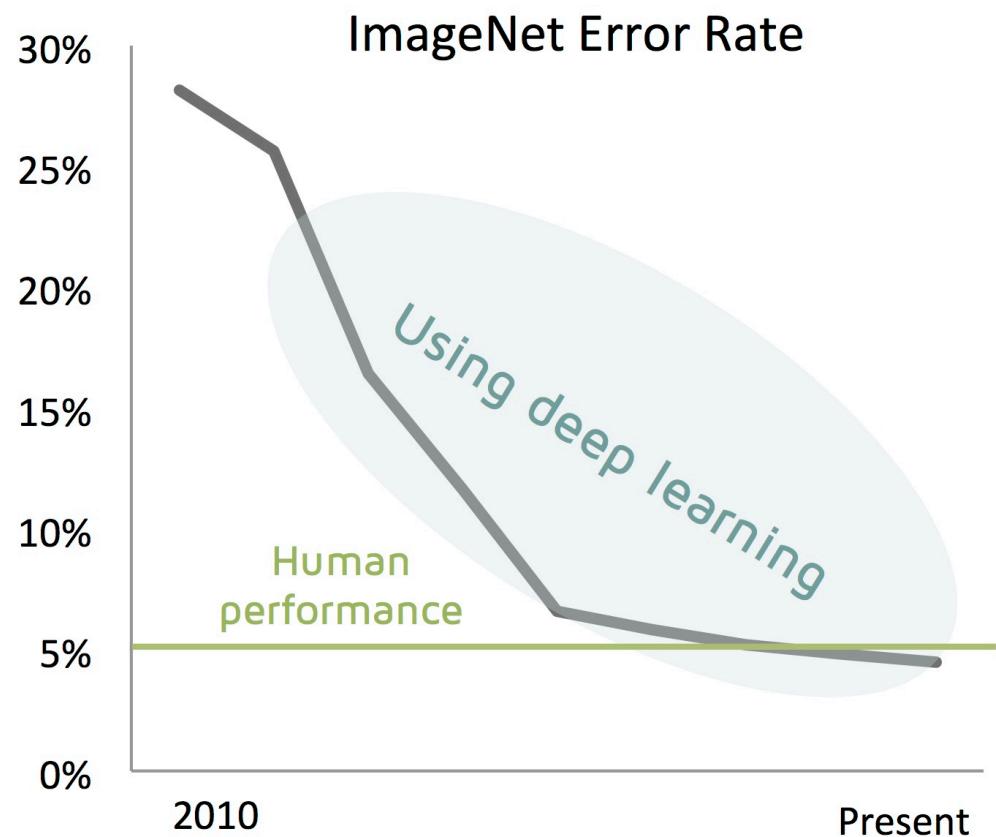
[1] Karlić, R. et al. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences* (2010)

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

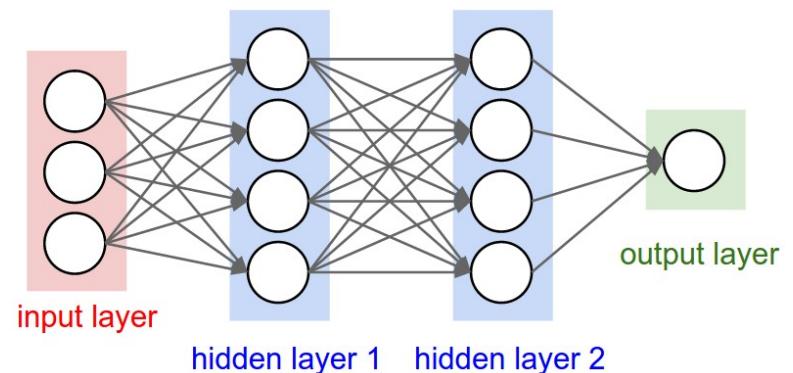
[3] Dong, X. et al. Modeling gene expression using chromatin features in various cellular contexts. *Genome Biology* (2012)

2/22/18

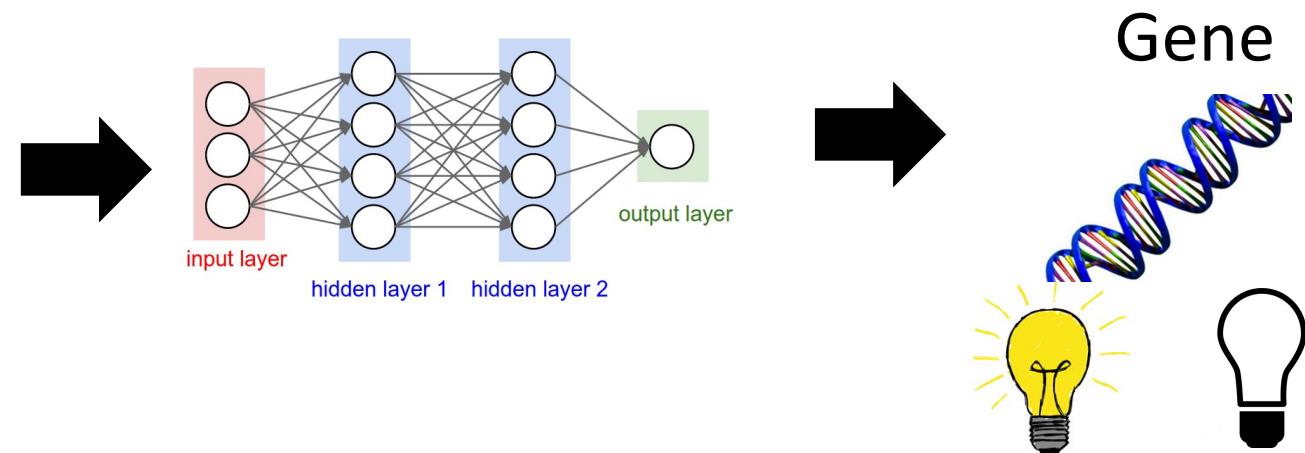
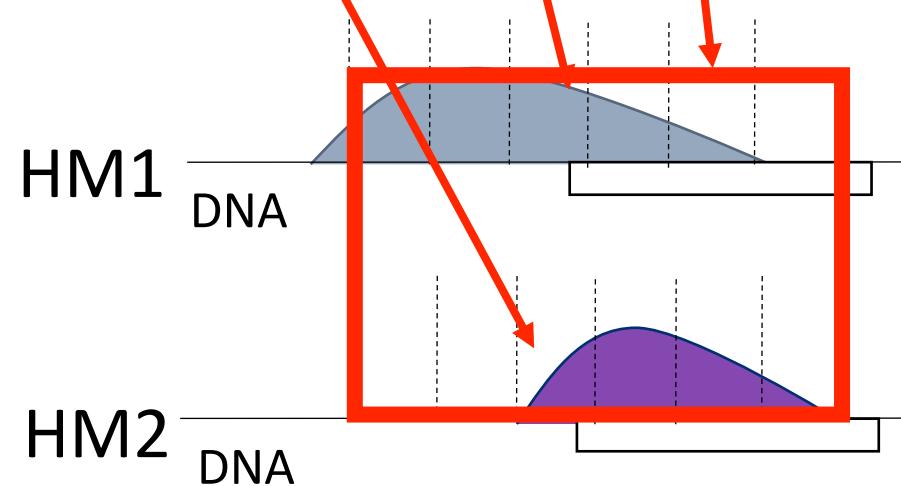
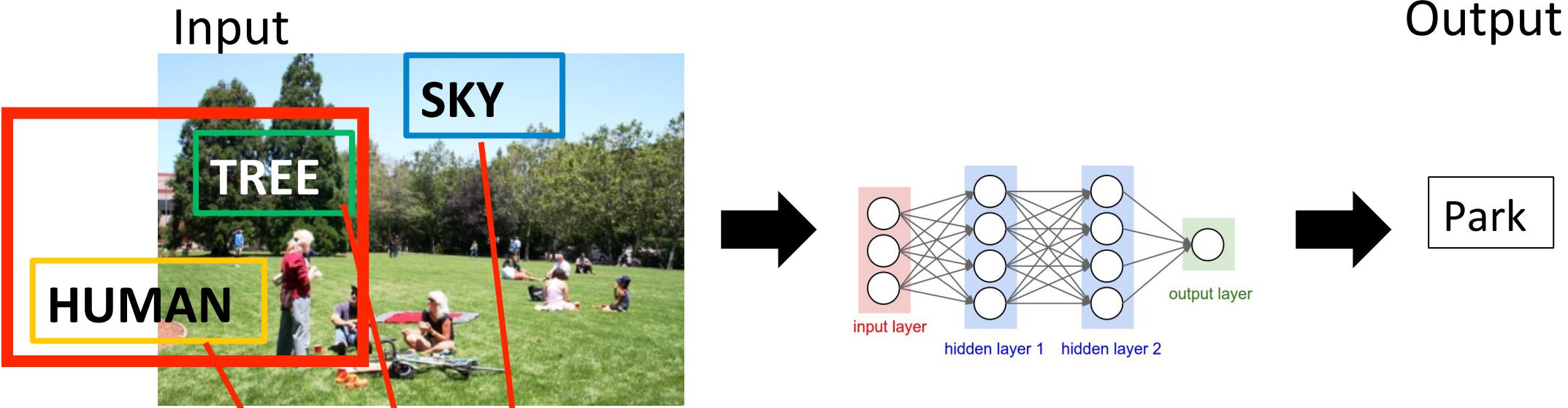
Solution



Deep Neural Network (DNN)



Solution : CNN



CNN models Locality and Translation Invariance



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!

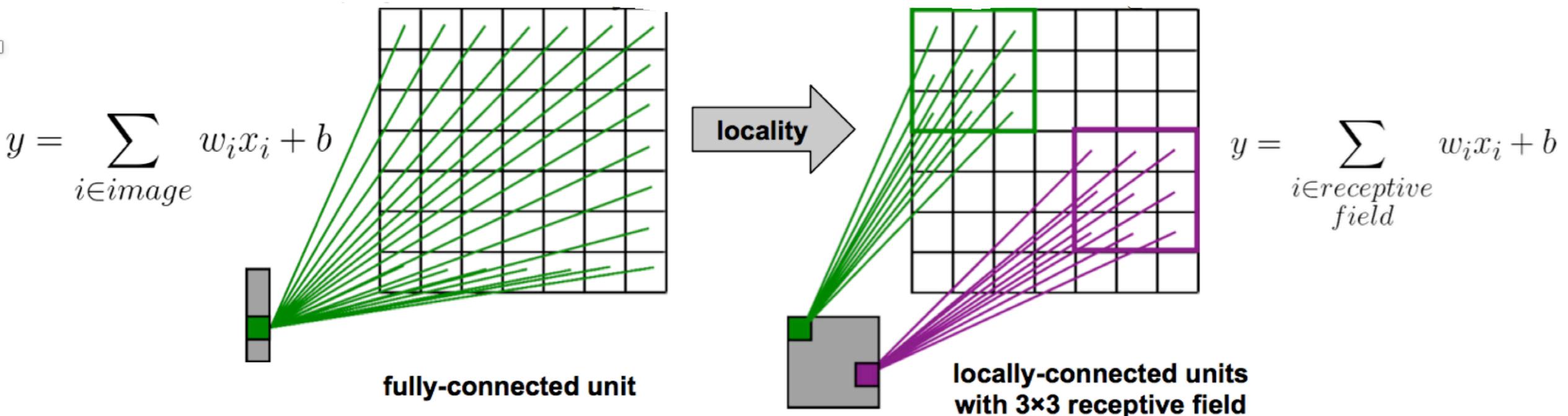
Locality and Translation Invariance

- **Locality:** objects tend to have a local spatial support
- **Translation invariance:** object appearance is independent of location
- Can define these properties on a grid/lattice
 - CNN machinery applicable to other data with such properties, e.g. audio/text

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

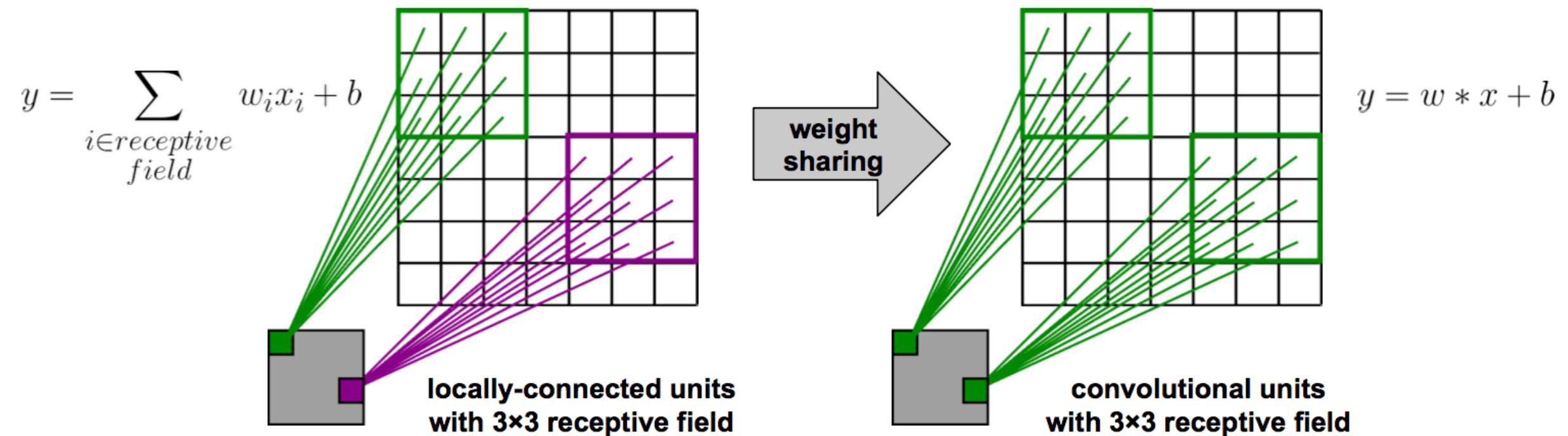
Incorporating Assumptions: Locality

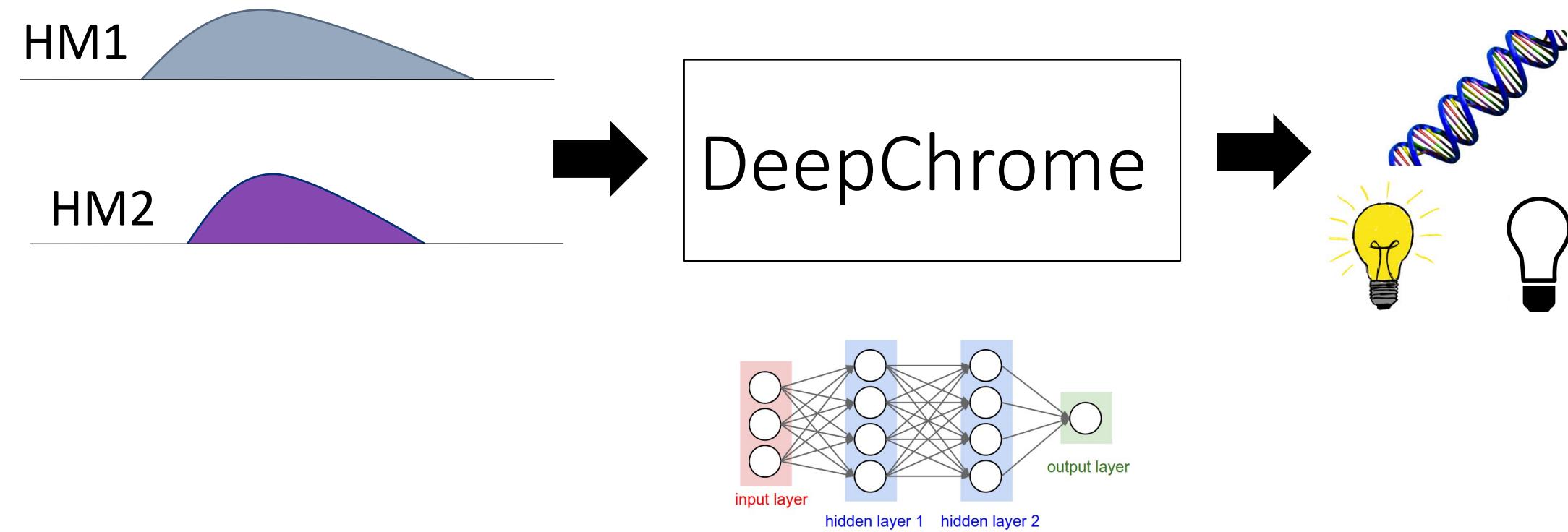
- Make **fully-connected** layer **locally-connected**
- Each neuron/unit connects to a local rectangular area – receptive field
- Different neuron units connect to different local regions (rarely used)



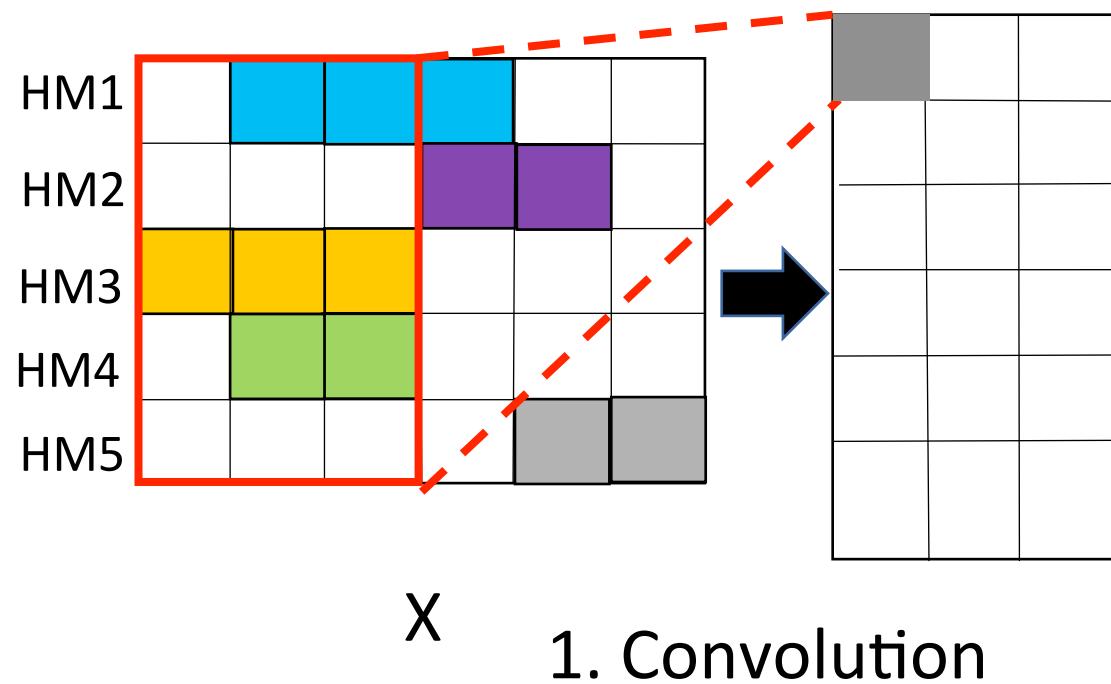
Incorporating Assumptions: translation invariance

- **Weight sharing**
 - units connected to different locations have the same weights
 - equivalently, each unit is applied to all locations
- *Convolutional layer – locally-connected layer with weight sharing (translation invariance)*
- The weights are invariant, the output is equivariant

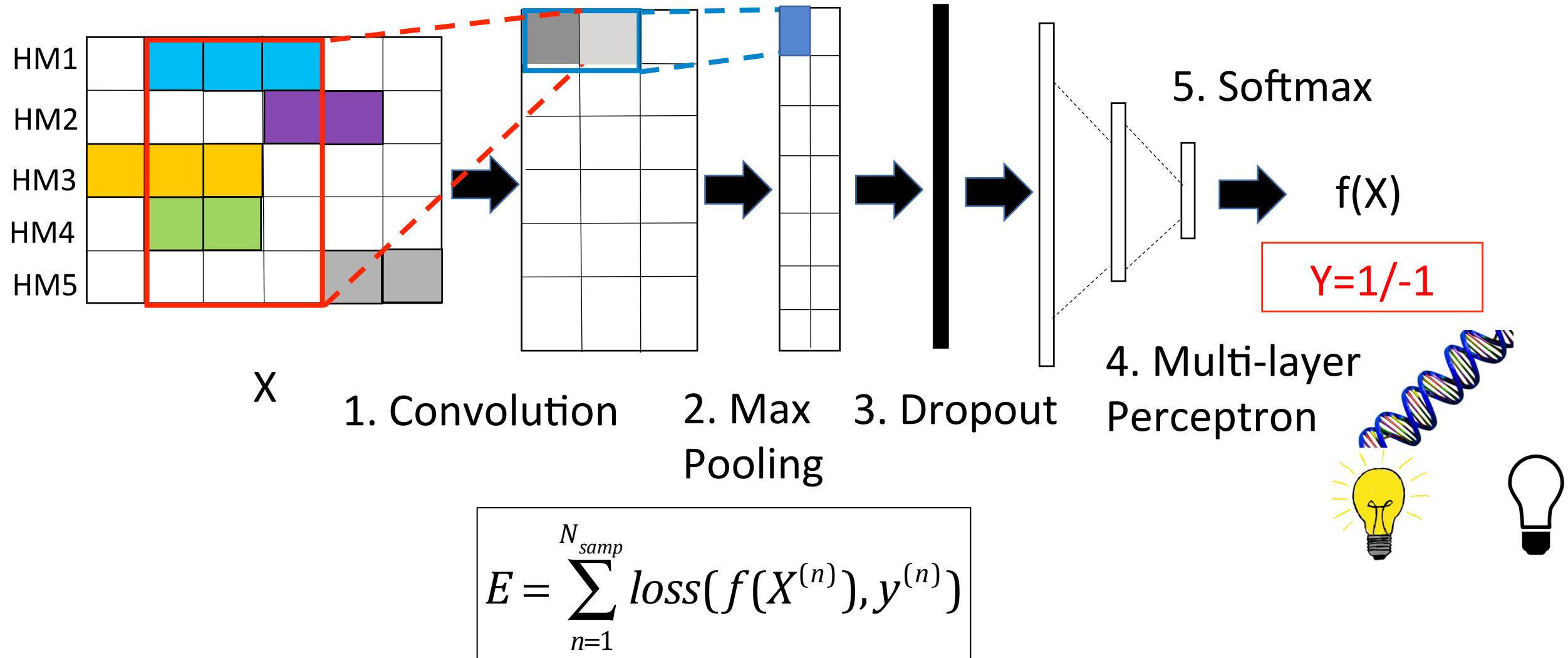




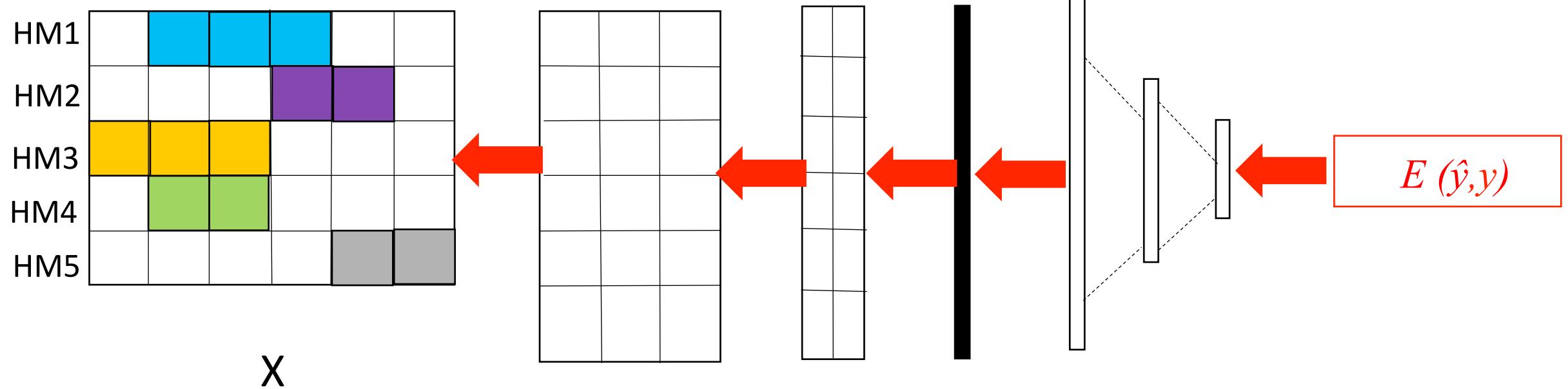
DeepChrome: Convolutional Neural Network (CNN)



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
- **Output (Gene Expression):** Discretized RNA-Seq

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

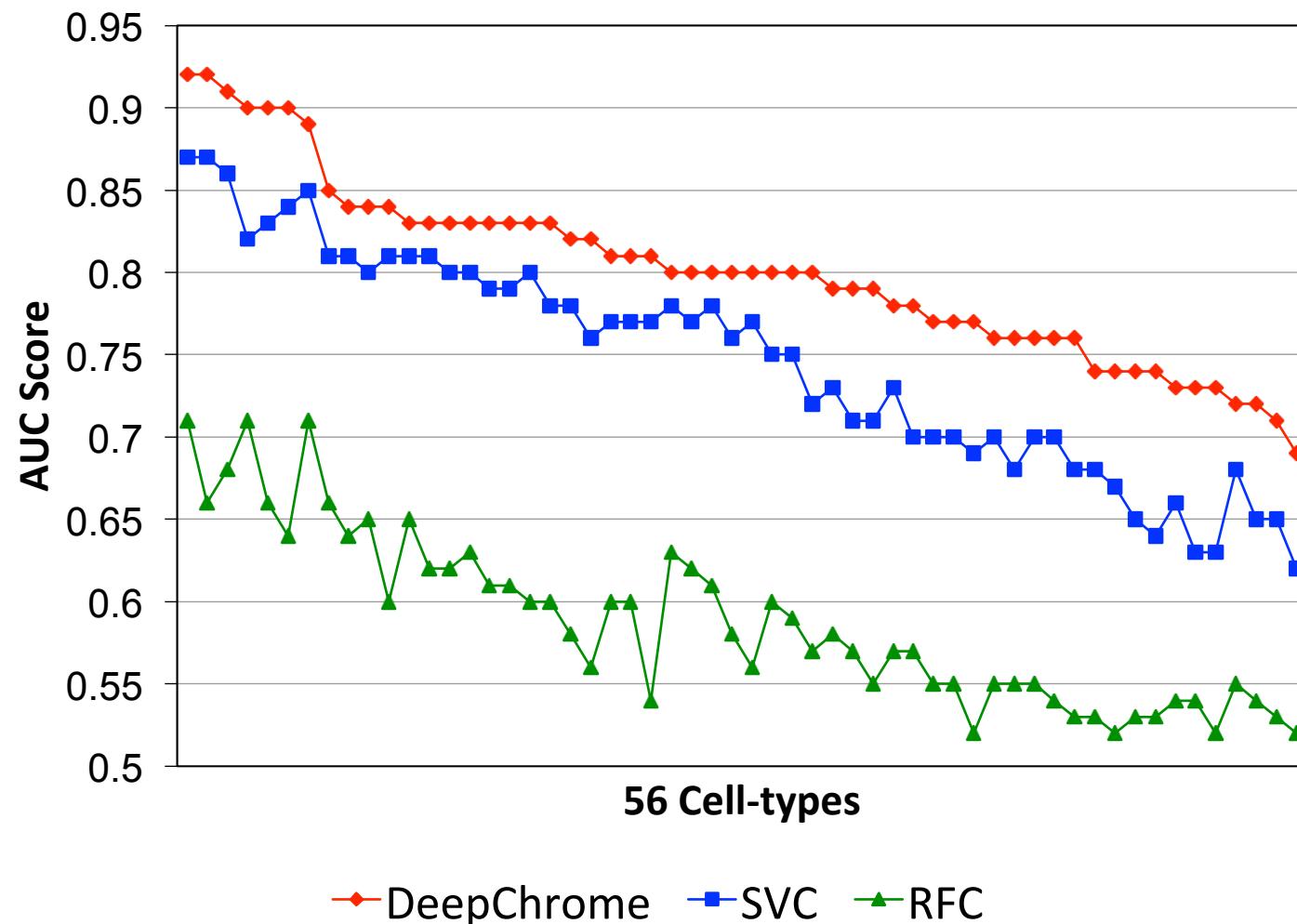
- **Baselines:** Support Vector Classifier (SVC) and Random Forest Classifier (RFC)

Training Set
6601 Genes

Validation Set
6601 Genes

Test Set
6600 Genes

Results: Accuracy



Summary

Accurate



DeepChrome: Convolution
Neural Network (CNN)

First DNN
implementation

Understandable

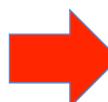
R. Singh, et al. Deep-learning for predicting gene expression from histone modifications". *Bioinformatics*. (2016)

Today

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

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

- Deep Learning Tools for analyzing **Sequential Data** about Regulation:



- DeepChrome
- AttentiveChrome
- DeepMotif

<https://www.deepchrome.org>

- More Tools: learning graphs from data

<https://www.jointggm.org>

Solution to Black-box DNN: Interpretability by Attention

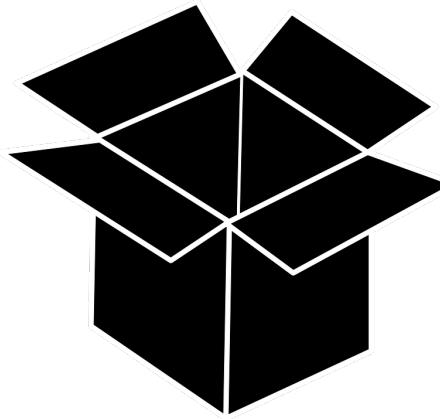
Input



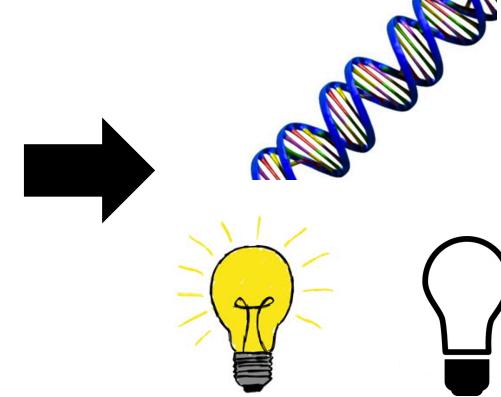
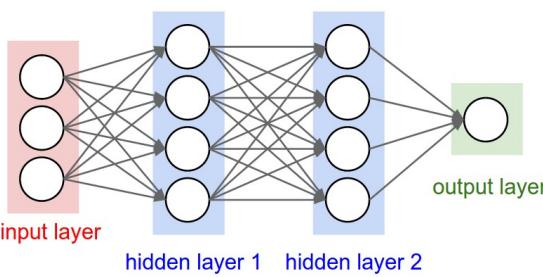
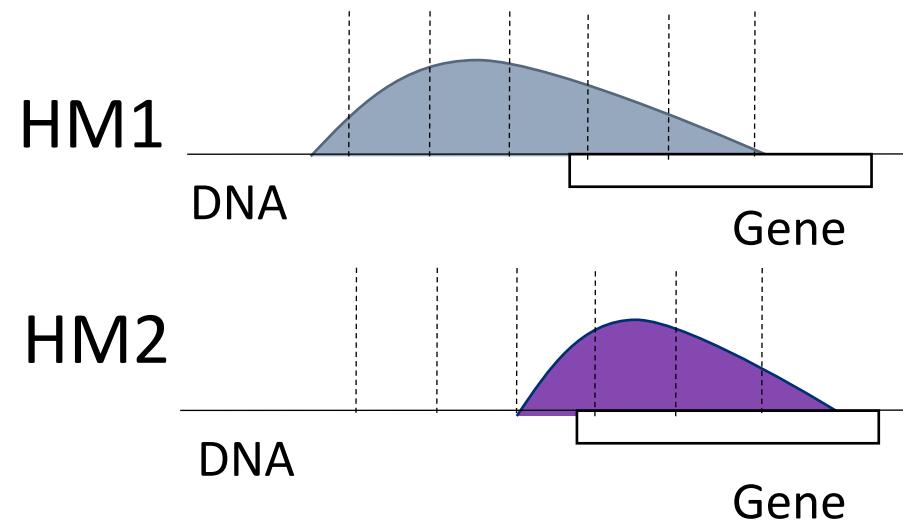
Output

Park

Attention
Mechanism



Gene



Solution : Interpretability

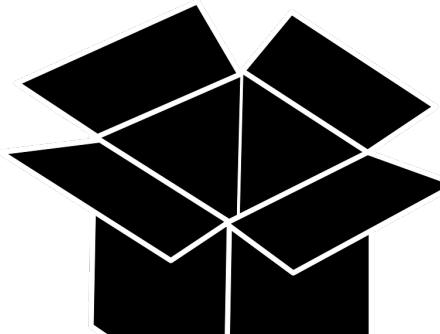
Input



Output

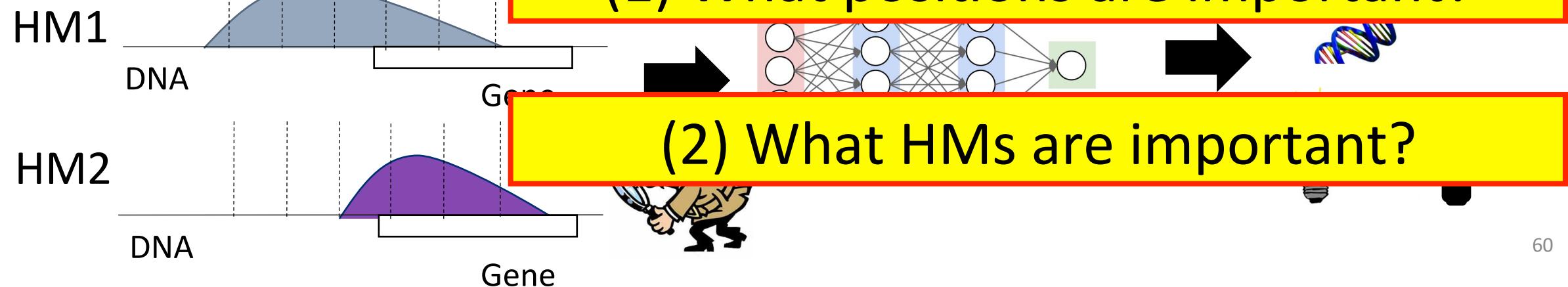
Park

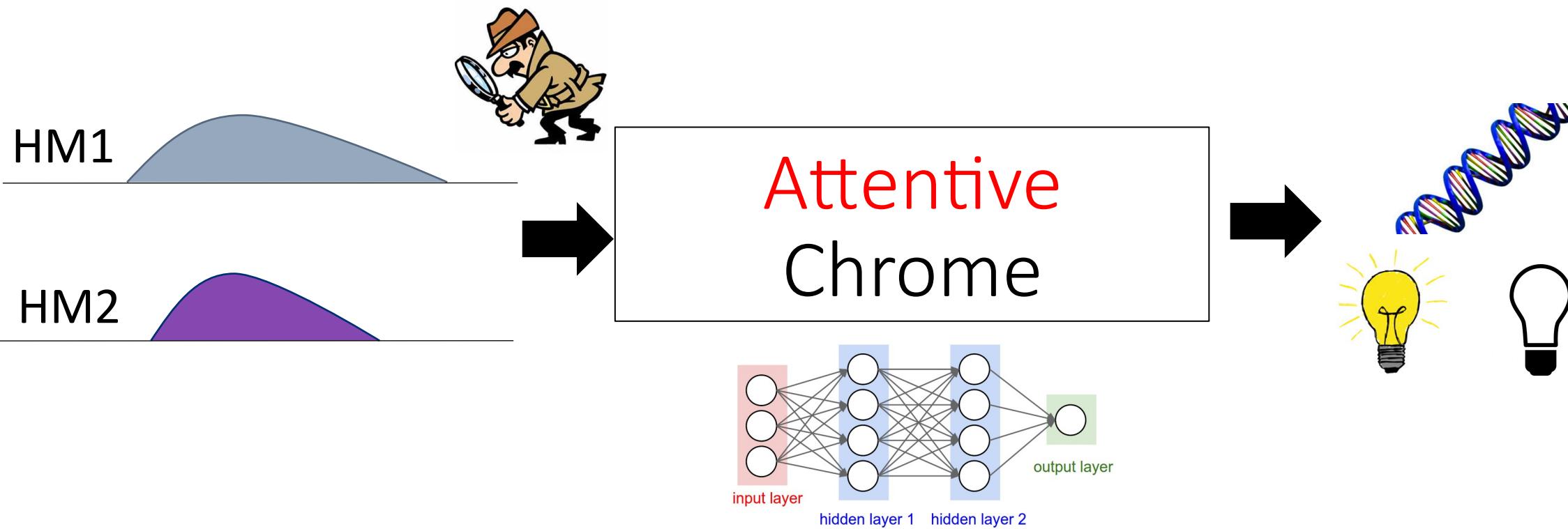
Attention
Mechanism



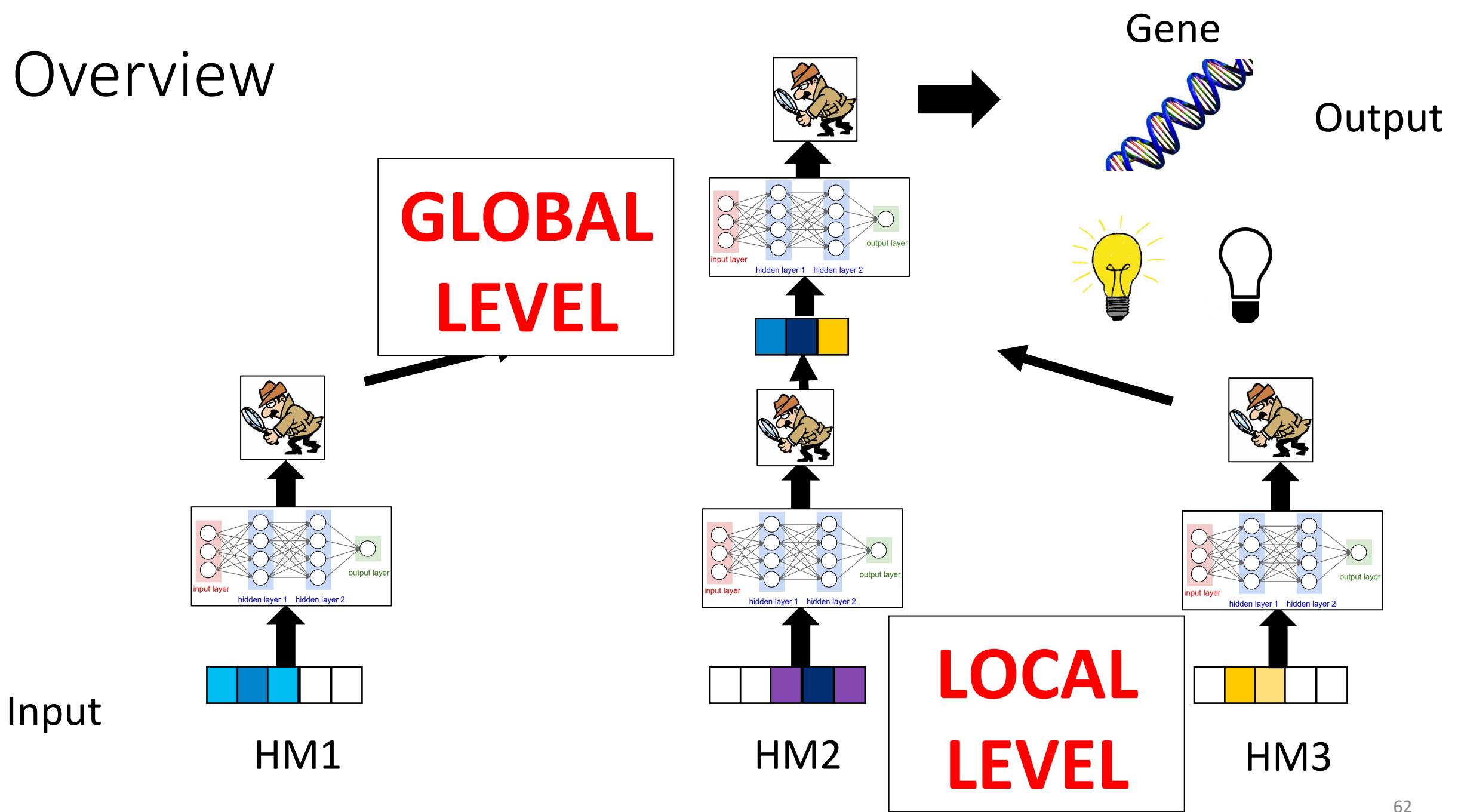
Gene

(1) What positions are important?





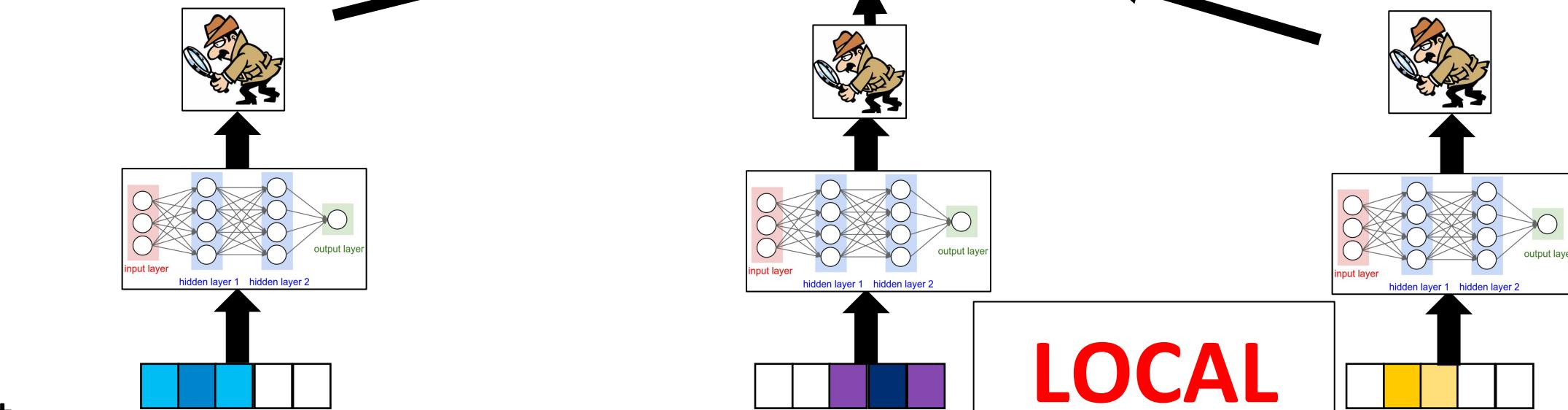
Overview



Overview

GLOBAL

(2) What HMs are important?

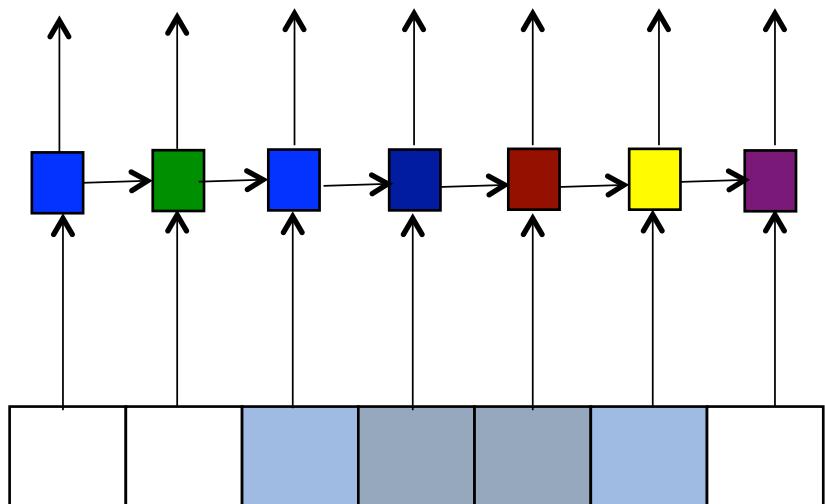


**LOCAL
LEVEL**

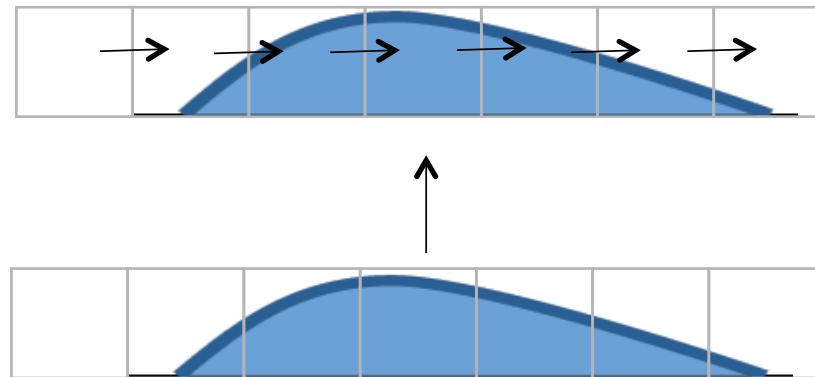
(1) What positions are important?

HM3

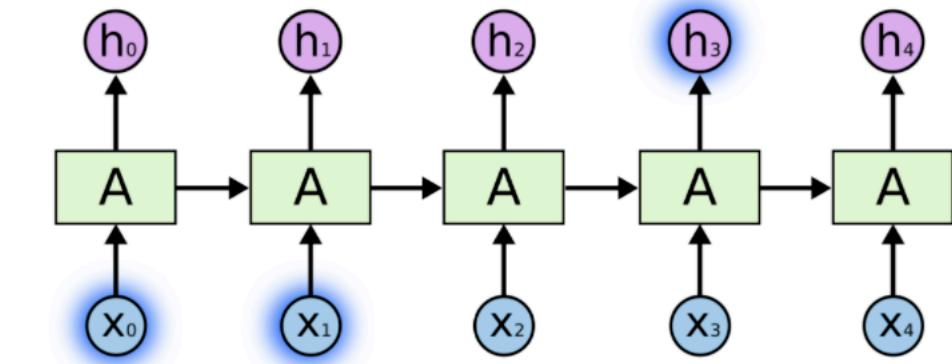
Recurrent Neural Network (RNN)



HM1



Using Attention to Select RNN per-time-step outputs



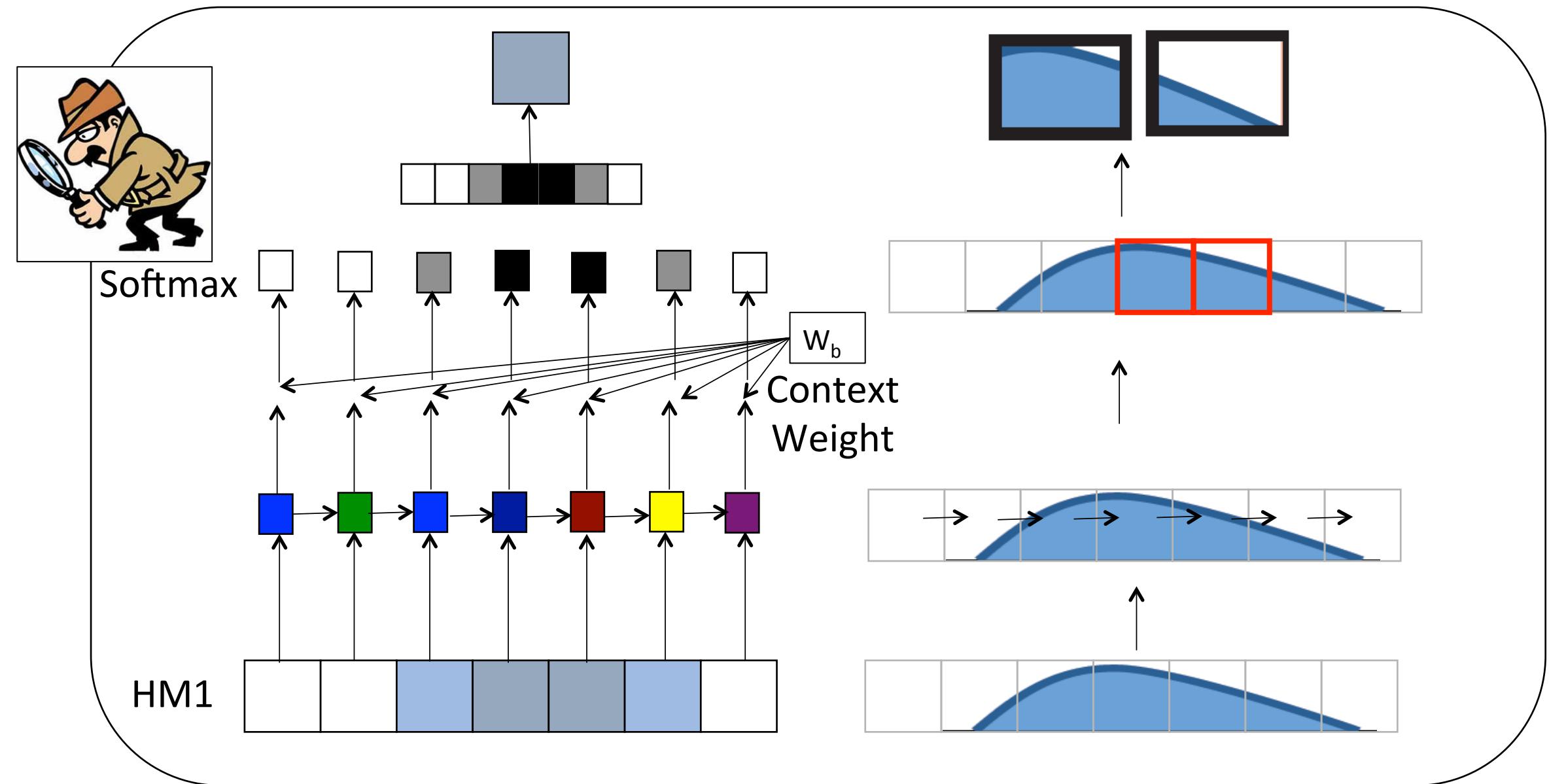
$$h_t = f_W(h_{t-1}, x_t)$$

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

Attention Mechanism at the local HM level (attention weights per gene)

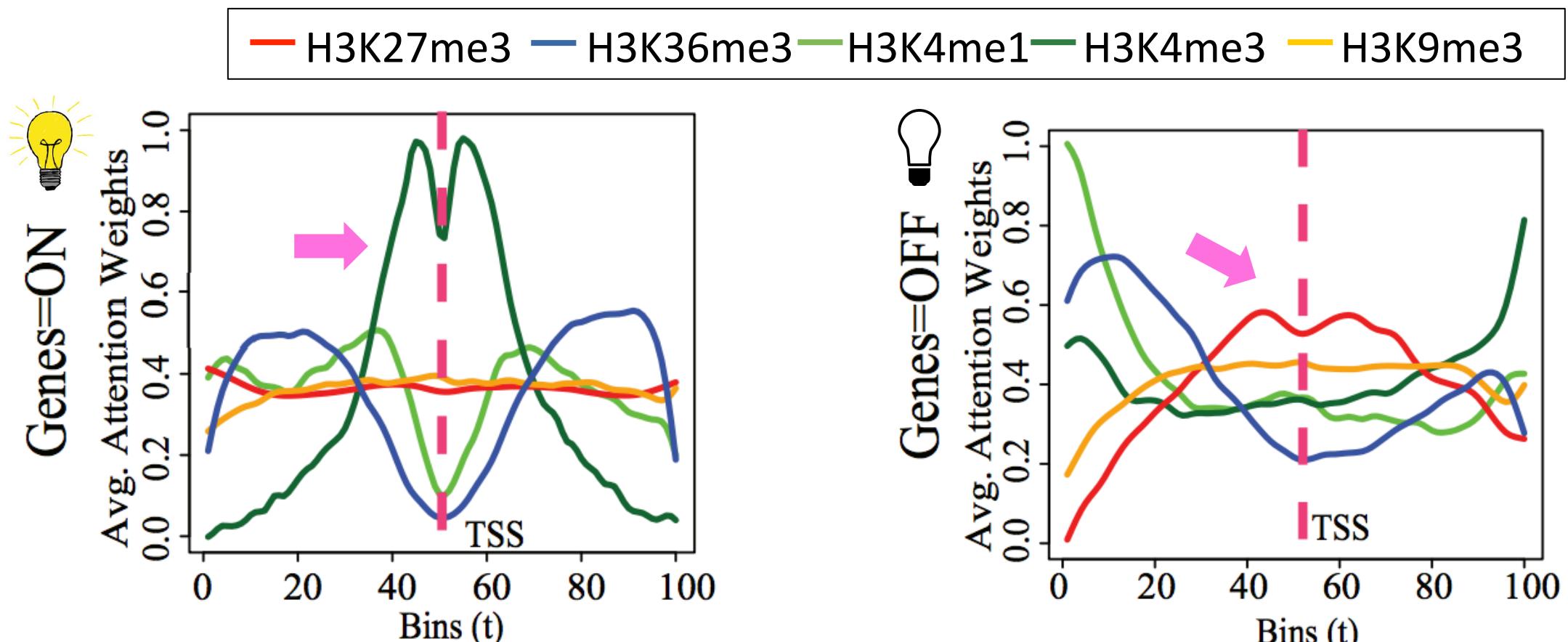


Results

	Baselines		Our Model
Models	DeepChrome (CNN)	RNN	AttentiveChrome
Mean	0.8008	0.8052	0.8115
Median	0.8009	0.8036	0.8123
Max	0.9225	0.9185	0.9177
Min	0.6854	0.7073	0.7215
Improvement over DeepChrome (out of 56 cell types)		36	49

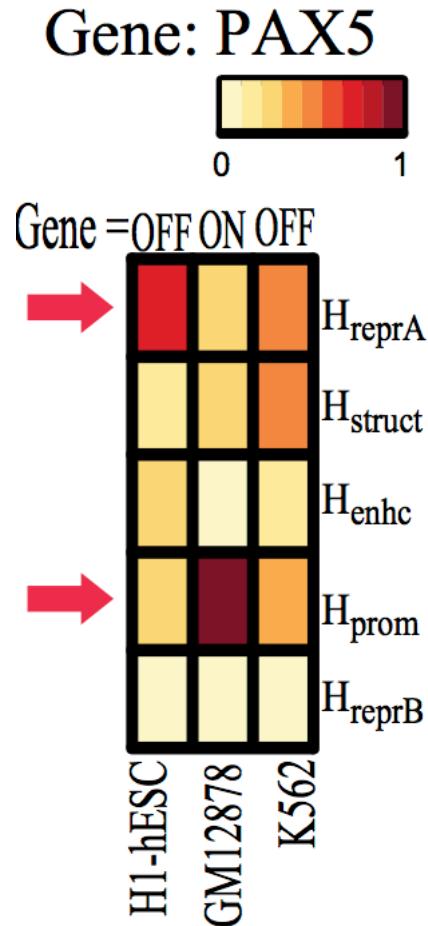
Results: Local level attention

CELL TYPE: GM12878 (Blood Cell)



(1) What positions are important?

Results: Global level attention



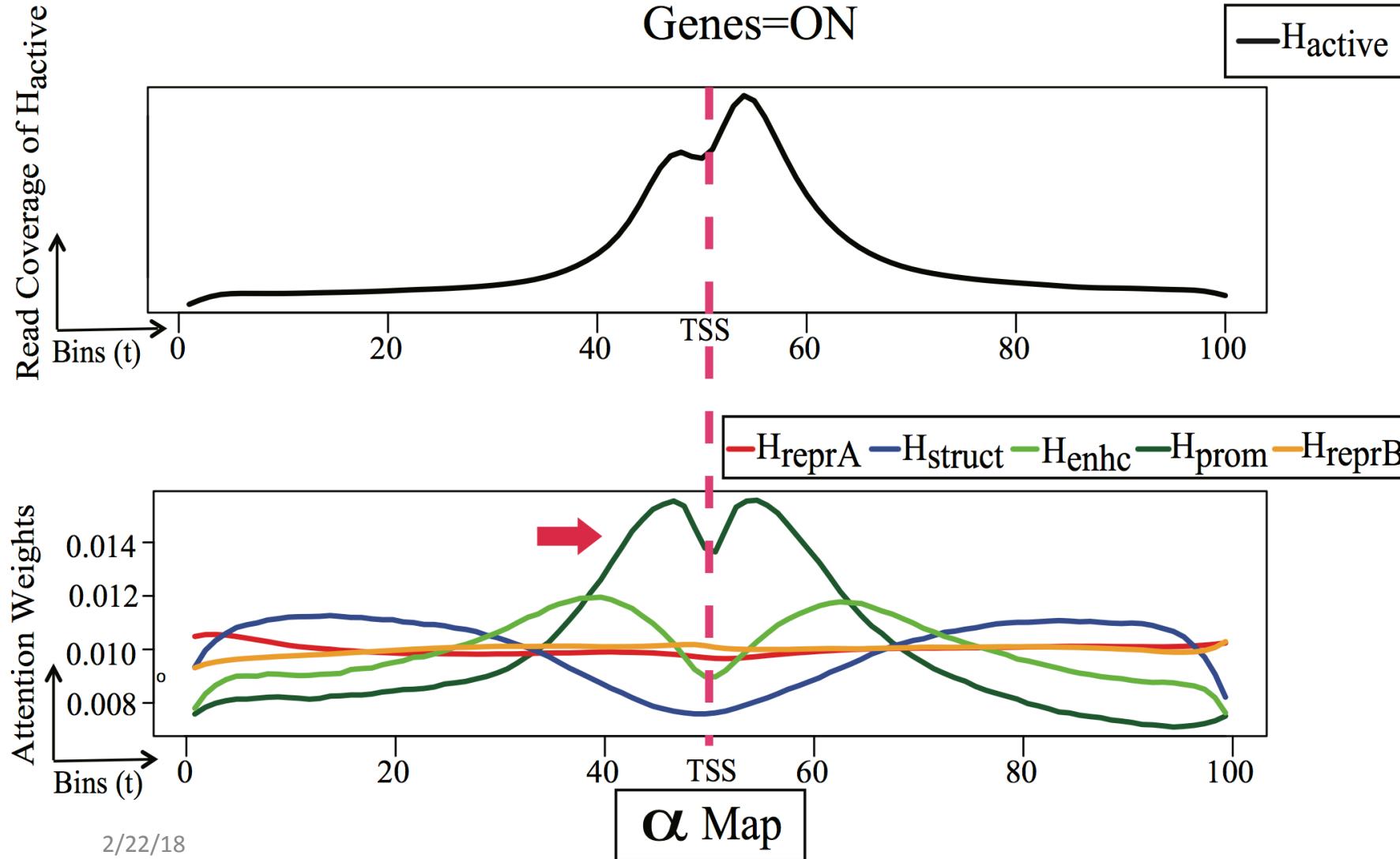
➤ An important differentially regulated gene (PAX5) across three blood lineage cell types:

- 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)



- 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

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

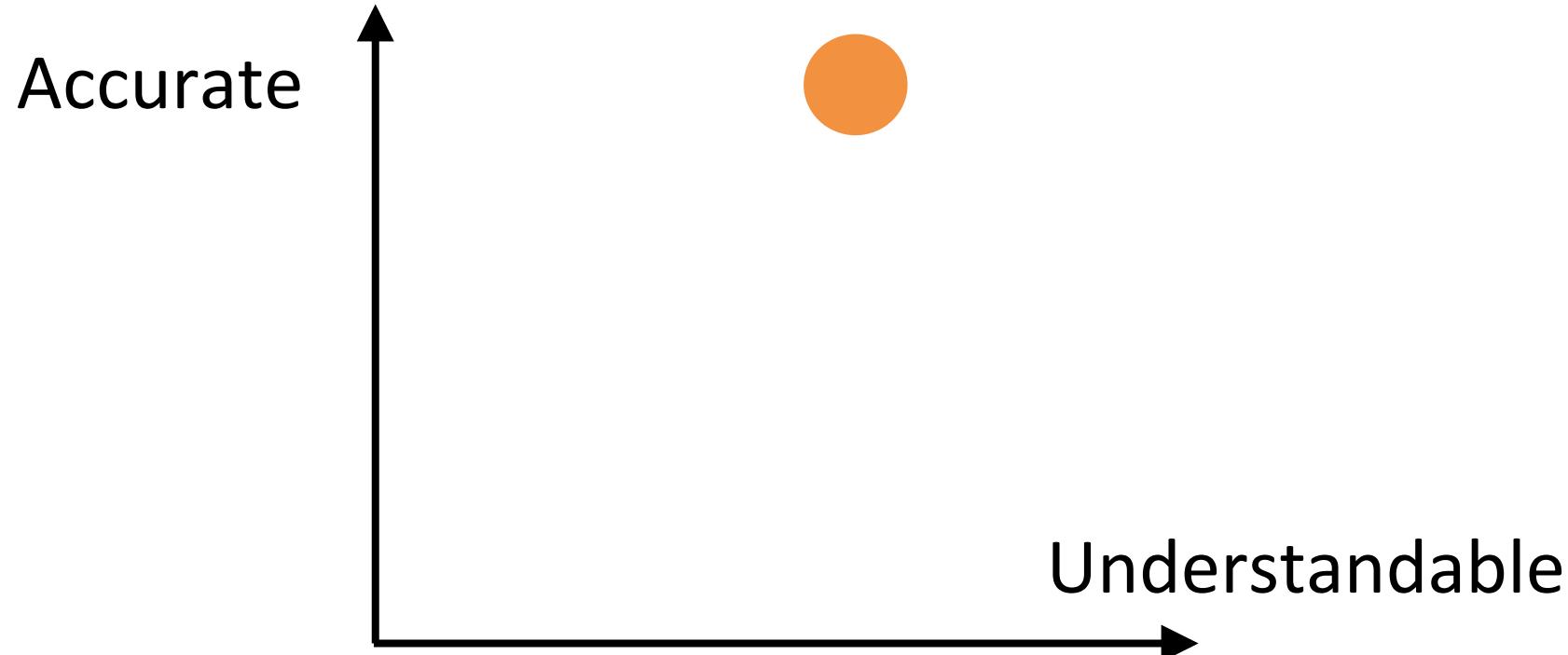
Histone Modification (HM) Mark	Renamed as	Functional Category
H3K4me3	H_{prom}	Promoter mark
H3K4me1	H_{enhc}	Distal Enhancer mark
H3K36me3	H_{struct}	Structural mark
H3K9me3	H_{reprA}	Repressor mark
H3K27me3	H_{reprB}	Repressor mark

Summary

code available at: deepchrome.org

➤ Attentive DeepChrome

- Both accurate and interpretable
- Novel implementation of deep attention mechanism
- Importance analysis at both HM and HM-HM level



Today

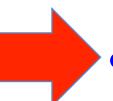
- Deep Learning
 - Basics
 - History
 - Why is this a breakthrough ?
 - Recent trends

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

- Deep Learning Tools for analyzing Sequential Data about Regulation:

- DeepChrome
- AttentiveChrome
- DeepMotif

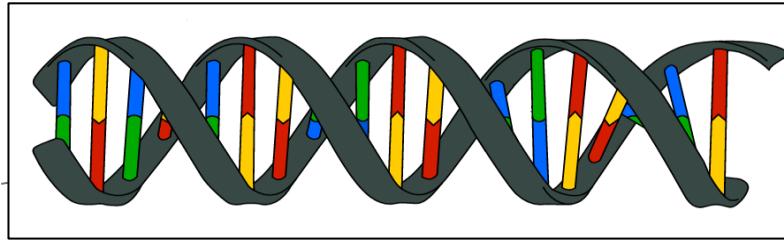
<https://www.deepchrome.org>



- More Tools: learning graphs from data

<https://www.jointggm.org>

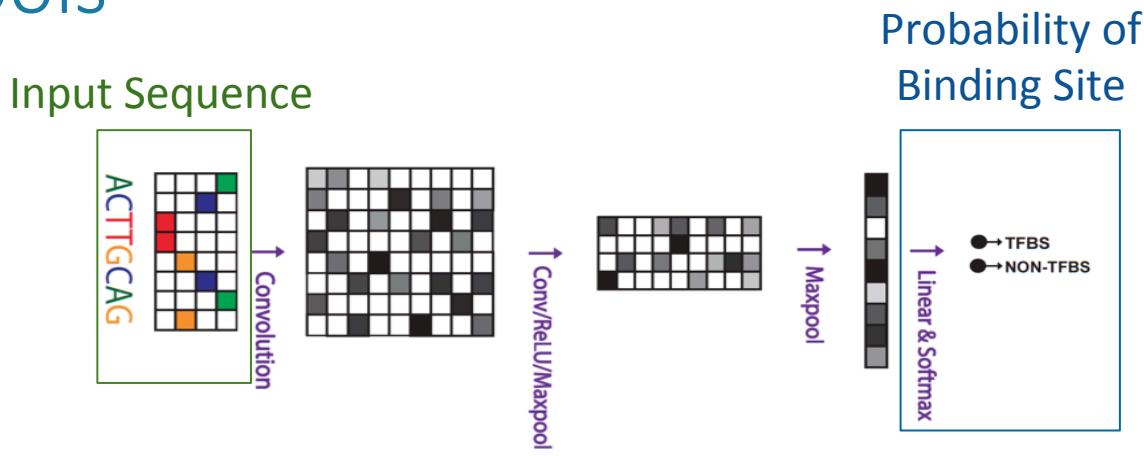
Task: Sequence Based Functional Annotation Tasks



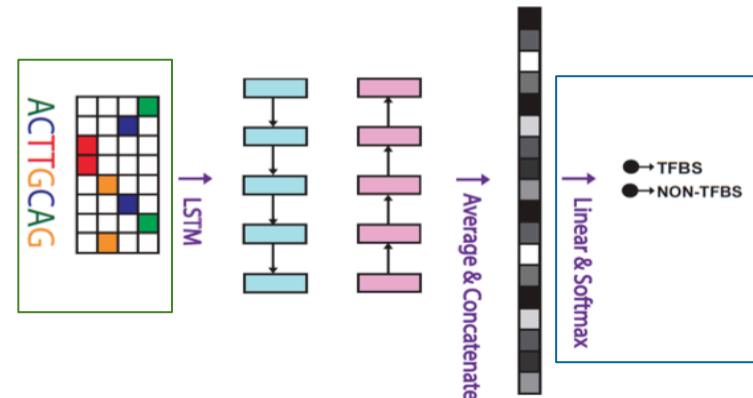
TF1 TF2 TF3 TF1 TF2 TF3 TF2
 diamond diamond diamond diamond diamond diamond diamond
...GCG**ACGAATCG...**AAC**GATATGCT...**CAT**ATCATTTC...**TGT**CAAG...**CTCG**GAGTC...**TAT**CAA**G**CTG...**

Literature: Various DNN Tools

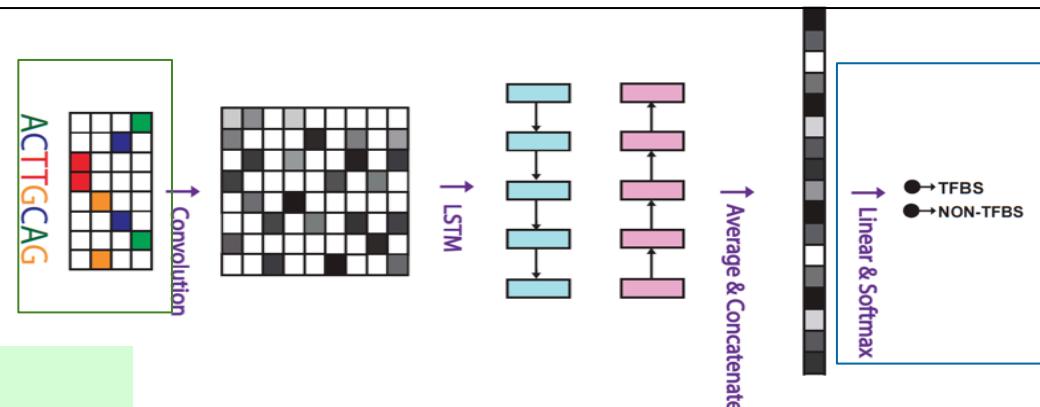
1. Convolutional (CNN)



2. Recurrent (RNN)



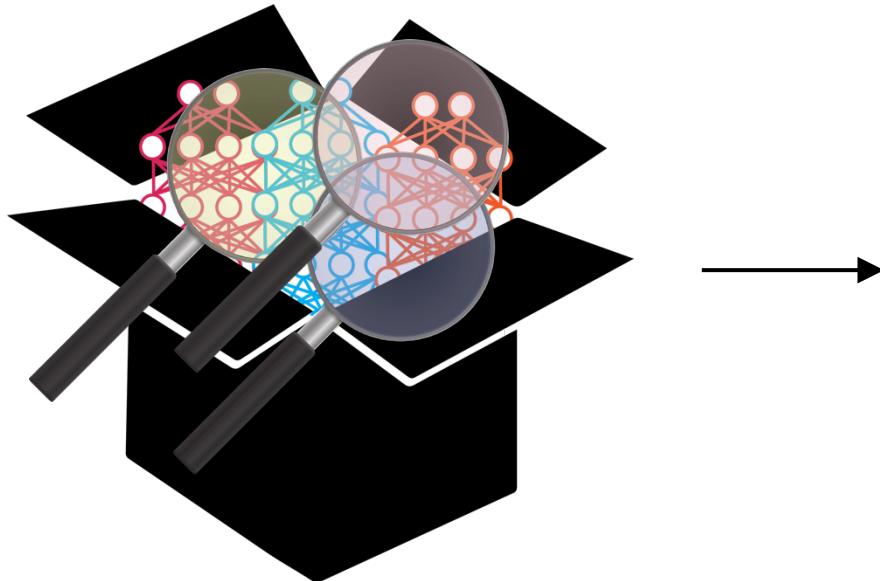
3. Convolutional-Recurrent (CNN-RNN)



DeepSEA, DeepBind, BASSET, DanQ,

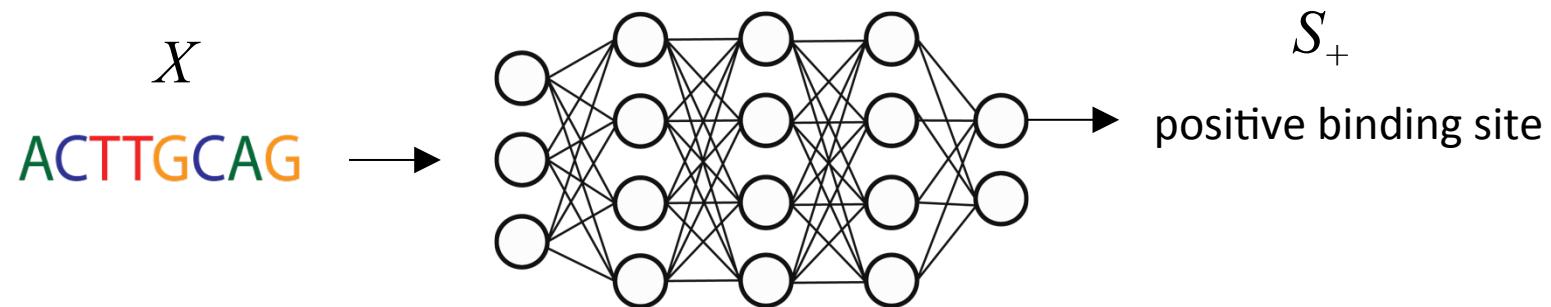
Deep Motif Dashboard: Understand DNNs by Post Analysis

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



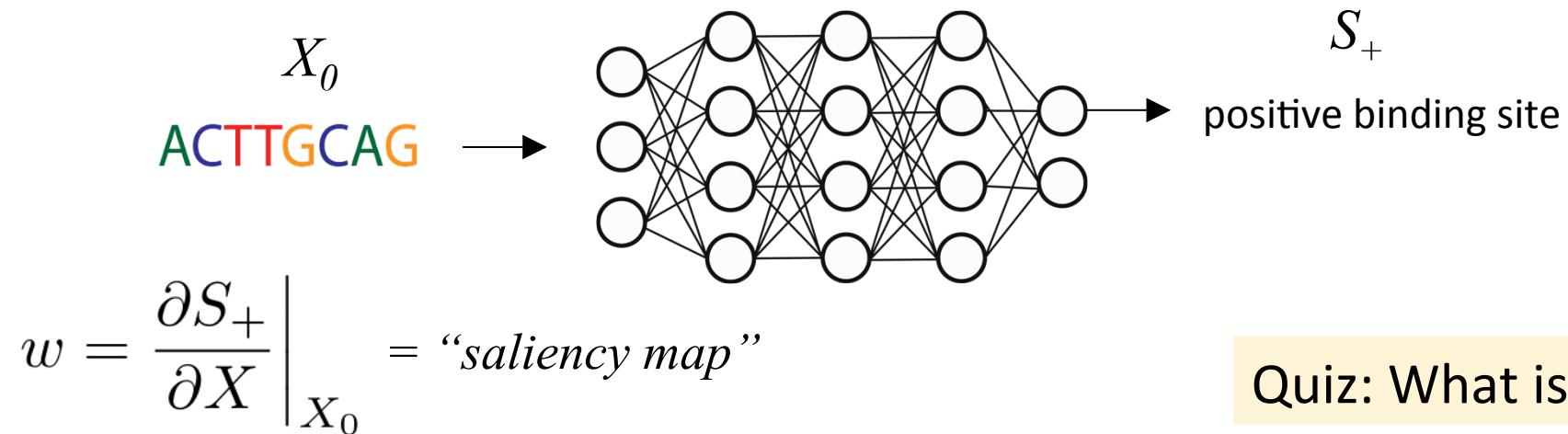
1. Saliency Maps
 2. Temporal Output Values
 3. Class Optimization

1. Saliency Map



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

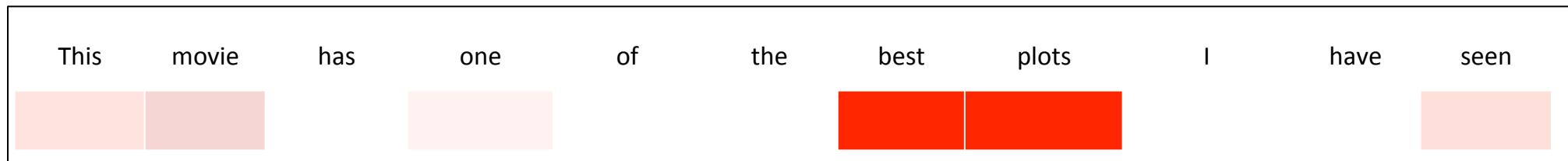
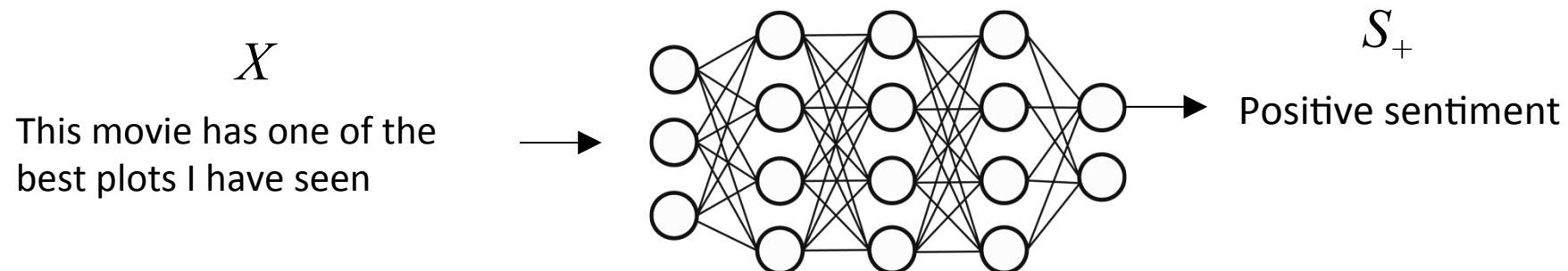
1. Saliency Map



Quiz: What is gradient?

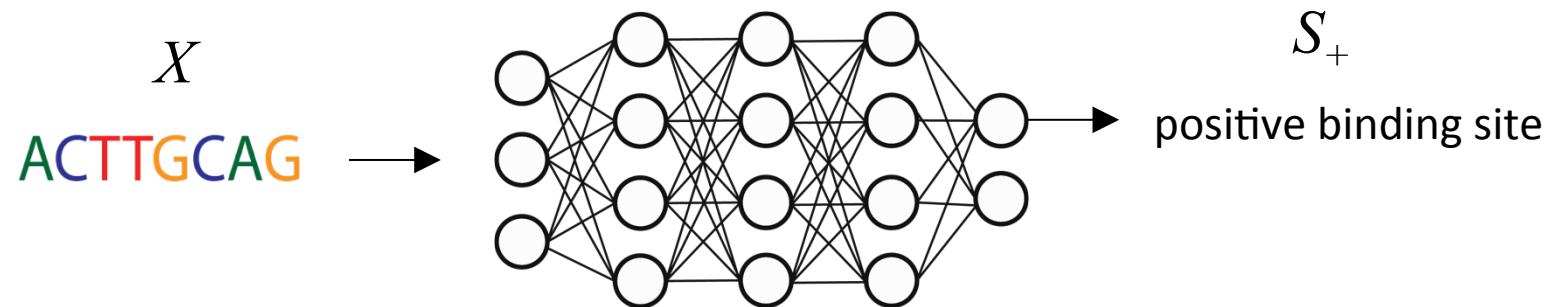
[Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR 2013](#)

1. Saliency Map



█ = important for classification

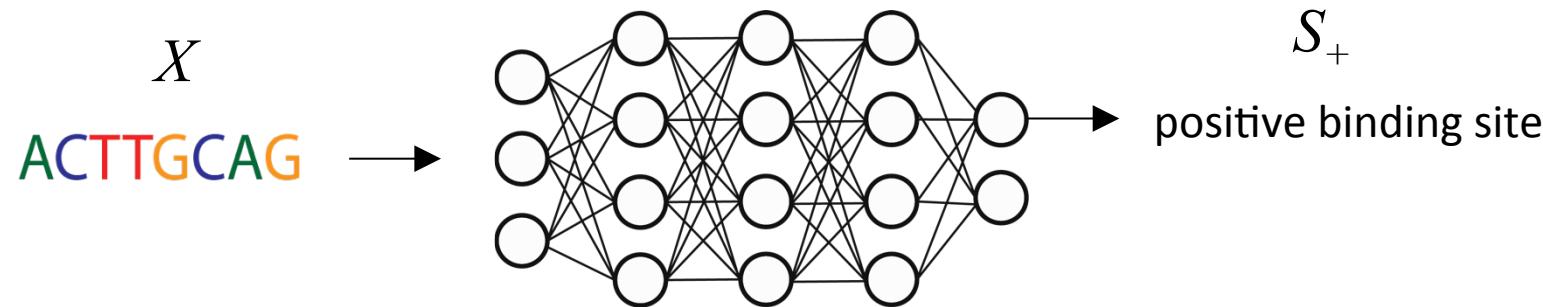
1. Saliency Map



Positive Test Sequence	TGCTCGCATCC TATTGGCCACGTTAGTCACATGGCCCCACCTGGCTGCAAAGCACGCTGGAAACGTAGTC TTTCTT
Saliency Map	

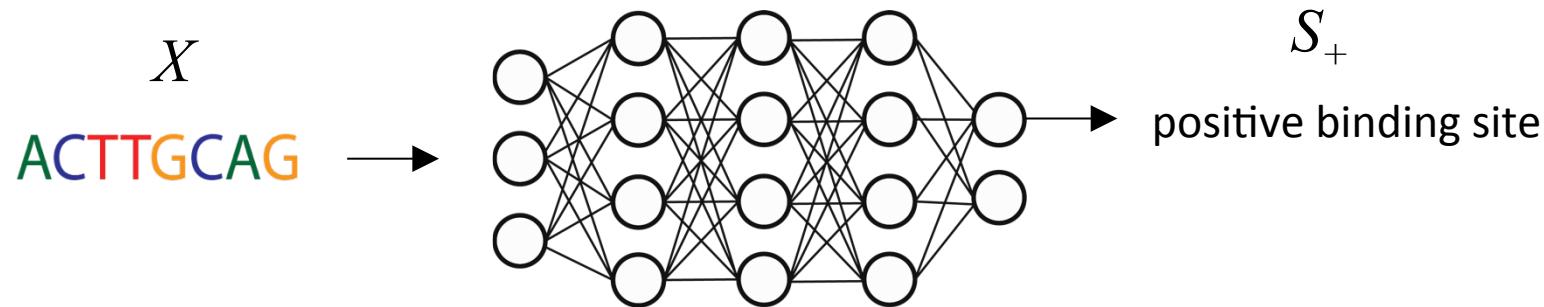
█ = important nucleotide for prediction

2. Temporal Output Values



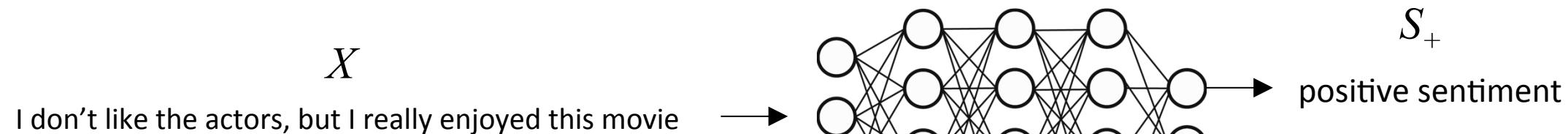
What are the model's predictions at each timestep of the DNA sequence?

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.

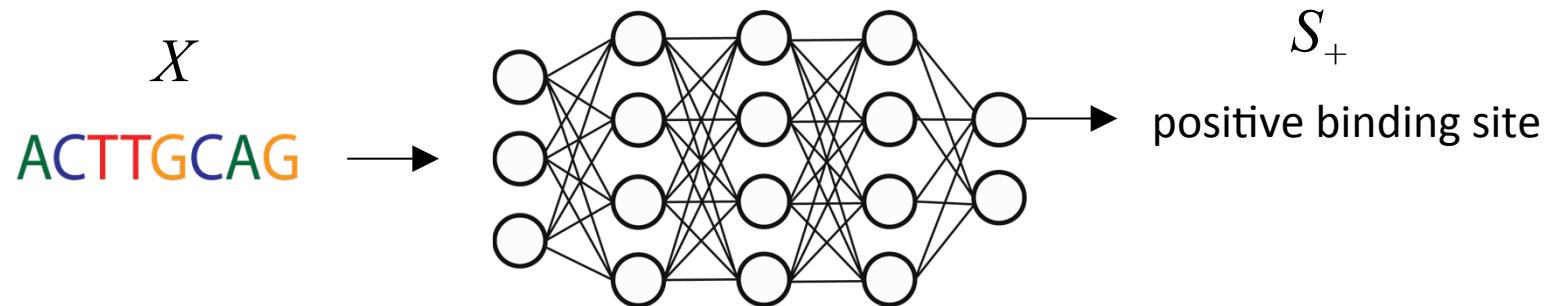
2. Temporal Output Values



 = negative sentiment

 = positive sentiment

2. Temporal Output Values

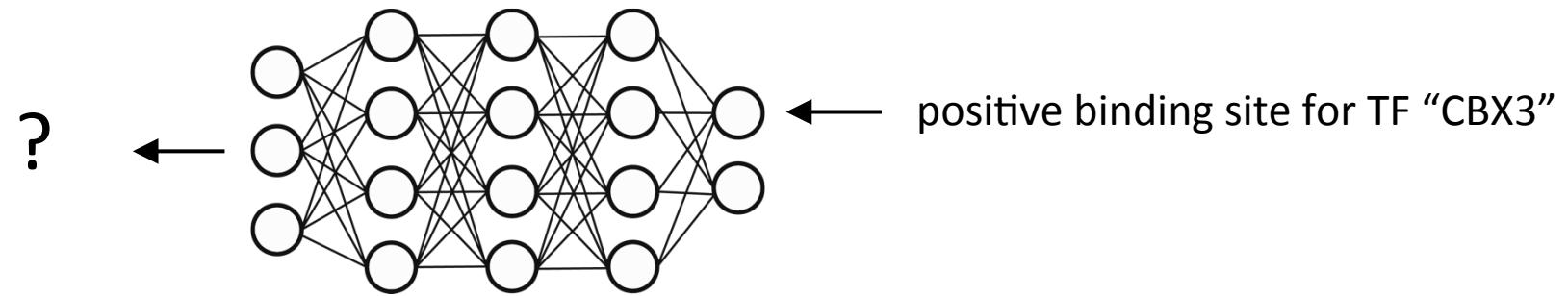


Positive Test Sequence	CTT CTGCTCGCATCCTATTGCCACGTTAGTCACATGGCCCCACCTGGCTGCAAAGCACGCTGGAAACGTAGTCTTC TT
RNN Forward Output	
RNN Backward Output	

= negative binding site prediction

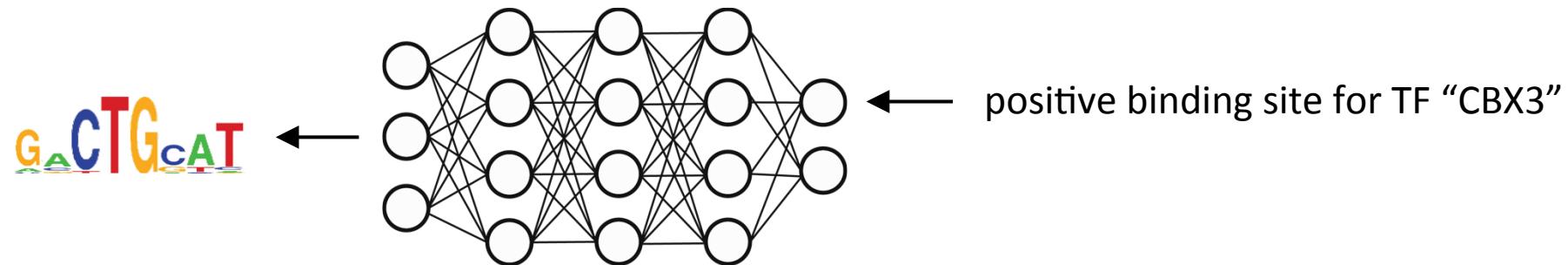
= positive binding site prediction

3. Class Optimization



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

3. Class Optimization

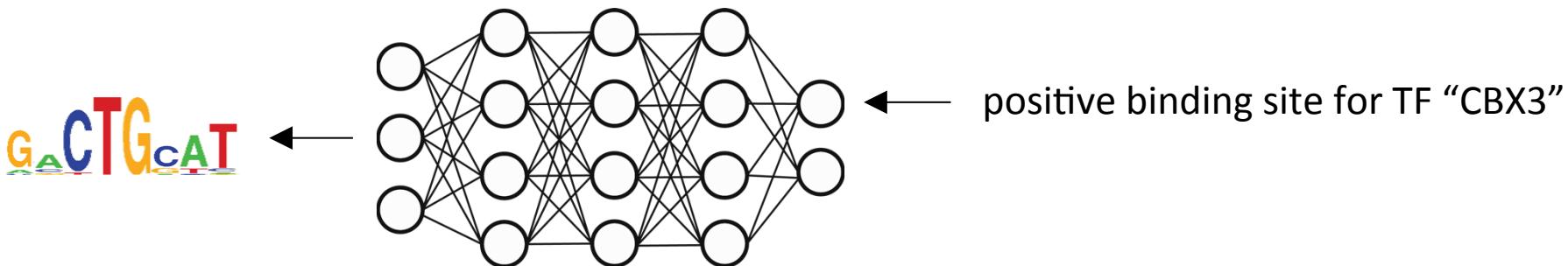


$$\arg \max_X S_+(X) + \lambda \|X\|_2^2$$

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
 - 2. Temporal Output Values
- TF Specific {
 - 3. Class Optimization

code available at: **deepmotif.org**

Related Work to Understand DNN

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

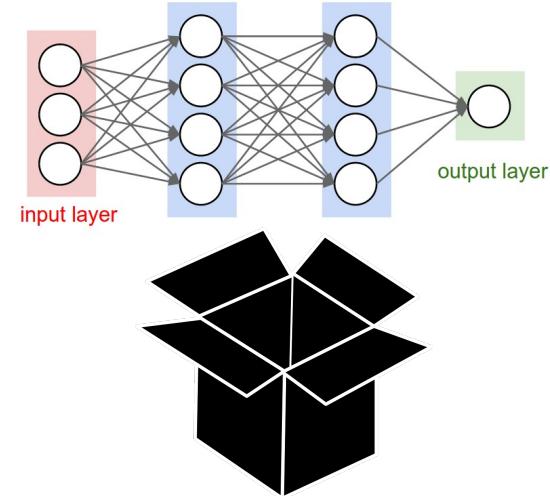
Temporal Output Values

Saliency Map

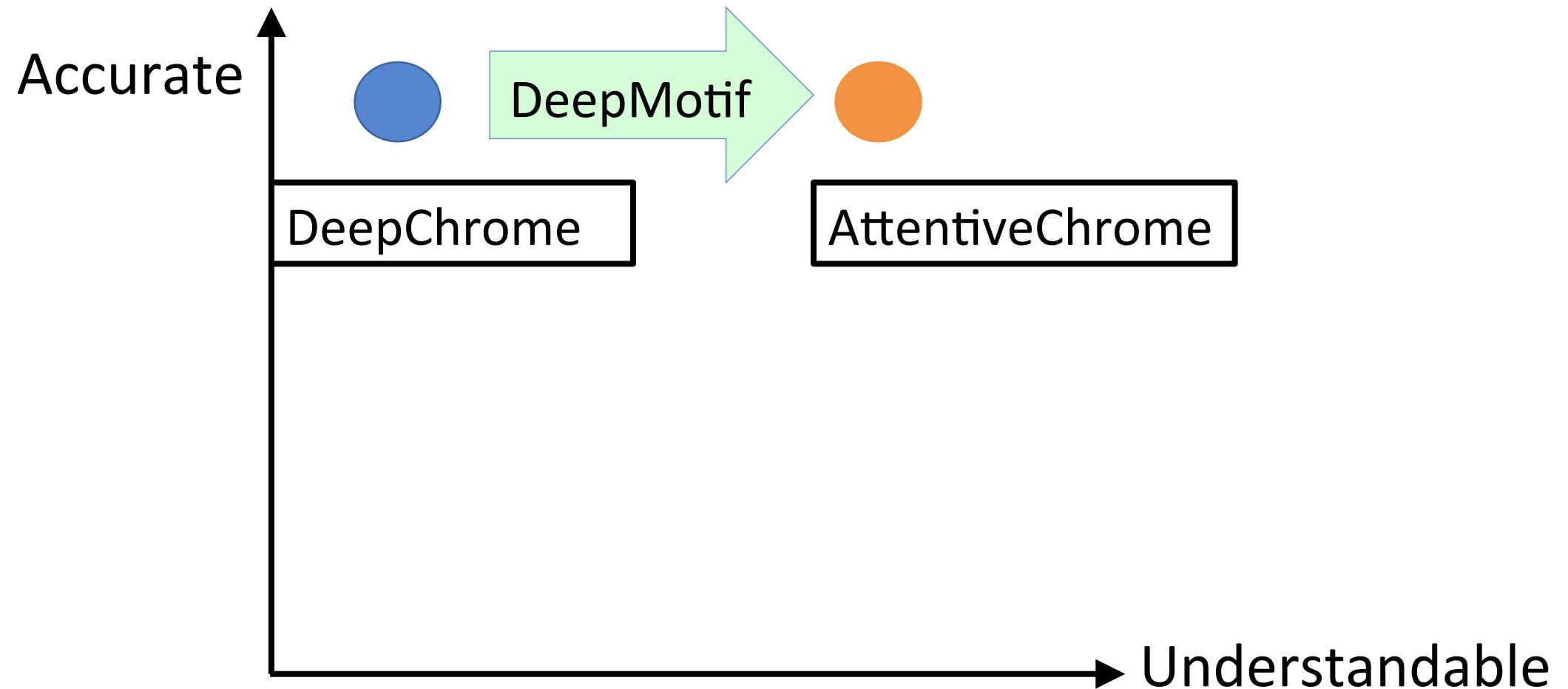
Class Optimization

DeepLift

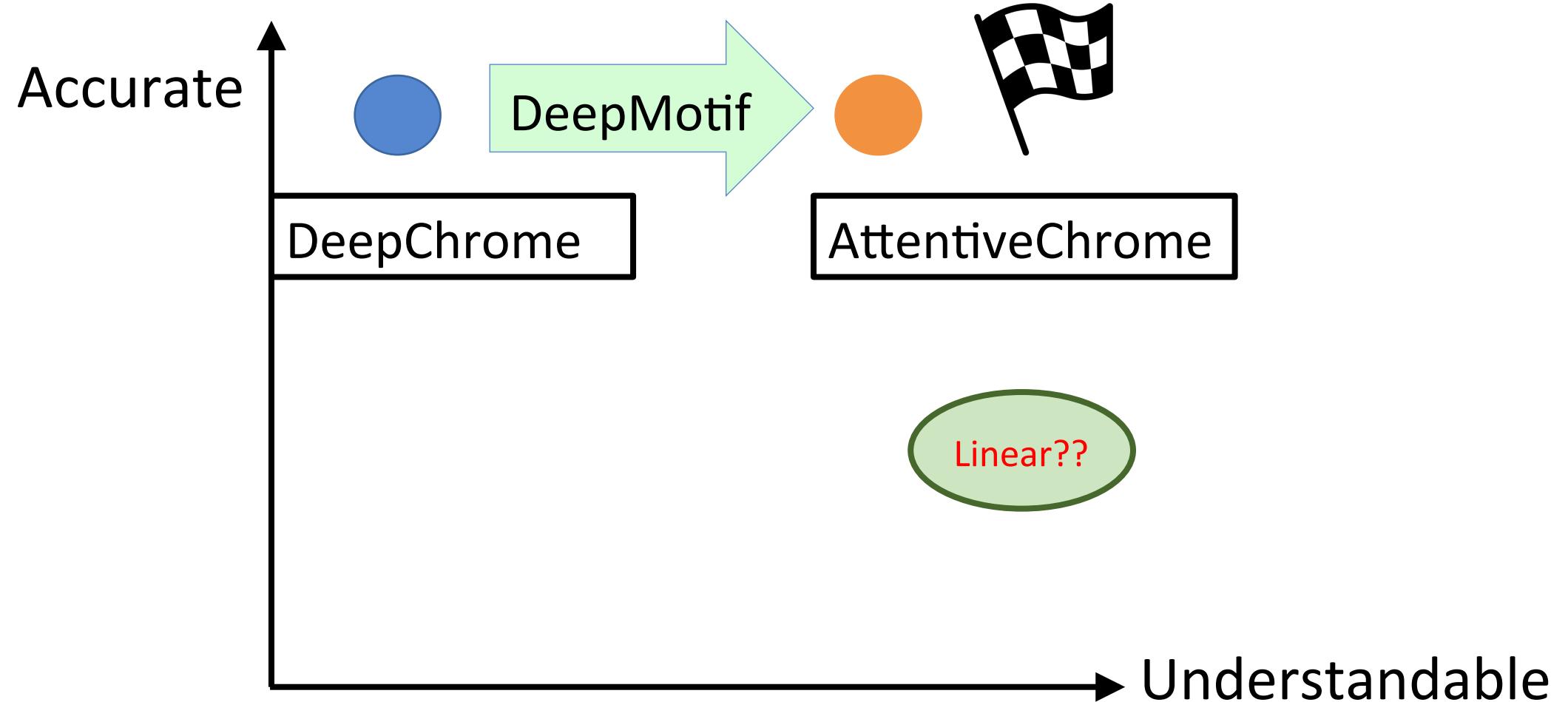
Influential Func / ICML27 Best Paper



Summary:



Summary:



Today

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

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

- Deep Learning Tools for analyzing **Sequential Data** about Regulation:

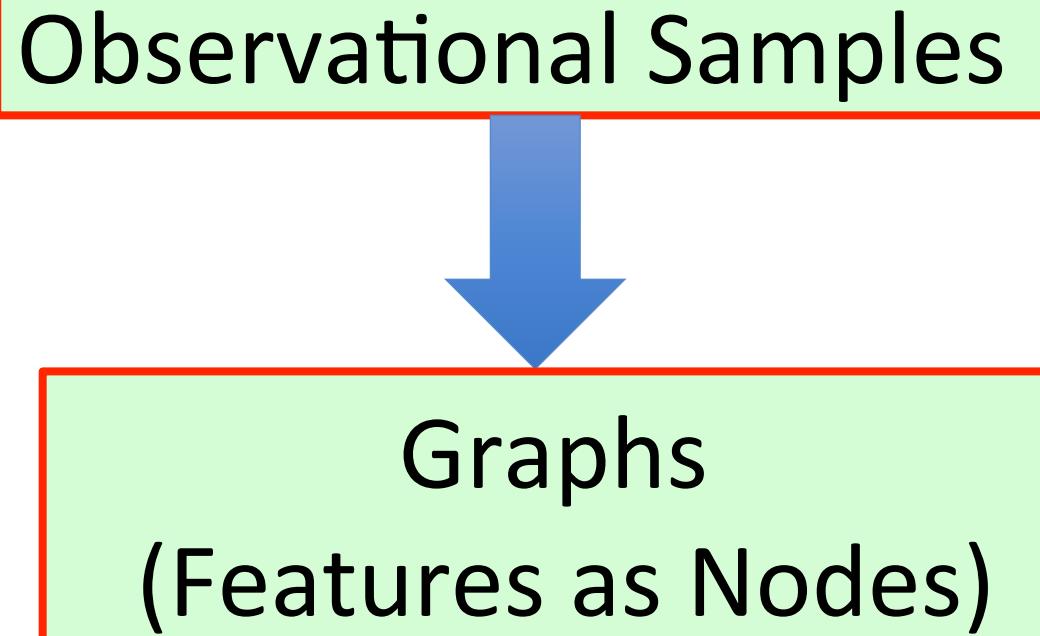
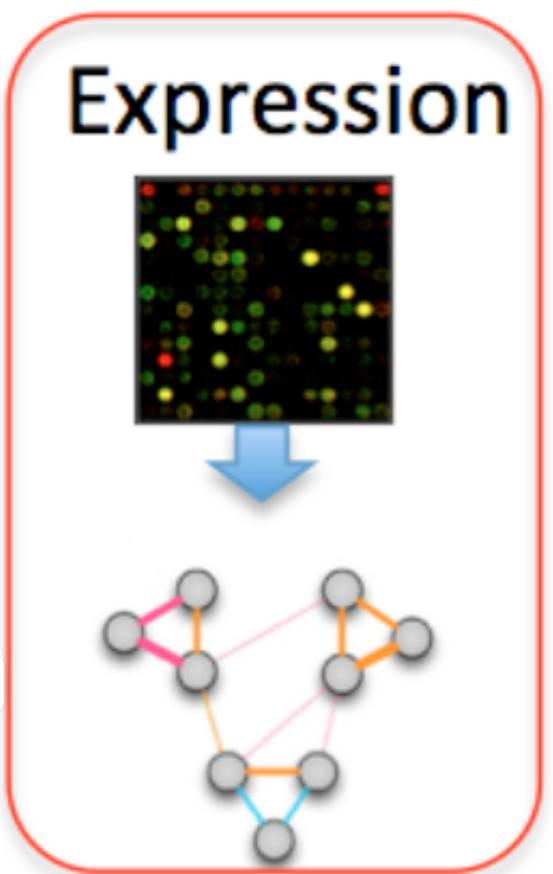
- DeepChrome
- AttentiveChrome
- DeepMotif

<https://www.deepchrome.org>

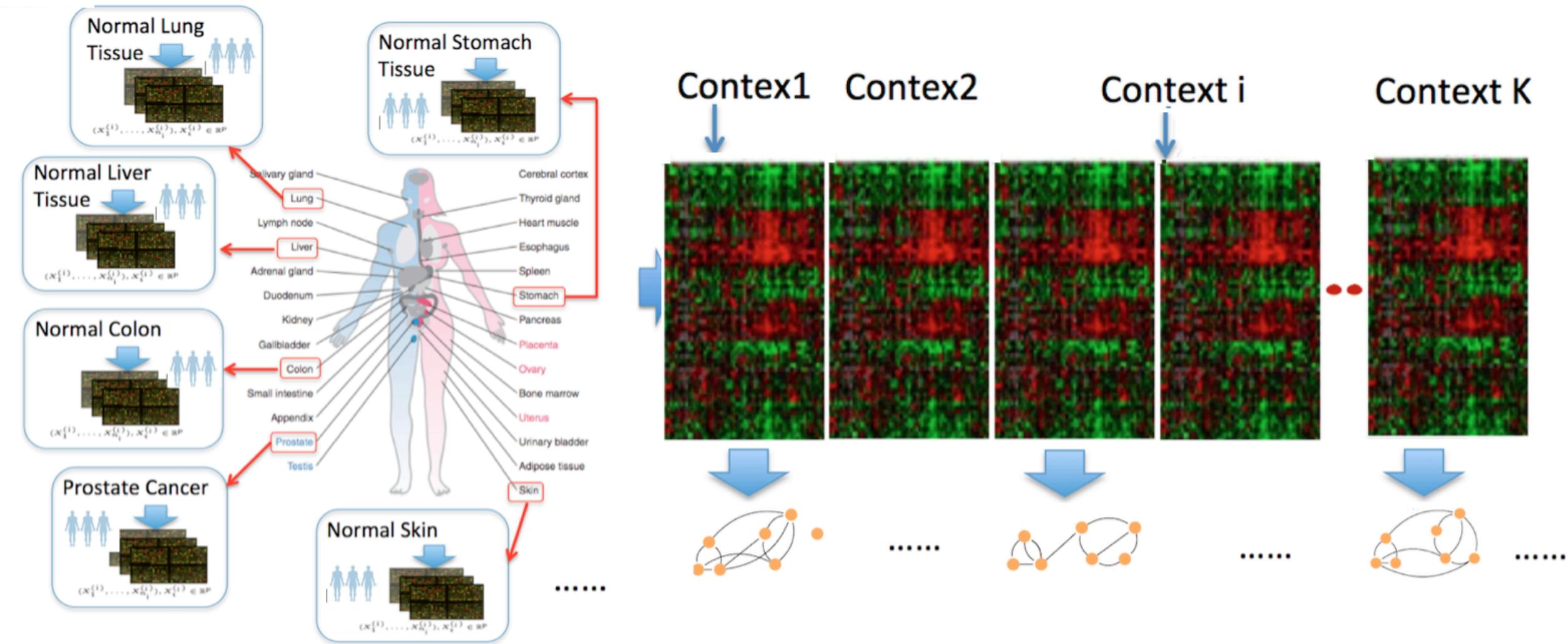
- More Tools: learning graphs from data

<https://www.jointggm.org>

Fast and Scalable Joint Estimators for Learning Sparse Gaussian Graphical Models from Heterogeneous Data

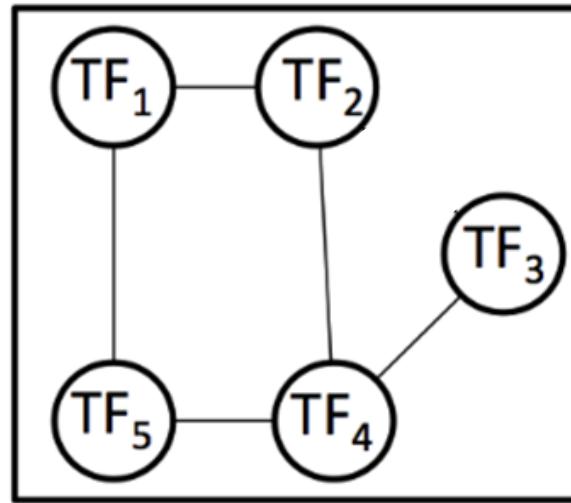


Motivation: Graphs vary across contexts

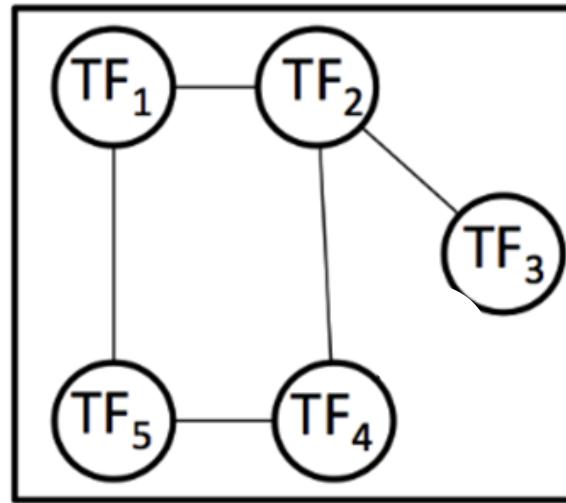


Motivation: Graphs vary across contexts

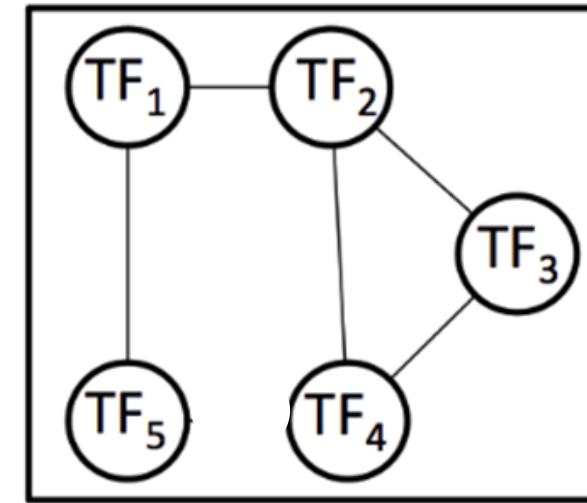
- Different but related TF co-binding patterns in the form of graphs
- e.g., estimated from Chi-Seq



Normal

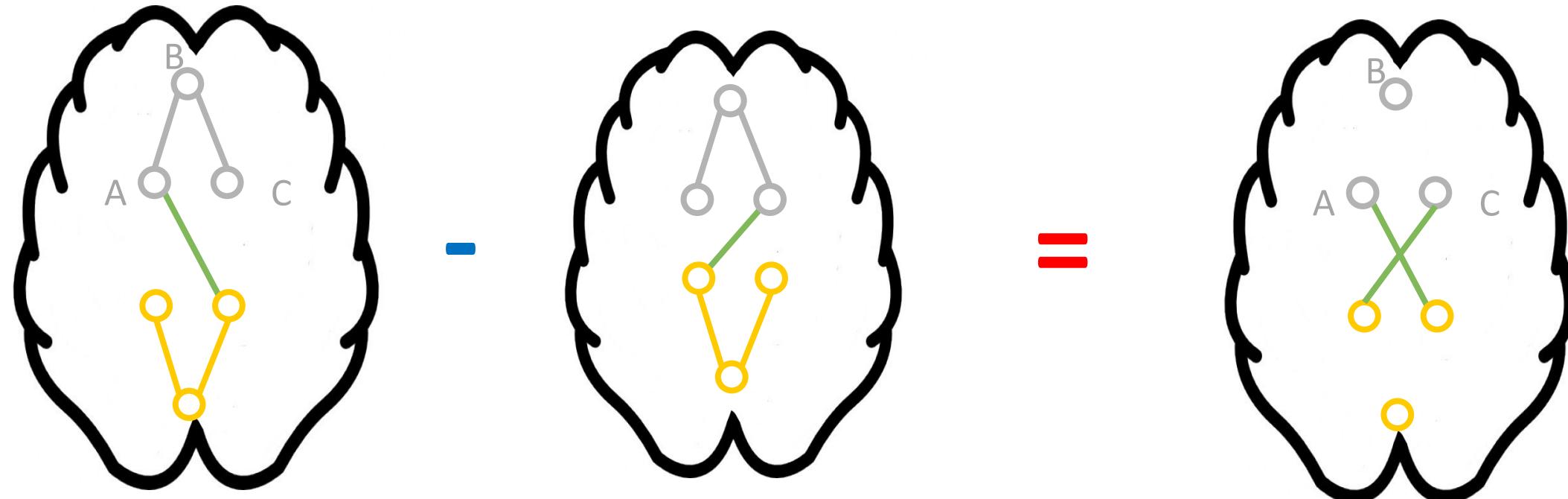


Leukemia



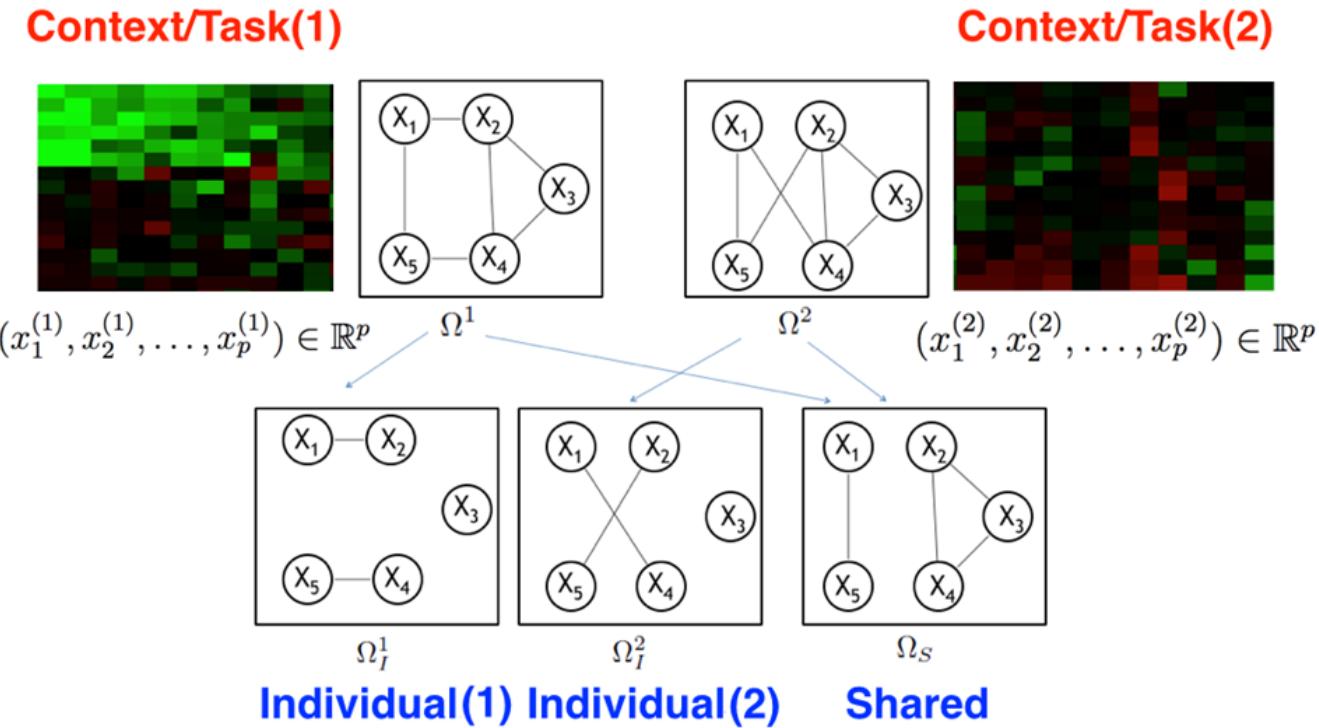
Stem

Task I: Learning sparse changes between two graphs



- For example:
 - Find differences in the brains of people with diseases, *e.g.* Autism, Alzheimer's
 - Used for understanding
 - Used for diagnosis

Task II: Learning both shared and context-specific graphs explicitly and simultaneously



- Able to Know both
 - House keeping interactions
 - Context-specific networks

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 same iteration number T

Traditional Optimization Method

---- Block Coordinate Descent :

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

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

more than **2 billion years**

Current Optimization:

---- Still needs SVD for each covariance matrix

SVD for the matrices needs

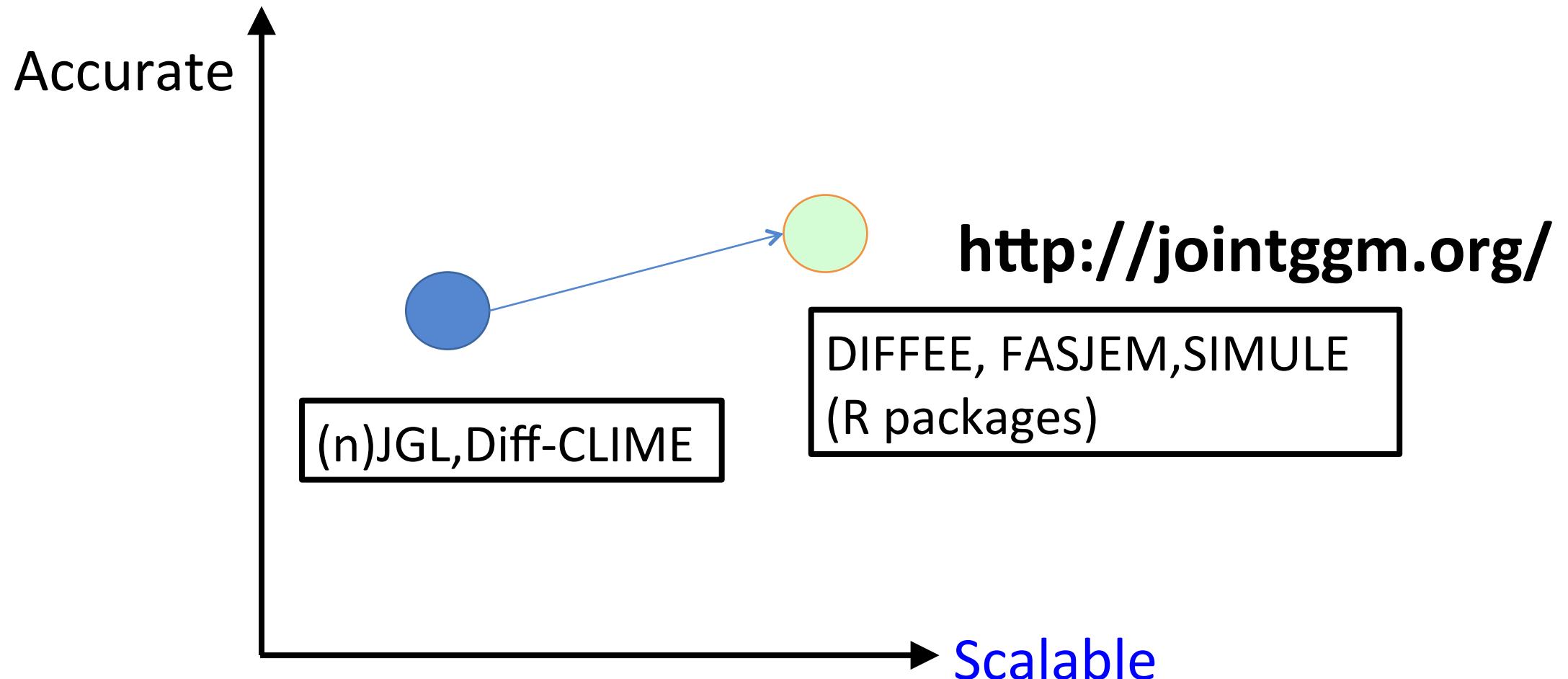
$$O(K p^3) \rightarrow 3.5 \text{ days} \\ / \text{Itera}$$

Our Tools

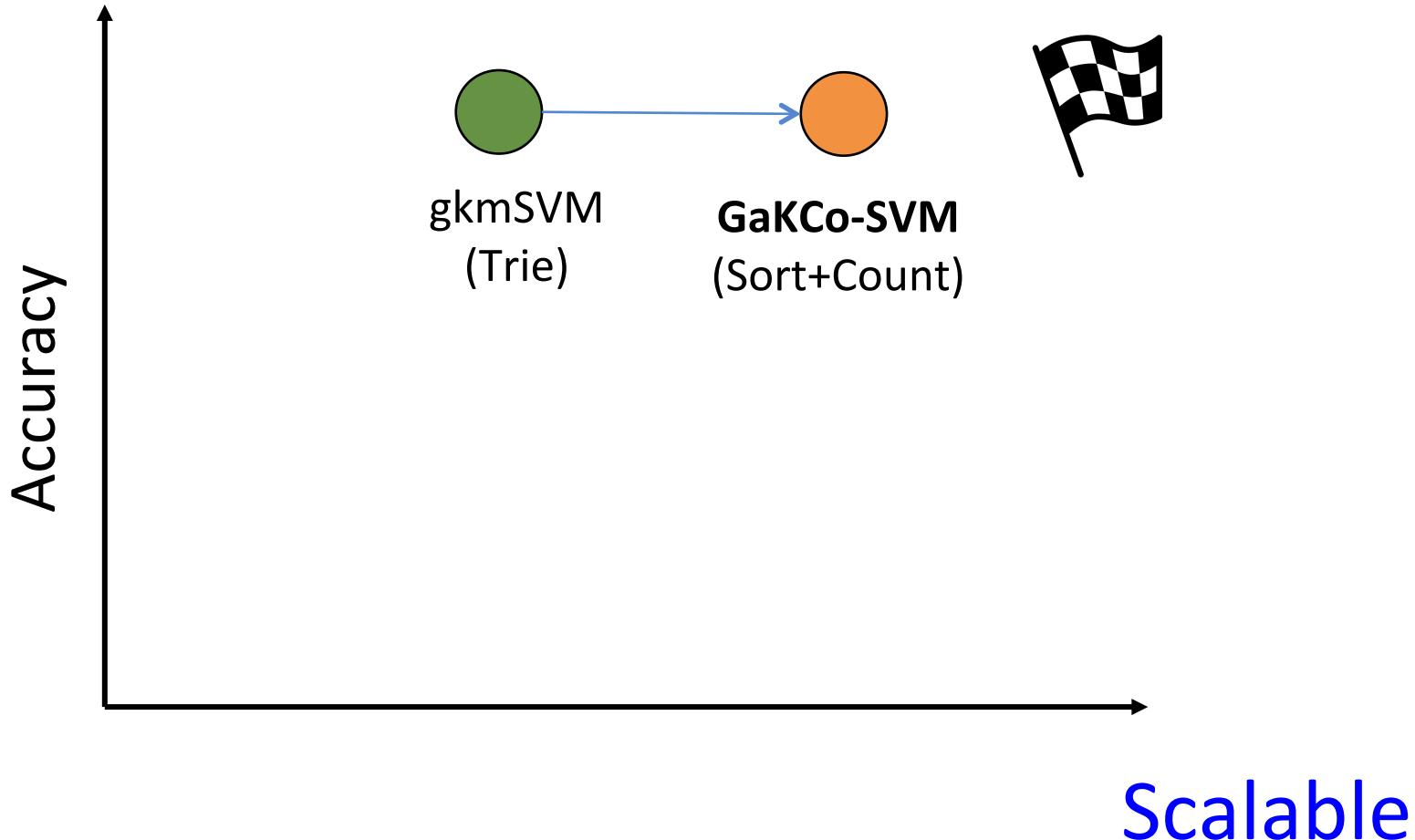
- Fast and scalable estimators for joint graph discovery from heterogeneous samples
- Parallelizable algorithms
- Sharp convergence rate (sharp error bounds)

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

Summary

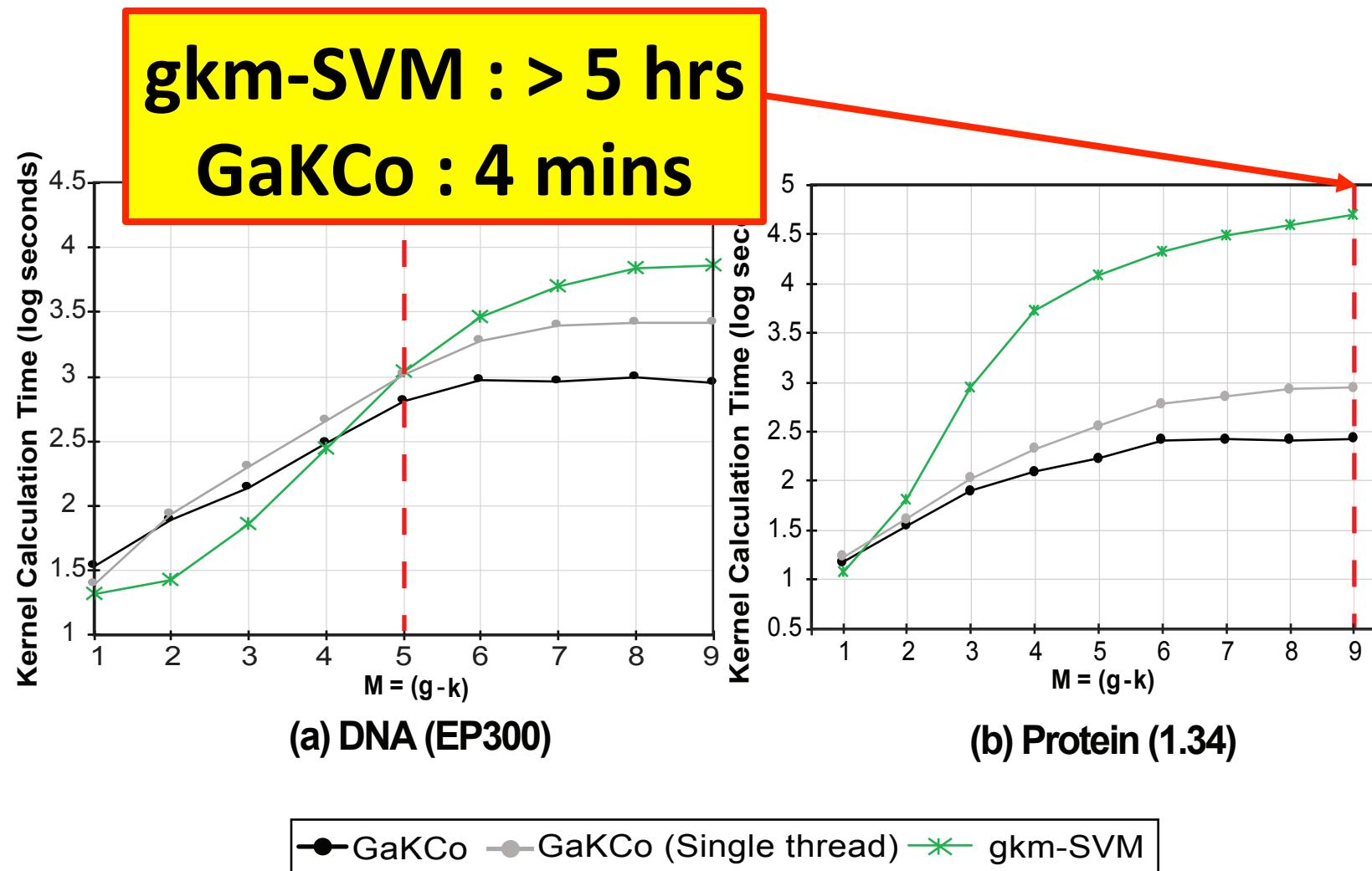


One more scalable tool: GaKCo-SVM for sequence classification

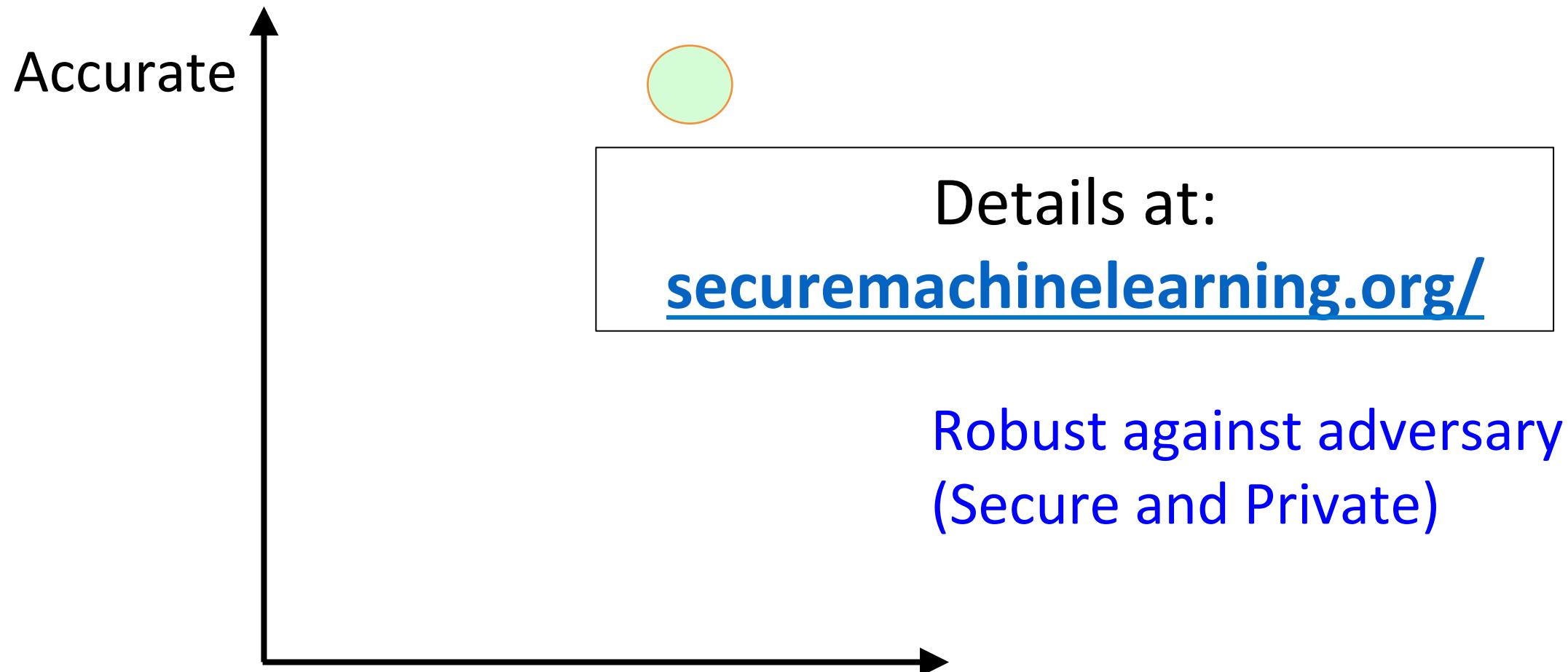


R. Singh, et al. "Gakco: a fast gapped k-mer string kernel using counting." *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. (2017)

Scales well with increasing Σ and m



Tools for Robustness of Machine Learning



Adversarial Examples to Fool DNN Models



“panda”



+

$0.007 \times [\text{noise}]$

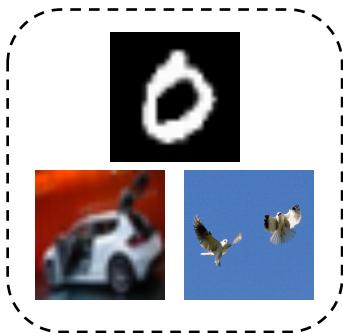
=



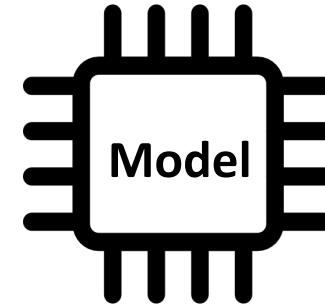
“gibbon”

Example from: Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy.
Explaining and Harnessing Adversarial Examples. ICLR 2015.

EvadeML-Zoo: a benchmark toolbox



- MNIST
- CIFAR-10
- ImageNet



- CNN
- DenseNet
- MobileNets



- FGSM, BIM,
- JSMA, DeepFool,
- CW₂, CW_i, CW₀



- Feature Squeezing

Acknowledgements



Ritambhara Singh



Jack Lanchantin



Weilin Xu



Arshdeep Sekhon



Beilun Wang

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

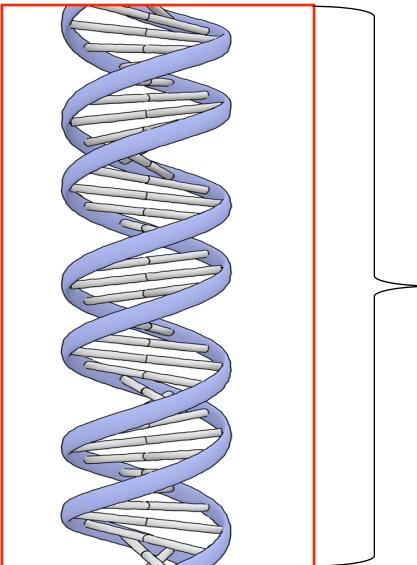
UVA Computer Science Dept. Security Research Group: Prof. David Evans



Thank you

Backup

DNA and Diseases



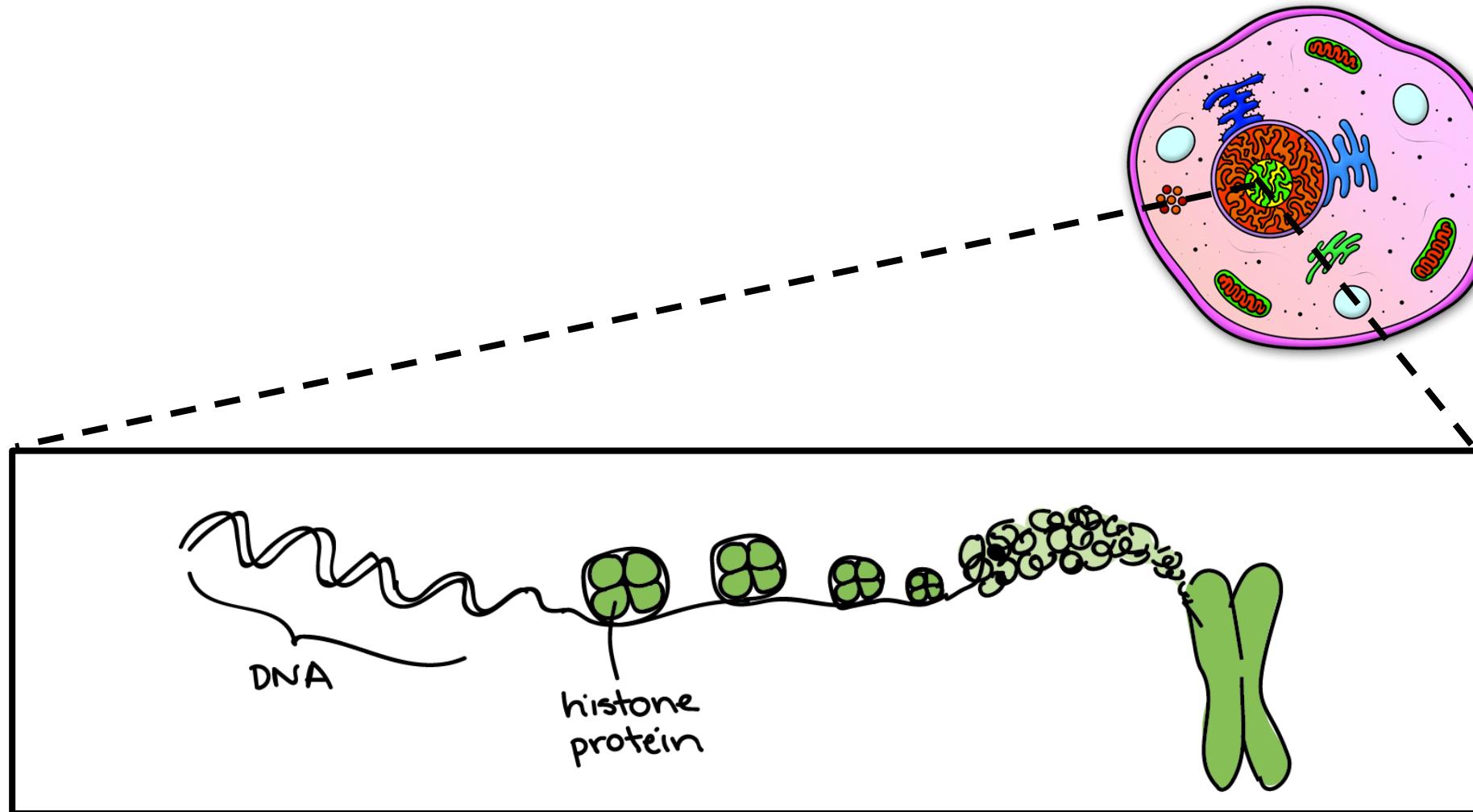
DNA

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

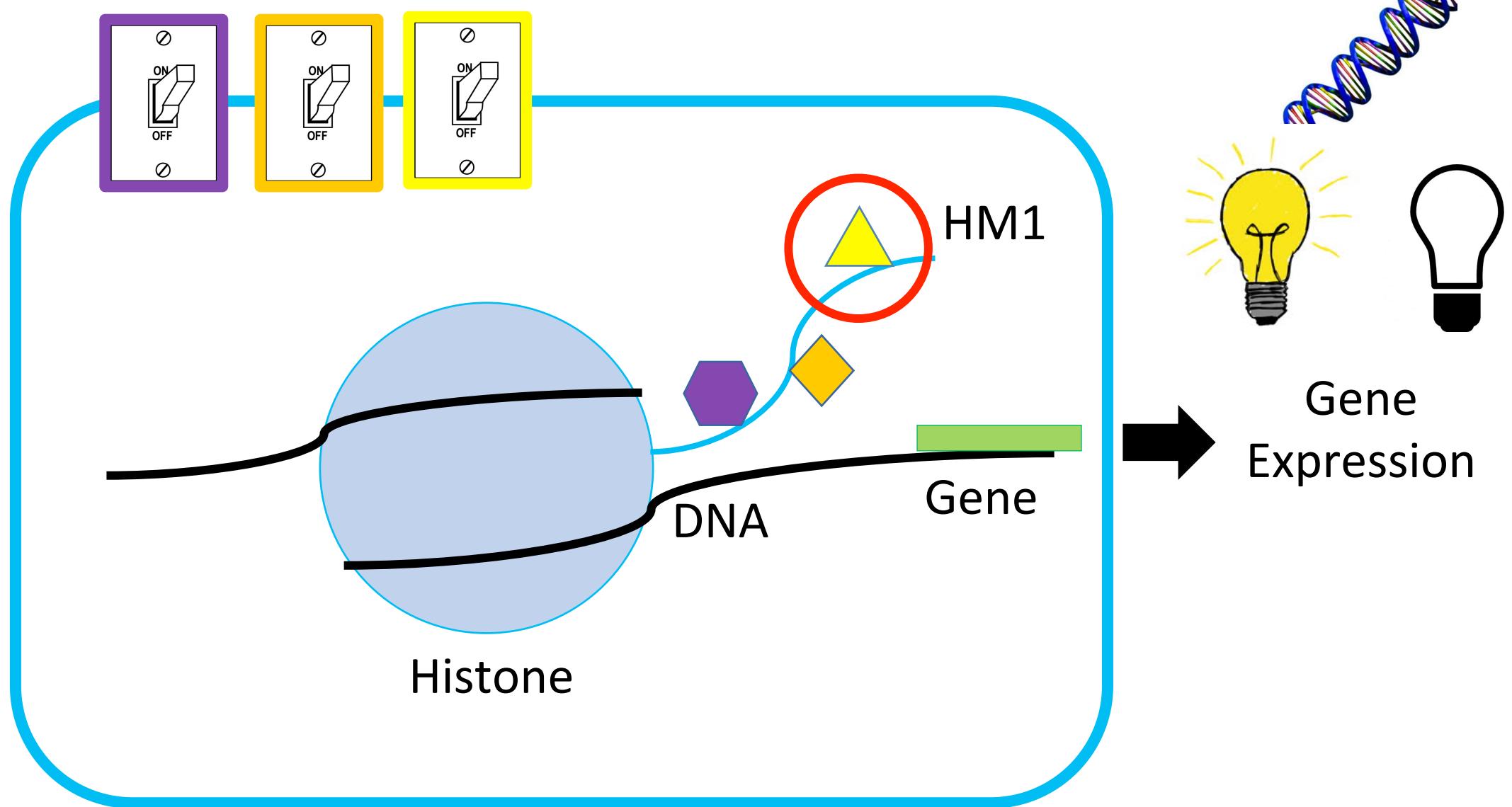
Epigenetics
**“Environment
of the DNA”**

Histone Proteins

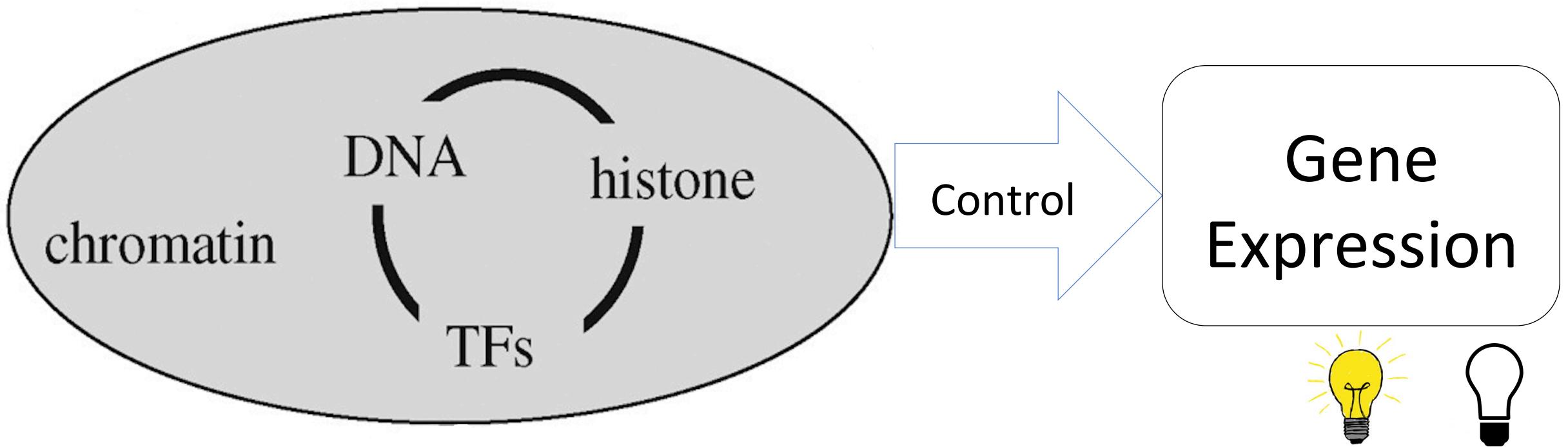
CELL



Histone Modifications (HM)



Chromatin



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 be defined 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”.