

Deep learning for genomics

Matt Ploenzke

BST 261: Data Science II
Department of Biostatistics
Harvard University

A horizontal timeline diagram illustrating the publication years of three deep learning models. A blue arrow points from left to right, representing time. Vertical blue lines drop from the arrow to specific years on the x-axis. The x-axis is labeled with the years 2015, 2016, 2017, and 2018. Above each year, the names of the models and their respective authors are listed.

DeepSEA (Zhou, et al)

DeepBind (Alipanahi, et al)

Basset (Kelly, et al)

2015

2016

2017

2018

DeepSEA (Zhou, et al)

The diagram features a horizontal blue arrow pointing to the right, representing time. Vertical blue lines extend upwards from the arrow to mark specific years: 2015, 2016, 2017, and 2018. Above each year, the names of deep learning models are listed, connected by thin vertical lines to the corresponding year markers.

DeepBind (Alipanahi, et al)

Basset (Kelly, et al)

DanQ (Quang, et al)

2015

2016

2017

2018

DeepSEA (Zhou, et al)

DeepBind (Alipanahi, et al)

Basset (Kelly, et al)

DanQ (Quang, et al)

Separable FC Layer (Alexandari, et al)



2015

2016

2017

2018

DeepSEA (Zhou, et al)

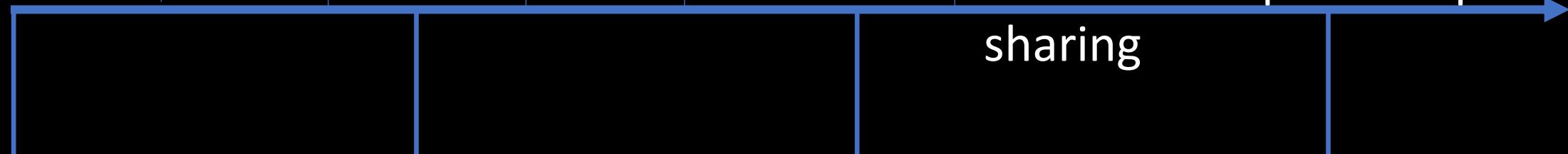
DeepBind (Alipanahi, et al)

Basset (Kelly, et al)

DanQ (Quang, et al)

Separable FC Layers

Reverse-complement parameter
sharing



2015

2016

2017

2018

DeepSEA (Zhou, et al)

DeeperBind

DeepBind (Alipanahi, et al)

Basset (Kelly, et al)

DanQ (Quang, et al)

Separable FC Layers

Reverse-complement parameter
sharing

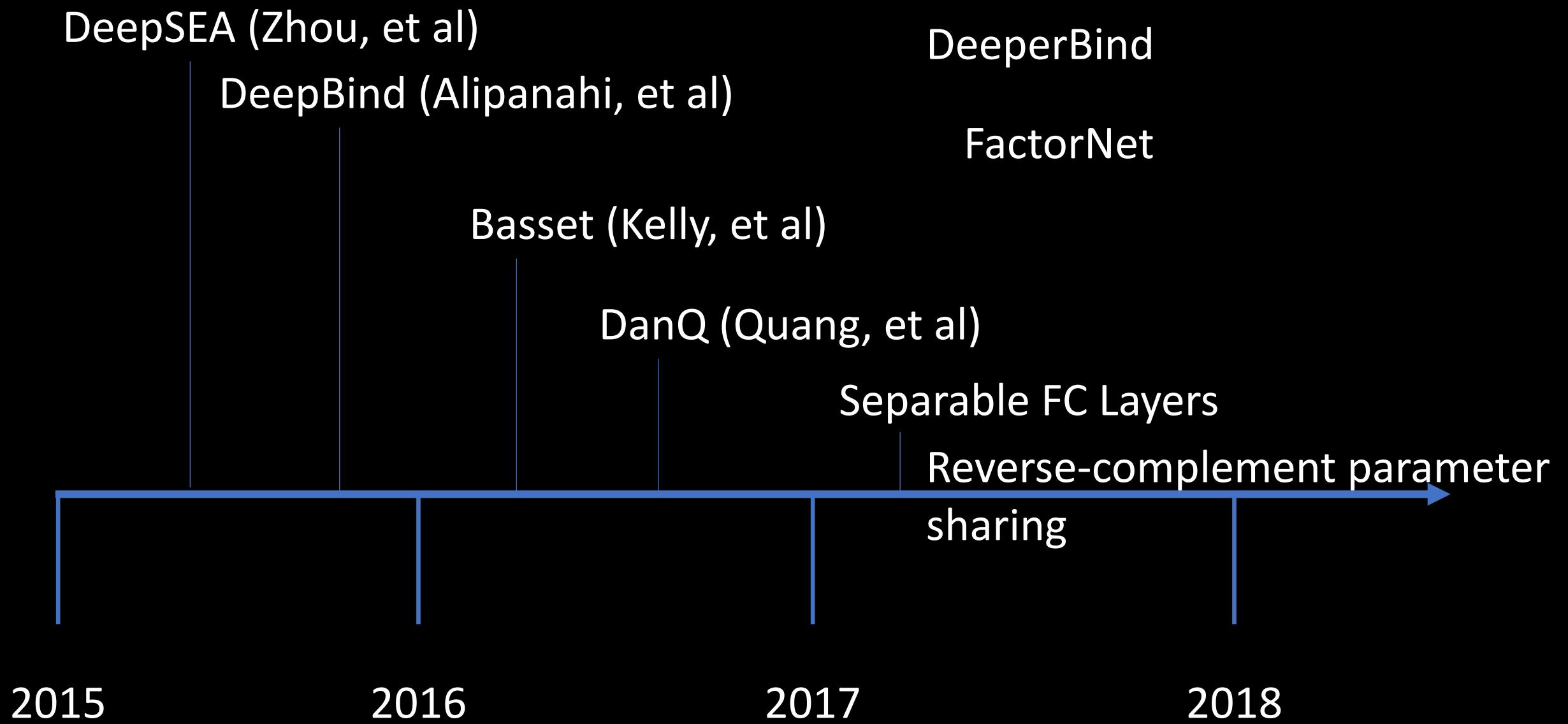


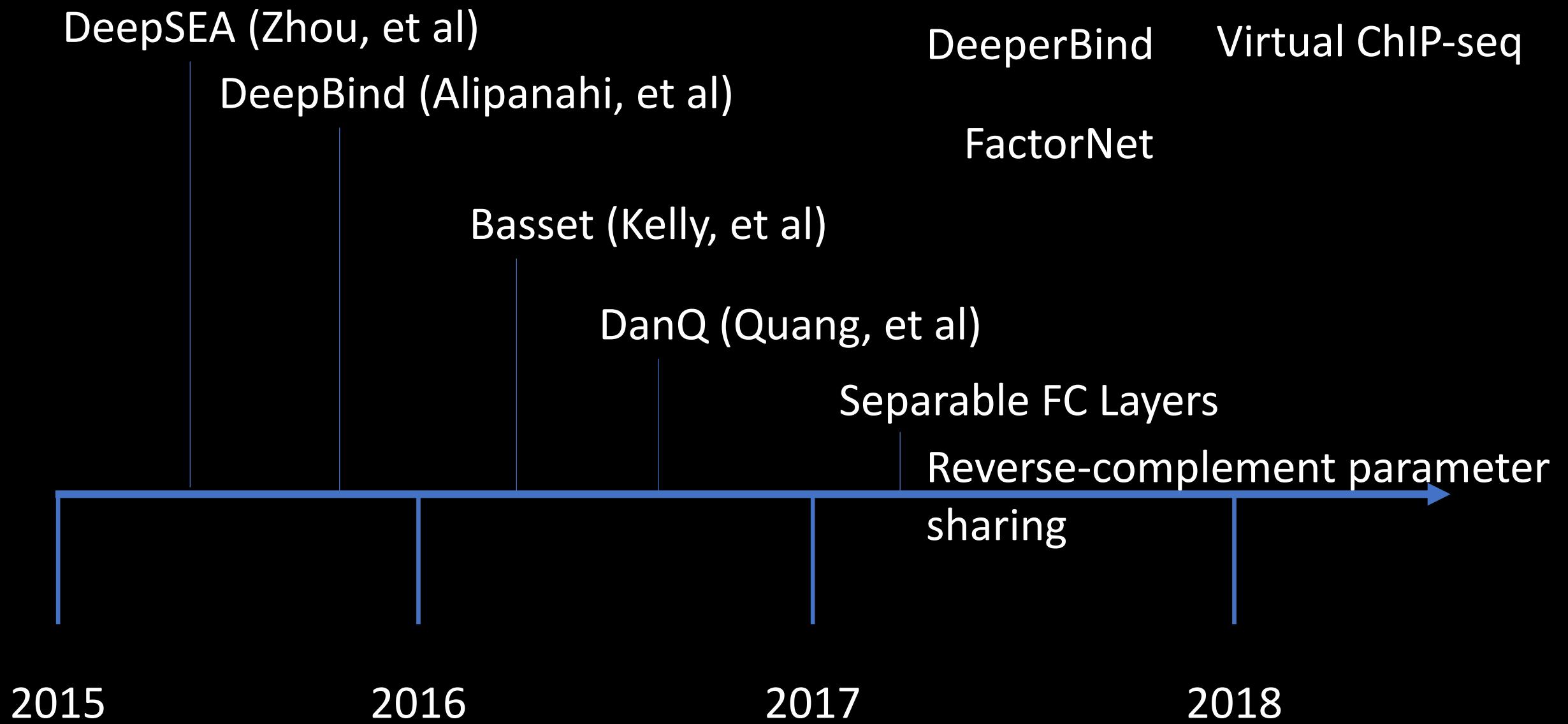
2015

2016

2017

2018





DeepSEA (Zhou, et al)

DeepBind (Alipanahi, et al)

Base

DeepBind

Virtual ChIP-seq

PERSPECTIVE

nature
biotechnology

Deep learning in biomedicine

Michael Wainberg^{1,2}, Daniele Merico¹, Andrew Delong¹ & Brendan J Frey¹

Deep learning is beginning to impact biological research and

(Fig. 1b,c) using high-throughput technologies has created an oppor-

Separable FC Layers

Reverse-complement parameter
sharing

2015

2016

2017

2018



DeepSEA (Zhou, et al)

DeepBind (Alipanahi, et al)

Base

DeepBind Virtual ChIP-seq

PERSPECTIVE

nature
biotechnology

Deep learning in biomedicine

Michael Wainberg^{1,2}, Daniele Merico¹, Andrew Delong¹ & Brendan J Frey¹

Deep learning is beginning to impact biological research and (Fig. 1b,c) using high-throughput technologies has created an opportunity for

Review Article | Published: 10 April 2019

Deep learning: new computational modelling techniques for genomics

Gökçen Eraslan, Žiga Avsec, Julien Gagneur & Fabian J. Theis

Nature Reviews Genetics (2019) | Download Citation

2015

2016

2017

2018

rs
plement parameter



DeepSEA (Zhou, et al)

DeepBind (Alipanahi)

Base

Review Article | Published: 10 April 2019

Deep learning: new computational modelling techniques for genomics

Gökçen Eraslan, Žiga Avsec, Julien Gagneur & Fabian J. Theis

Nature Reviews Genetics (2019) | Download Citation

2015

2016

2017

2018

nature
biotechnology

DeepBind | Virtual ChIP | co

PERSPECTIVE

Deep learning in biomedicine

Michael Wainberg^{1,2}, Daniele Merico¹, Andrew Delong¹ & Brendan J Frey¹

Deep learning is beginning to impact biological research and (Fig. 1b,c) using high-throughput technologies has created an opportunity for

DeepSEA (Zhou, et al)

DeepBind (Alipanahi)

Base

DeepBind Virtual CbID.org

PERSPECTIVE

nature
biotechnology

Deep learning in biomedicine

Michael Wainberg^{1,2}, Daniele Merico¹, Andrew Delong¹ & Brendan J Frey¹

Deep learning is beginning to impact biological research and (Fig. 1b) using high-throughput technologies has created an opportunity for

Review Article | Published: 10 April 2019

Deep learning: new computational modelling techniques for genomics

Gökçen Eraslan, Žiga Avsec, Julien Gagneur & Fabian J. Theis

Nature Reviews Genetics (2019) | Download Citation

2015

2016

2017

2018

<https://github.com/gokceneraslan/awesome-deepbio>

<https://github.com/hussius/deeplearning-biology>

DeepSEA (Zhou, et al)

DeepBind (Alipanahi, et al)

Base

DeepBind Virtual CbID.org

PERSPECTIVE

nature
biotechnology

Deep learning in biomedicine

Michael Wainberg^{1,2}, Daniele Merico¹, Andrew Delong¹ & Brendan J Frey¹

Review Article | Published: 10 April 2019

Deep learning: new computational modelling techniques for genomics

Gökçen Eraslan, Žiga Avsec, Julien Gagneur & Fabian J. Theis

Nature Reviews Genetics (2019) | Download Citation

2015

2016

2017

2018

rs
plement parameter

And many, many more!

Today's focus:

CNNs for genomics

Snapshot

- Problem formulation
 - *Where do we get the data?*
- Data setup
 - *What do the data look like?*
- Opening the black box
 - *What has the model learned?*

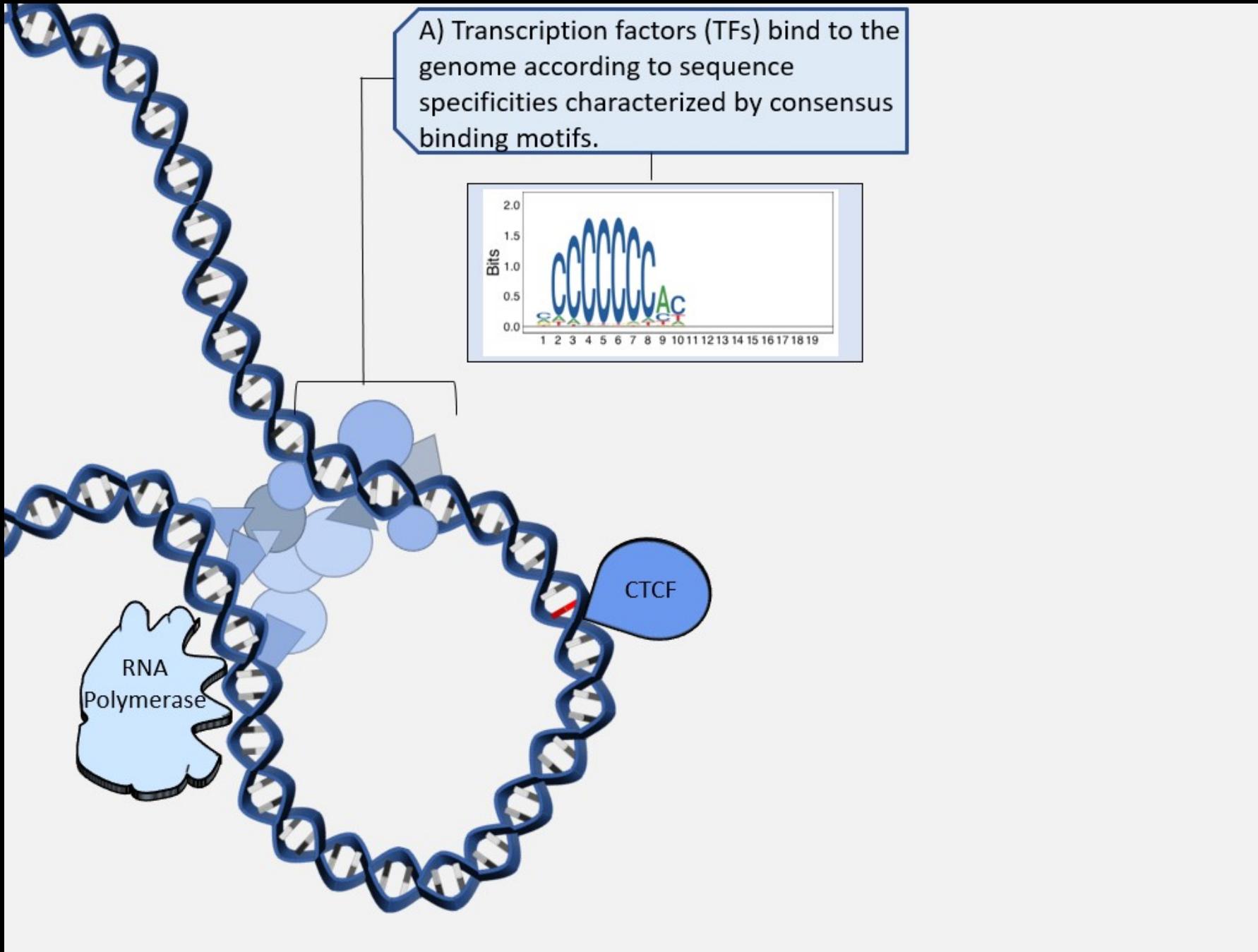
Next generation sequencing

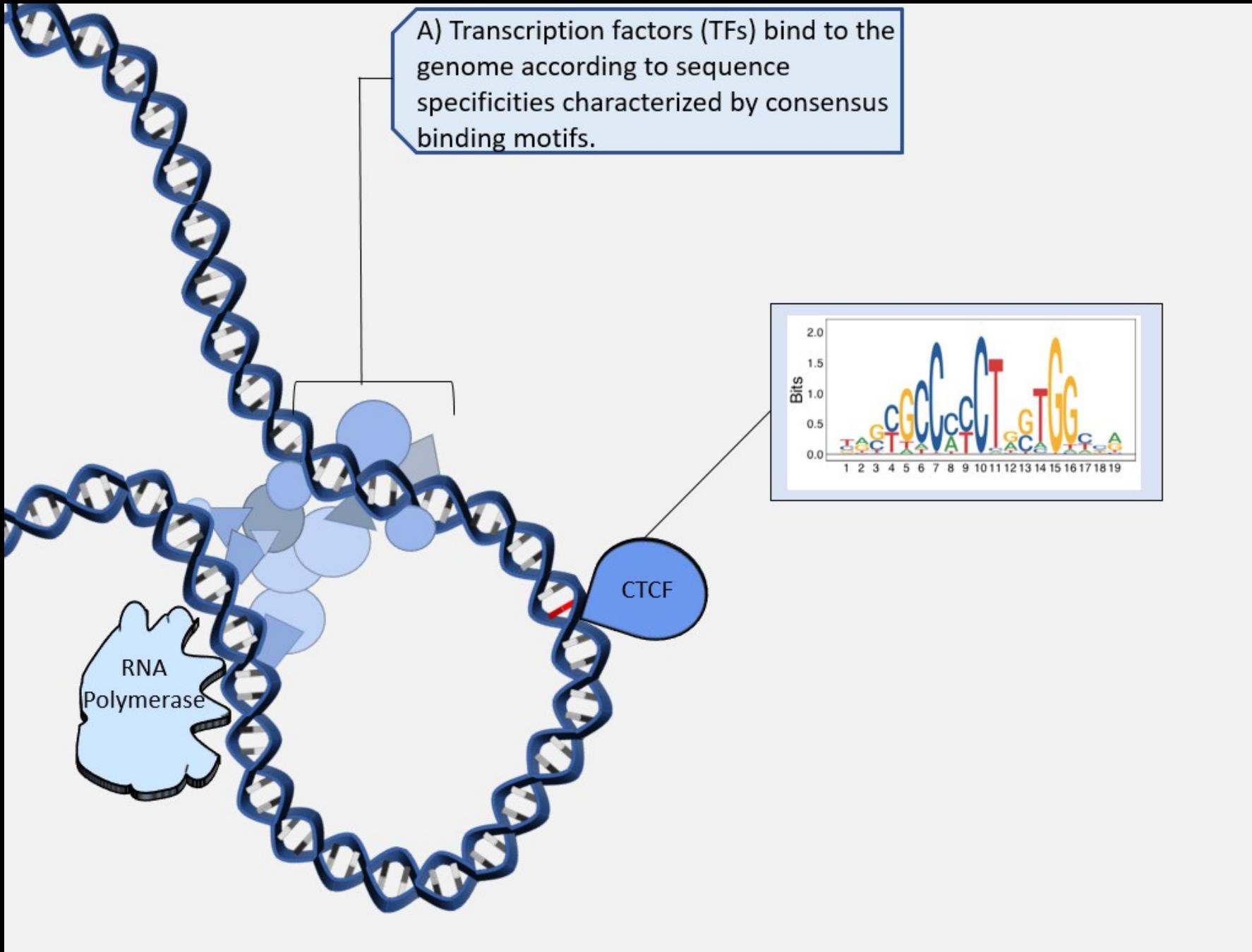
Snapshot

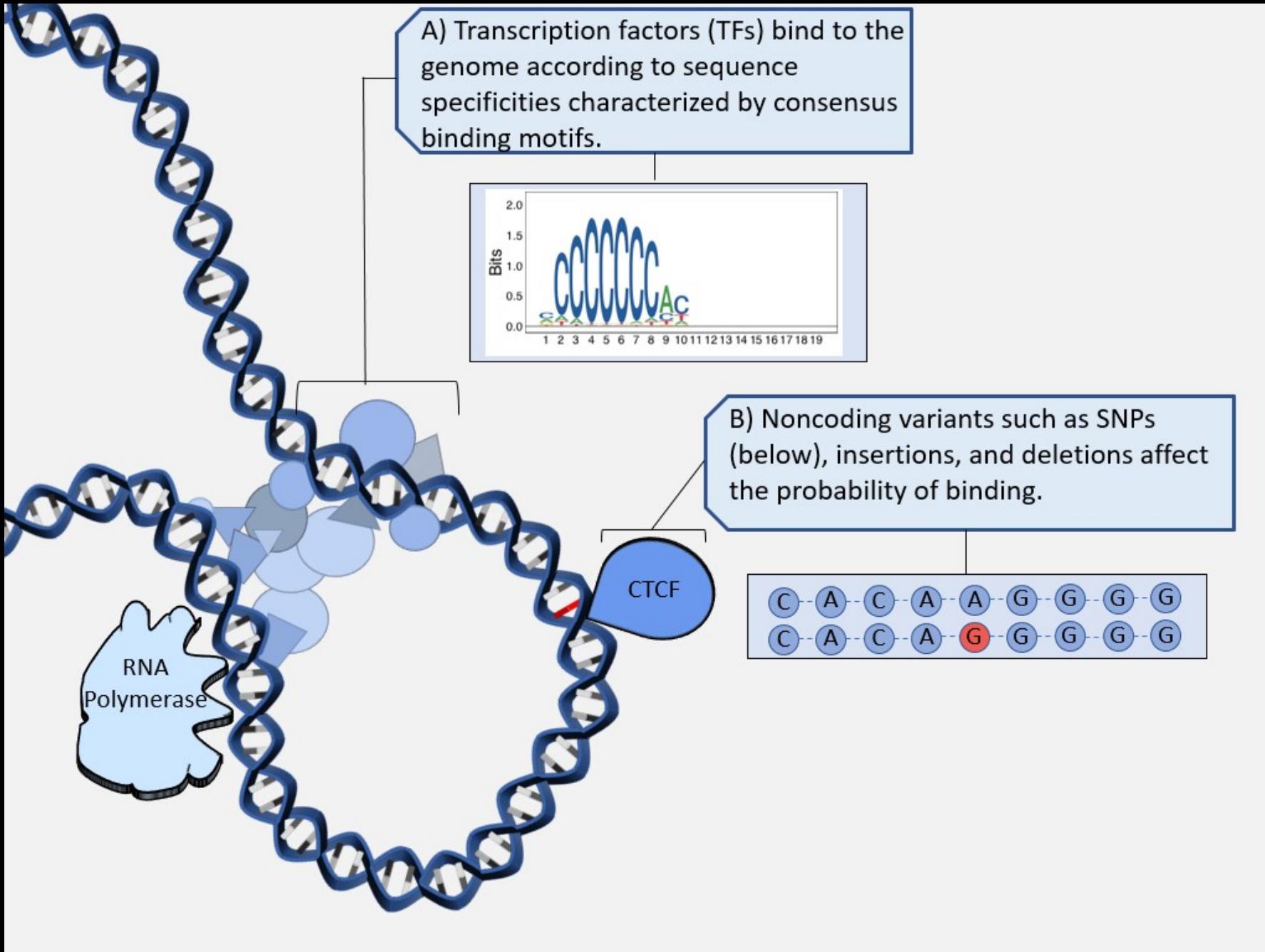
- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- Next generation sequencing
- ChIP-seq
 - Dnase-seq

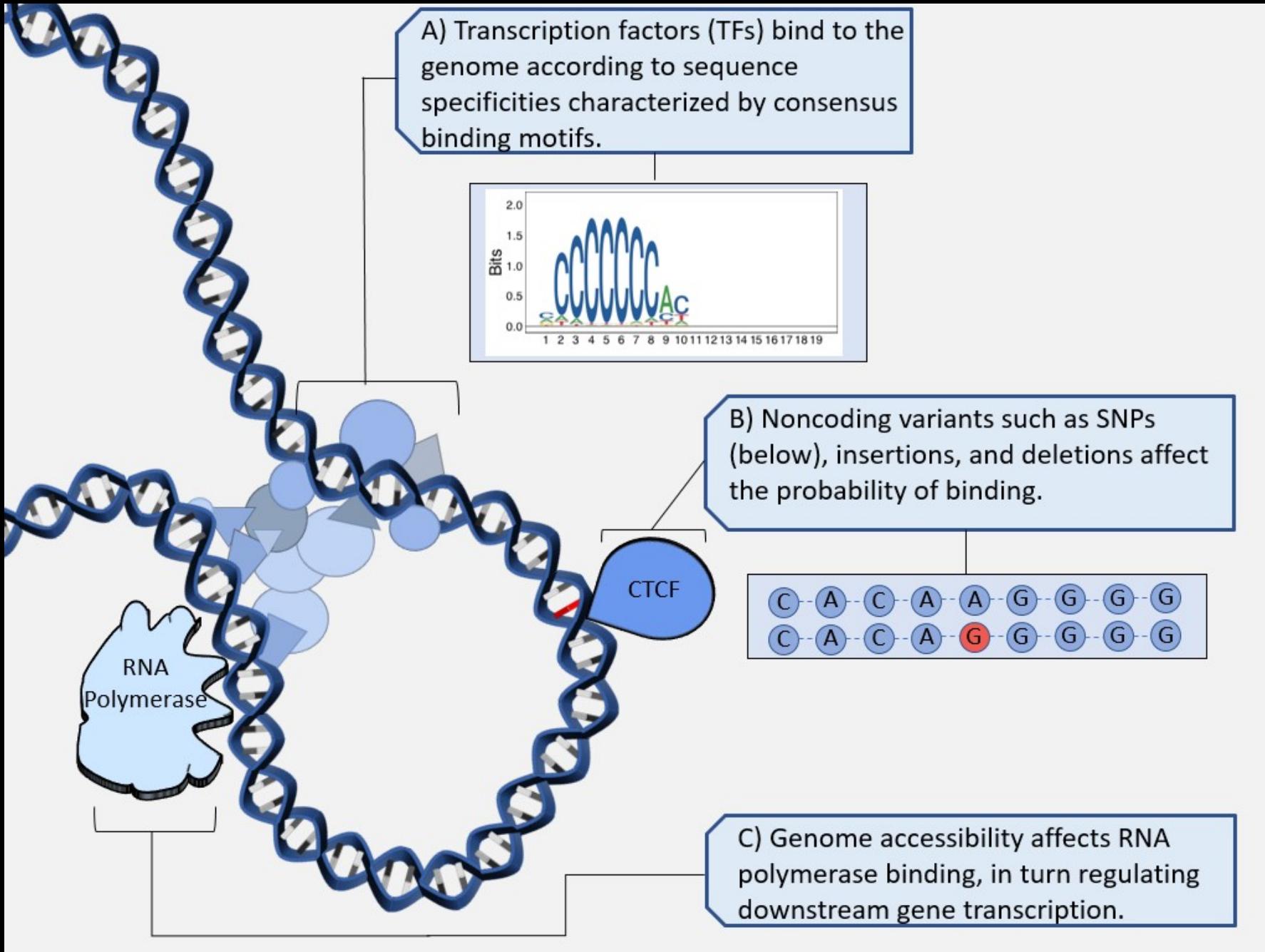
Snapshot

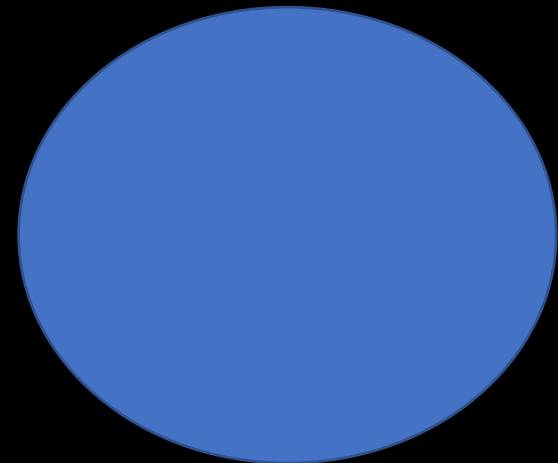
- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- Next generation sequencing
- ChIP-seq
 - Dnase-seq
 - RNA-seq
 - Single cell RNA-seq
 - Whole genome sequencing
 - ...



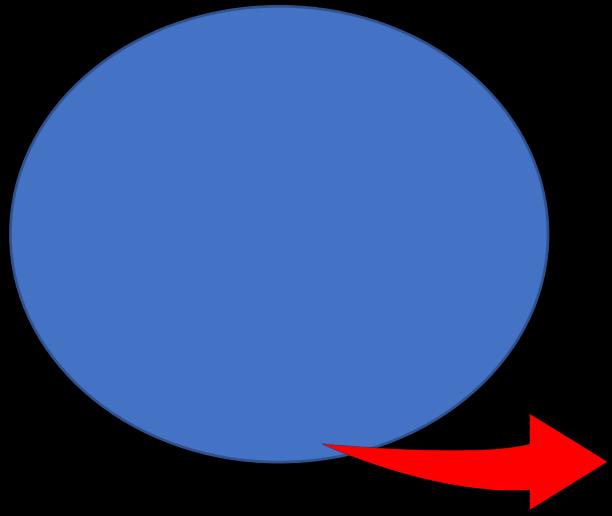








TATAGACGTATGCGGAATTCCCCCCCCCAAGCACCTCCGCACGGTATAACATTAACATTTTATCCATCTCTTGATACAGAGAACGTCTTAACTTA



TATAGACGTATGCGGAATT**CCCGCCCC**CAAGCACCTCCGCACGGTATAACATTAACATTAACTTTATCCCATCTCTTGATACAGAGAACGTCTTAACTTA

In ChIP-seq, only sequences bound to the shape
(DNA-binding protein/transcription factor) are
sequenced

In ChIP-seq, only sequences bound to the shape
(DNA-binding protein/transcription factor) are
sequenced

What sequences does the shape bind to?

In Dnase-seq, all sequences bound to any shape
are sequenced

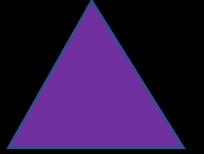
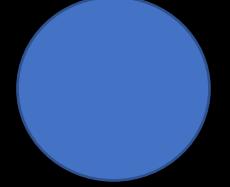
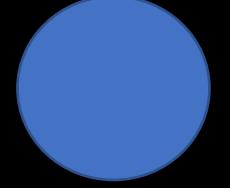
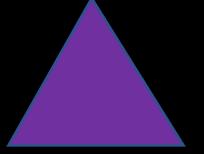
What shapes are bound in different cells/conditions?

In Dnase-seq, all sequences bound to any shape
are sequenced

In ChIP-seq, only sequences bound to the shape
(DNA-binding protein/transcription factor) are
sequenced

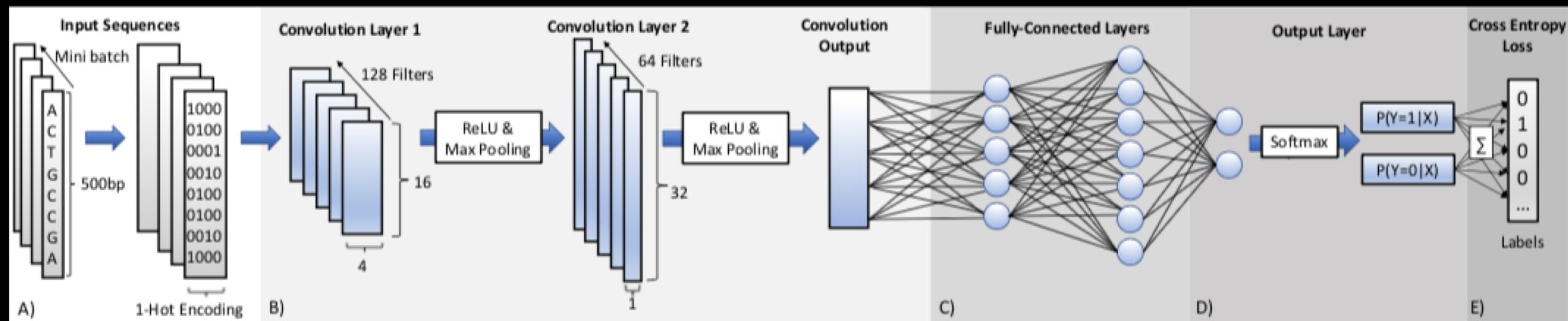
Want to learn how DNA sequences arising from
different samples are different

In Dnase-seq, all sequences bound to any shape
are sequenced

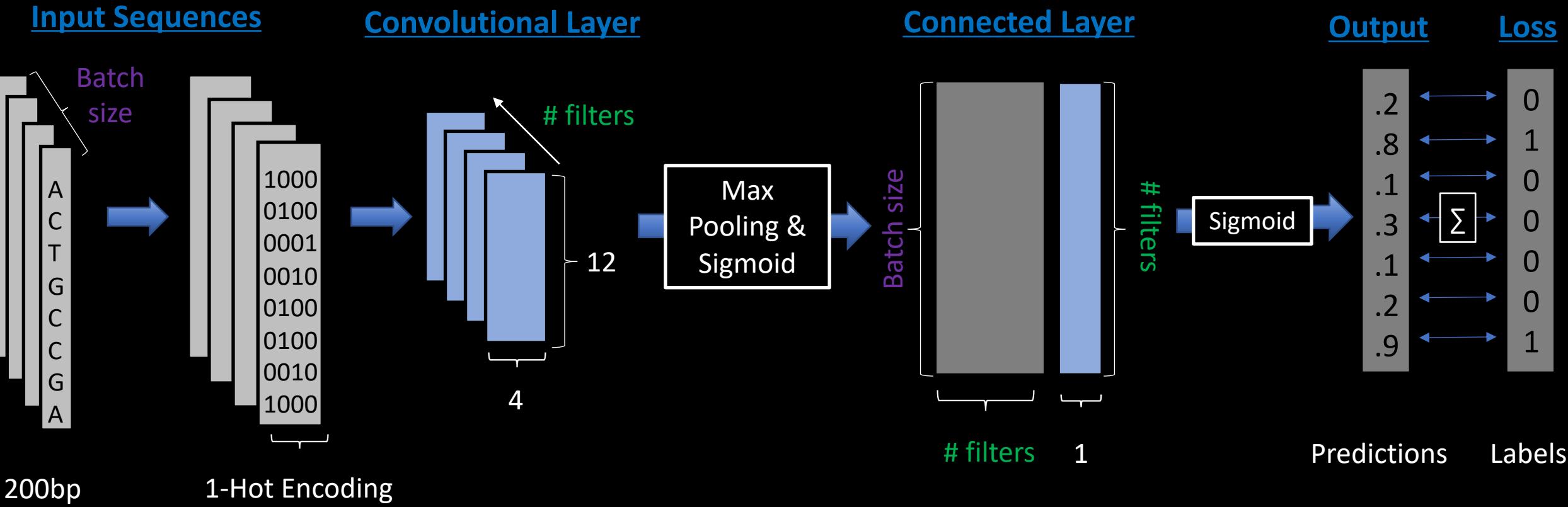
Sequence (X)	Label (Y)
ACTAATGAGAGTAGTTTATACTGGTATCGCTGCTAACAGAGCTTGATATAAACATCTAACACAGCCTGATGGAATAAAGGGGGGGGCAAATACTCCCGCGTTAGGAGTATGTTGCCTCATAGCTTGTAGGATAGAGCAGGGGCCGATAAAATTAACCATTCAAGGTTTGCCACTAACATAGAACACCGTTAAGA	
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATTCCCCCCCCAACGCACCTCCGCACGGTATAACATTAAACATTTTATCCCATCTCTGTATAACAGAGAACGTCTTAACTTATTCGTATTACTGTGTCTATGTGACTCTCACCTTACAGAACCGAAAGTGCAATTGGATATCTCGACGAGTTAAA	
GCGAAAGACCCCCCTAATAGCCTCAGACTTCATTACGCTTGTGGGTGGATTCCACCTATAGGACGTACTTACGGCCTATTAAAGCAAAGCCAGAGAGAGGGTTTTAATCGATGTTGGTCACACATTCAAGCTATGTCGGTTAACGTTAATGAGTTATTCTTGGAGACATCAGTCTATGTGGCTTACGAAGGAAA	
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATAGGAAAATAAAAGCAGATTACTTCAGATAGCAACAAGAACAGTGACTCCAGCATTCAAGTCAAACAAATTACGCAGTATGGGGGGGGTATTAAGCGTTATGTGGGAAC TGCGAGATCATCTCATTGCCAGCAA	
• • •	• • •

Sequence (X)	Label (Y)
ACTAATGAGAGTAGTTTATACTGGTATCGCTGCTAACAGAGCTTGATATAAACATCTAACACAGCCTGATGGAATAAAGGGGGGGGCAAATACTCCCGCGTTAGGAGTATGTTGCCTCATAGCTTGTAGGATAGAGCAGGGGCCGATAAAATTAACCATTCAAGGTTTGCCACTAACATAGAACACCGTTAAGA	1
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATTCCCCCCCCCAAGCACCTCCGCACGGTATAACATTAAACATTTTATCCCATCTCTTGTATACAGAGAACGTCTTAACTTATTCGTATTACTGTGTCTATGTGACTCTCACCTTACAGAACCGAAAGTGCAATTGGATATCTCGACGAGTTAAA	0
GCGAAAGACCCCCCTAATAGCCTCAGACTTCATTACGCTTGTGGGTGGATTCCACCTATAGGACGTACTTACGGCCTATTAAAGCAAAGCCAGAGAGAGGGTTTTAATCGATGTTTGGGTACACATTCAAGCTATGTCGGTTAATGAGTTATTCTTGGAGACATCAGTCTATGTGGCTTACGAAGGAAA	0
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATAGGAAAATAAAAGCAGATTACTTCAGATAGCAACAAGAACAGTGACTCCAGCATTCAAGTCAAACAAATTACGCAGTATGGGGGGGGTATTAAGCGTTATGTGGGAAC TGCGAGATCATCTCATTGCCAGCAA	1
• • •	• • •

Sequence (X)	Label (Y)
ACTAATGAGAGTAGTTTATACTGGTATCGCTGCTAACAGAGCTTGATATAAACATCTAACAGCCTGATGGAATAAAGGGGGGGGCAAATACTCCCGCGTTAGGAGTATGTTGCCCTAGCTTGTAGGATAGAGCAGGGGCCGATAAAATTAACCATTCAAGGTTTGCCACTAACATAGAACACCGTTAAGA	1
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATTCCCCCCCCCAAGCACCTCCGCACGGTATAACATTAAACATTTTATCCCATCTCTGTATAACAGAGAACGTCTTAACTTATTCGTATTACTGTGTCTATGTGACTCTCACCTTACAGAACCGAAAGTGCAATTGGATATCTCGACGAGTTAAA	0
GCGAAAGACCCCCCCCCTAATAGCCTCAGACTTCATTACGCTTGTGGGTGGATTCCACCTATAGGACGTACTTACGGCCTATTAAAGCAAAGCCAGAGAGAGGGTTTTAATCGATGTTTGGGTACACATTCAAGCTATGTCGGTTAATGAGTTATTCTTGGAGACATCAGTCTATGTGGCTTACGAAGGAAA	0
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATAGGAAAATAAAAGCAGATTACTTCAGATAGCAACAAGAACAGTGACTCCAGCATTCAAGTCAAACAAATTACGCAGTATGGGGGGGGTATTAAGCGTTATGTGGGAAC TGCGAGATCATCTCATTGCCAGCAA	1
• • •	• • •



*Perhaps the most common architecture used in genomics (2 conv layers, 2 FC layers).



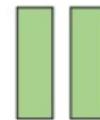
Sequence length = 200 nucleotides

A w11 w12 w13 w14 w15 w16 ...
C w21 w22 w23 w24 w25 w26 ...
G w31 w32 w33 w34 w35 w36 ...
T w41 w42 w43 w44 w45 w46 ...

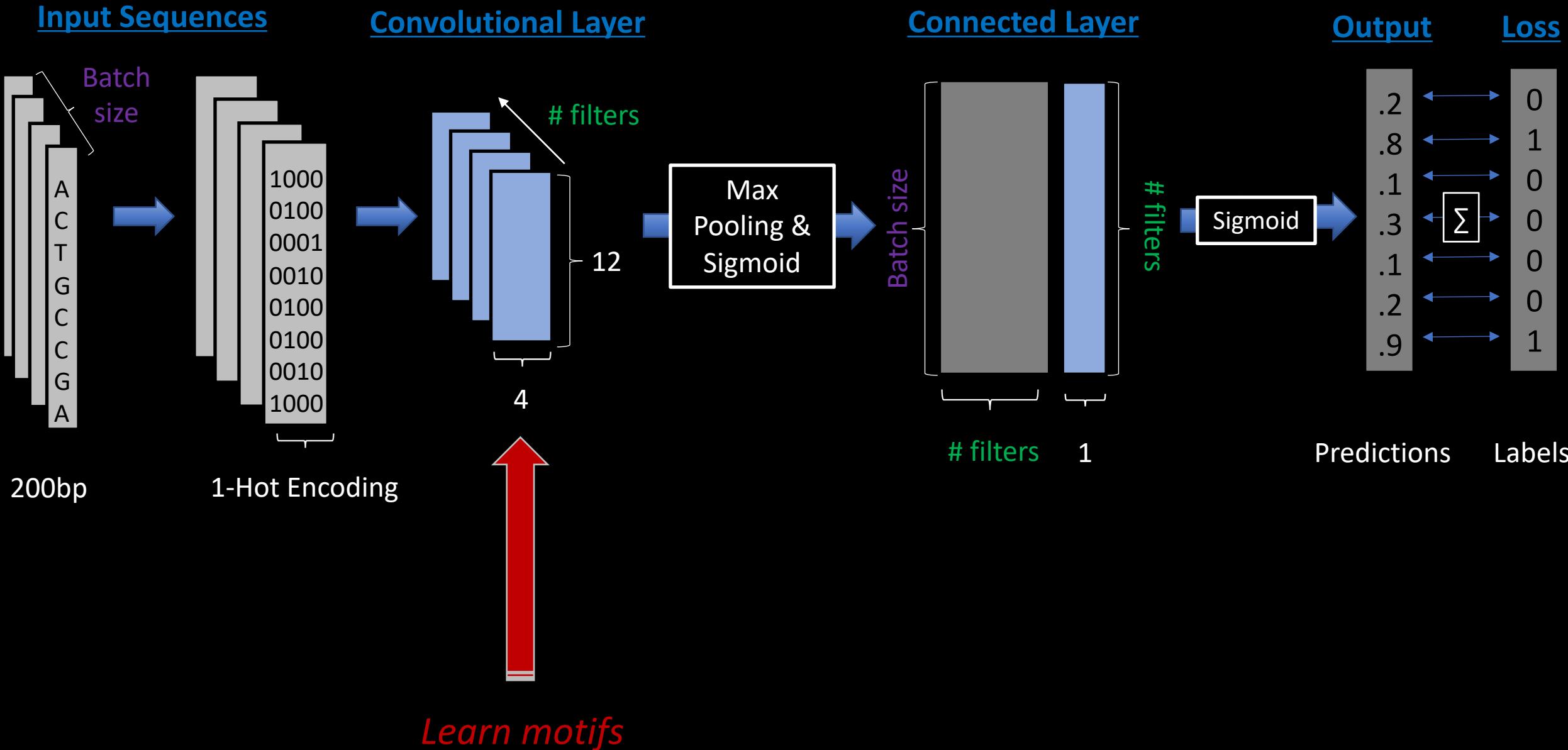
Slide to every position
Outputting a value at each

w11 w12 w13 w14 w15 w16 ...
w21 w22 w23 w24 w25 w26 ...
w31 w32 w33 w34 w35 w36 ...
w41 w42 w43 w44 w45 w46 ...

Filter length = 12 nucleotides



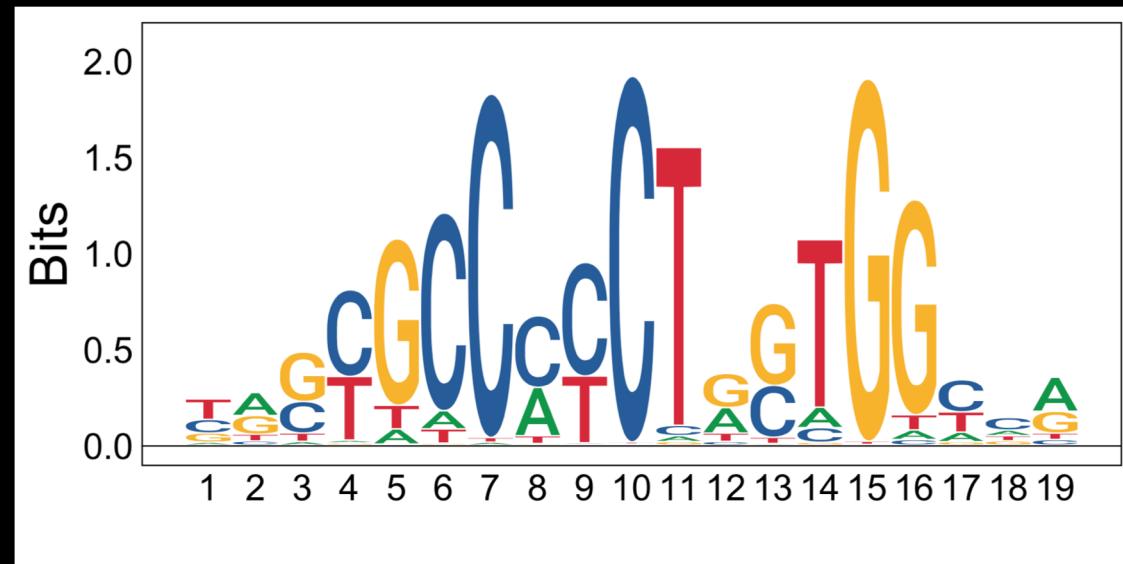
Vector of activation values



Learn motifs

But first! What is a motif?

But first! What is a motif?



Position weight matrix (PWM)

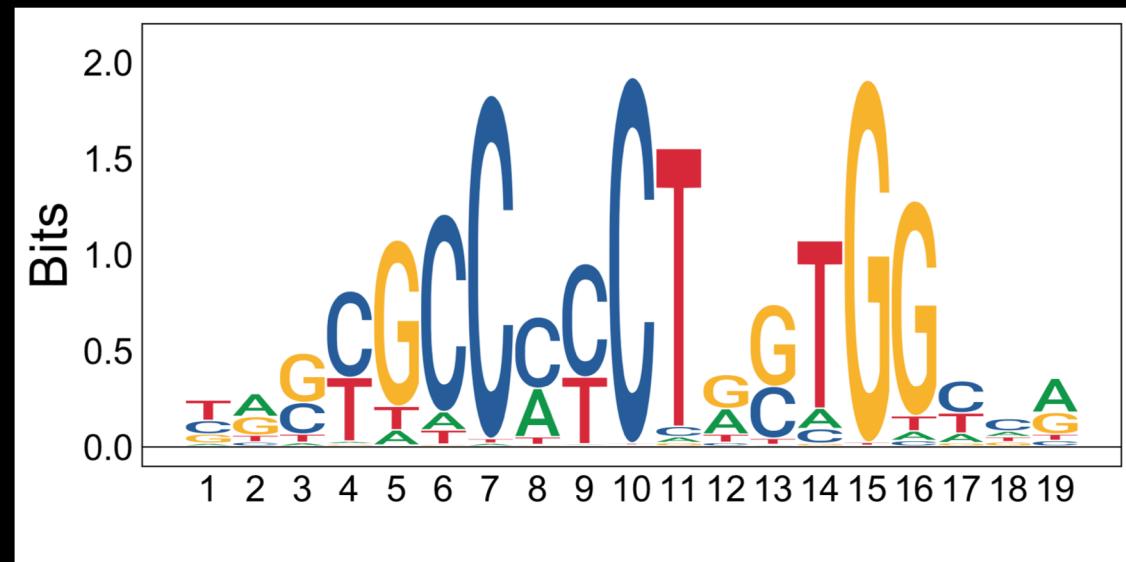
Sequence logo

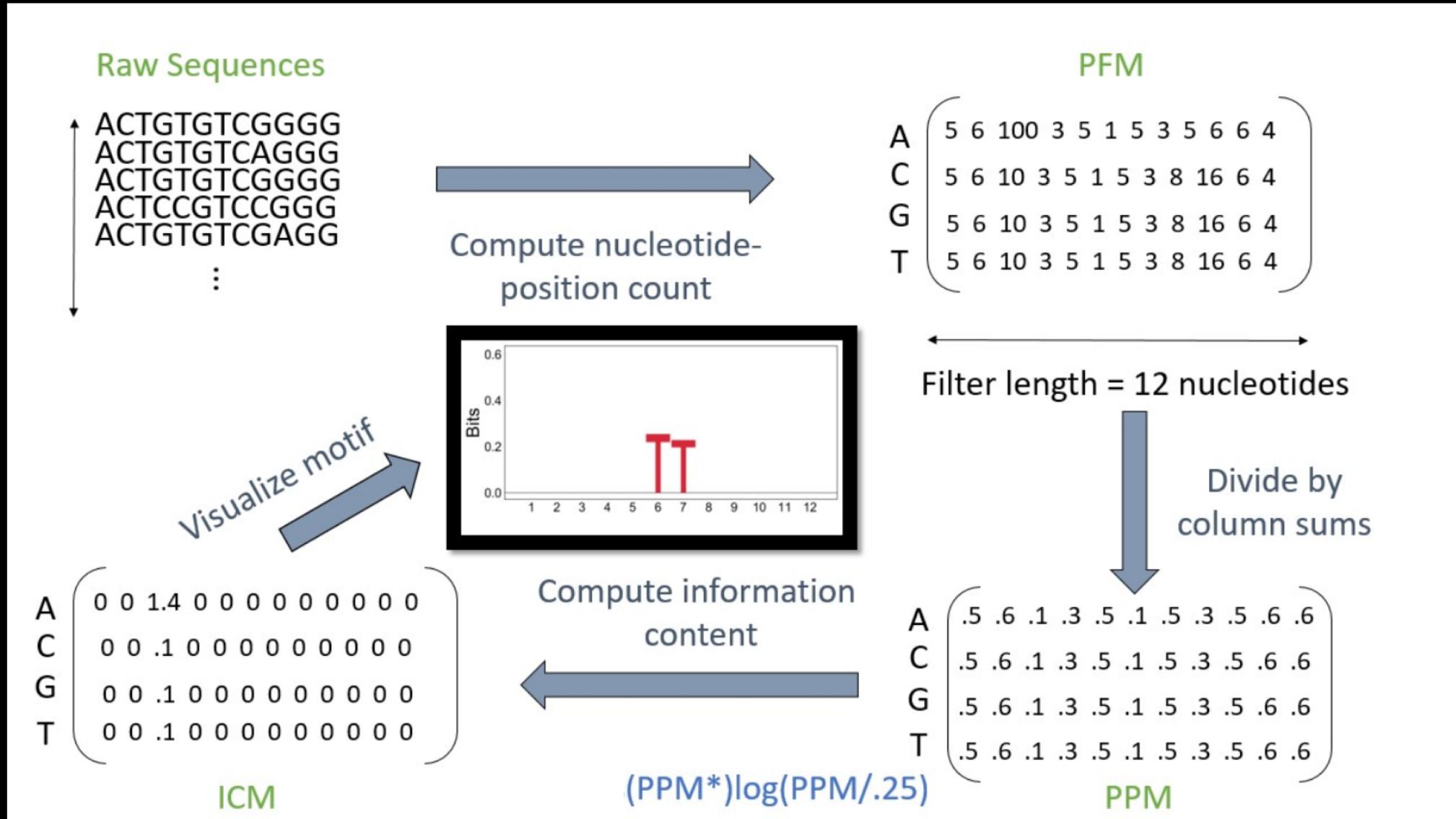
Position probability matrix (PPM)

Position sequence specificity matrix (PSSM)

Information content matrix

But first! What is a motif?

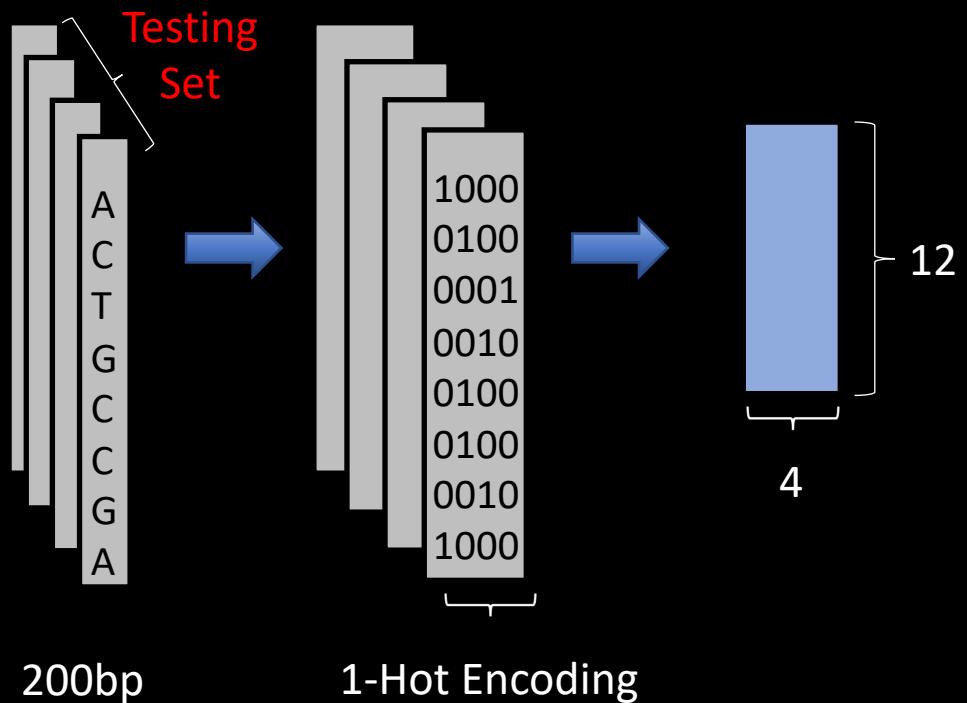




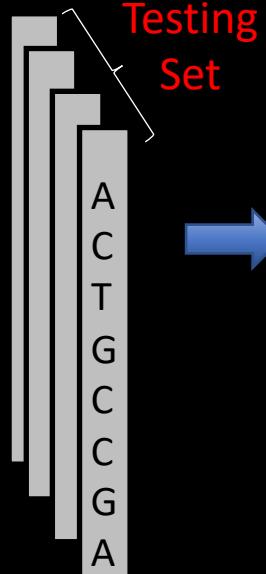
So how can we compute these for our CNN?

Input Sequences

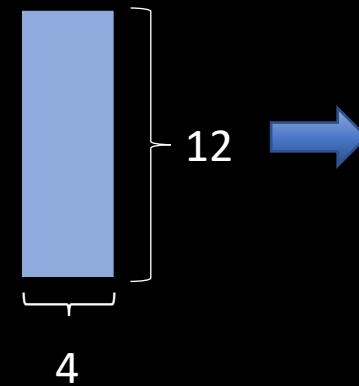
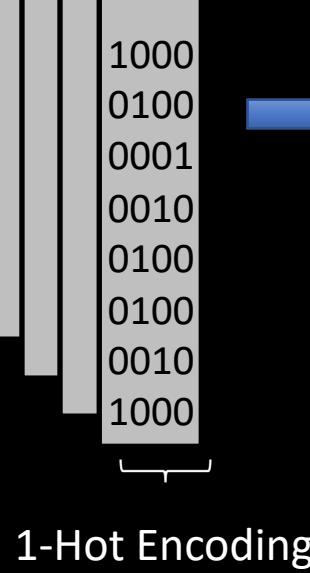
Convolutional Layer



Input Sequences



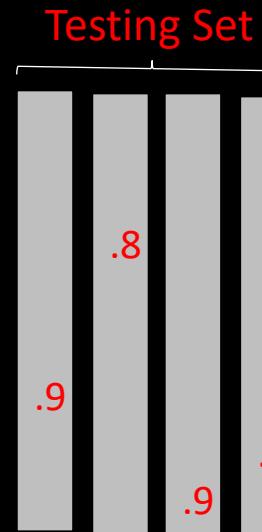
Convolutional Layer



Vector of activations

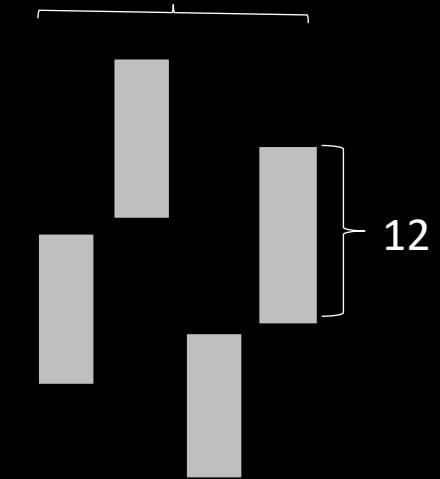


Rank Activations

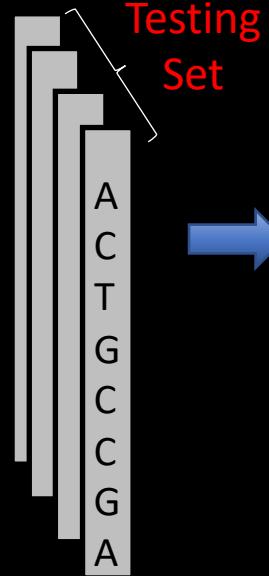


Select top 10%

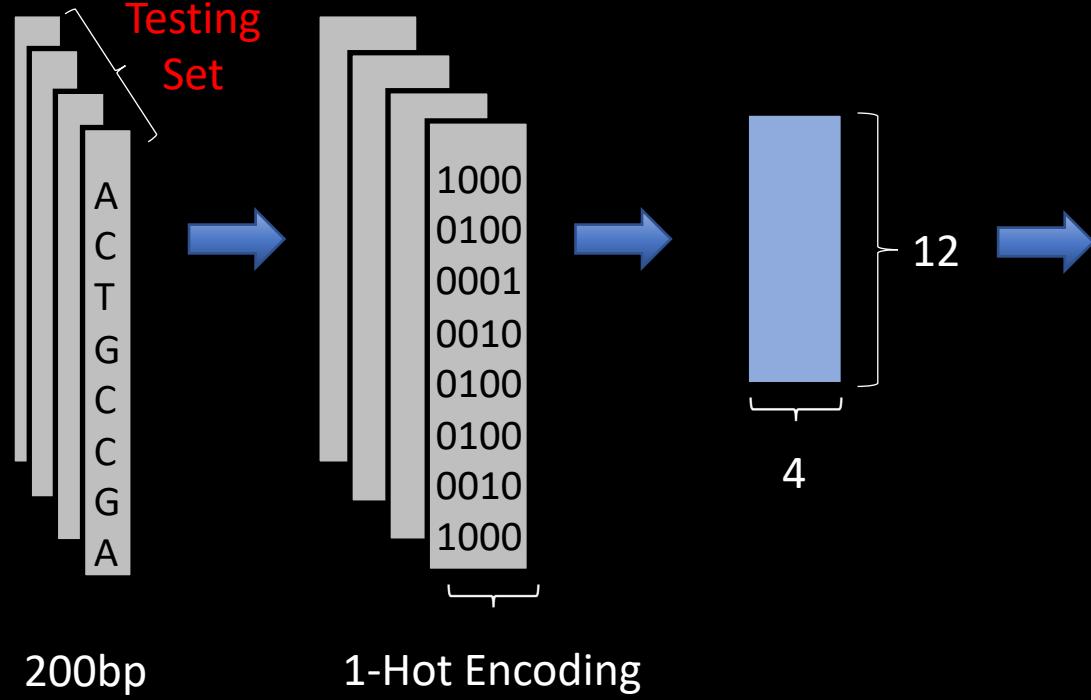
Extract Sequences



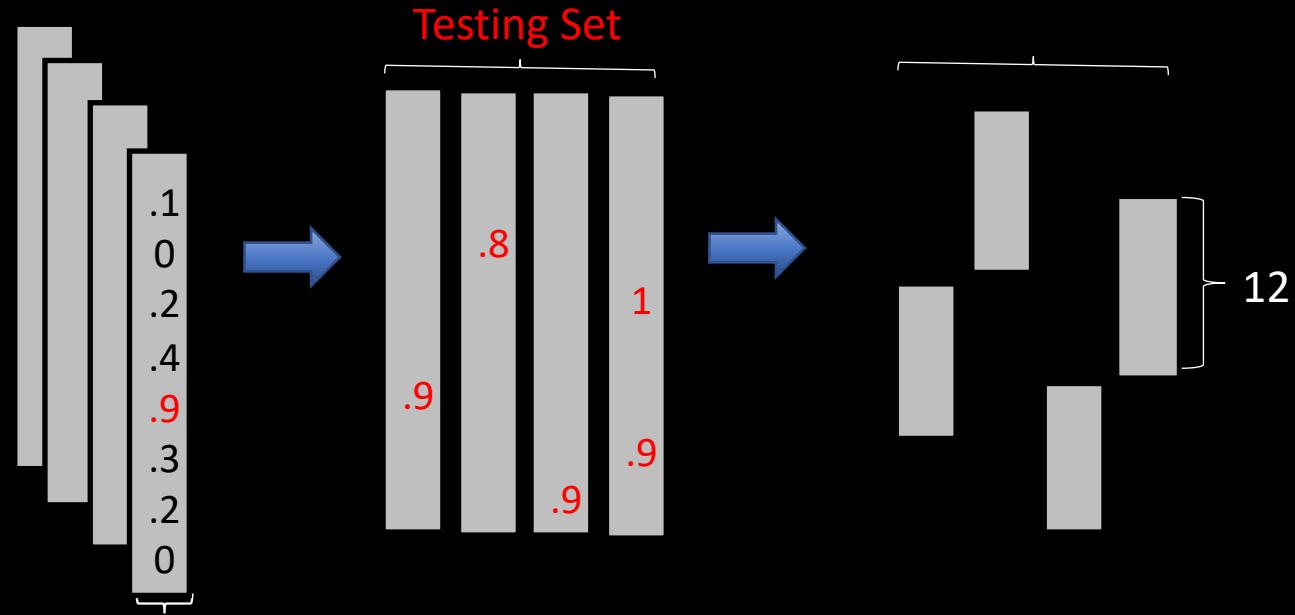
Input Sequences



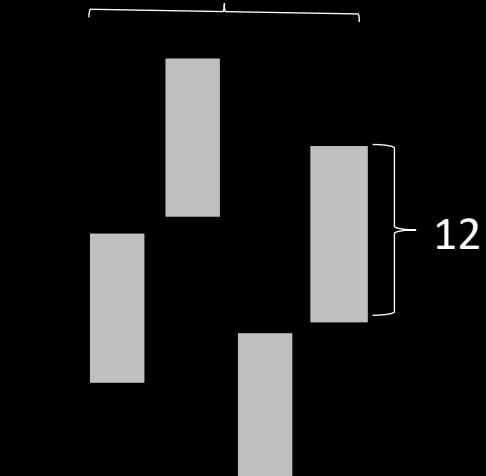
Convolutional Layer



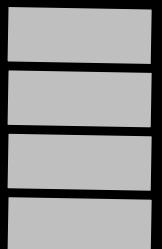
Rank Activations



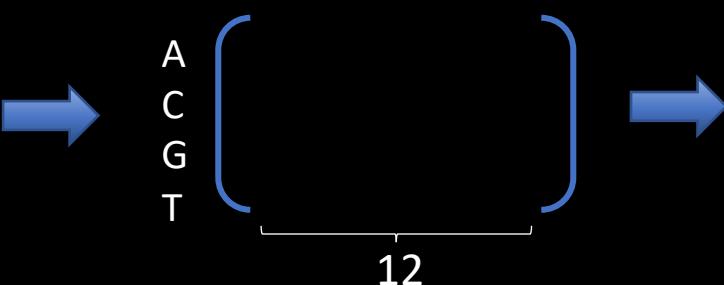
Extract Sequences



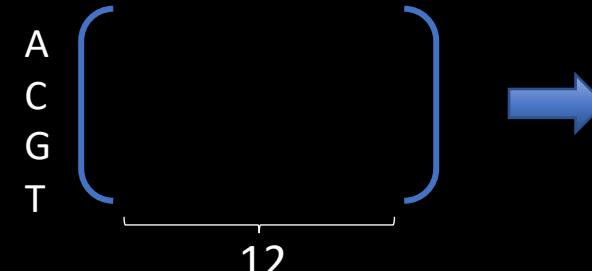
Stack Sequences



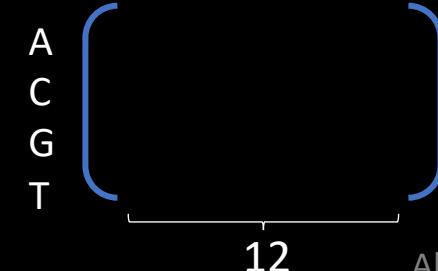
Compute Position Frequency Matrix



Compute Position Probability Matrix

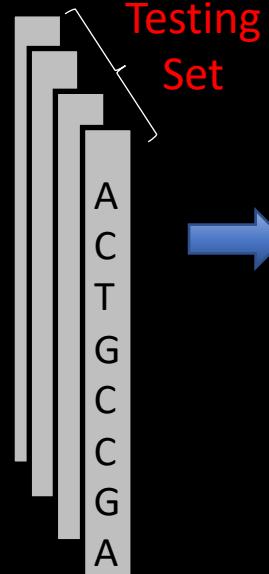


Compute Position Weight or Information Content

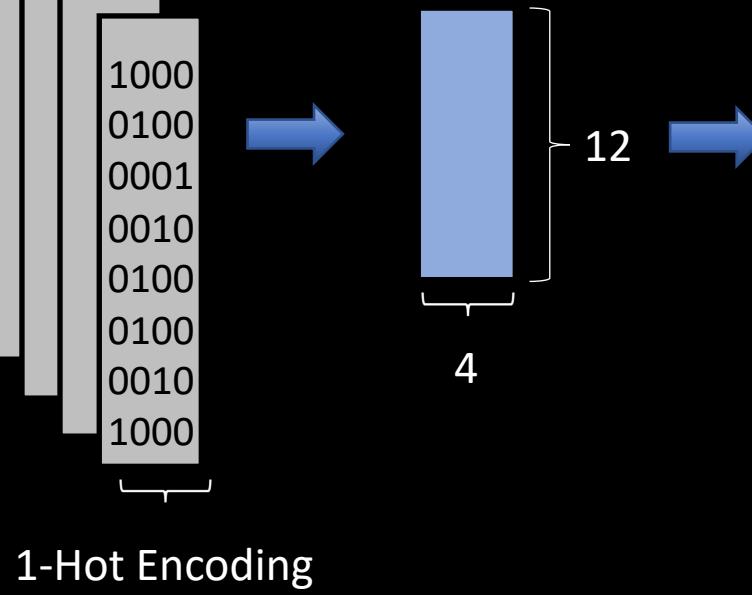


Alipinahi, et al., 2015

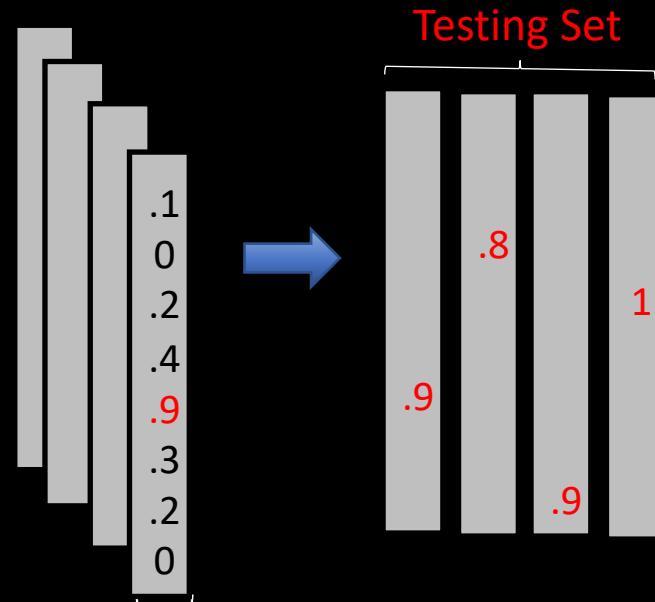
Input Sequences



Convolutional Layer



Rank Activations



Testing Set

.1

0

.2

.4

.9

.9

.3

.2

0

.1

0

.2

.4

.9

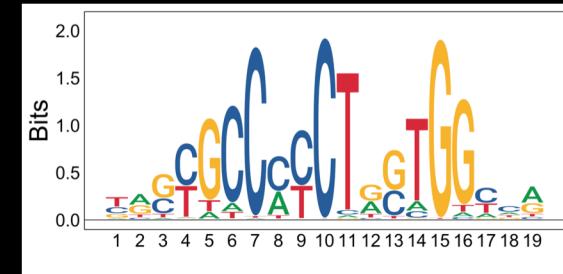
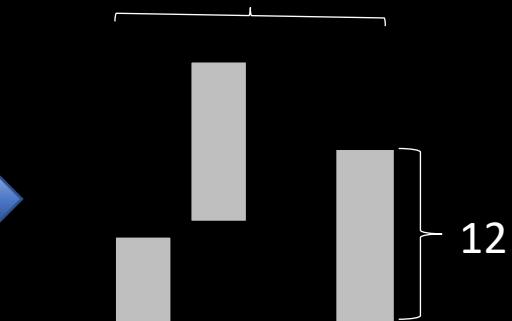
.9

.3

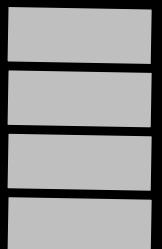
.2

0

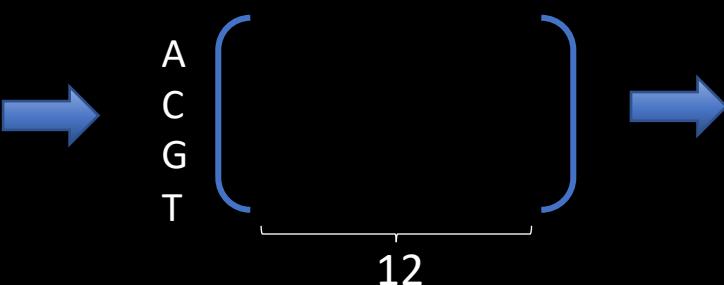
Extract Sequences



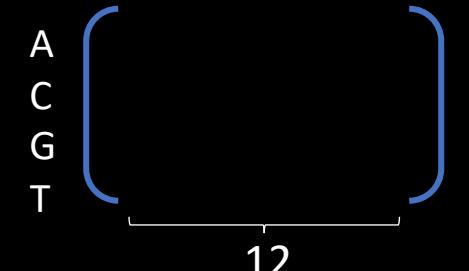
Stack Sequences



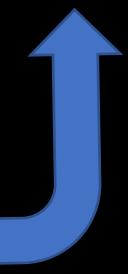
Compute Position Frequency Matrix



Compute Position Probability Matrix

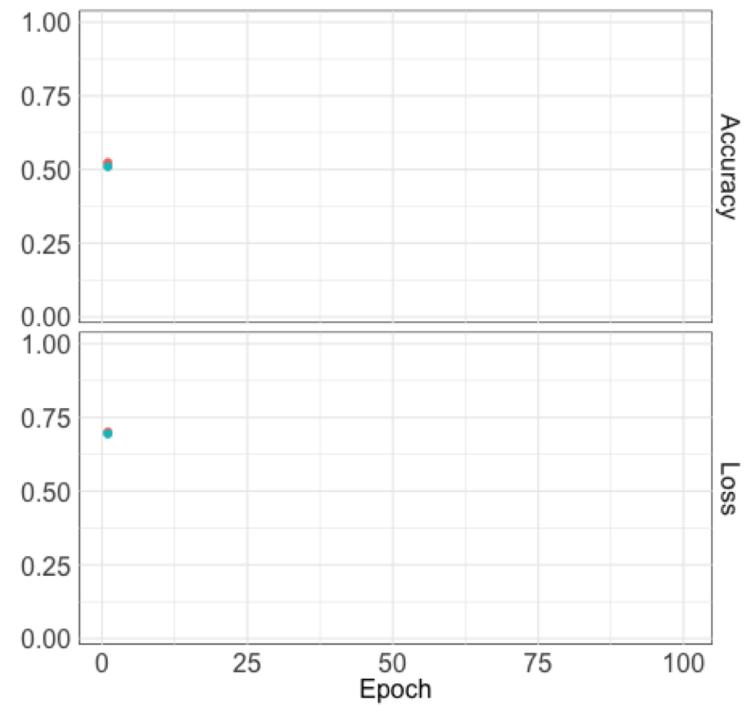


Compute Position Weight or Information Content

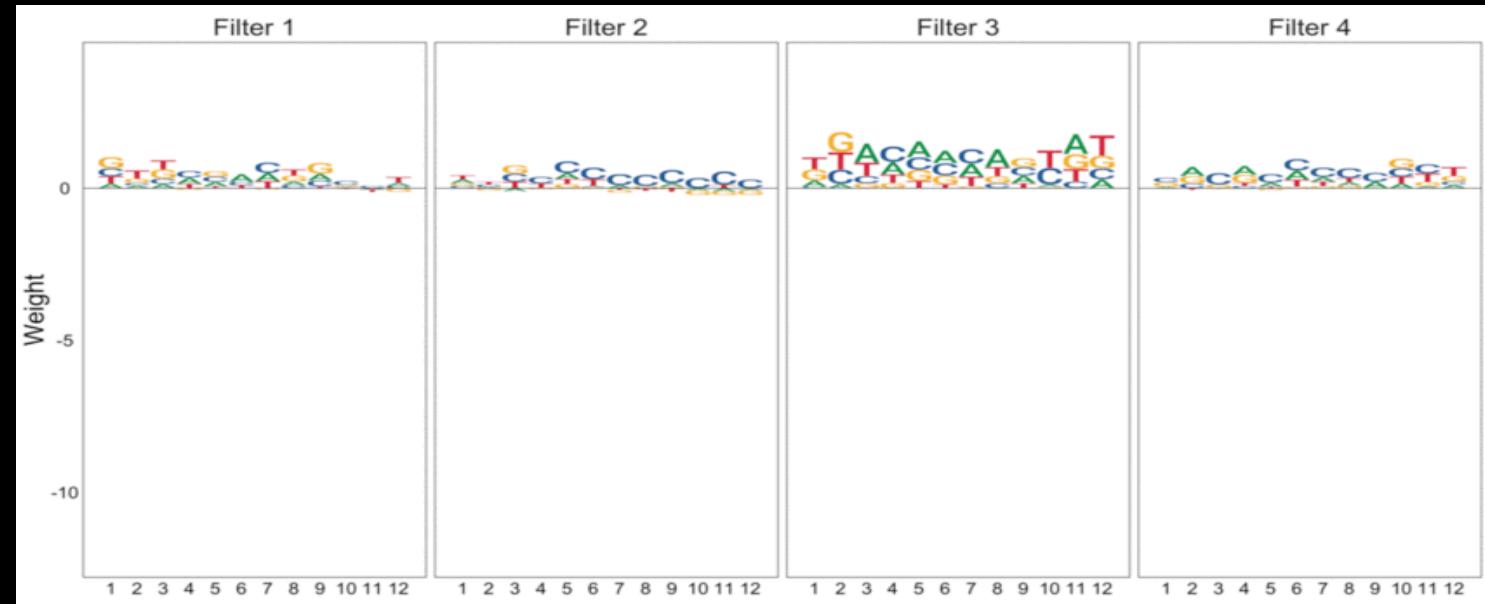
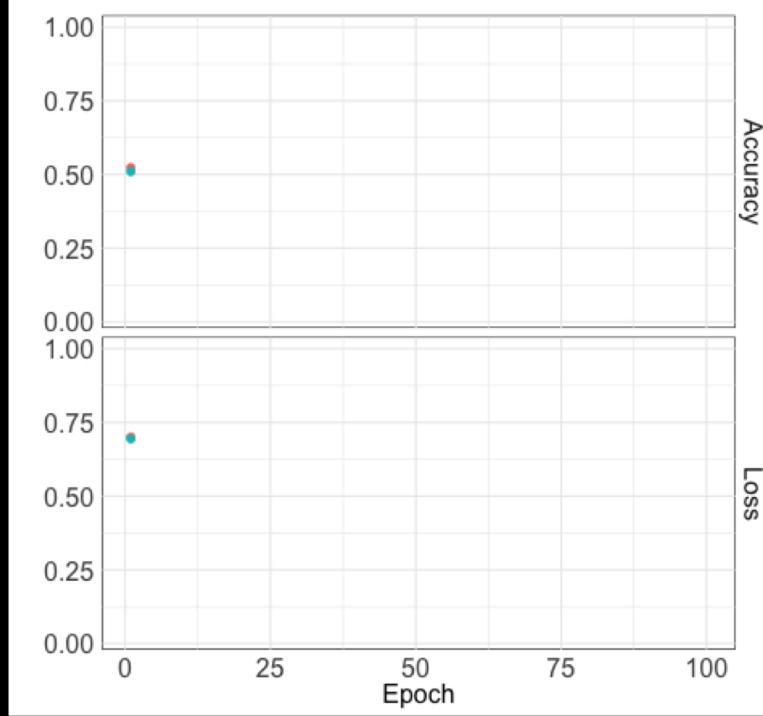


1. Train model
2. Pass through test-set observations
3. Retain high-scoring feature maps for a given filter
4. Compute PWM for motif
5. Repeat for every filter
6. Compare to some database of known motifs (e.g. jaspar)

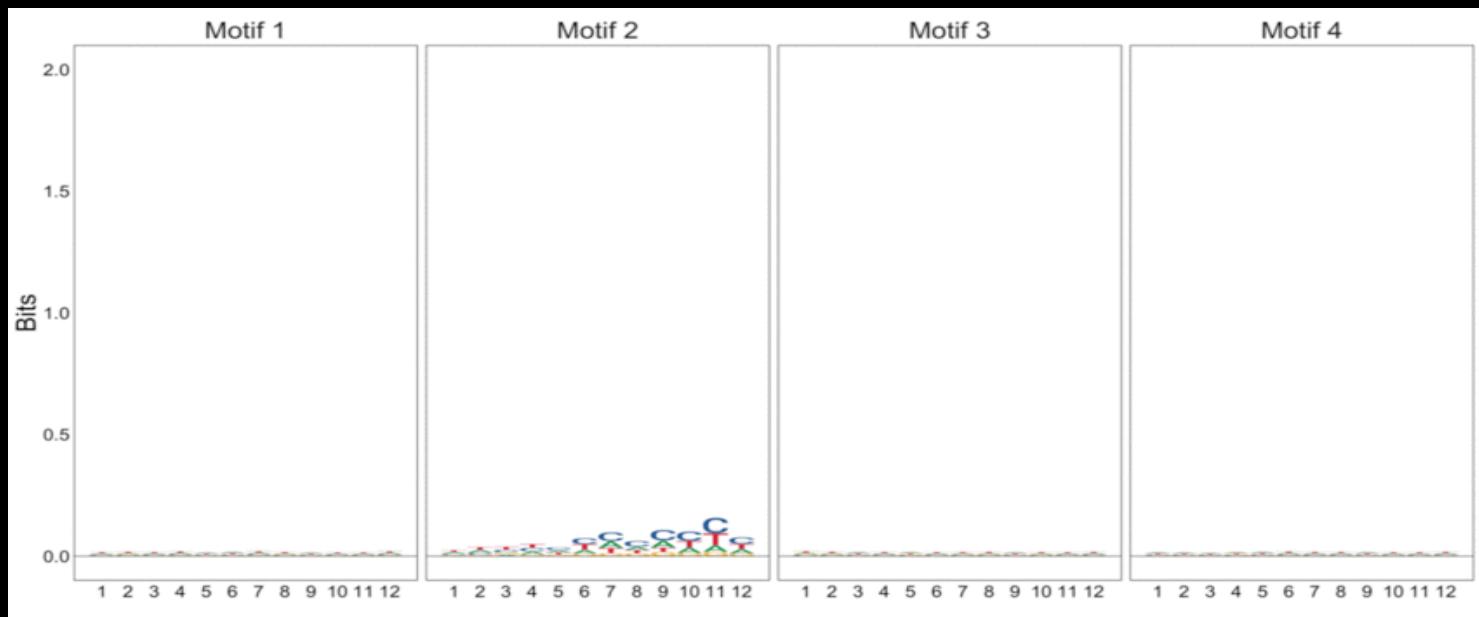
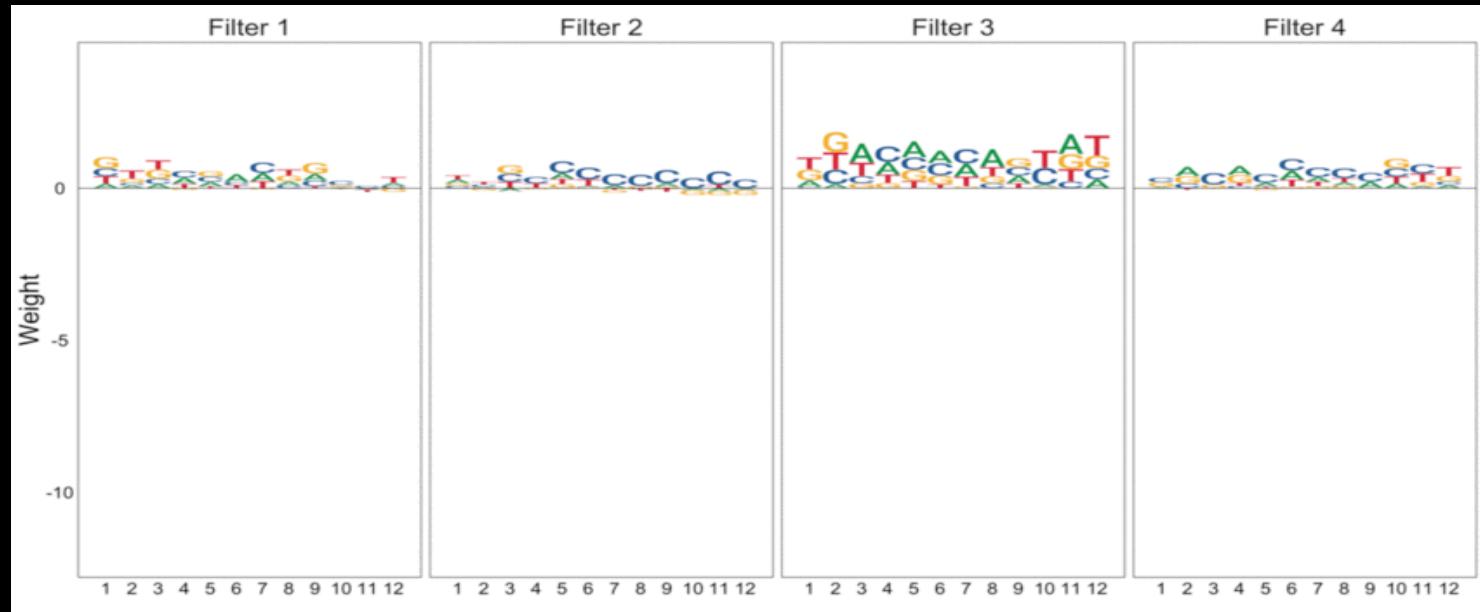
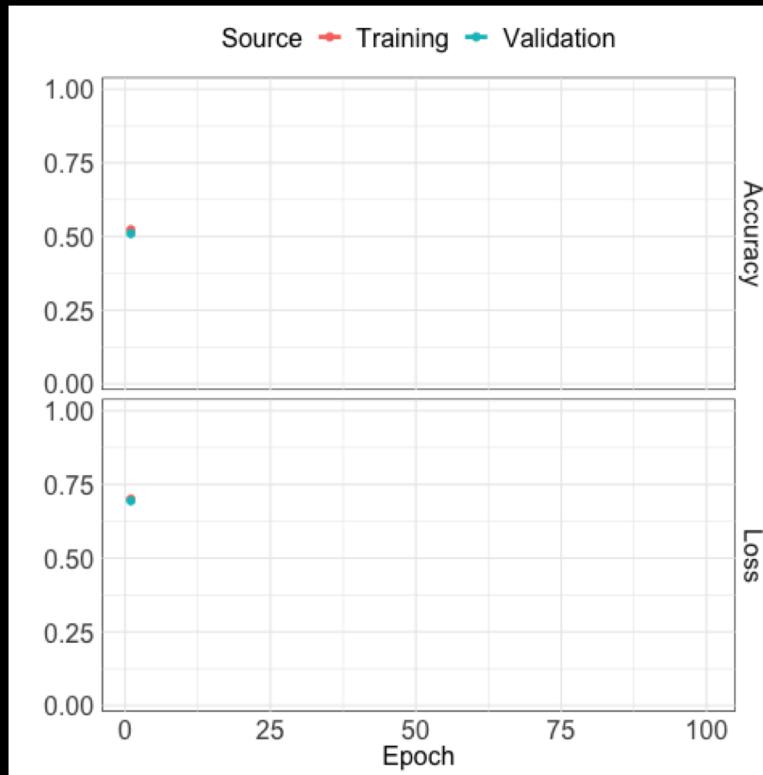
Source Training Validation



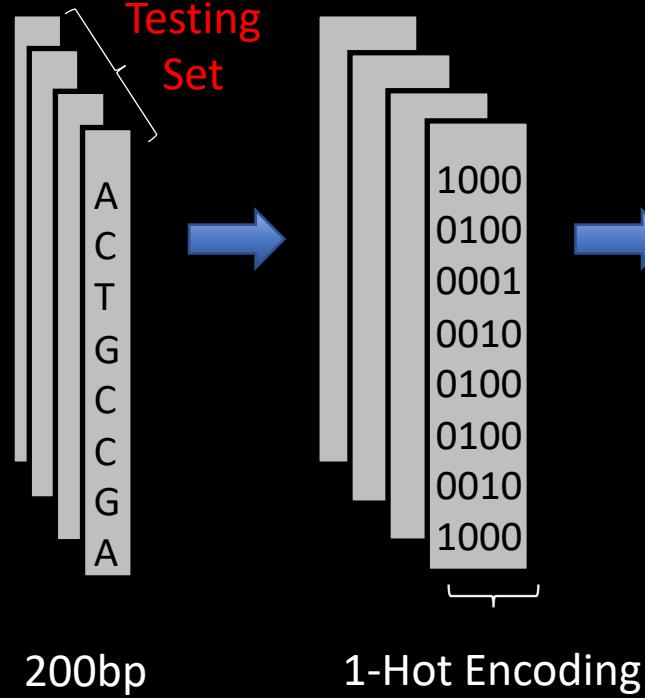
Source - Training - Validation



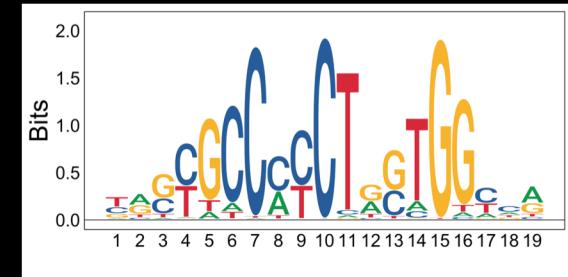
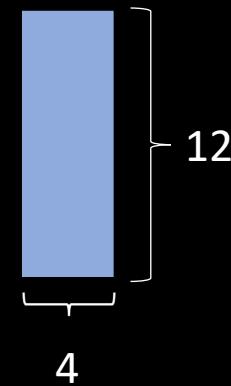
Source - Training - Validation



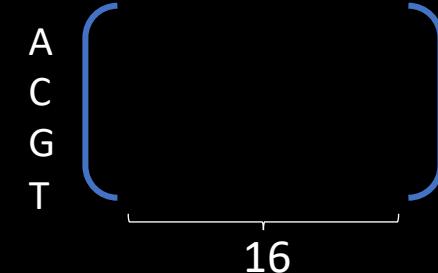
Input Sequences



Convolutional Layer

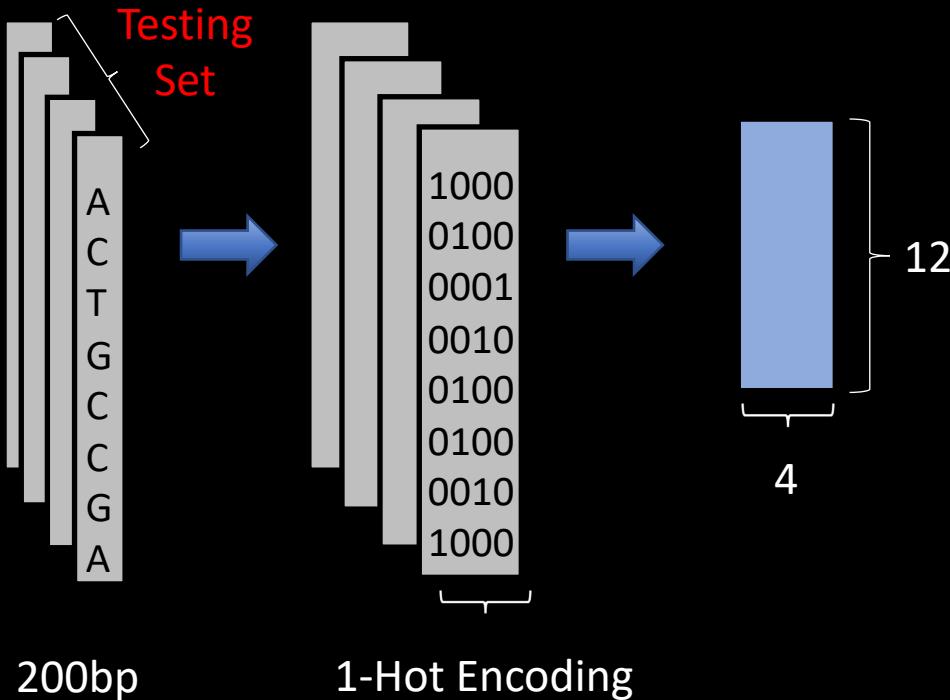


Learn Position Weight or Information Content



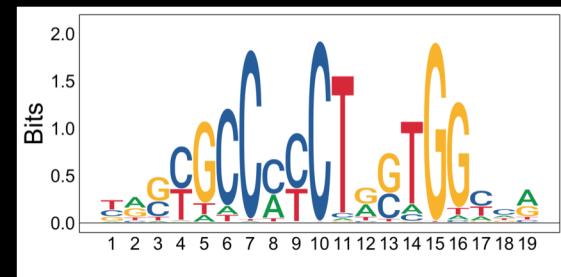
Input Sequences

Convolutional Layer

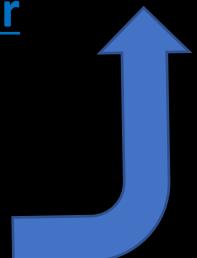
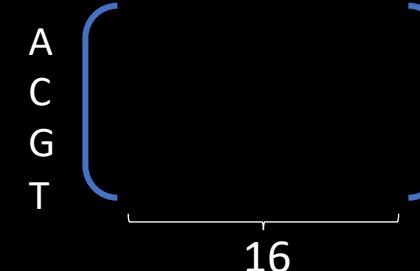


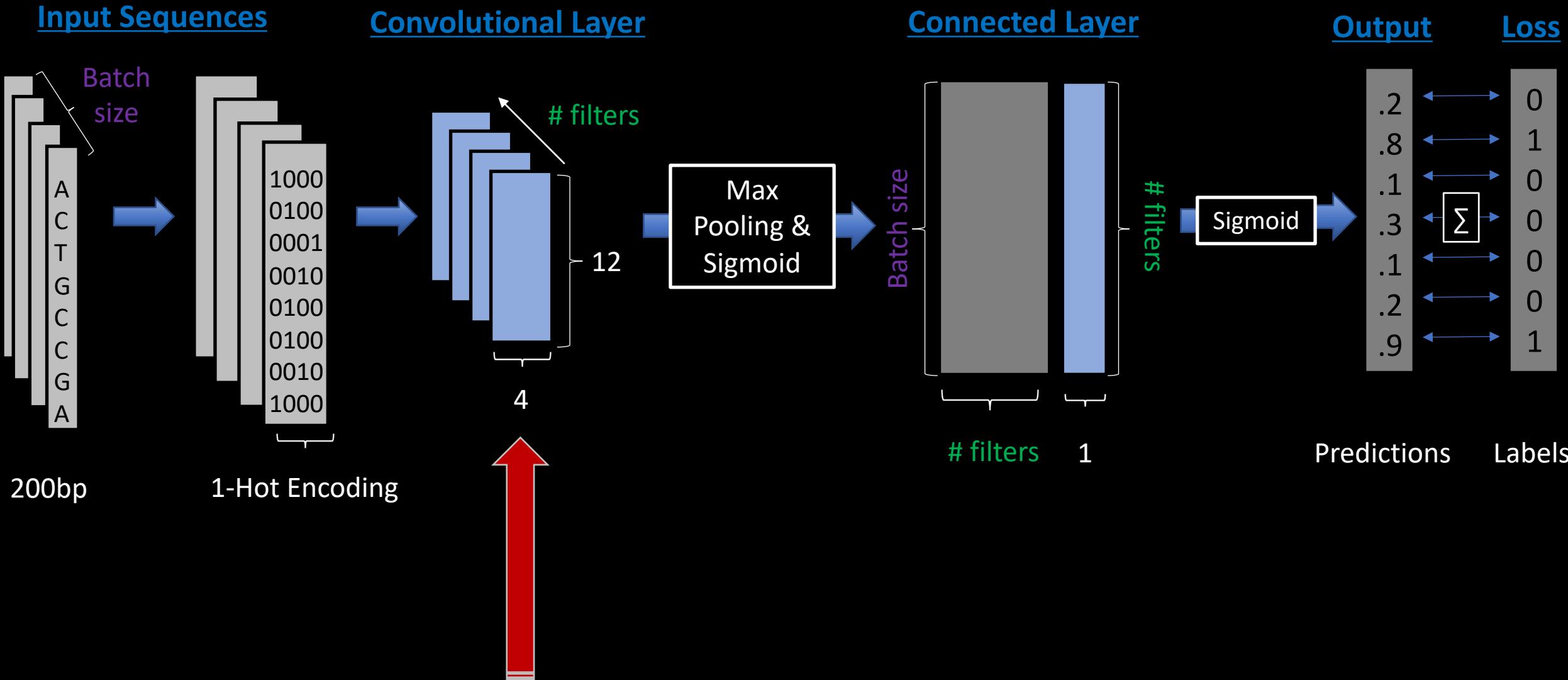
Achieved through:

1. Weight constraints
2. Peri-training weight transformation
3. Regularization

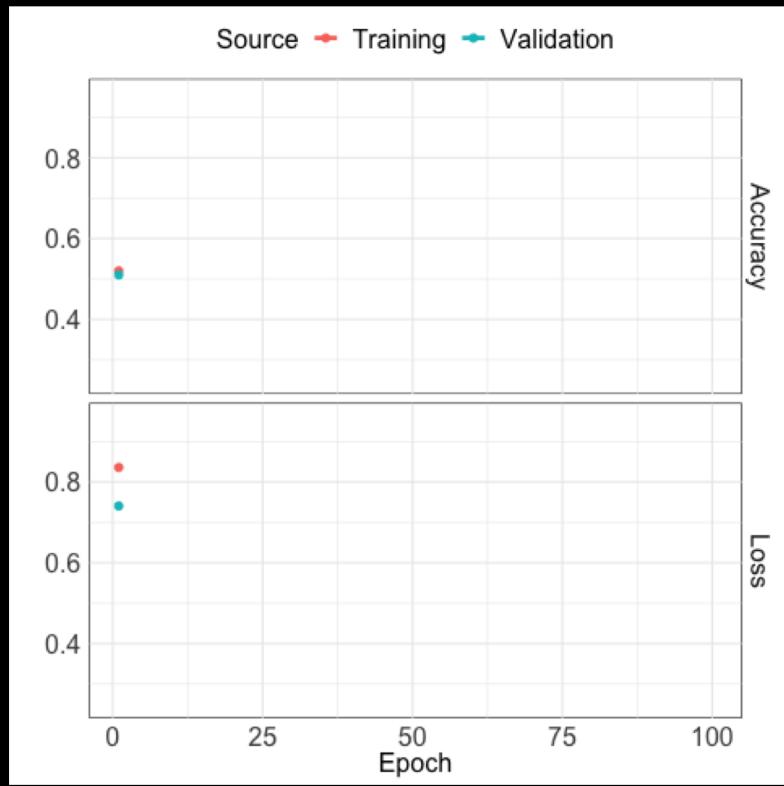


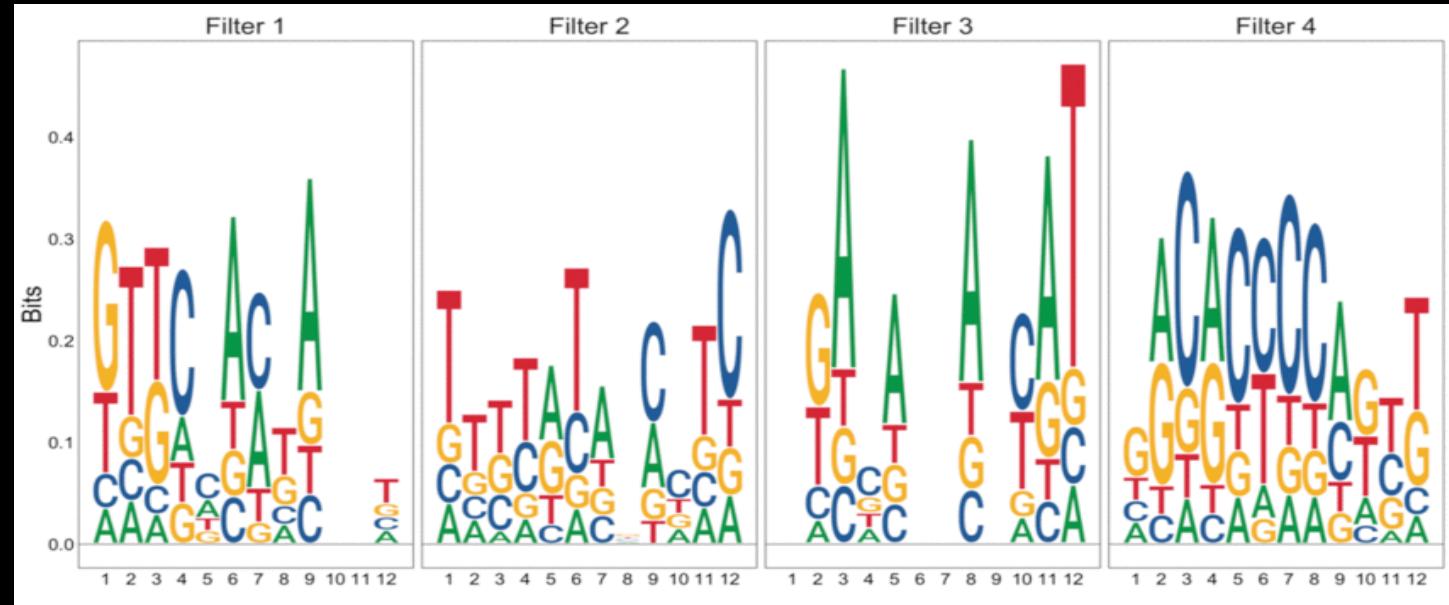
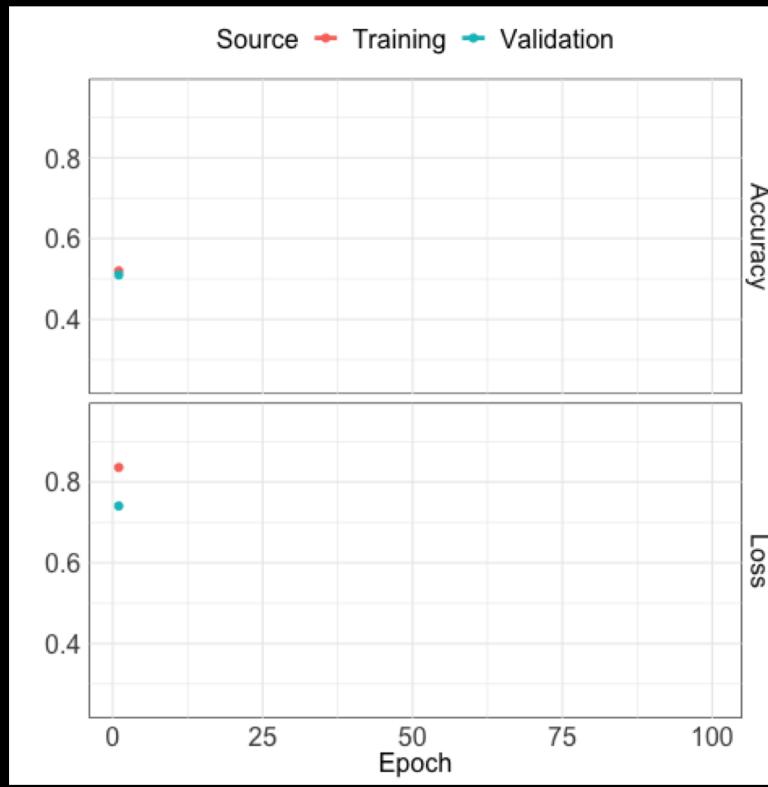
Learn Position Weight or
Information Content

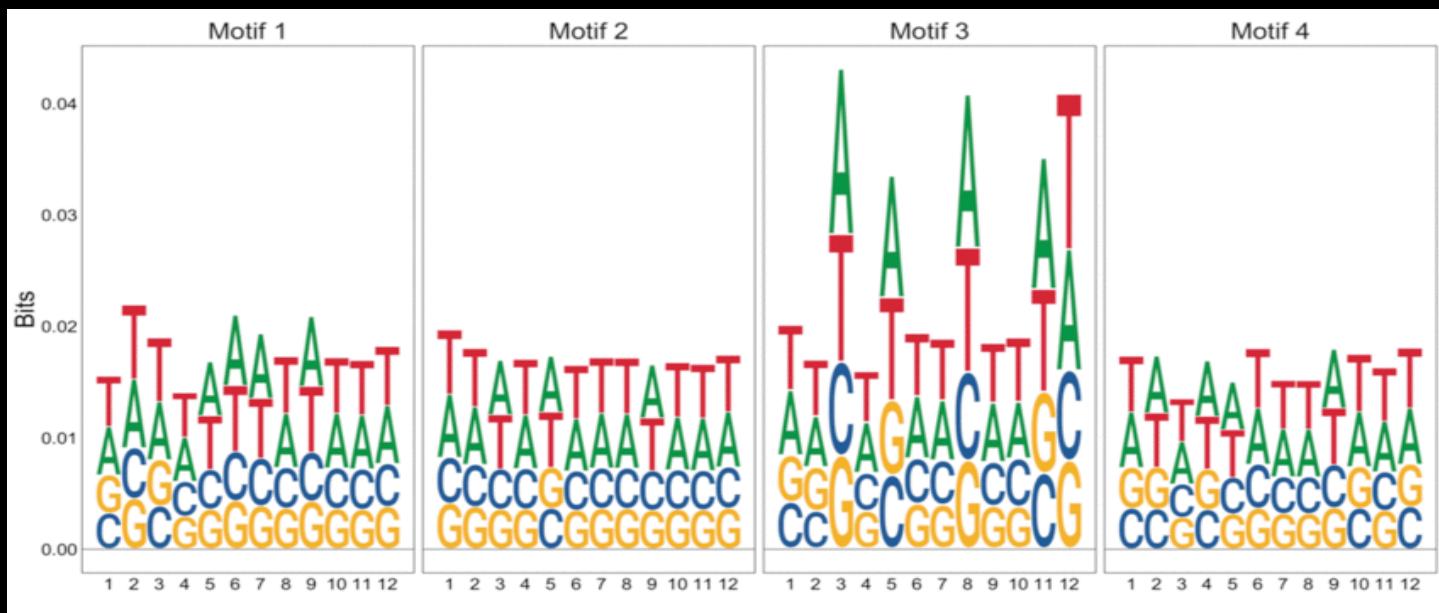
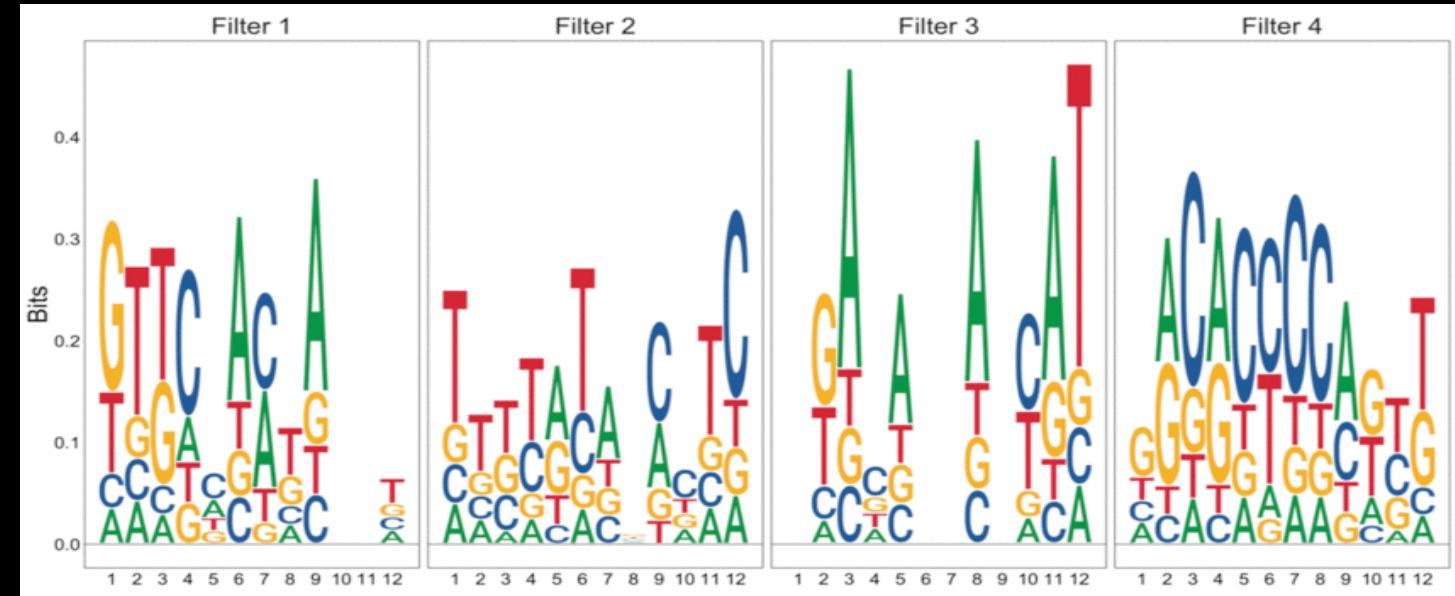
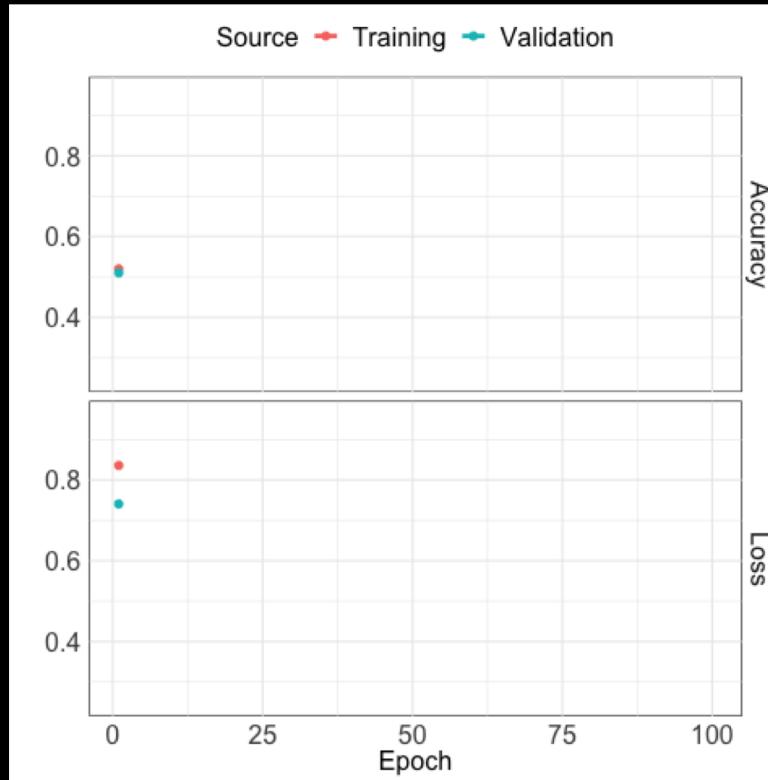


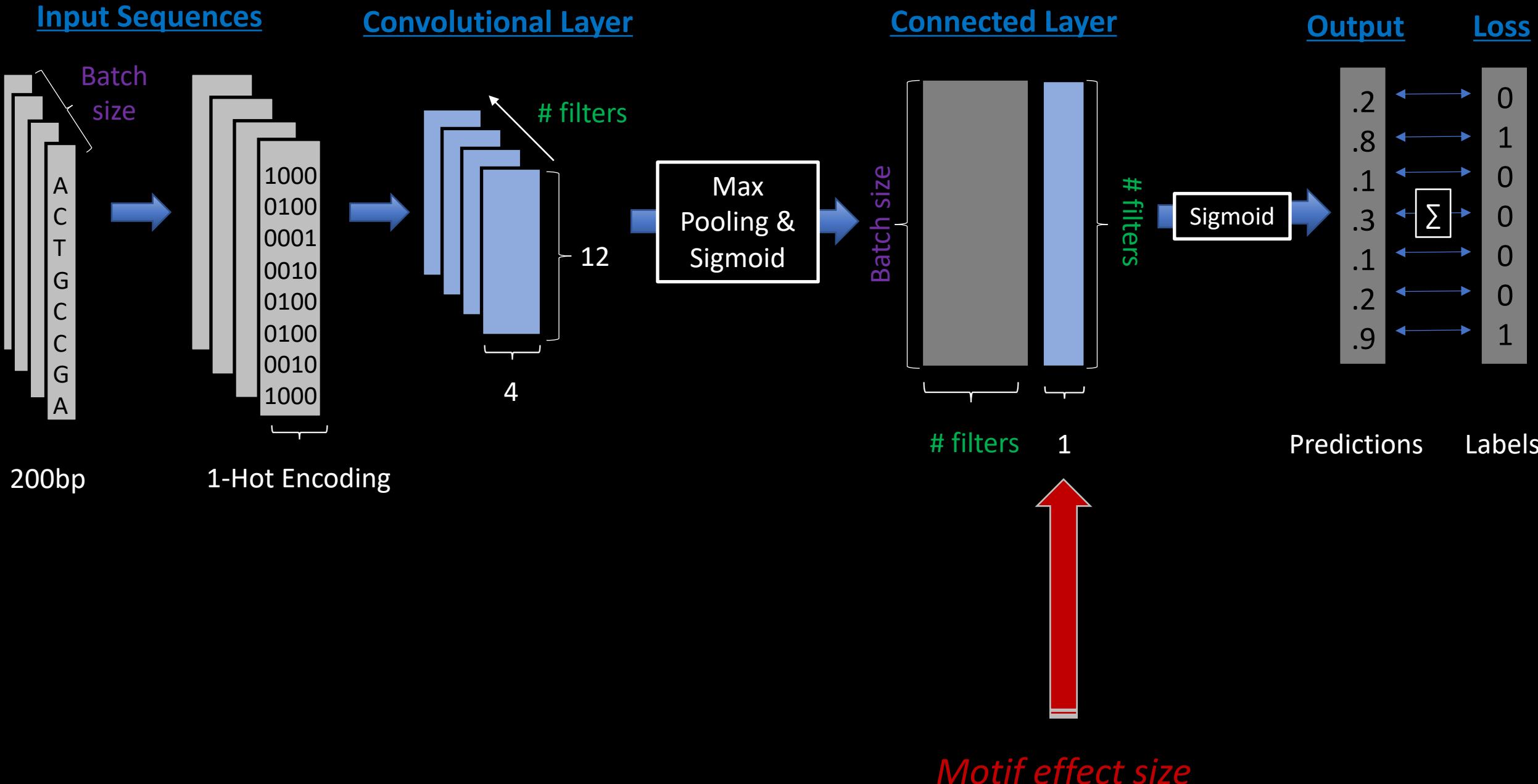


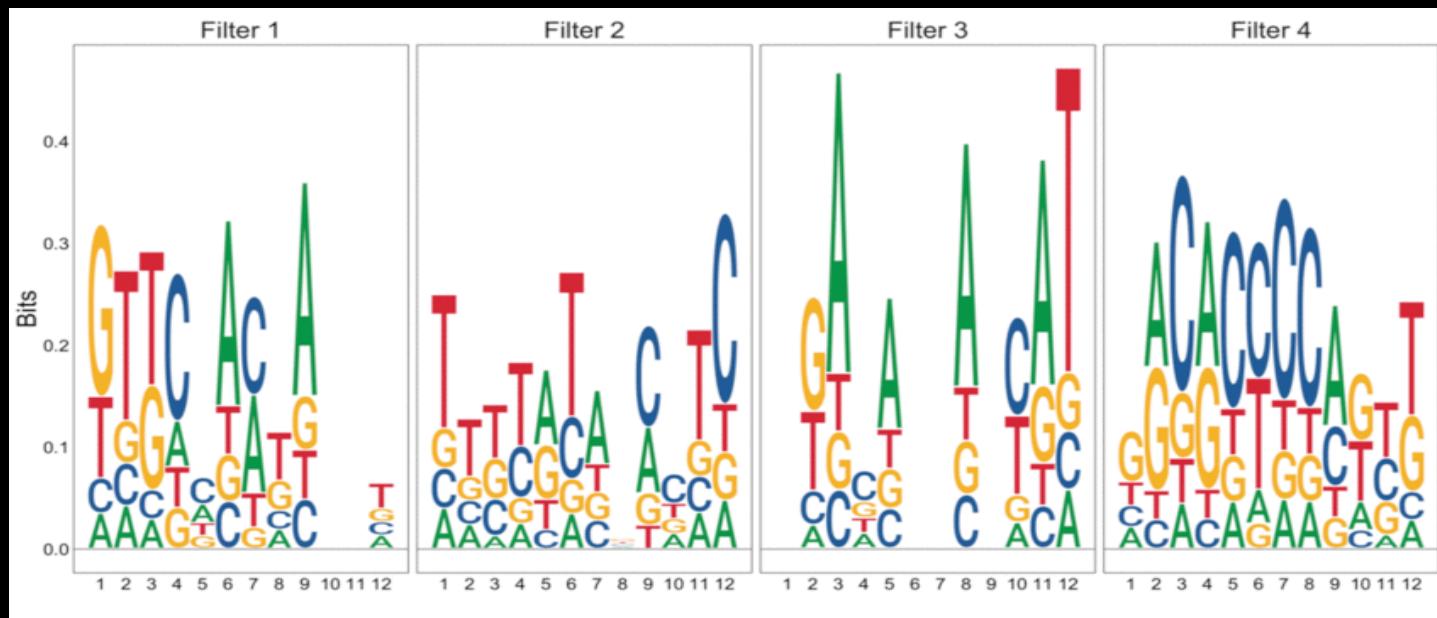
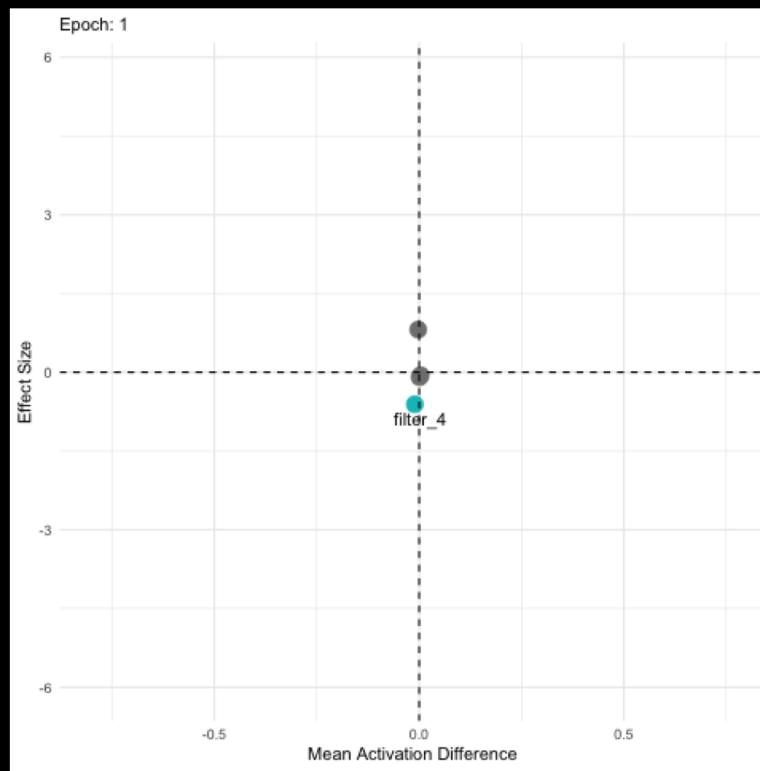
Learn motifs directly





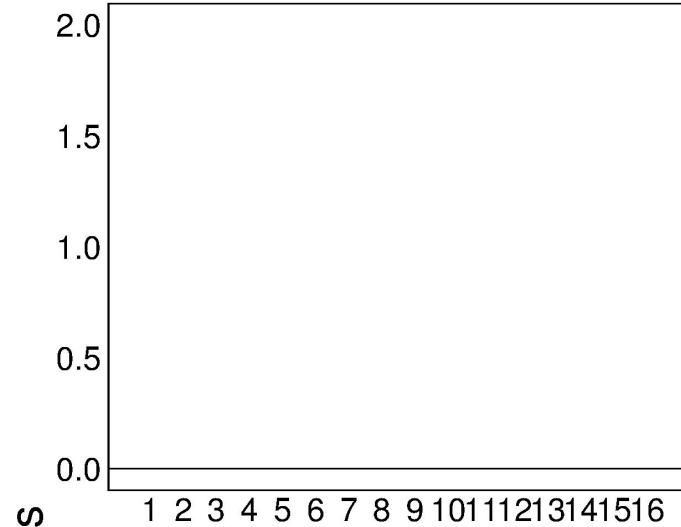


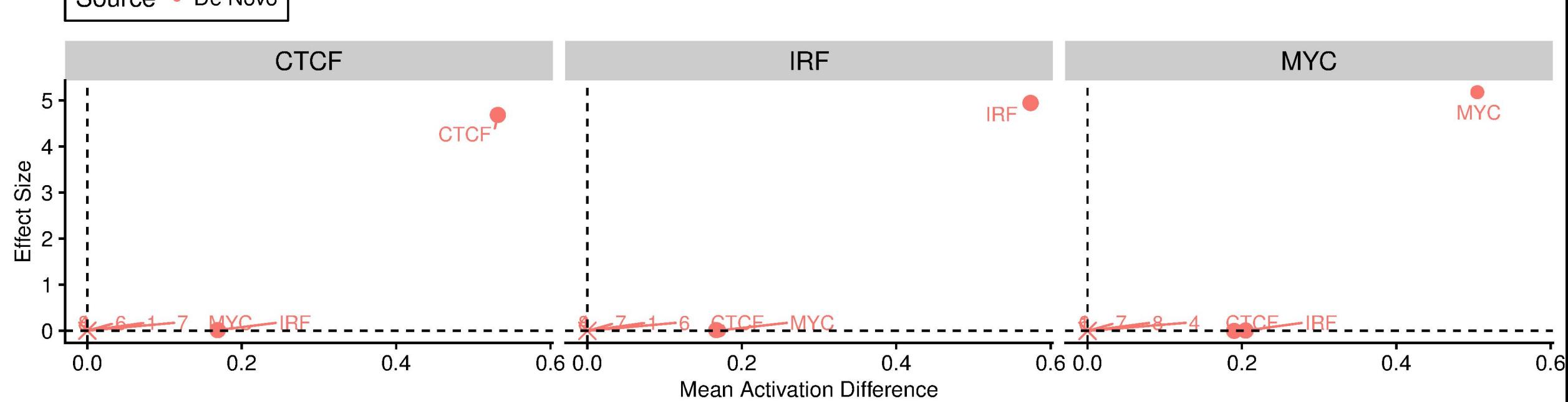
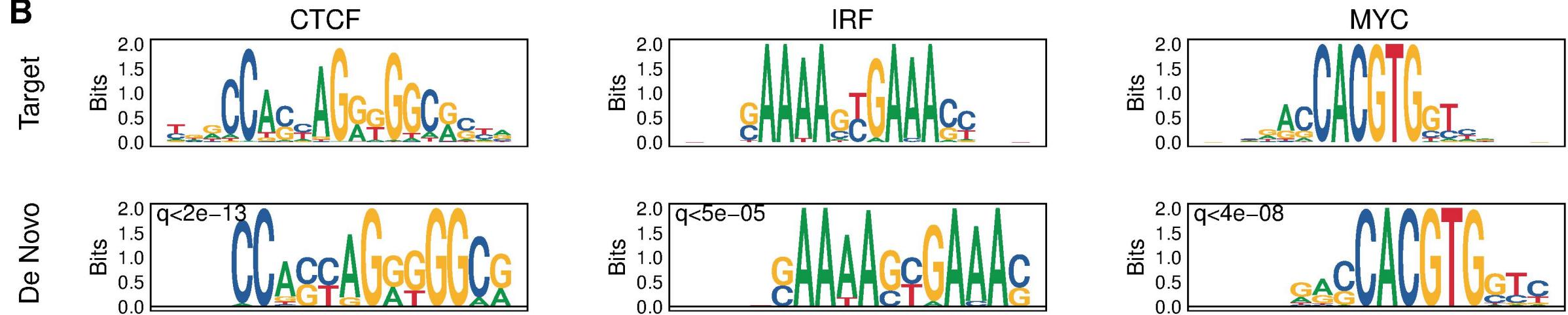


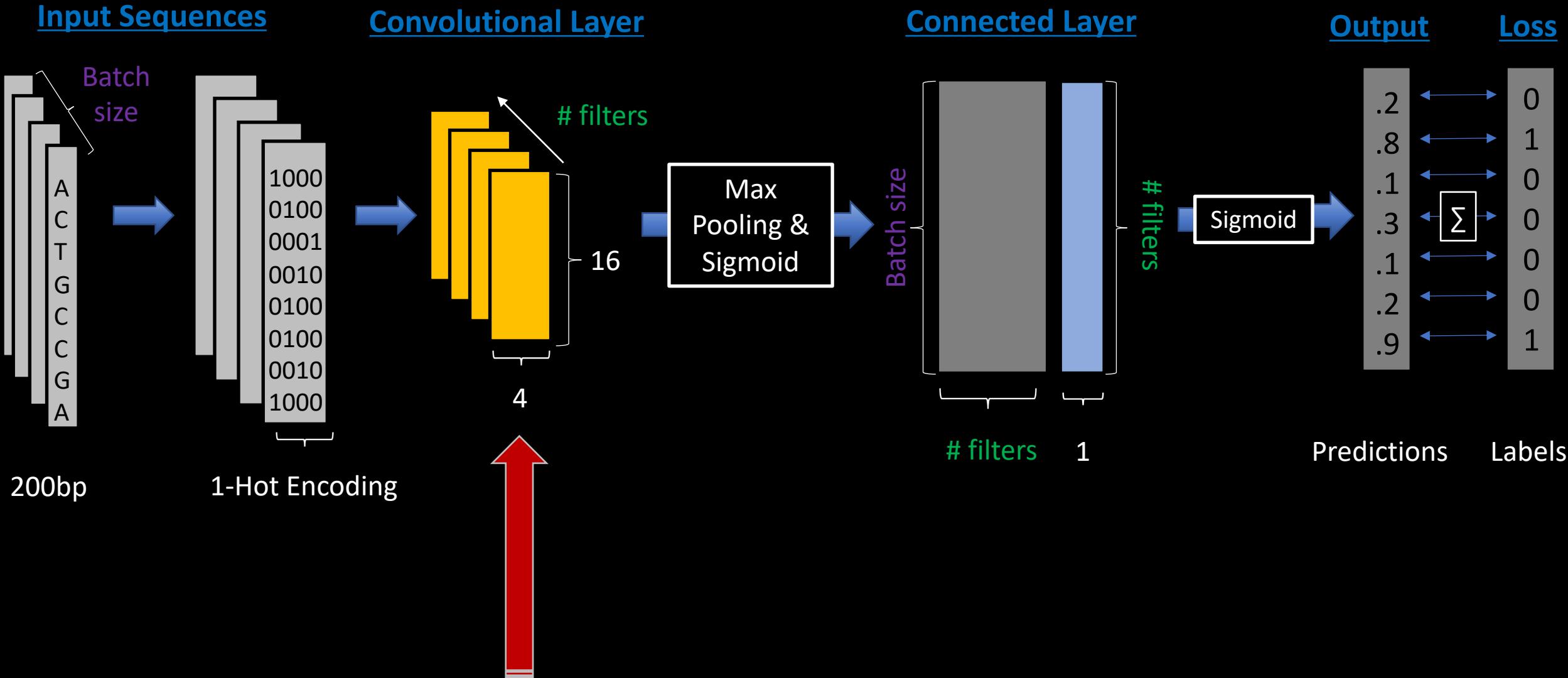


Sequence (X)	CTCF	IRF	MYC
ACTAATGAGAGTAGTTTATACTGGCCACCCAGGTGGCGTATCGCTGCTAAGAGCTTGATATAAAC ATCTCAACAGCCTTGATGGAATAAACAAACTCCCGCGTTAGGAGTATGTTGCCTCATAGCTTT GTAGGATAGCAGGGGCCGATAAAATTAAACCATTCAAGGTGCCACTAACATAGAACACGA	1	0	0
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATTAAAAGTGAAA CAAGCACCTCCG CACGGTATAACATTAACATTTTATCCCATCTCTGTATACAGAGAACGTCTTAACCTATTCGTATT ACTGTGTGTCTATGTGACTCTCACCTTACAGAACCAAGAAAGTGCATTGGATATCTCGACGA	0	1	0
GCGAAAGATAATGCCTCAGACTTCATTCACGCTTGTGGGTGGATTCCACCTATAGGACGTACTT ACGGCCTATTAAAGCAAAGCCAGAGAGAGGTTCACGTGATGTTAATTGGGTACACATTCAAG CTATGTCGGCTTAATGAGTCCACCAGAGGGCTTATTCCCTGGAGACATCAGTCTATGTGGCTTA	1	0	1
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATA GGAAAAATAAAAGCAGATTACTTCAGATAGCAACAAGAACAGTGAUTCCAGCATTCAAGTCAA ACAAATTACGCAGTATGGGGGGGGTATTAAGCGTTATGTGGAACTGCGAGATCATCTCATT GCCAGCAA	0	0	0
• • •	• • •	• • •	• • •

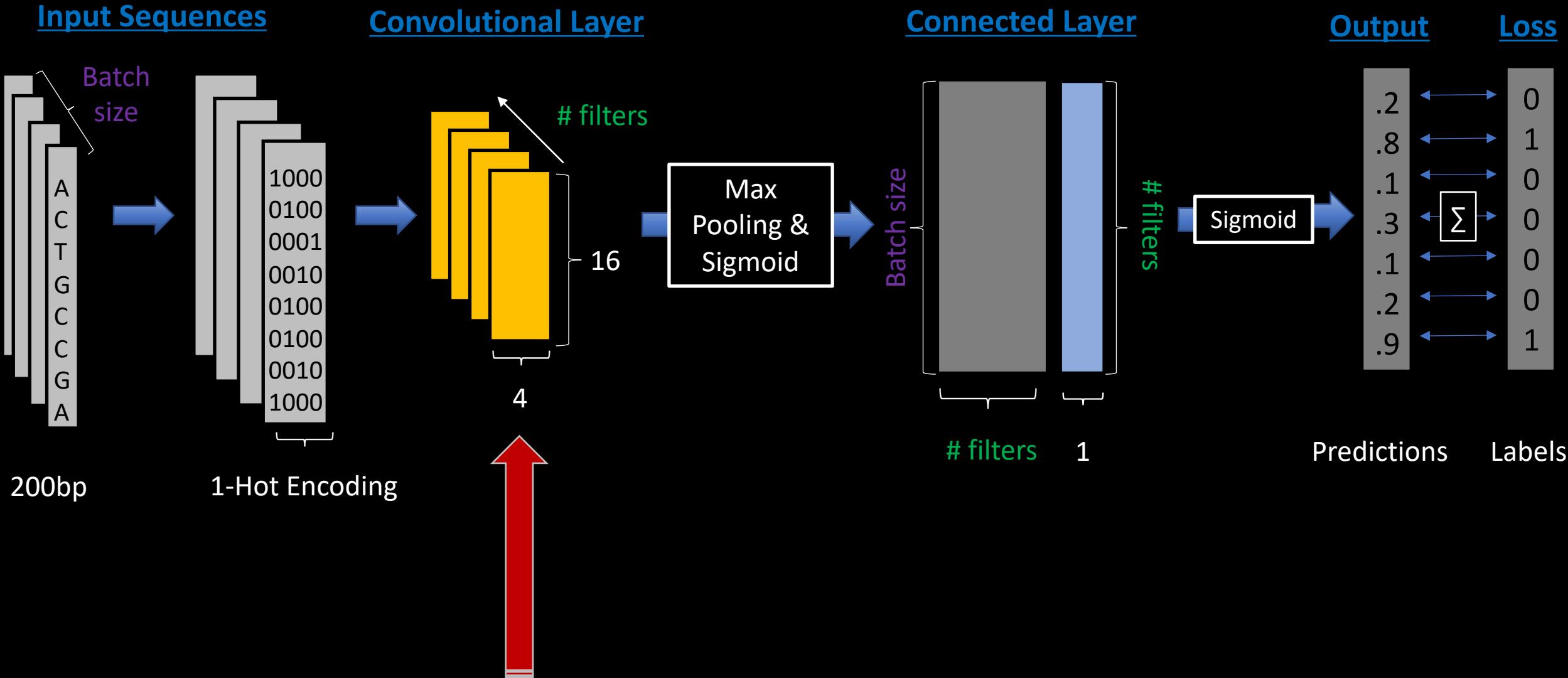
Filter 1



A**B**

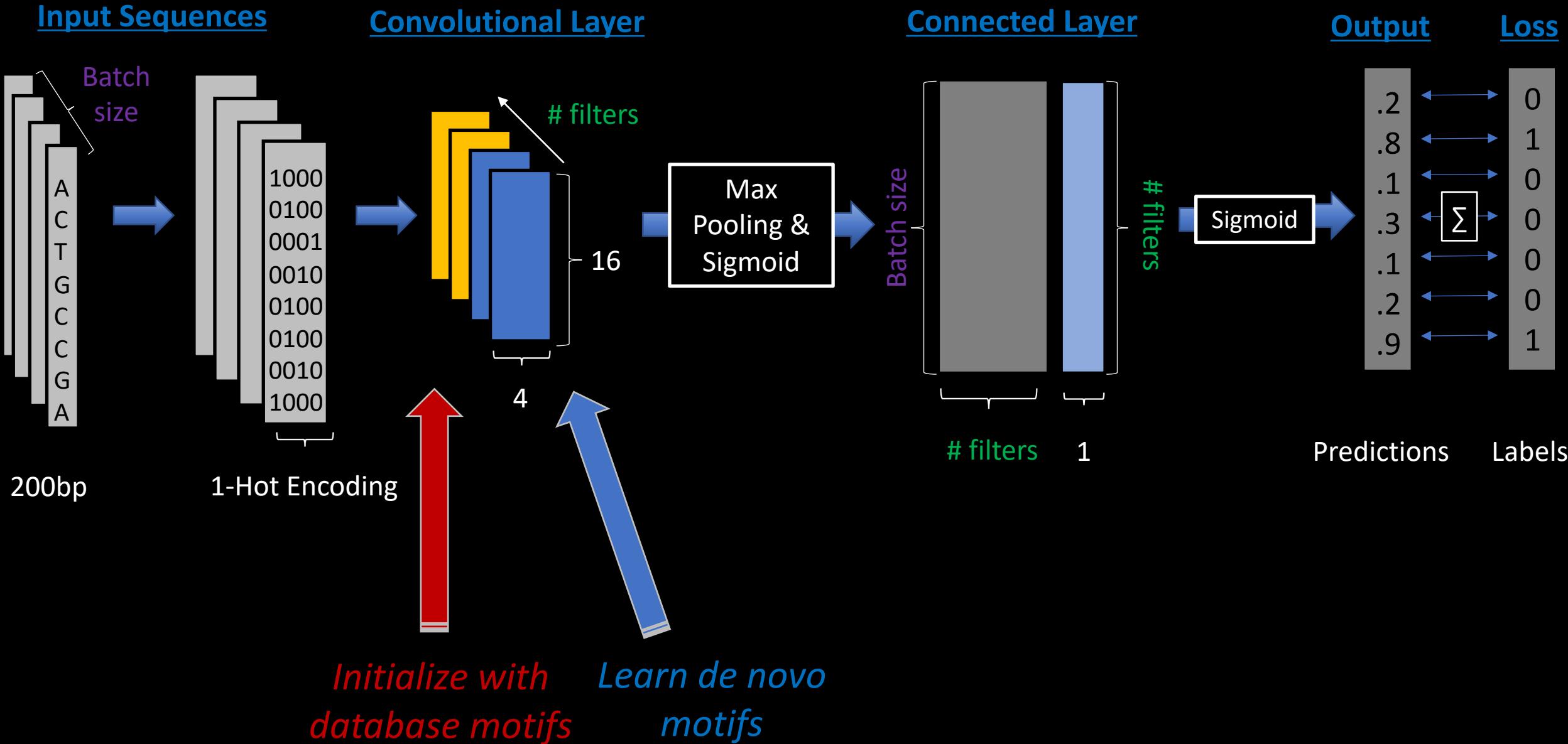


*Initialize & fix with
database motifs
(Jaspar)*



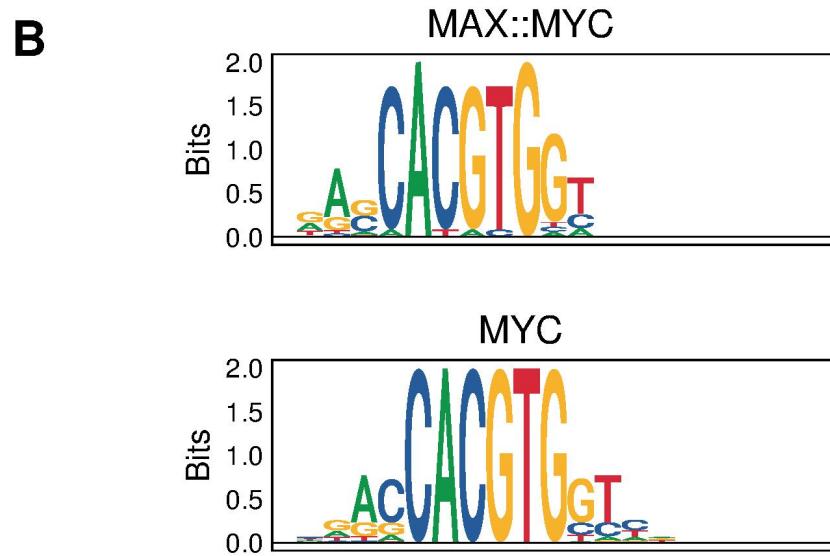
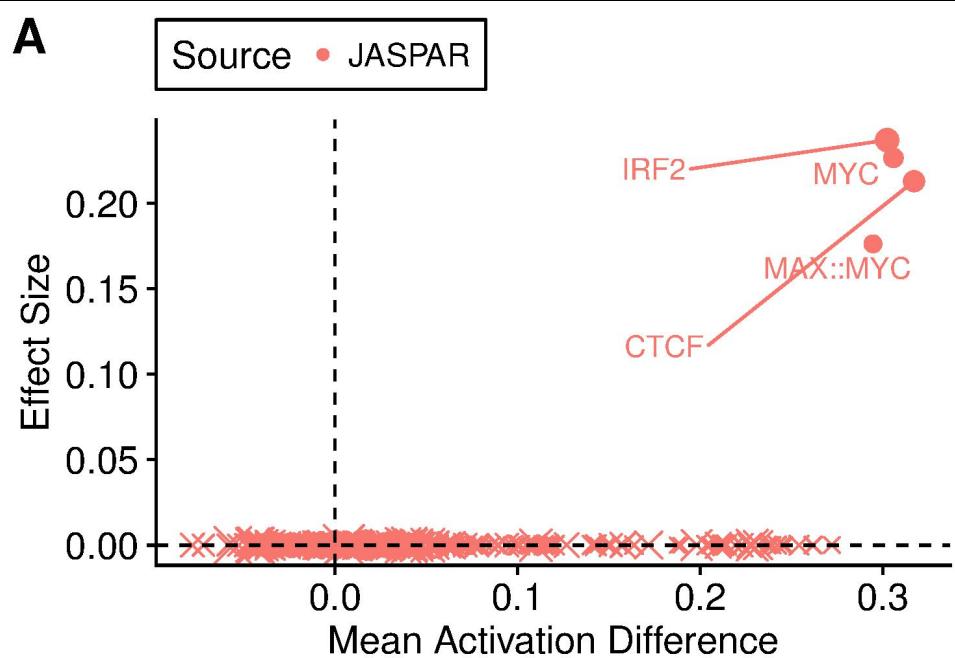
*Initialize & fix with
database motifs
(Jaspar)*

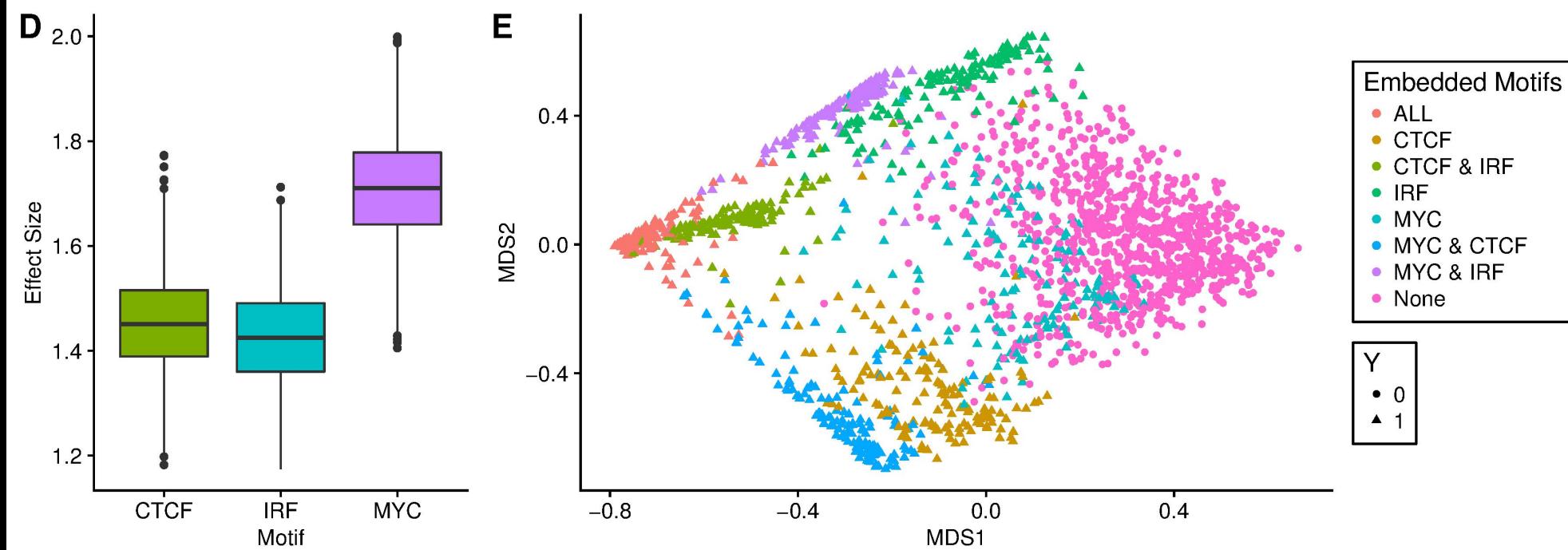
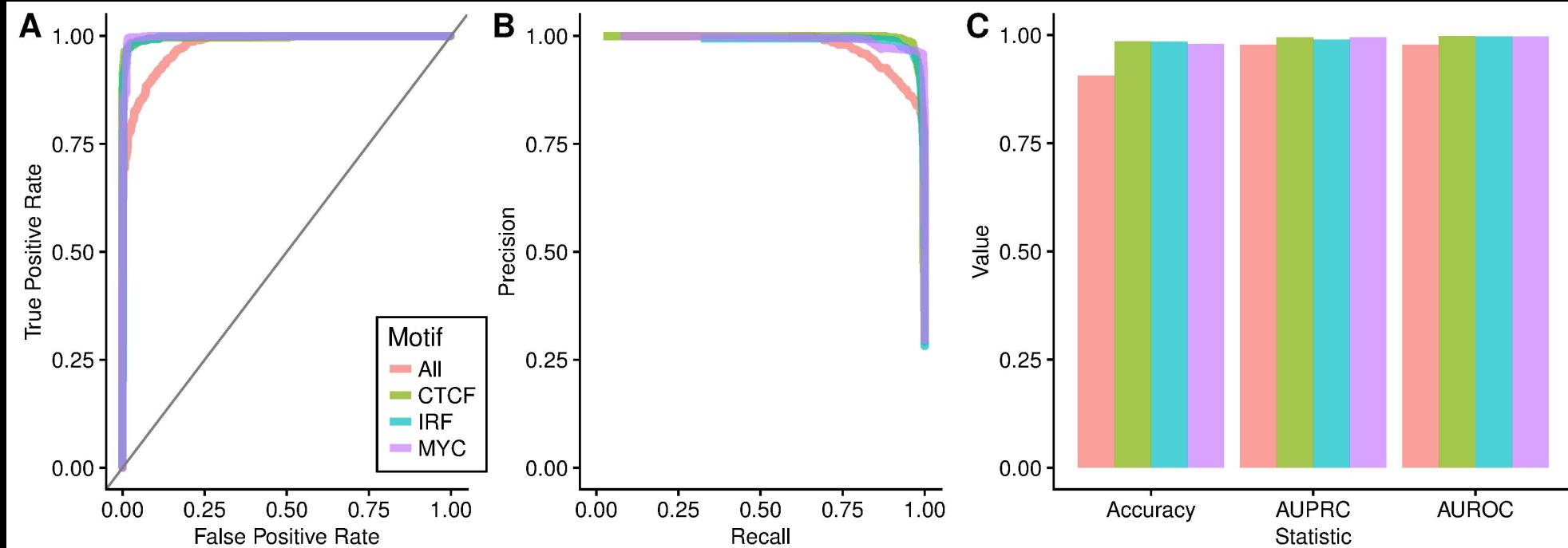
Transfer learning

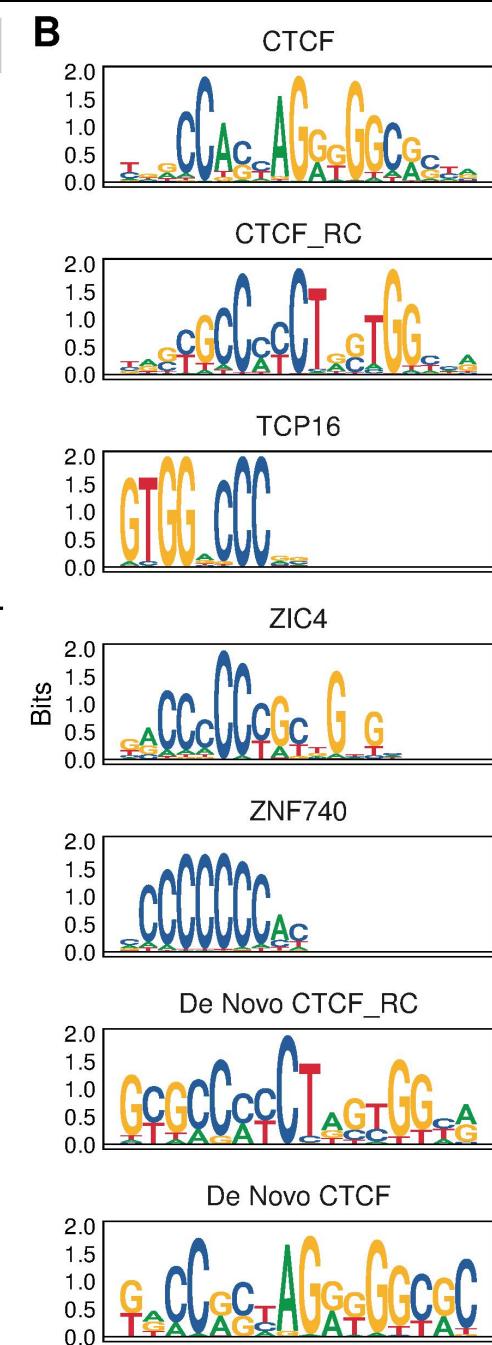
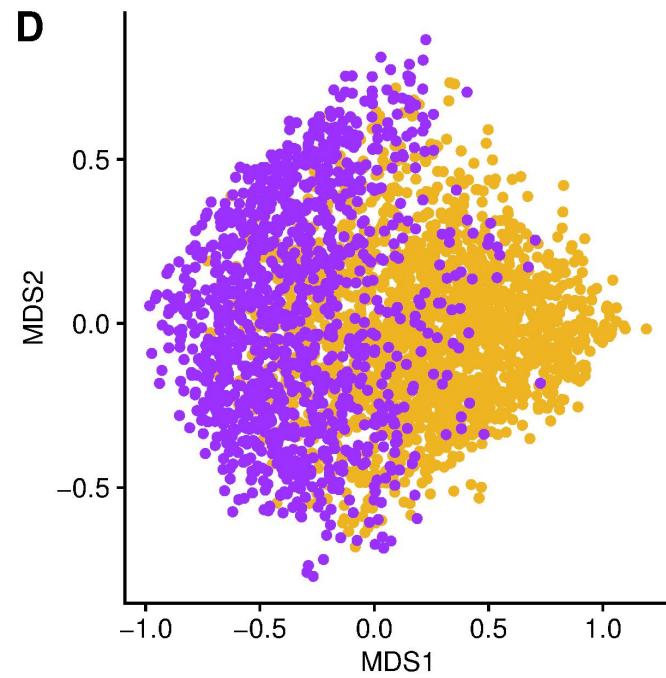
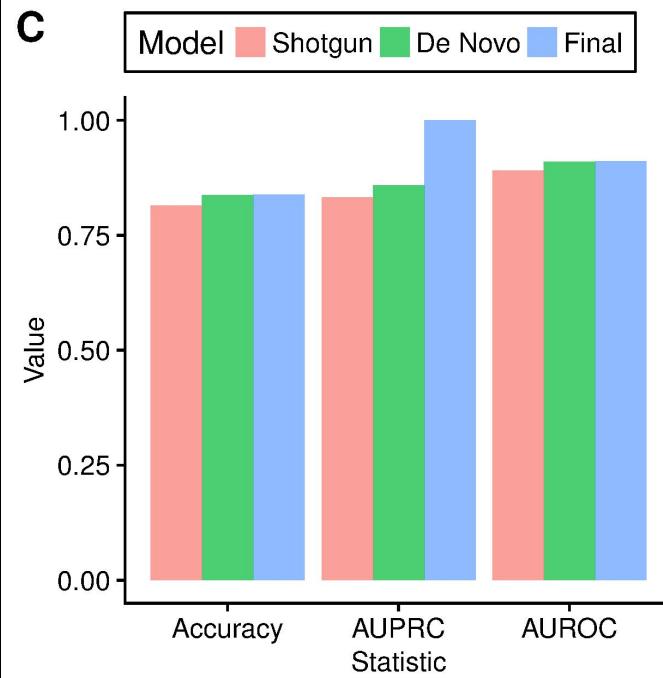
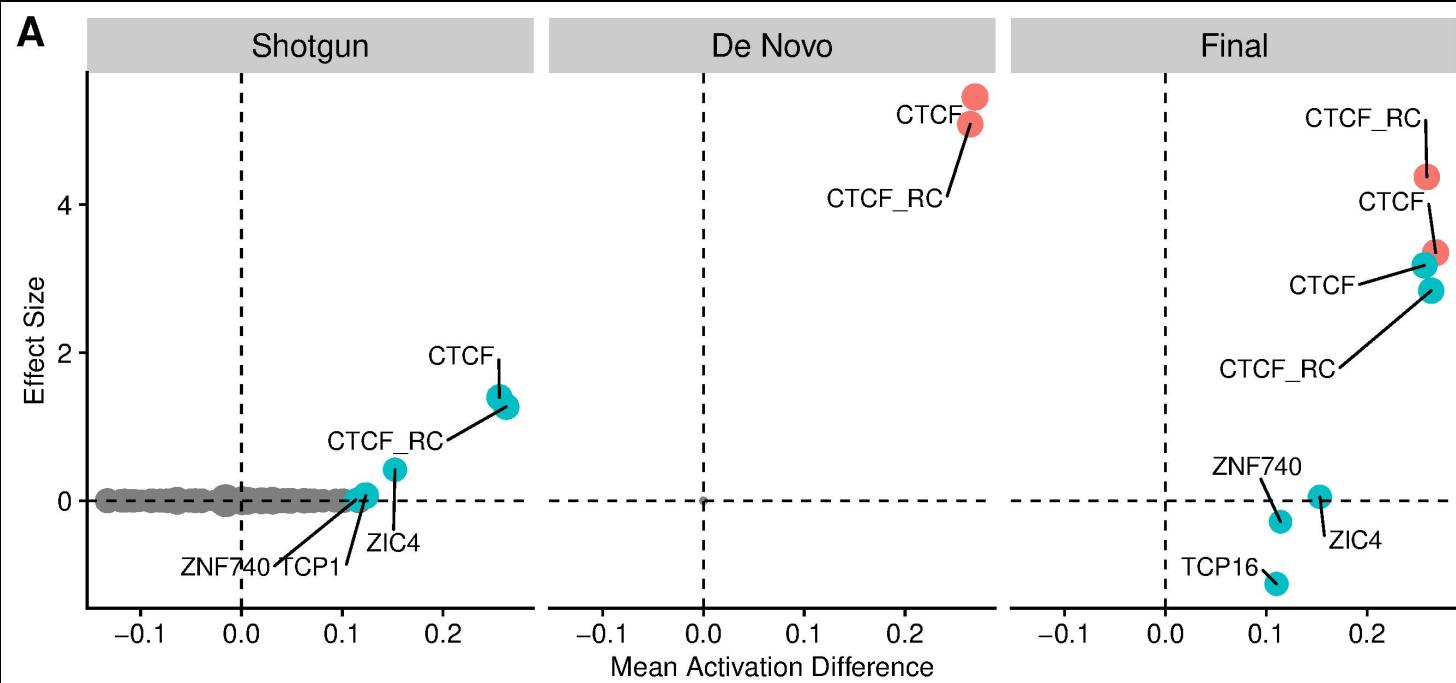


Sequence (X)	CTCF or IRF or MYC
ACTAATGAGAGTAGTTTATACTGGCCACCCAGGTGGCGTATCGCTGCTAAGAGCTTGATATAAACATCTCAAC AGCCTTGATGGAATAAAACAAACTCCCGCGTTAGGAGTATGTTGCCTCATAGCTTGTAGGATAGCAGGGGC CGATAAAAATTAAACCATTCAAGGTGCCACTAACATAGAACACGA	1
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATTAAAAGTGAAA CAAGCACCTCCGCACGGTAT AACATTAACATTTTATCCCATCTCTTGATACAGAGAACGTCTTAACTTATTTCGTATTACTGTGTGTATGTG ACTCTCACCTTACAGAACCAAGAAAGTGAATTGGATATCTCGACGA	1
GCGAAAGATAATGCCTCAGACTTCATTCACGCTTGTGGGTGGATTCCACCTATAGGACGTACTTACGGCCTA TTAAGCAAAGCCAGAGAGAGGGTTT CACGTGATGTTAATTGGGTACACATTCAAGCTATGTCGGCTTAATG AGTCCACCAAGAGGGCTTATTCCCTGGAGACATCAGTCTATGTGGCTTA	1
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATAGGAAAATA AAAGCAGATTACTTCAGATAGCAACAAGAACAGTGACTCCAGCATTCAAGTCAAACAAATTACGCAGTATG GGGGGGGGTATTAAGCGTTATGTGGGAACTGCGAGATCATCTCATTGCCAGCAA	0
• • •	• • •

Sequence (X)	Cell type A
ACTAATGAGAGTAGTTTATACTGG CCACCCAGGTGGCG TATCGCTGCTAAGAGCTTGATATAAACATCTCAAC AGCCTTGATGGAATAAAACAAATACTCCCGCGTTAGGAGTATGTTGCCTCATAGCTTGTAGGATAGCAGGGGC CGATAAAAATTAAACCATTCAAGGTGCCACTAACATAGAACATGAACACGA	1
TGCAATGACGATACTACAAAATTCTATAGACGTATGCGGAATT AAAAGTGAAA CAAGCACCTCCGCACGGTAT AACATTAACATTTTATCCCATCTCTTGATACAGAGAACGTCTTAACTTATTTCGTATTACTGTGTGTATGTG ACTCTCACCTTACAGAACCAAGAAAGTGAATTGGATATCTCGACGA	1
GCGAAAGATAATGCCTCAGACTTCATTCACGCTTGTGGGTGGATTCCACCTATAGGACGTACTTACGGCCTA TTAAGCAAAGCCAGAGAGAGGGTTT CACGTG ATGTTAATTGGGTACACATTCAAGCTATGTCGGCTTAATG AGT CCACCCAGAGGGCTT ATTCCCTTGGAGACATCAGTCTATGTGGCTTA	1
ATCAAAATAGAACGCCAGACACTTGACCAAAAATTACTTGGTCATTGCTAAAATAGCCCTACATAGGAAAATA AAAGCAGATTACTTCAGATAGCAACAAGAACAGTGACTCCAGCATTCAAGTCAAACAAATTACGCAGTATG GGGGGGGGTATTAAGCGTTATGTGGGAACTGCGAGATCATCTCATTGCCAGCAA	0
• • •	• • •







Snapshot

- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- 1. Feed-forward feature maps (activation-based)

Snapshot

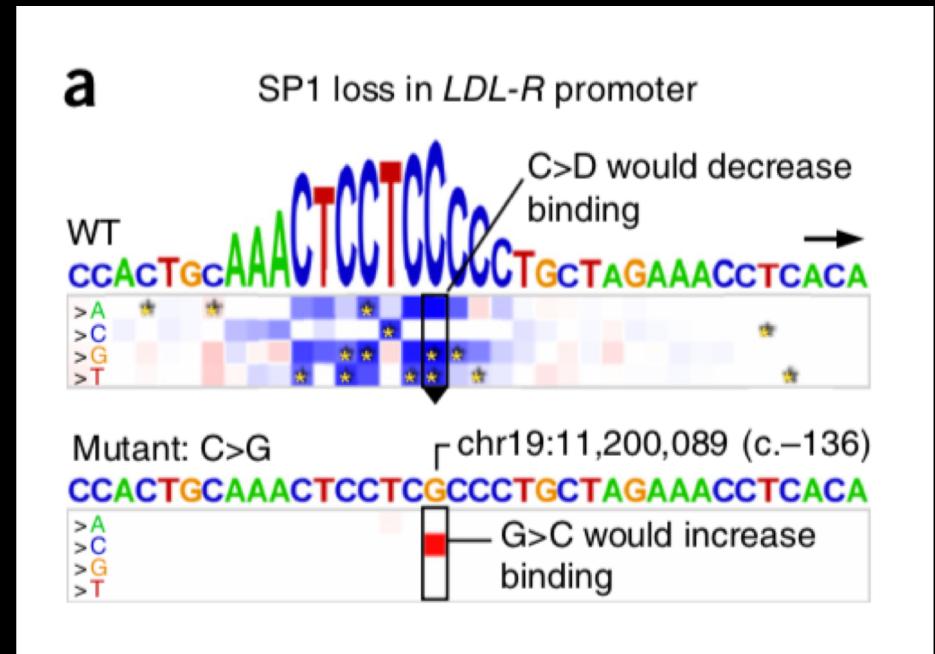
- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- 1. Feed-forward feature maps (activation-based)
 - 2. Model architecture (design-based)

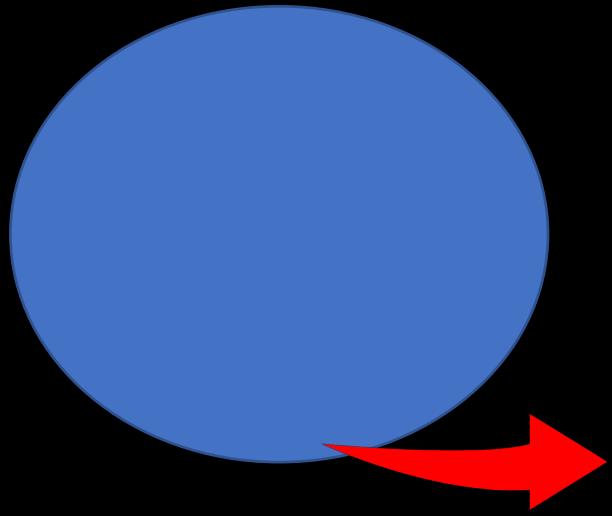
Snapshot

- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- 1. Feed-forward feature maps (activation-based)
 - 2. Model architecture (design-based)
 - 3. In-silico mutagenesis (perturbation-based)

1. For a given observation of interest, compute prediction from model or node of interest
 - Observation could be a promotor associated with cancer, for example
2. Modify a given value of the input
 - Swap A to C
3. Obtain a new prediction
4. Calculate and visualize the difference

1. For a given observation of interest, compute prediction from model or node of interest
 - Observation could be a promotor associated with cancer, for example
2. Modify a given value of the input
 - Swap A to C
3. Obtain a new prediction
4. Calculate and visualize the difference





TATAGACGTATGCGGAATT**CCCGCCCC**CAAGCACCTCCGCACGGTATAACATTAACATTAACTTTATCCCATCTCTTGATACAGAGAACGTCTTAACTTA

Snapshot

- Problem formulation
 - *Where do we get the data?*
 - Data setup
 - *What do the data look like?*
 - Opening the black box
 - *What has the model learned?*
- 
- 1. Feed-forward feature maps (activation-based)
 - 2. Model architecture (design-based)
 - 3. In-silico mutagenesis (perturbation-based)
 - 4. Backpropagation-based

1. Many algorithms exist

- Saliency maps/gradient times input (Simonyan et al 2013)
- Guided backpropagation (Springenberg et al 2014)
- Layerwise relevance propagation (Bach et al 2015)
- Integrated gradients (Sundararajan et al 2016)
- Grad-CAM, Guided CAM (Selvaraju et al 2016)
- DeepLIFT (Shrikumar et al 2017)

1. Many algorithms exist

- Saliency maps/gradient times input (Simonyan et al 2013)
- Guided backpropagation (Springenberg et al 2014)
- Layerwise relevance propagation (Bach et al 2015)
- Integrated gradients (Sundararajan et al 2016)
- Grad-CAM, Guided CAM (Selvaraju et al 2016)
- DeepLIFT (Shrikumar et al 2017)

1. Many algorithms exist
 - Saliency maps/gradient times input (Simonyan et al 2013)
 - Guided backpropagation (Springenberg et al 2014)
 - Layerwise relevance propagation (Bach et al 2015)
 - Integrated gradients (Sundararajan et al 2016)
 - Grad-CAM, Guided CAM (Selvaraju et al 2016)
 - DeepLIFT ([Shrikumar et al 2017](#))
2. Connections exist between the methods
3. Pros and cons to different approaches but DeepLIFT recommended for genomics applications due to the discrete nature of nucleotide data

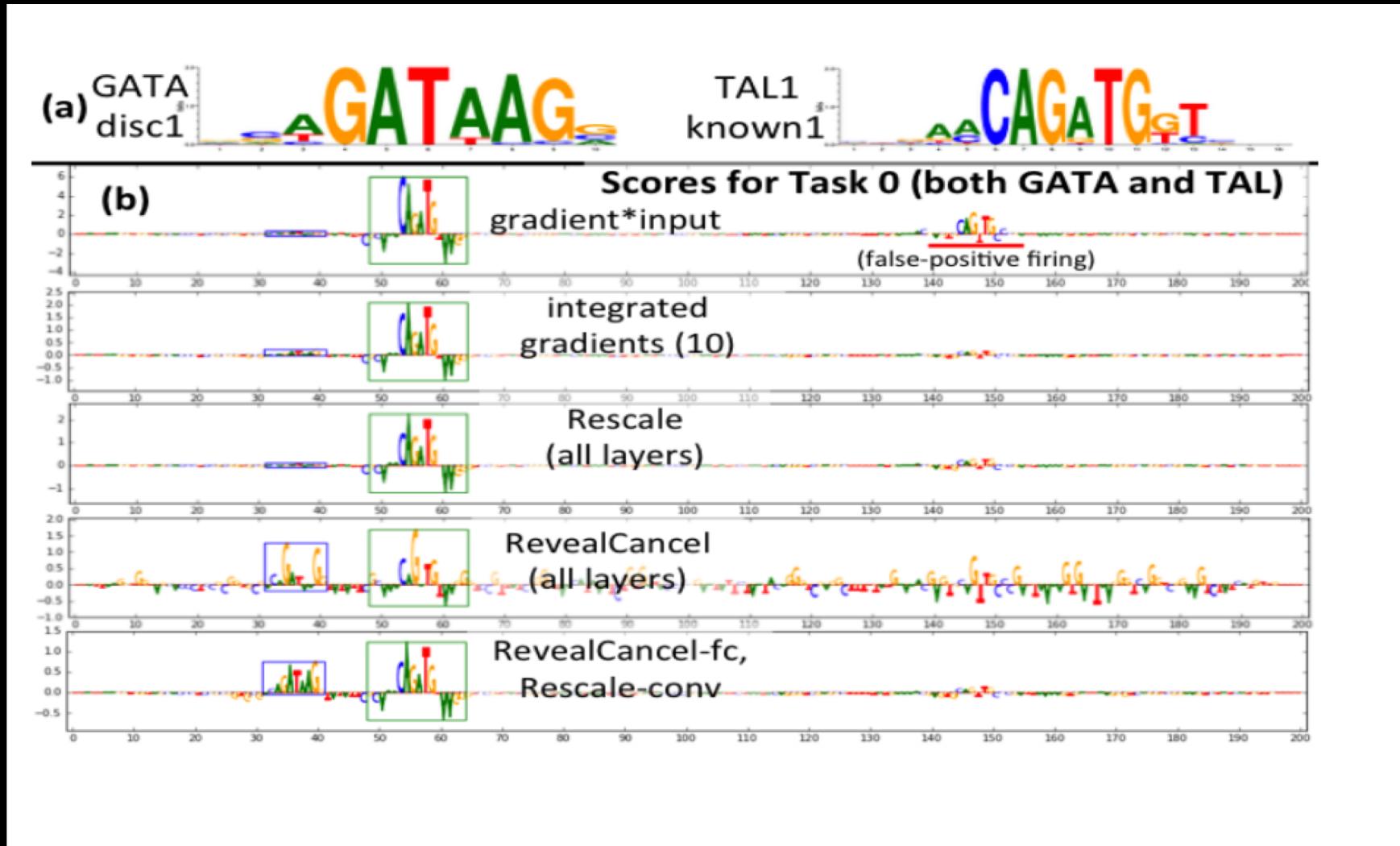
1. Many algorithms exist

- Saliency maps/gradient times input (Simonyan et al 2013)
 - Pick an observation
 - Calculate the gradient of the output w.r.t. the input observation
 - Multiply by the input
 - Visualize
- Guided backpropagation (Springenberg et al 2014)
 - Set negative gradients to zero when passing a ReLU
- Layerwise relevance propagation (Bach et al 2015)
 - Scaled version of gradients times input (Shrikumar et al 2016, Knidermans et al 2016)
- Integrated gradients (Sundararajan et al 2016)
- Grad-CAM, Guided CAM (Selvaraju et al 2016)
 - Feature map times gradient w.r.t. output class

- DeepLIFT (Shrikumar et al 2017)
 - Excellent software tool (see lab and homework)
 - Works with Keras models
 - Includes implementation of other methods
 - Requires a reference observation
 - E.g. genome background or an observation of all zeros

- DeepLIFT (Shrikumar et al 2017)

1. Calculate activations for each layer for input observation and reference
2. Compute difference between activations and input (X)
3. Use rules for backpropagating the differences and assigning importance scores to preceding neurons
 1. Ensures summation to delta property
 1. Contribution scores sum to the total difference from reference
4. Requires a single backward pass to calculate importance scores
5. Parallels to the chain rule with finite differences



Shrikumar, Avanti, Peyton Greenside, and Anshul Kundaje. "Learning important features through propagating activation differences." *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR.org, 2017.

Valuable resources

- Kipoi Model Zoo
 - Zoo for trained models
 - <https://kipoi.org/>
- Review articles provided on slide 10
- Github links provided on slide 12
- Selene genomics pytorch API
 - <https://selene.flatironinstitute.org/>
- Deepomics (Tensorflow for genomics)
 - <https://github.com/p-koo/deeconomics>

Homework #3

- Available after class
- Goal: train a CNN on genomic data and try to understand what your model has learned
- Due next Monday at midnight (5/6 @ 11:59PM EST)
- Friday's lab (5/3) will provide a tutorial