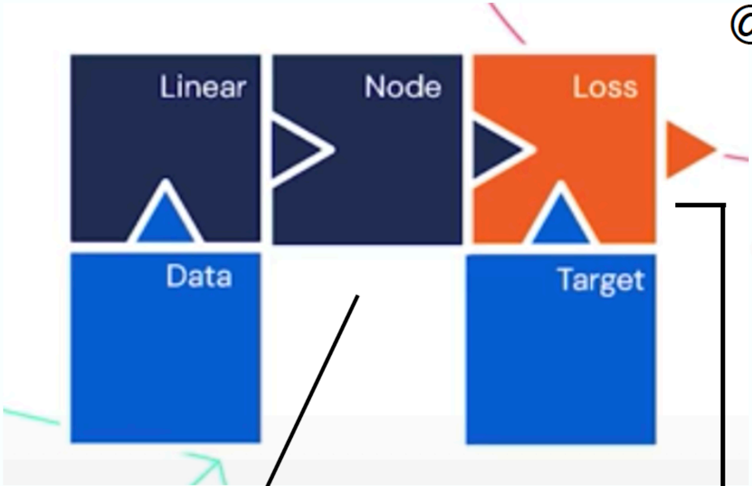
RECAP:

(deepmind
@Czarnecki)

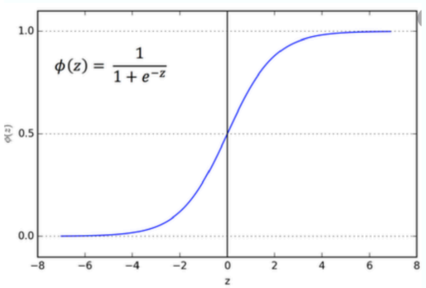


$$p_w(y|x)$$

$$\cancel{f_W(\vec{x}) = \vec{y}}$$

$$\boxed{g_3(W_3 g_2(W_2 g_1(W_1 \vec{x_0})))}$$

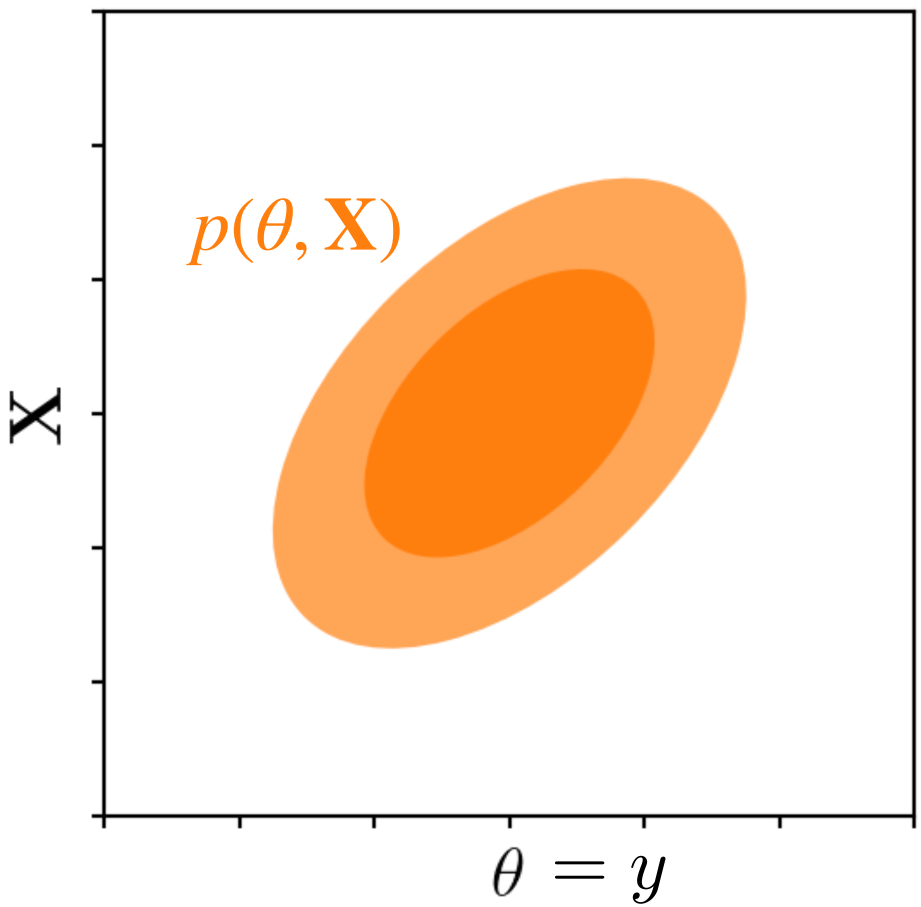
activation function



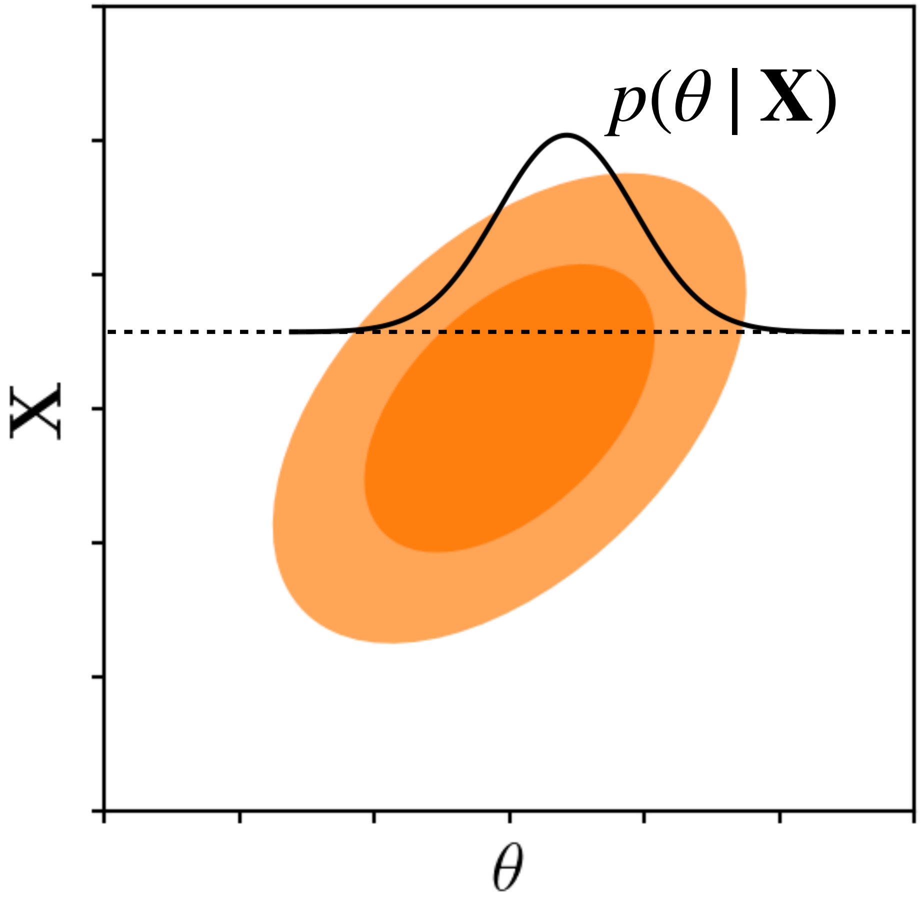
$$\frac{1}{N}\sum_{i=1}^{N}(y_i - f_w(x))^2$$

Minimized through gradient descent (backpropagation)

Consider the joint distribution of labels (model parameters) and data

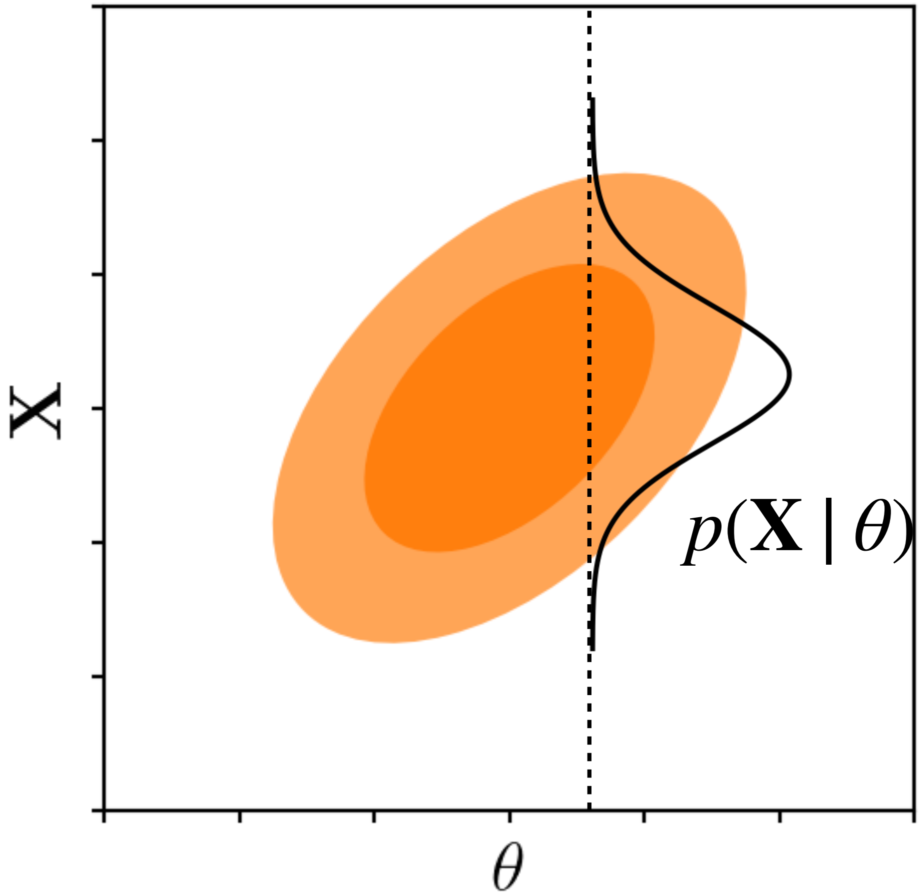


Discriminative models approximate a **posterior** $p(\theta \mid \mathbf{X})$



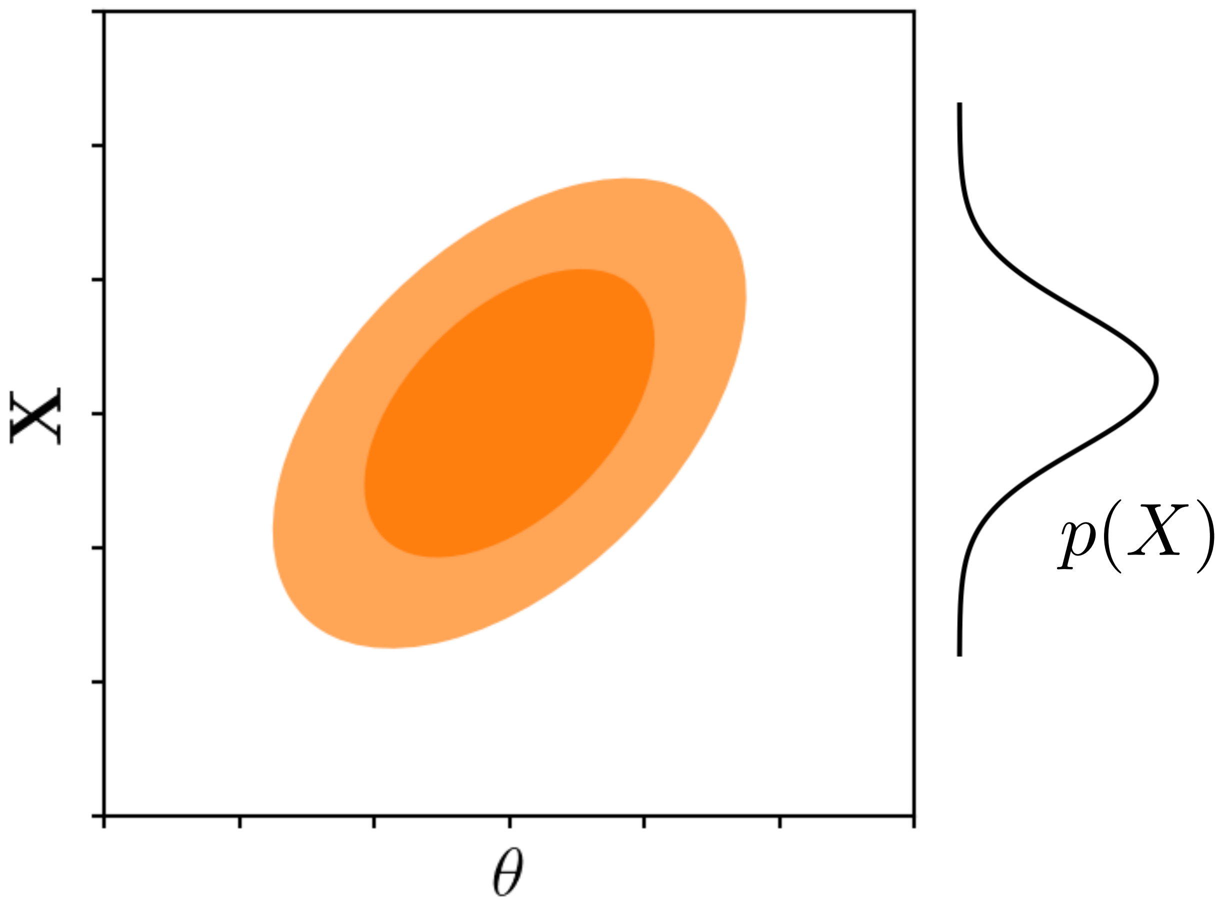
Approach	Discriminative	
Data type	Target Function	Method
Labels	$p(y\|x)$	All supervised networks with bottleneck

Conditional generative models approximate a **likelihood** $p(\mathbf{X} \mid \theta)$



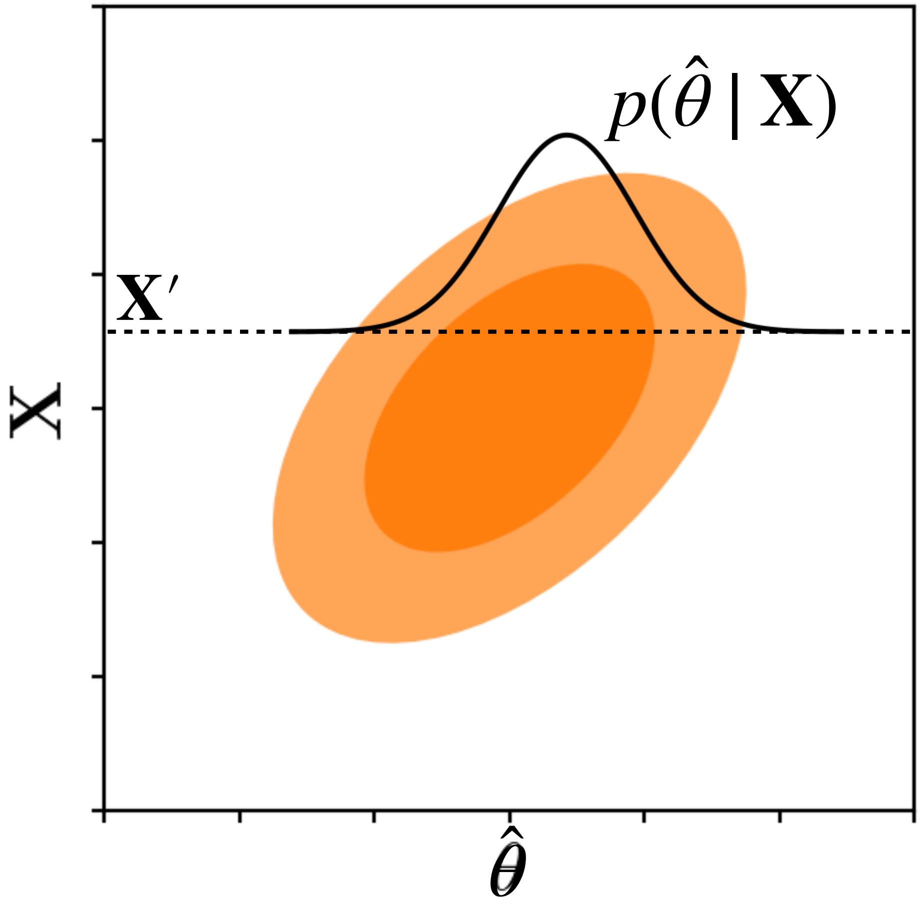
Approach	Discriminative		Generative	
Data type	Target Function	Method	Target Function	Method
Labels	$p(y\|x)$	All supervised networks with bottleneck	$p(x\|y)$	Conditional generative models

Generative models approximate the “evidence” $p(\mathbf{X})$



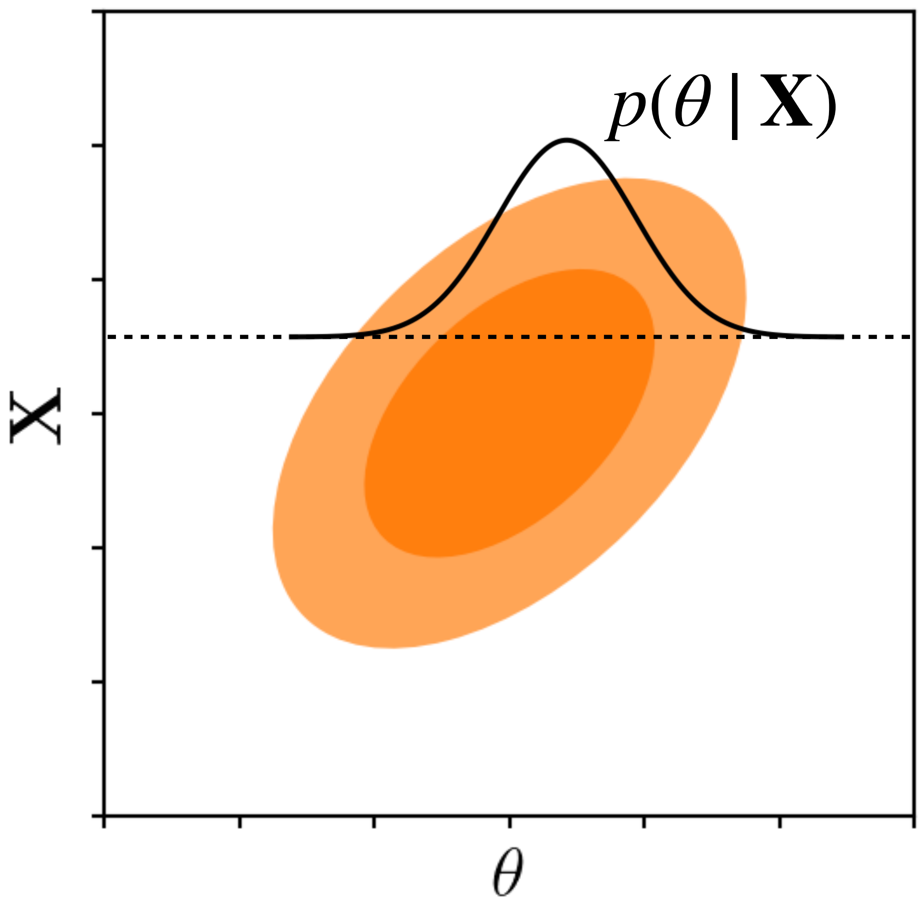
Approach Data type	Discriminative		Generative	
	Target Function	Method	Target Function	Method
Labels	$p(y\|x)$	All supervised networks with bottleneck	$p(x\|y)$ $p(x)$	Conditional generative models Generative models, Autoencoders

Self-supervised learning is a discriminative model, with a pretext task $\hat{\theta}$



Approach / Data type	Discriminative		Generative	
	Target Function	Method	Target Function	Method
Labels	$p(y\|x)$	All supervised networks with bottleneck	$p(x\|y)$	Conditional generative models
No Labels	$p(\hat{y}\|x)$	Self-supervised learning	$p(x)$	Generative models, Autoencoders

Let's focus first on discriminative models, where X is of high dimension, i.e. images, spectra, time series, sequences …



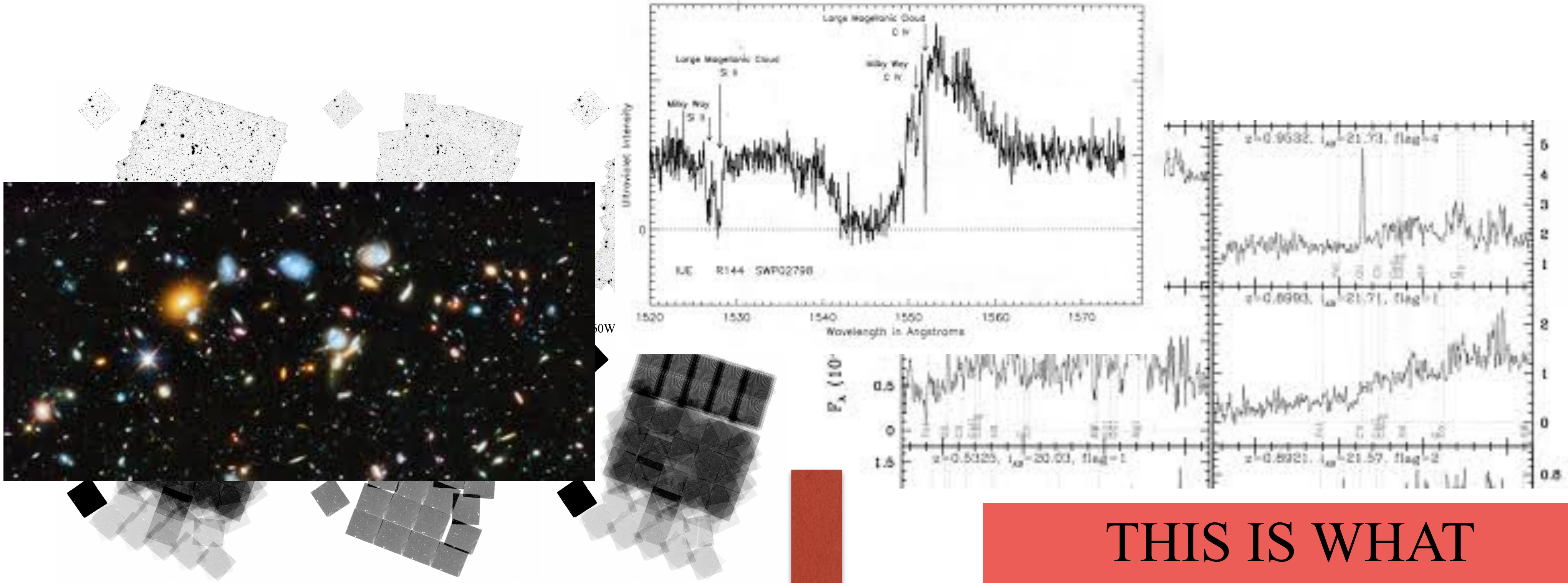
Approach Data type	Discriminative		Generative	
	Target Function	Method	Target Function	Method
Labels	$p(y\|x)$	All supervised networks with bottleneck	$p(x\|y)$	Conditional generative models
No Labels	$p(\hat{y}\|x)$	Self-supervised learning	$p(x)$	Generative models, Autoencoders

NEURAL NETWORKS FOR COMPUTER VISION

$$p(\theta|\boldsymbol{X})$$

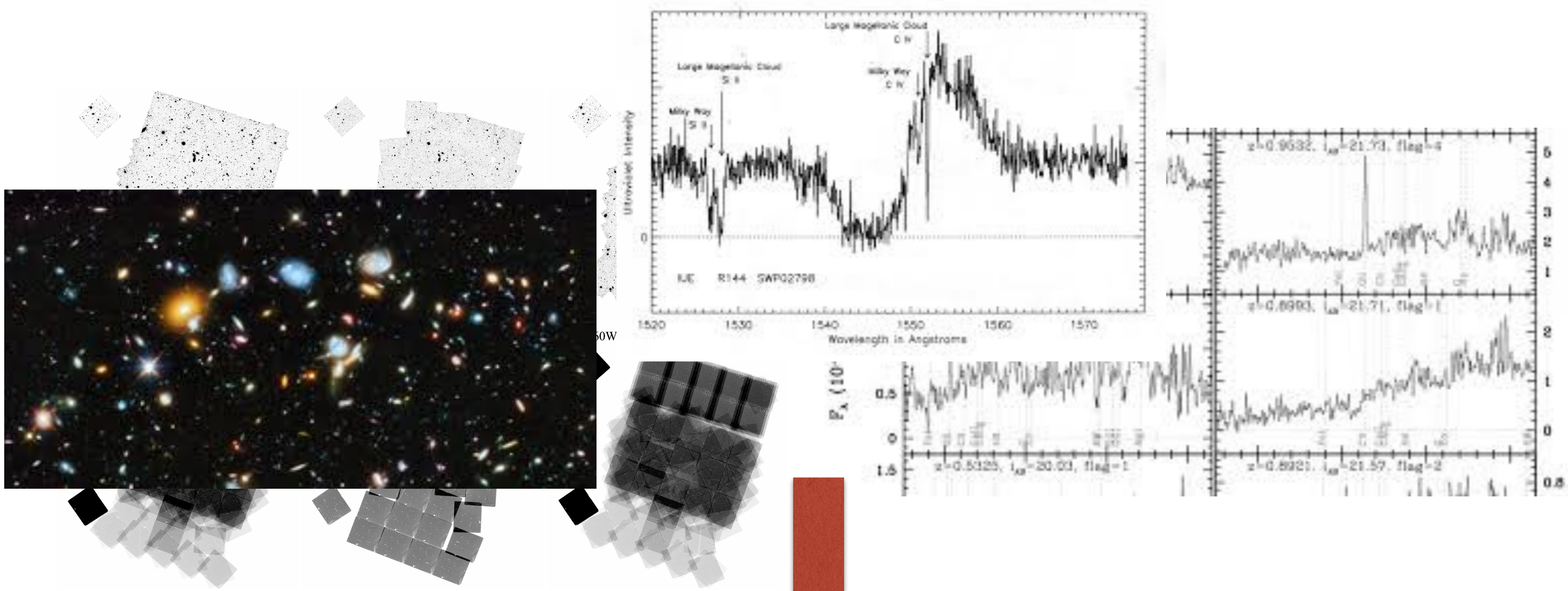
CAN WE GO DEEP NOW?

What do we put as input?



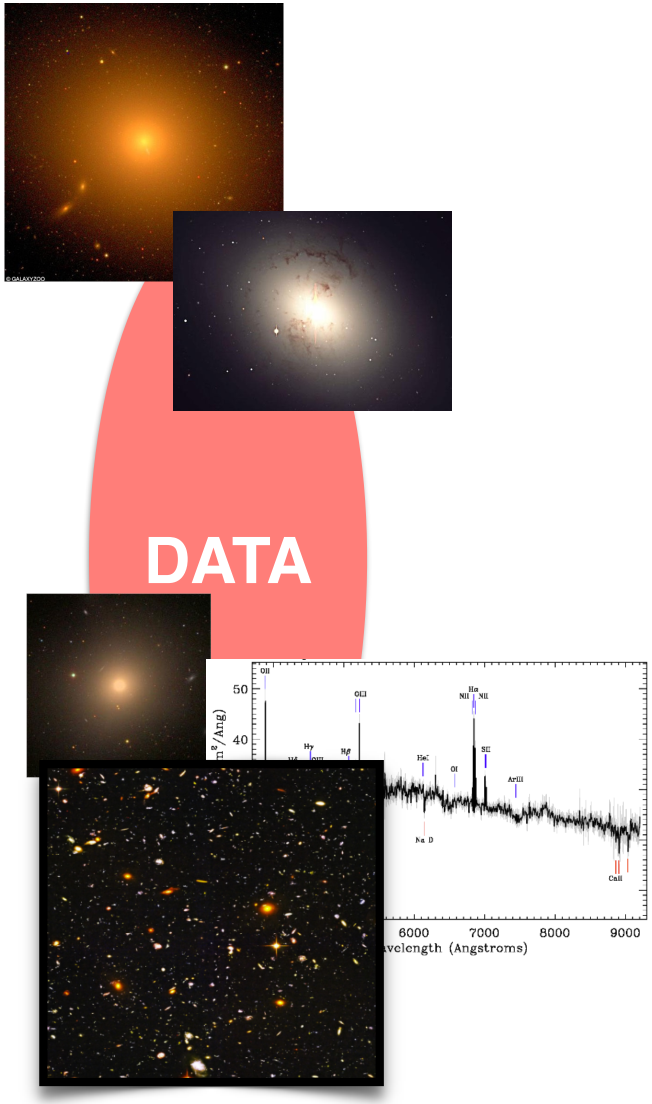
THIS IS WHAT MACHINES SEE

What do we put as input?



PRE-PROCESS DATA TO EXTRACT MEANINGFUL INFORMATION

THIS IS GENERALLY CALLED **FEATURE EXTRACTION**



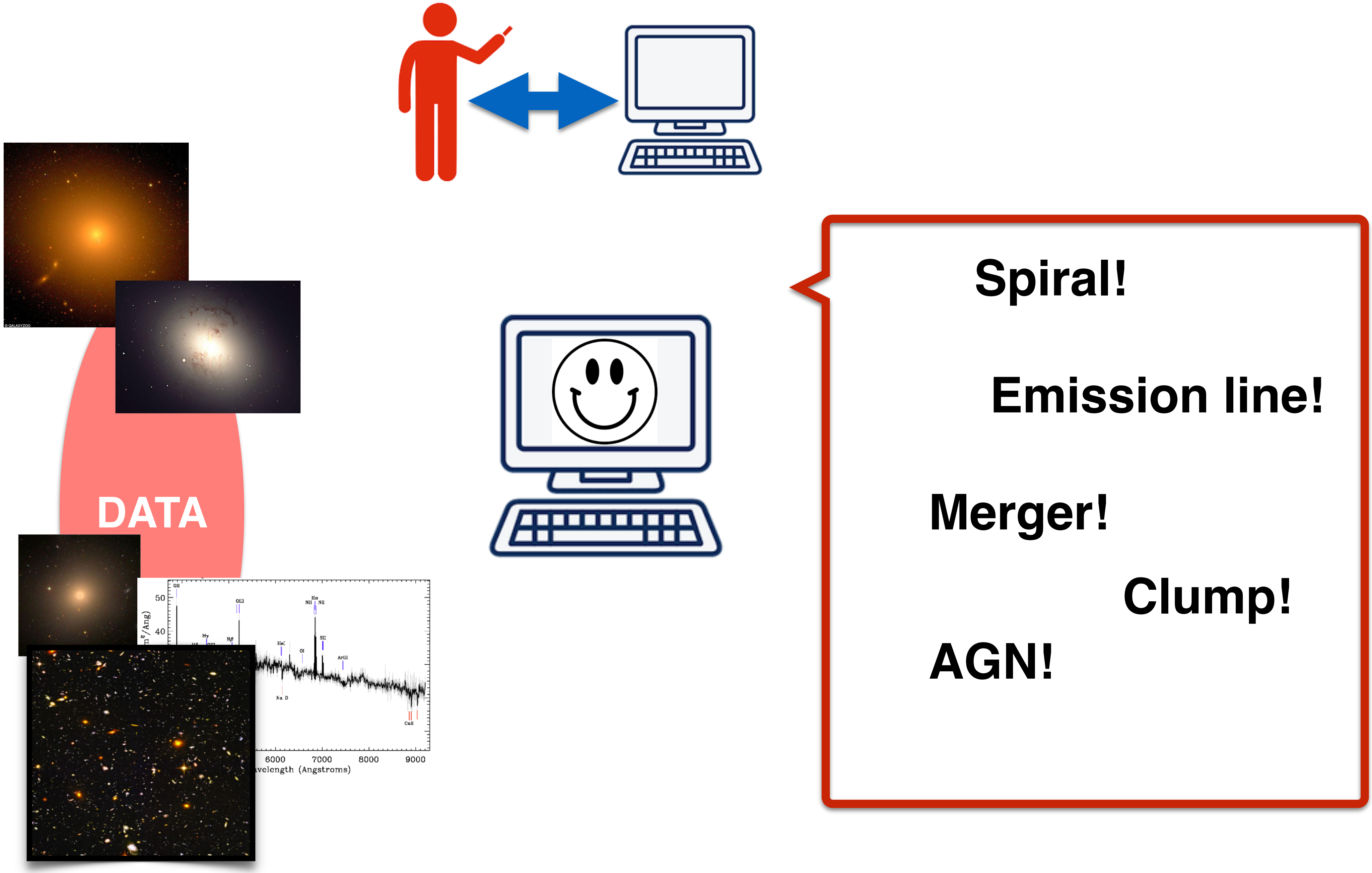
Spiral!

Emission line!

Merger!

Clump!

AGN!


DATA
Spiral!
Emission line!
Merger!
Clump!
AGN!

$$f_{\overrightarrow{W}}(\vec{x}) = \vec{y}$$

LABEL

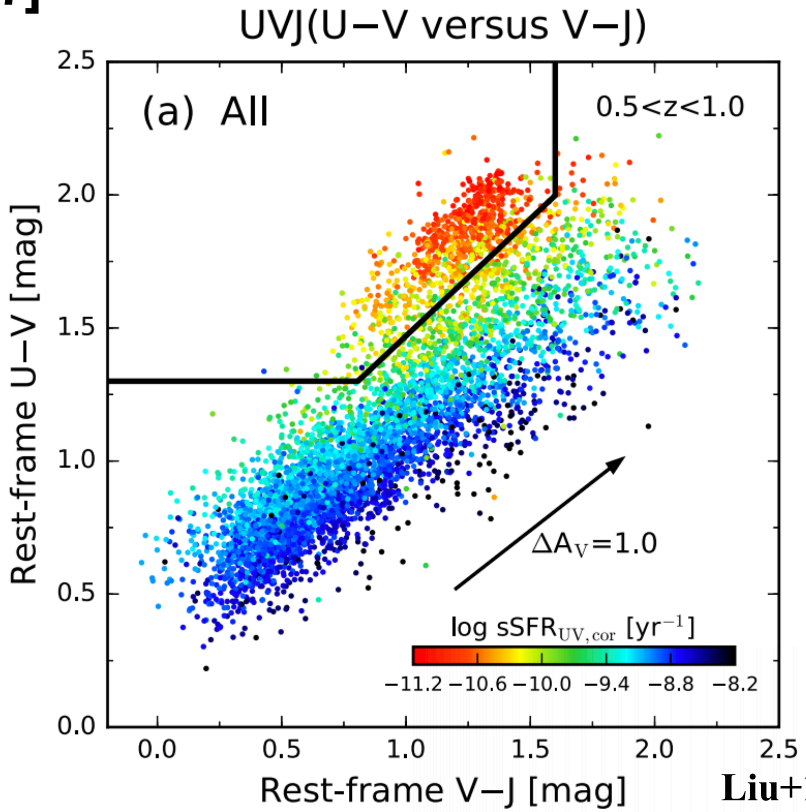
Q(0) , SF(1)

NETWORK FUNCTION

(U-V, V-J) FEATURES

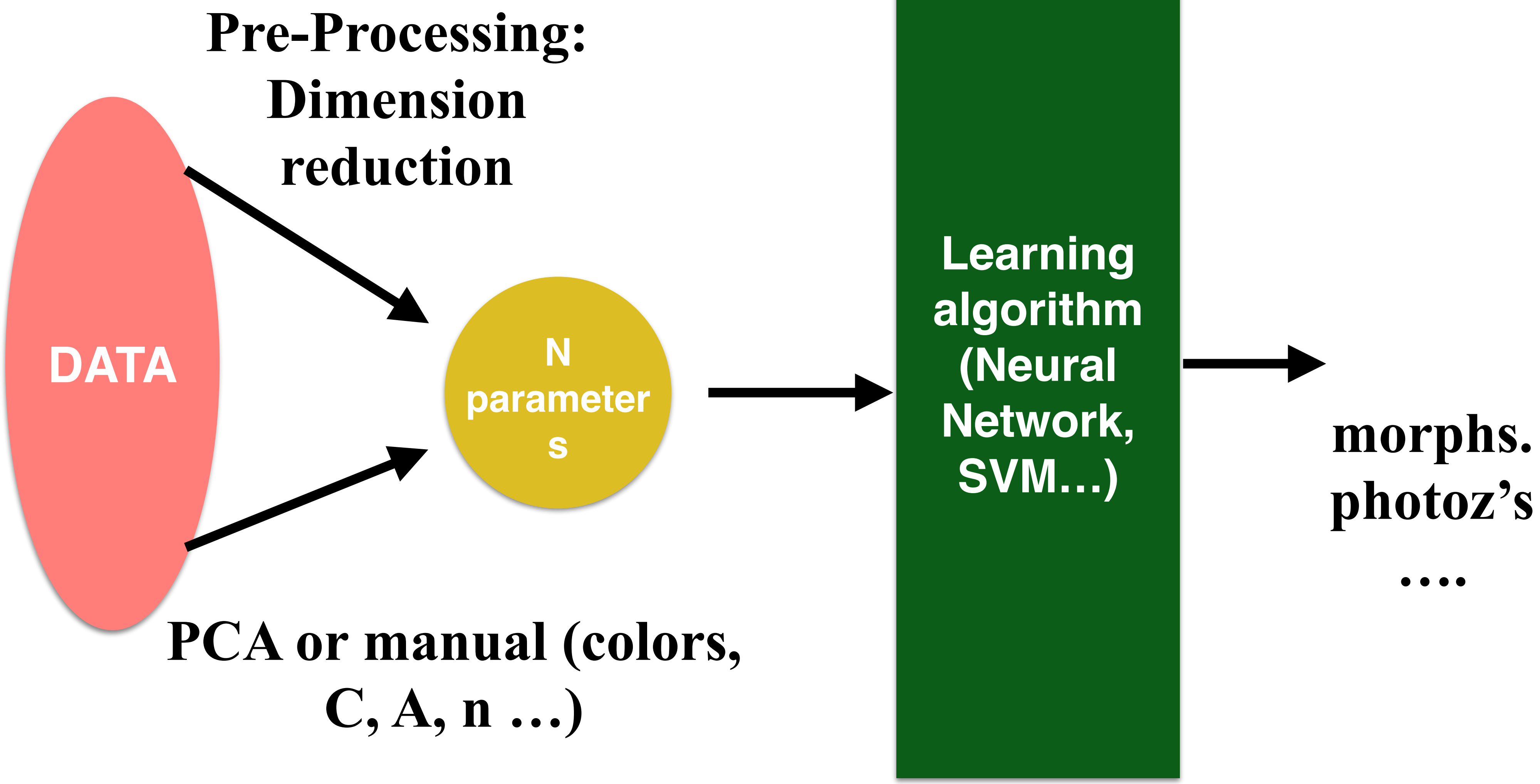
sgn[(u-v)-0.8*(v-j)-0.7]

WEIGHTS

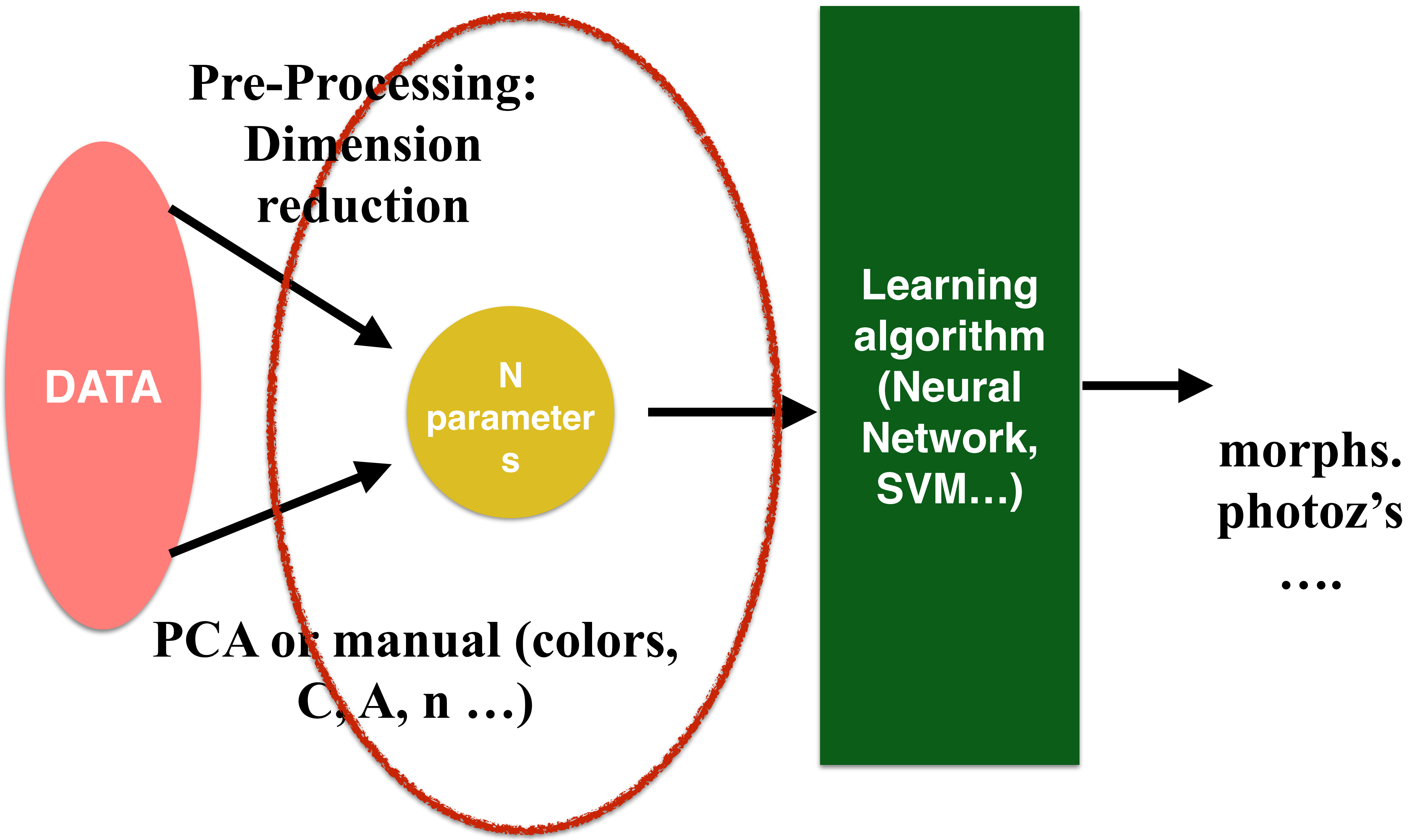


Liu+18

THE “CLASSICAL” APPROACH



"CLASSICAL" MACHINE LEARNING



In Astronomy

- Colors, Fluxes
- Shape indicators
- Line ratios, spectral features
- Stellar Masses, Velocity Dispersions

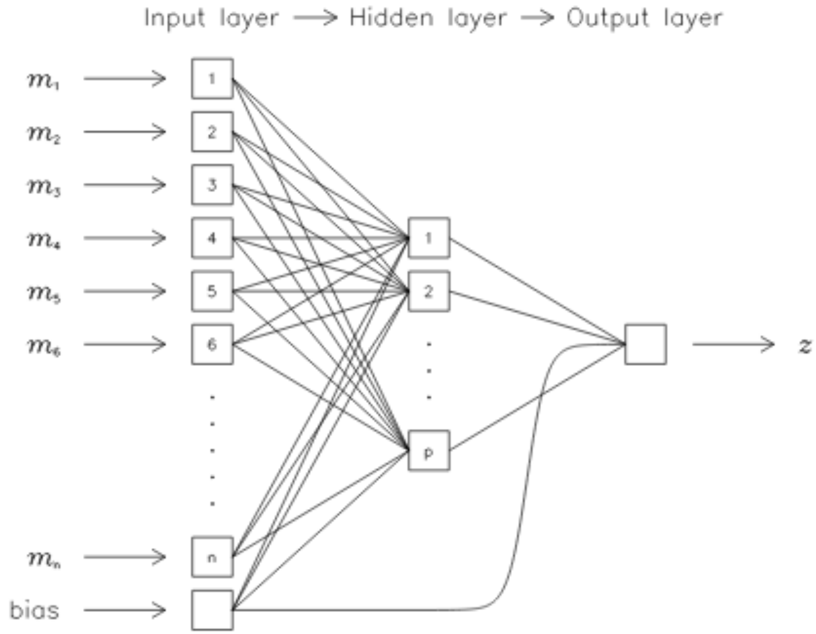
Requires specialized software before feeding the machine learning algorithm

IT IMPLIES A DIMENSIONALITY REDUCTION!

PHOTOMETRIC REDSHIFTS

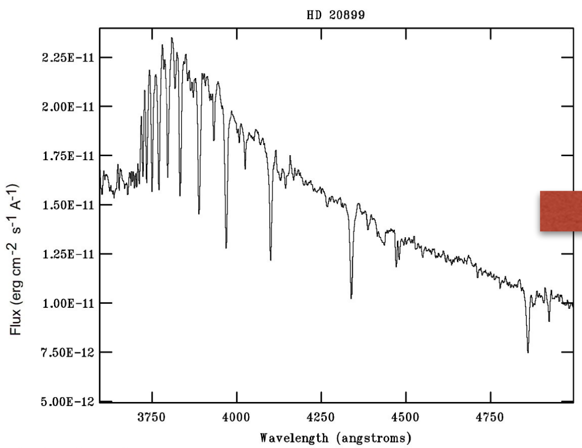
SDSS

g
r
i
z

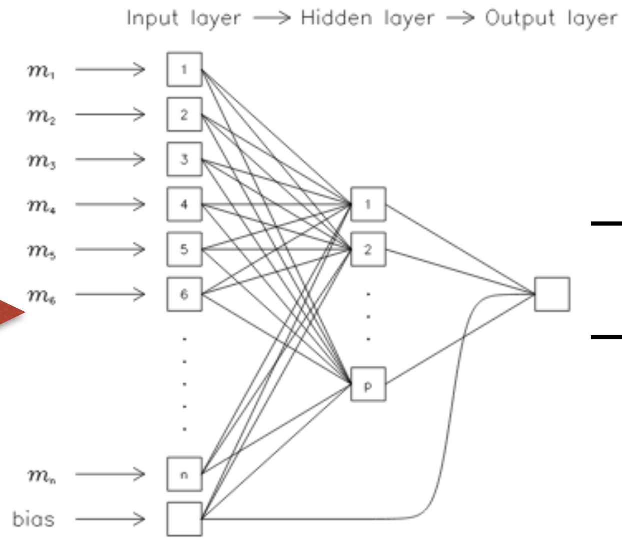


Collister+08

STELLAR PARAMETERS FROM MEDIUM BAND FILTERS



MEDIUM BAND FLUXES



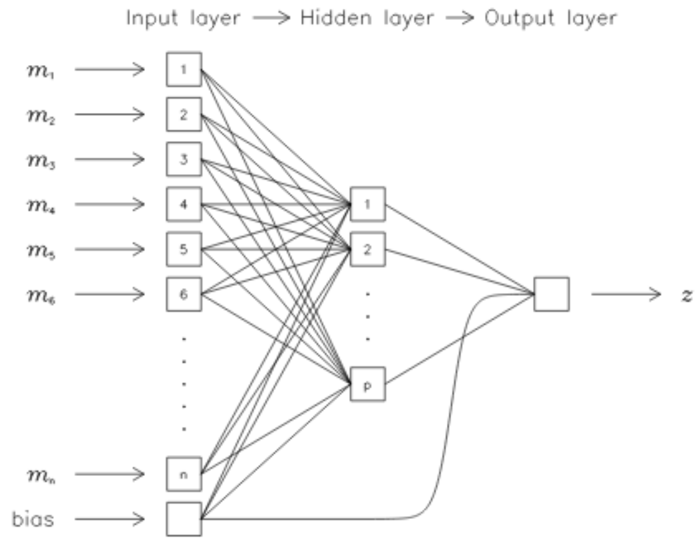
→ T

→ [M/H]

Bailer-Jones+00

No.	Symbol	Description	Scale[a]
1	CVD	Central velocity dispersion	~1 kpc
2	$M_{\rm bulge}$	Bulge stellar mass	0.5–4 kpc
3	Re	Bulge effective radius	0.5–4 Kpc
4	B/T	Bulge-to-total stellar mass ratio	0.5–8 kpc
5	M_*	Total stellar mass	2–8 kpc
6	$M_{\rm disc}$	Disc stellar mass	4–10 kpc
7	$M_{\rm halo}$	Group halo mass	0.1–1 Mpc
8	δ_5	Local density parameter	0.5–3 Mpc

Notes. [a] Approximate 1σ range from centre of galaxy. For photometric quantities half-light radii are used.



<u>**HEAVILY PROCESSED DATA**</u>

Teimiroonia+16

Other general computer vision features [for images!]

- Pixel Concatenation
- Color histograms
- Texture Features
- Histogram of Gradients
- SIFT

FOR MANY YEARS COMPUTER VISION
RESEARCHERS HAVE BEEN TRYING TO FIND THE MOST GENERAL FEATURES

Other general computer vision features [for images!]

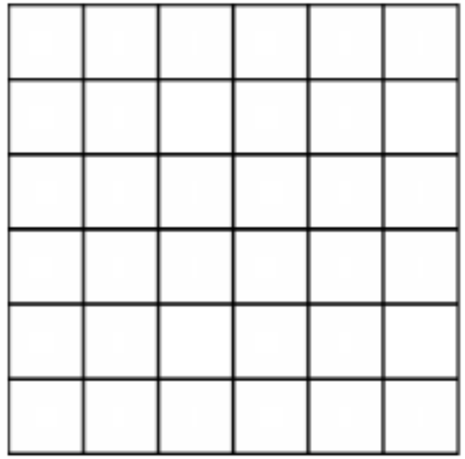
- Pixel Concatenation
- Color histograms
- Texture Features
- Histogram of Gradients
- SIFT

FOR MANY YEARS COMPUTER VISION
RESEARCHERS HAVE BEEN TRYING TO FIND THE MOST GENERAL FEATURES

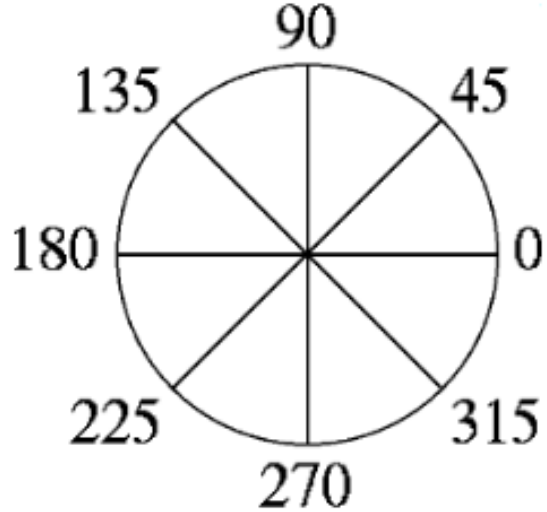
THE BEST CLASSICAL SOLUTION [BEFORE 2012] WHERE BASED **ON LOCAL FEATURES**

HISTOGRAM OF ORIENTED GRADIENTS (HoG)

1. DIVIDE IMAGE INTO SMALL SPATIAL REGIONS CALLED CELLS

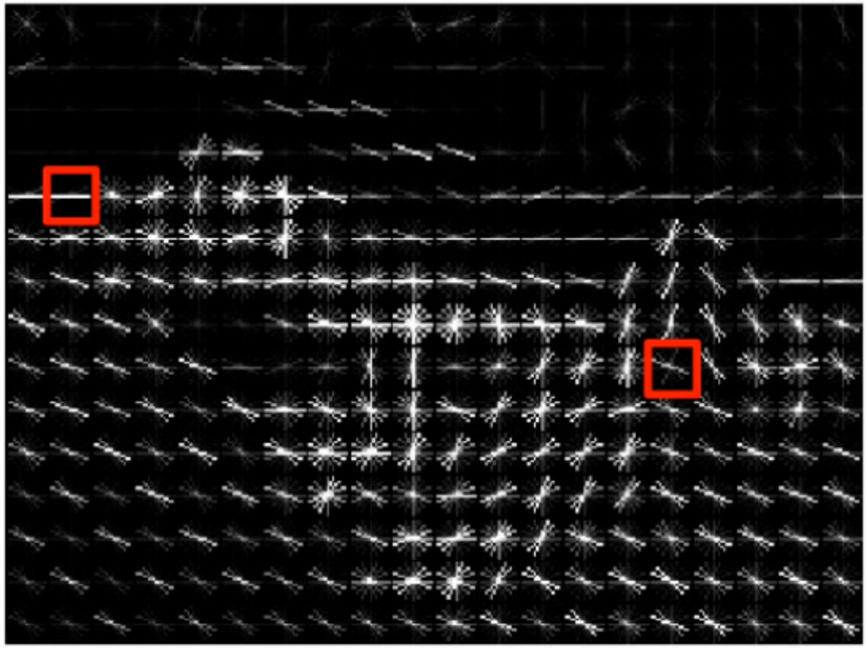


2. COMPUTE INTENSITY GRADIENTS OVER N DIRECTIONS [TYPICALLY 9 FOR IMAGE]



3. COMPUTE WEIGHTED 1-D HISTOGRAM OF ALL DIRECTIONS. A CELL IS REDUCED TO N NUMBERS

HISTOGRAM OF ORIENTED GRADIENTS (HoG)



EVERYTHING IS IN THE FEATURES….WHAT IF I IGNORED SOME IMPORTANT FEATURES?

EVERYTHING IS IN THE FEATURES….WHAT IF I IGNORED SOME IMPORTANT FEATURES?

WHAT ABOUT USING RAW DATA?

ALL INFORMATION IS IN THE INPUT DATA

WHY REDUCING ?

LET THE NETWORK FIND THE INFO

WHAT ABOUT USING RAW DATA?

ALL INFORMATION IS IN THE INPUT DATA

WHY REDUCING ?

LET THE NETWORK FIND THE INFO

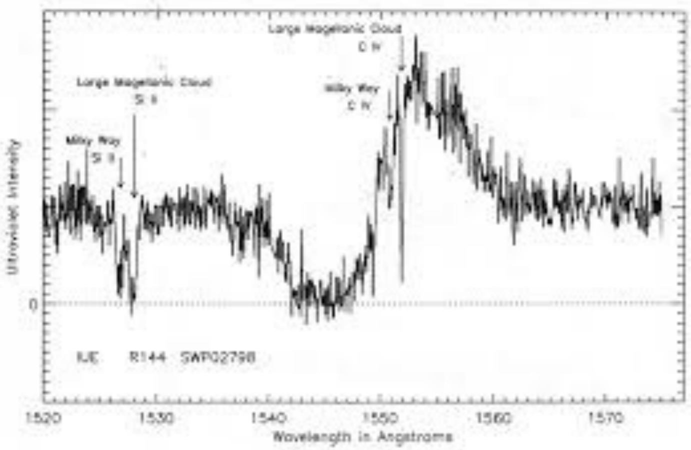
LARGE DIMENSION SIGNALS SUCH AS IMAGES OR SPECTRA WOULD REQUIRE TREMENDOUSLY LARGE MODELS

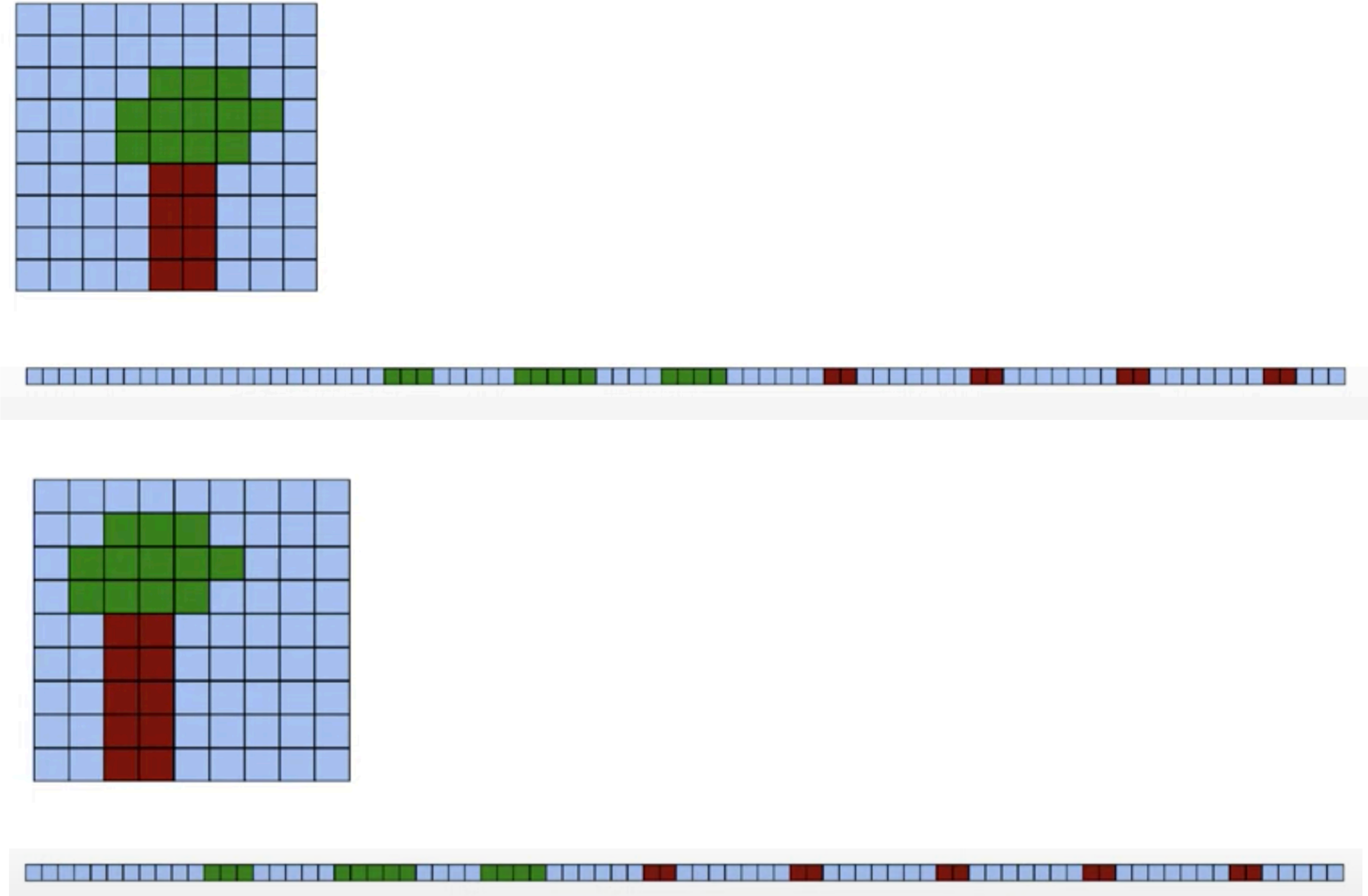
A 512x512 image as input of a fully connected layer producing output of same size:

$$(512 \times 512)^2 = 7e10$$

<u>BUT</u>

FEEDING INDIVIDUAL RESOLUTION ELEMENTS IS NOT VERY EFFICIENT SINCE IT LOOSES ALL INVARIANCE TO TRANSLATION AND <u>IGNORES CORRELATION IN THE DATA AT ALL SCALES</u>

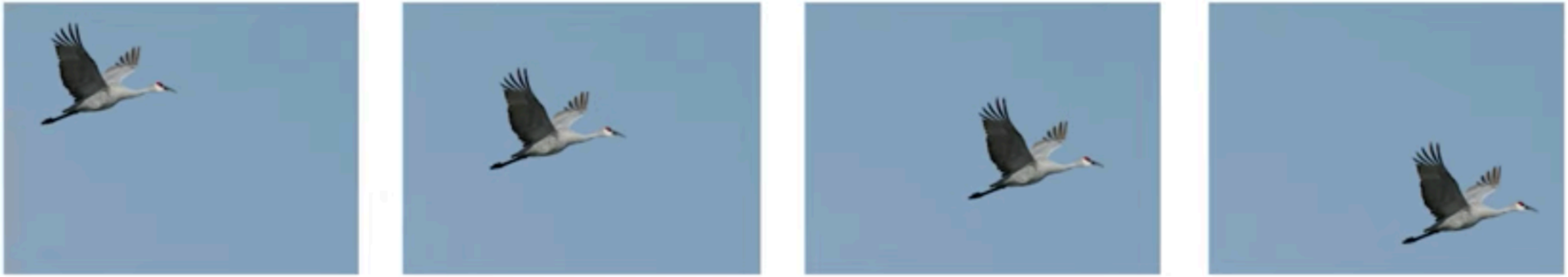




(Dielemann@Deepmind)

FEEDING INDIVIDUAL RESOLUTION ELEMENTS IS NOT VERY EFFICIENT SINCE IT LOOSES ALL INVARIANCE TO TRANSLATION

SO?



TWO BASIC PROPERTIES OF IMAGING DATA (BUT ALSO SPECTROSCOPY IN SOME SENSE) ARE **LOCALITY TRANSLATION INVARIANCE**

locality: nearby pixels are more strongly correlated

translational invariance: meaningful patterns can appear anywhere in the image

(Dielemann@Deepmind)

CONVOLUTIONAL NEURAL NETWORKS

Discrete Convolution

1D:
[Spectra]

$$f(x) * g(x) = \sum_{k=-\infty}^{k=+\infty} f(k).g(k-x)$$

2D:
[Images]

$$f(x,y) * g(x,y) = \sum_{k=-\infty}^{k=+\infty} \sum_{l=-\infty}^{l=+\infty} f(k,l).g(x-k, y-l)$$

DISCRETE CONVOLUTION

1D:
[Spectra]

$$f(x) * g(x) = \sum_{k=-\infty}^{k=+\infty} f(k).g(k-x)$$

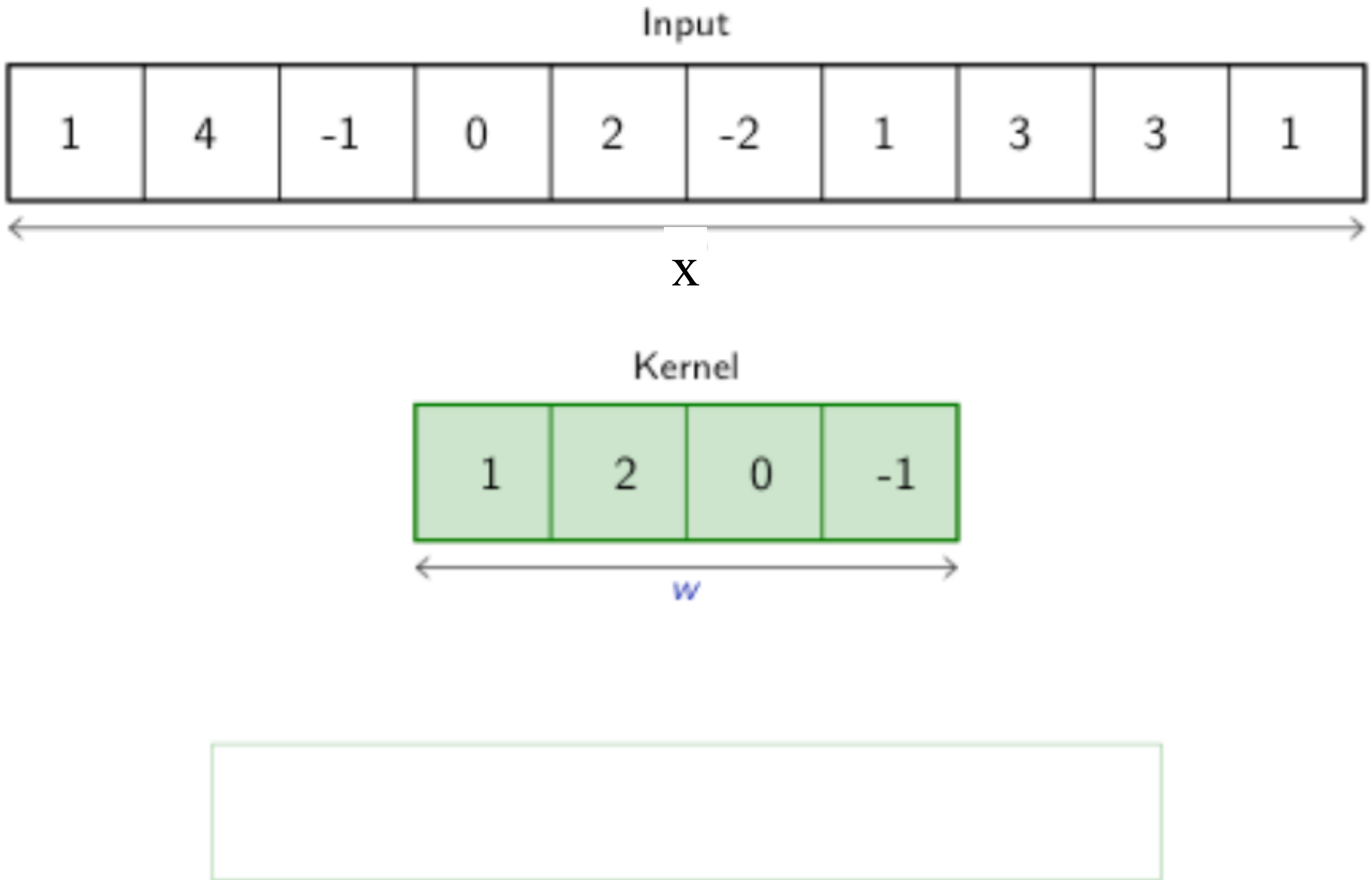
2D:
[Images]

$$f(x,y) * g(x,y) = \sum_{k=-\infty}^{k=+\infty} \sum_{l=-\infty}^{l=+\infty} f(k,l).g(x-k,y-l)$$

CONVOLUTION KERNEL

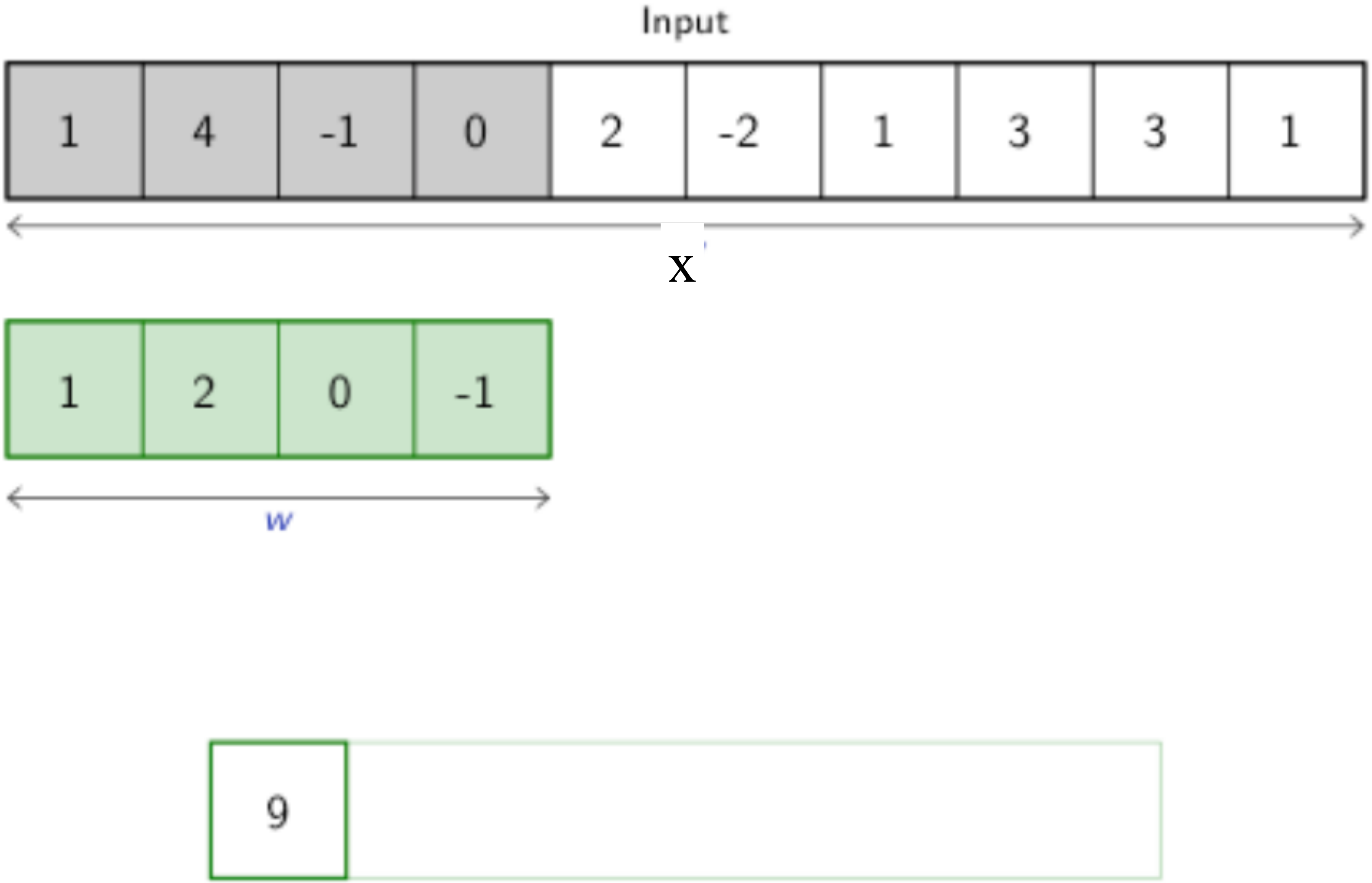
INPUT DATA

1-D CONVOLUTION



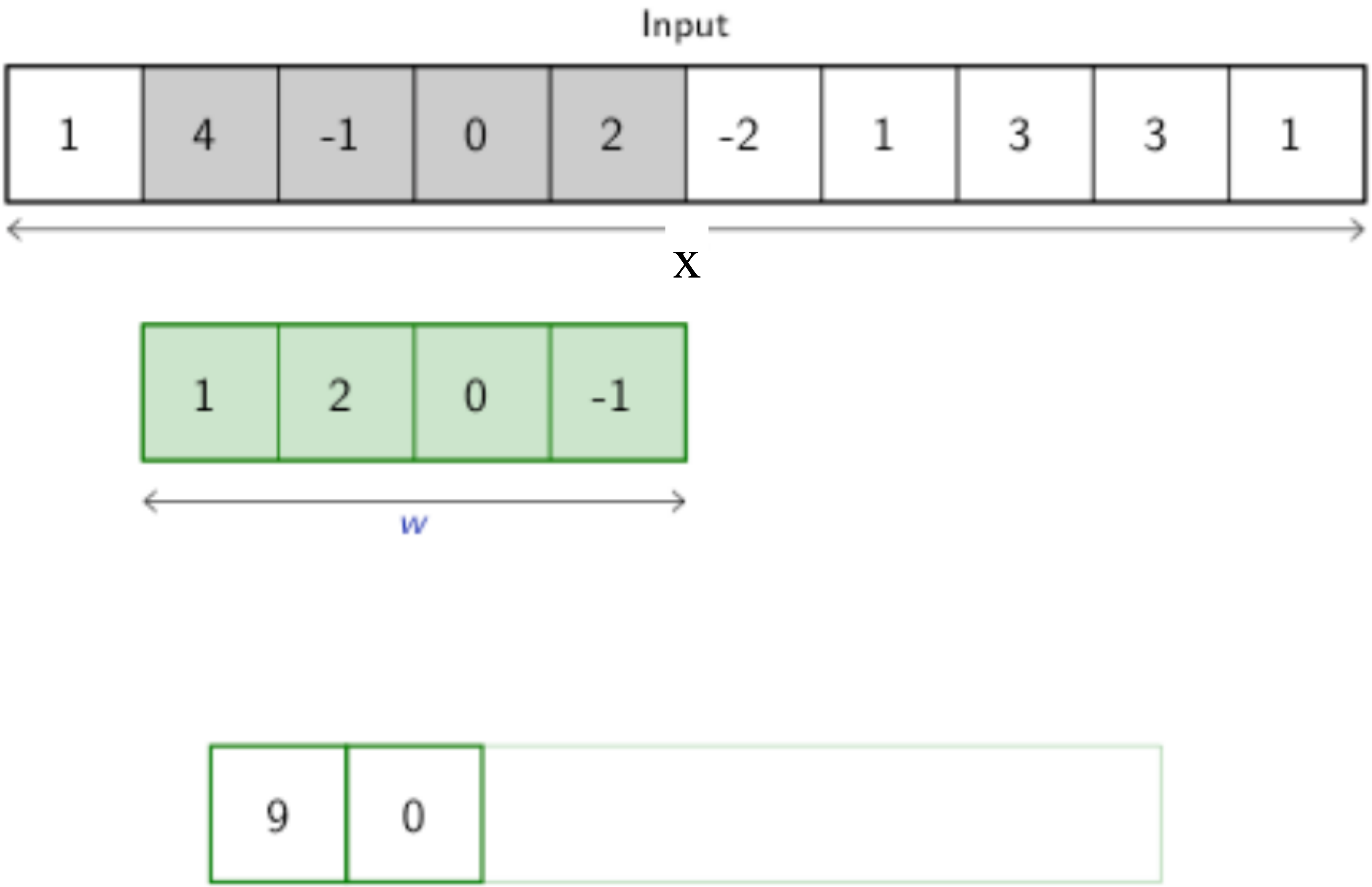
credit

1-D CONVOLUTION



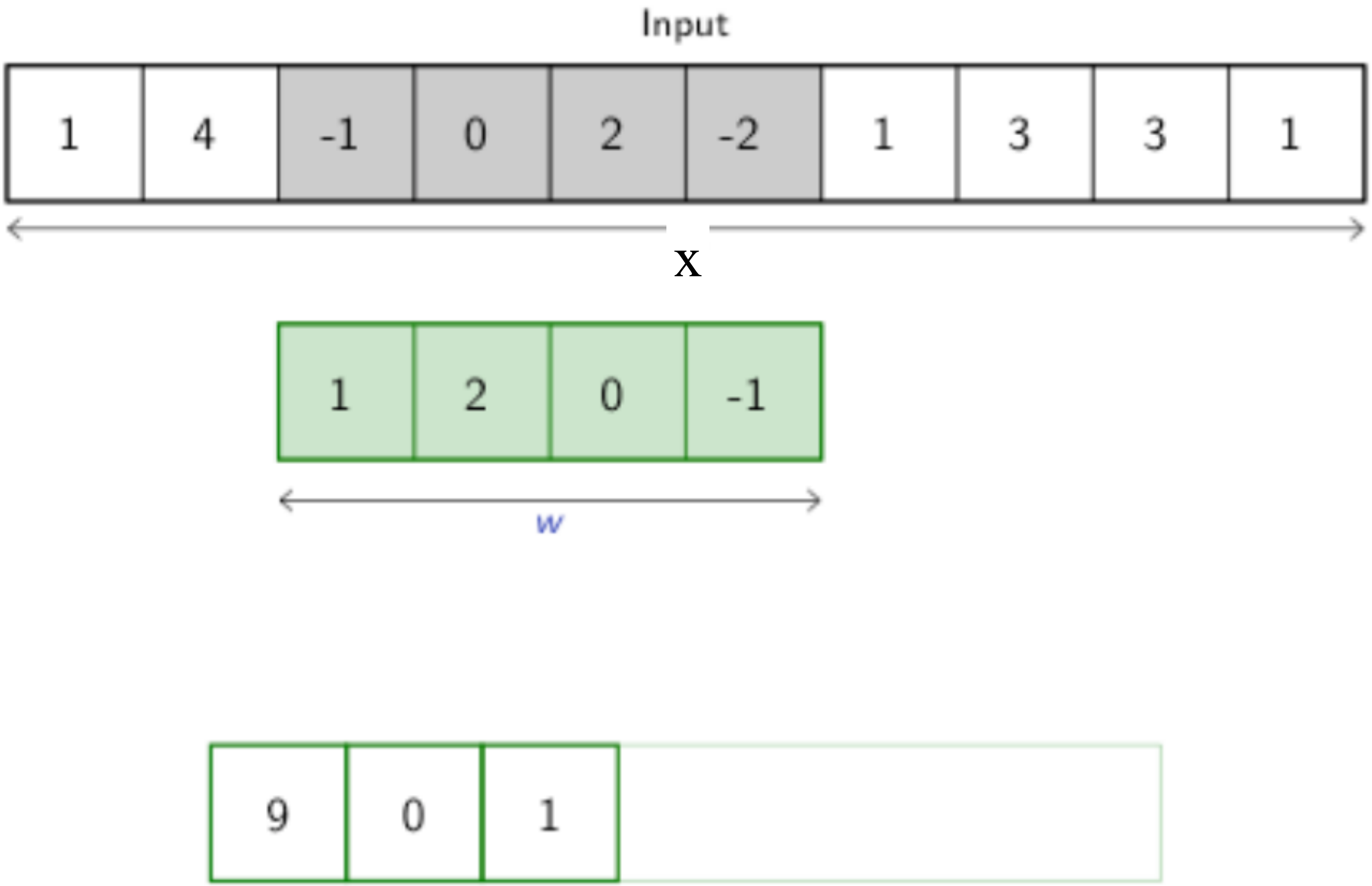
credit

1-D CONVOLUTION



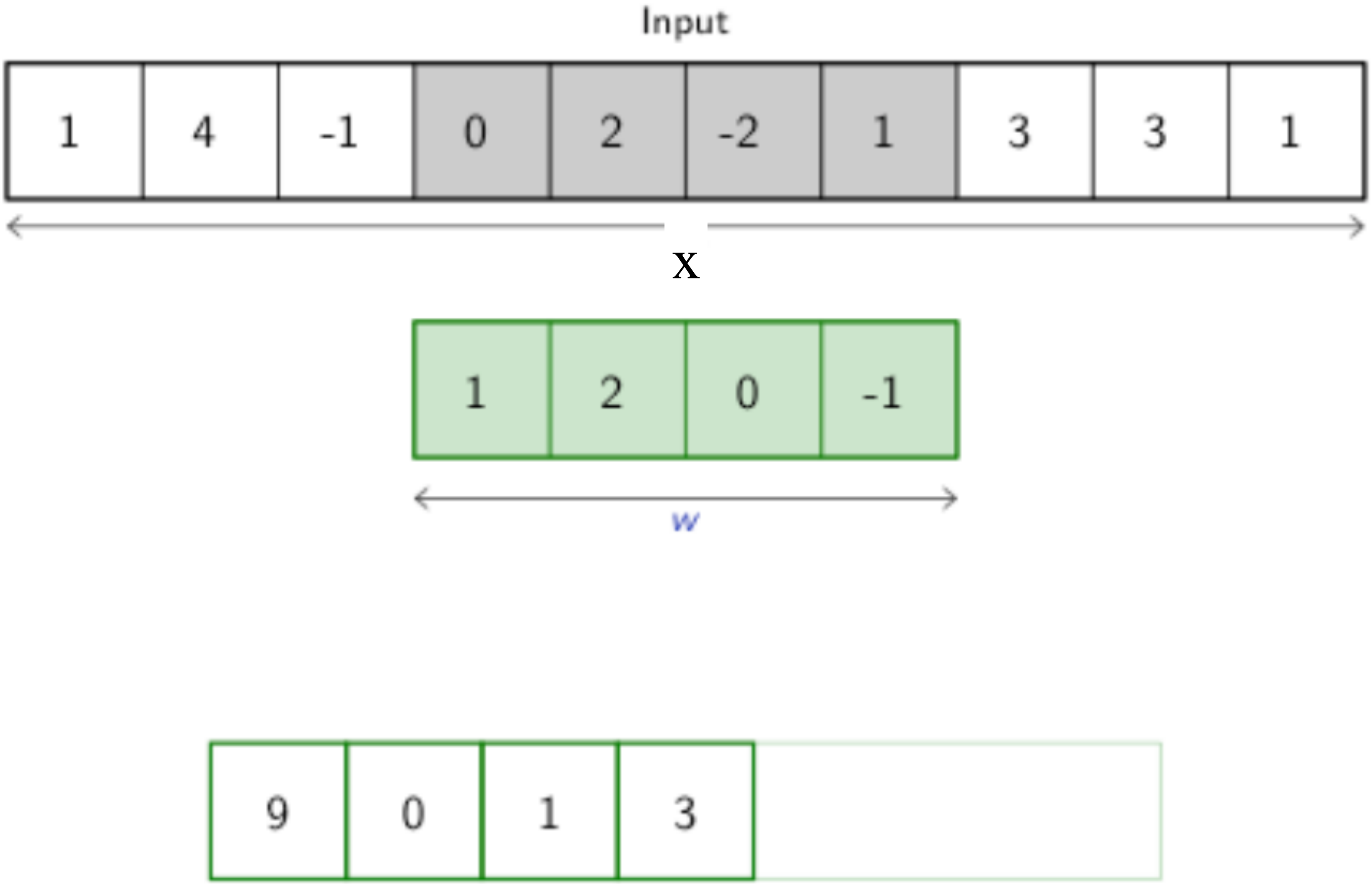
credit

1-D CONVOLUTION



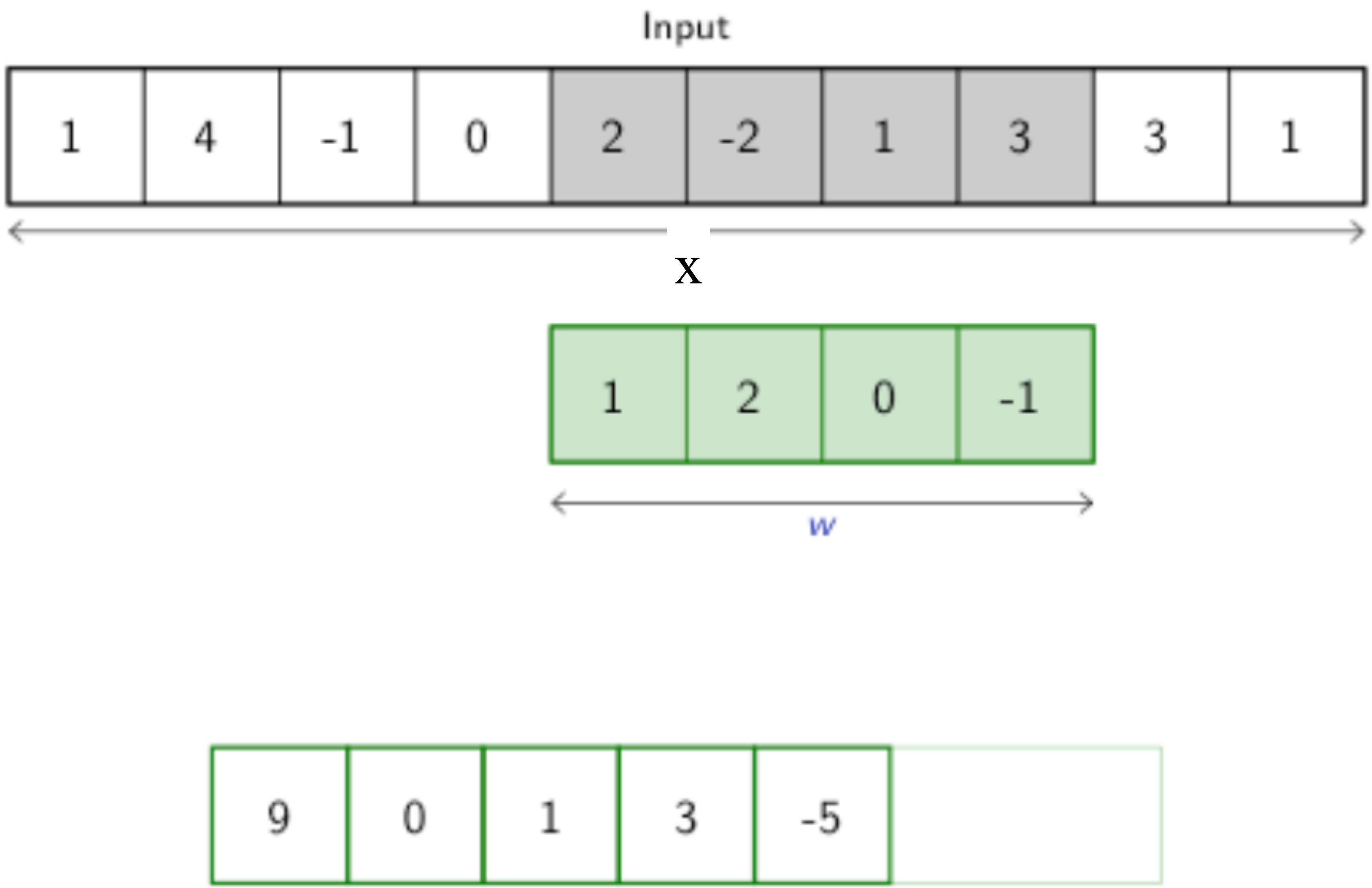
credit

1-D CONVOLUTION



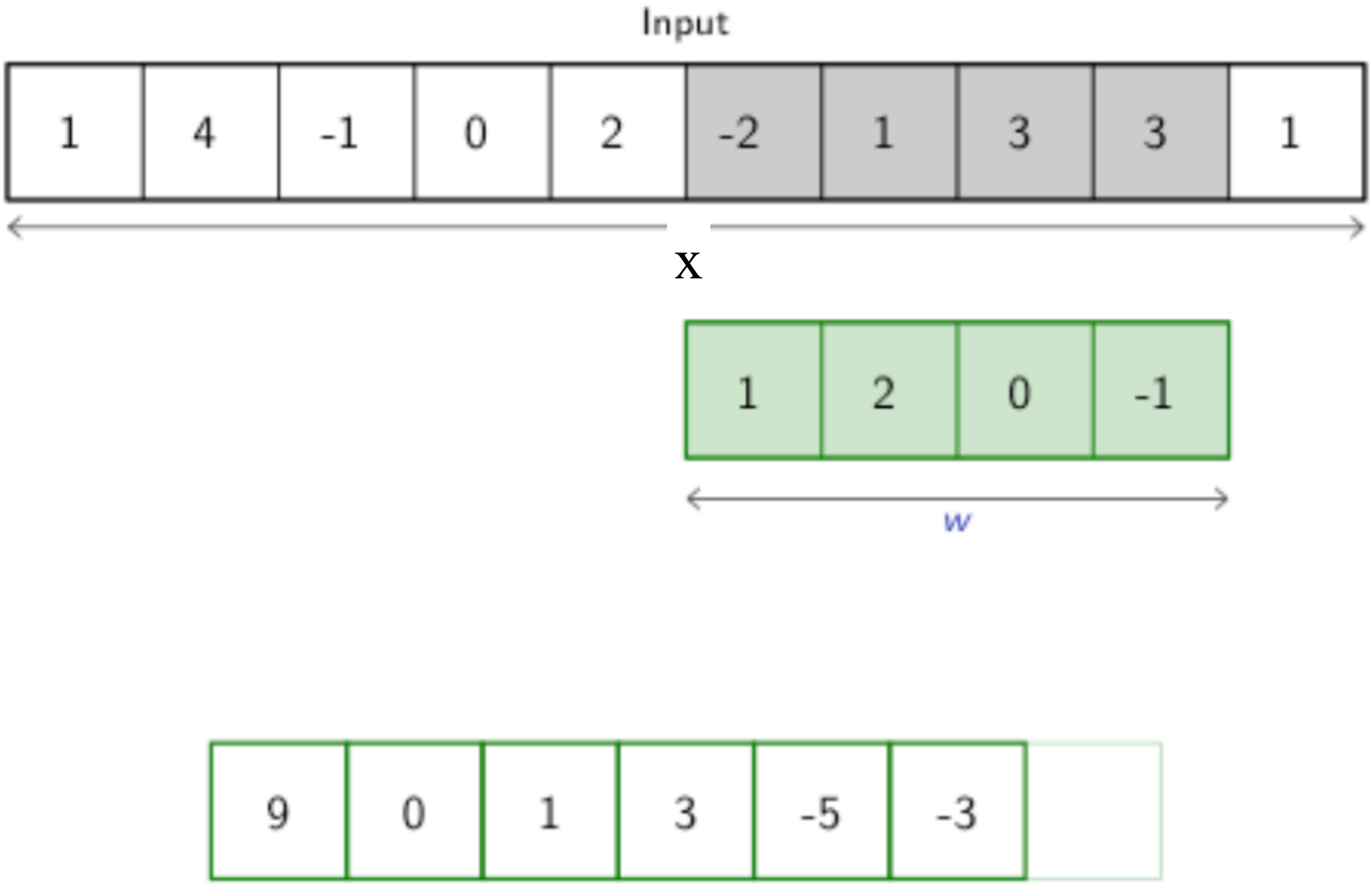
credit

1-D CONVOLUTION



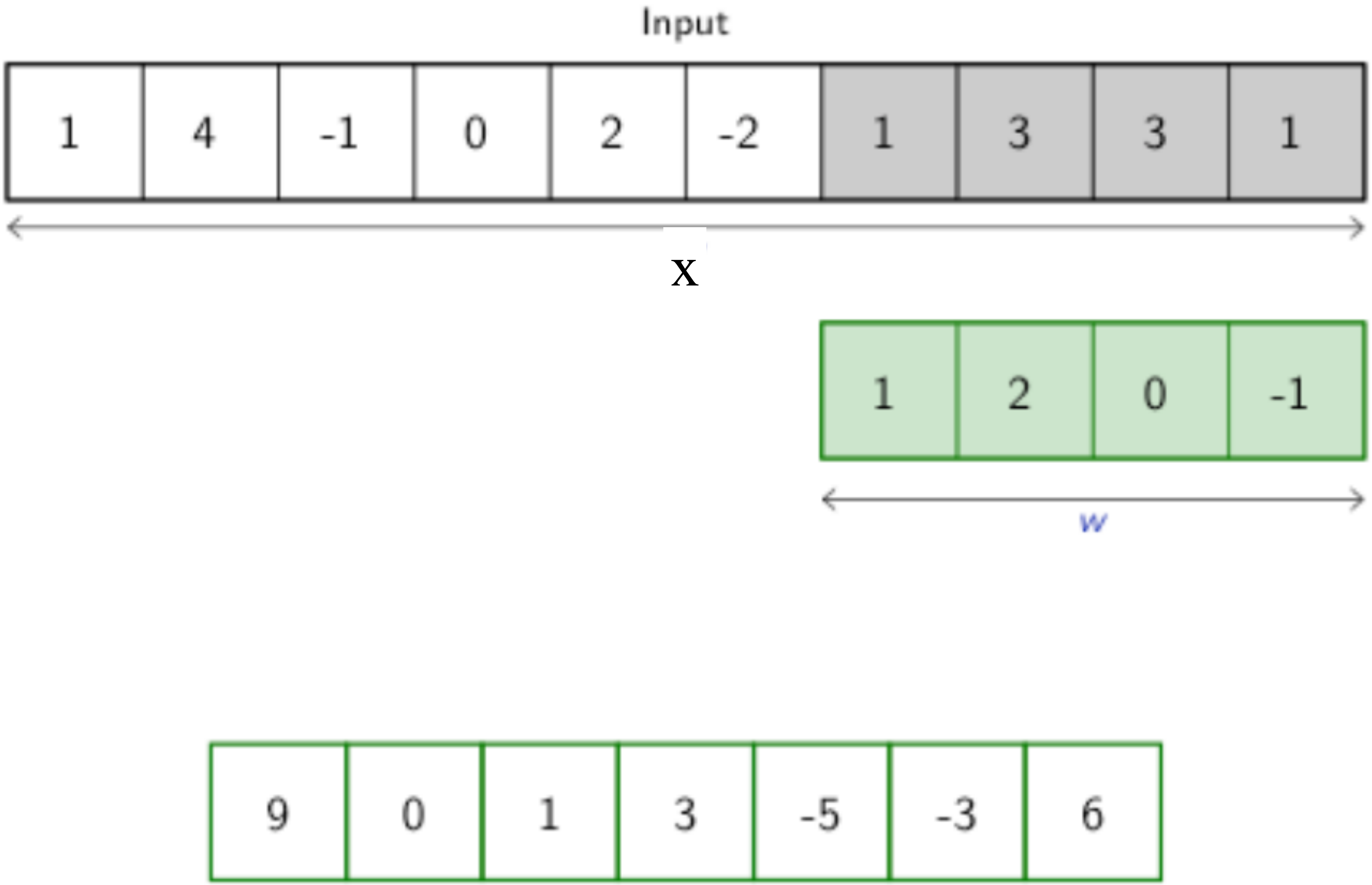
credit

1-D CONVOLUTION



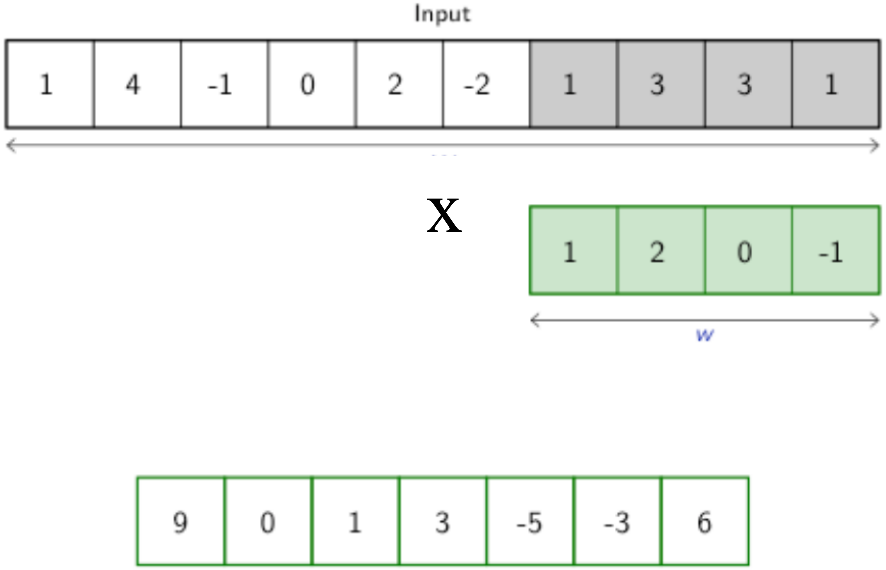
credit

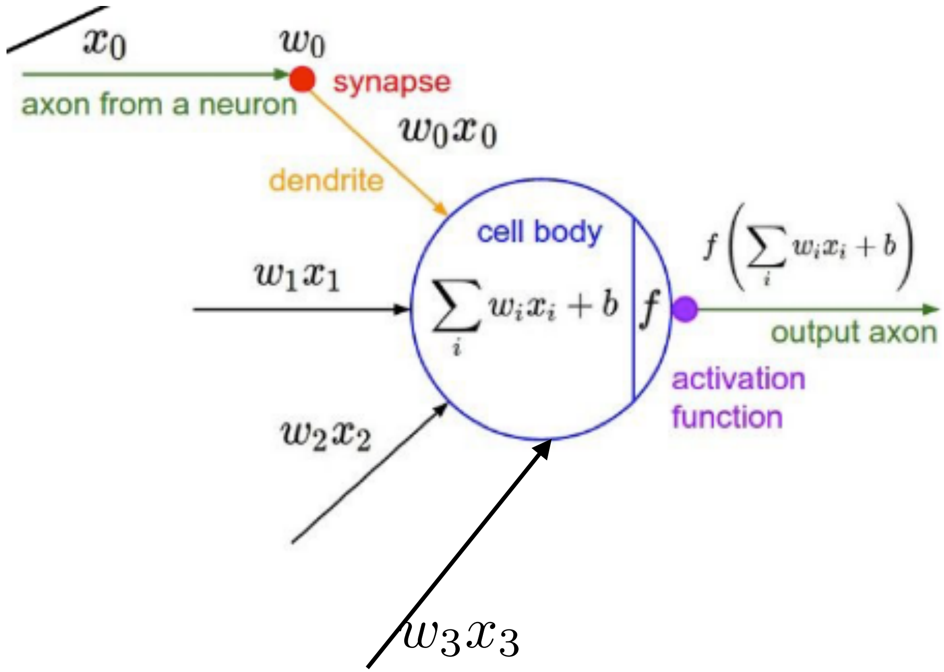
1-D CONVOLUTION



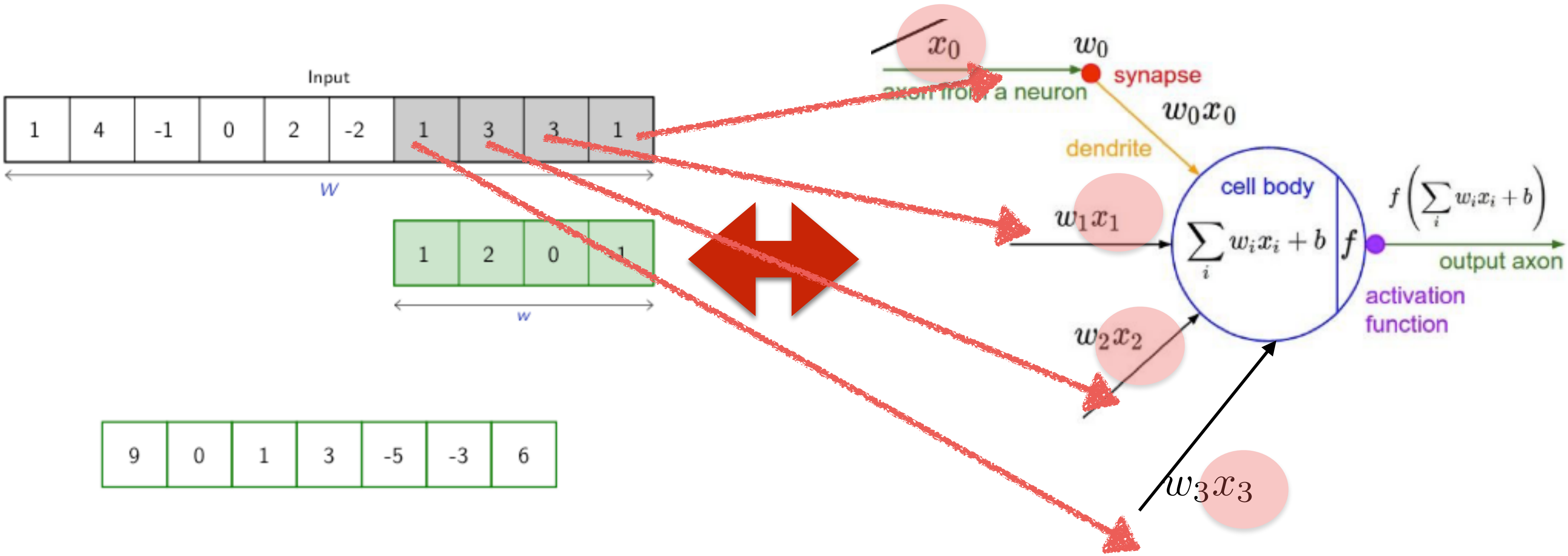
credit

THE CONVOLUTION BUILDING BLOCK OPERATION (BEFORE ACTIVATION) IS EQUIVALENT TO A NEURON WITH AS MANY INPUTS AS KERNEL ELEMENTS AND WEIGHTS EQUAL TO THE KERNEL

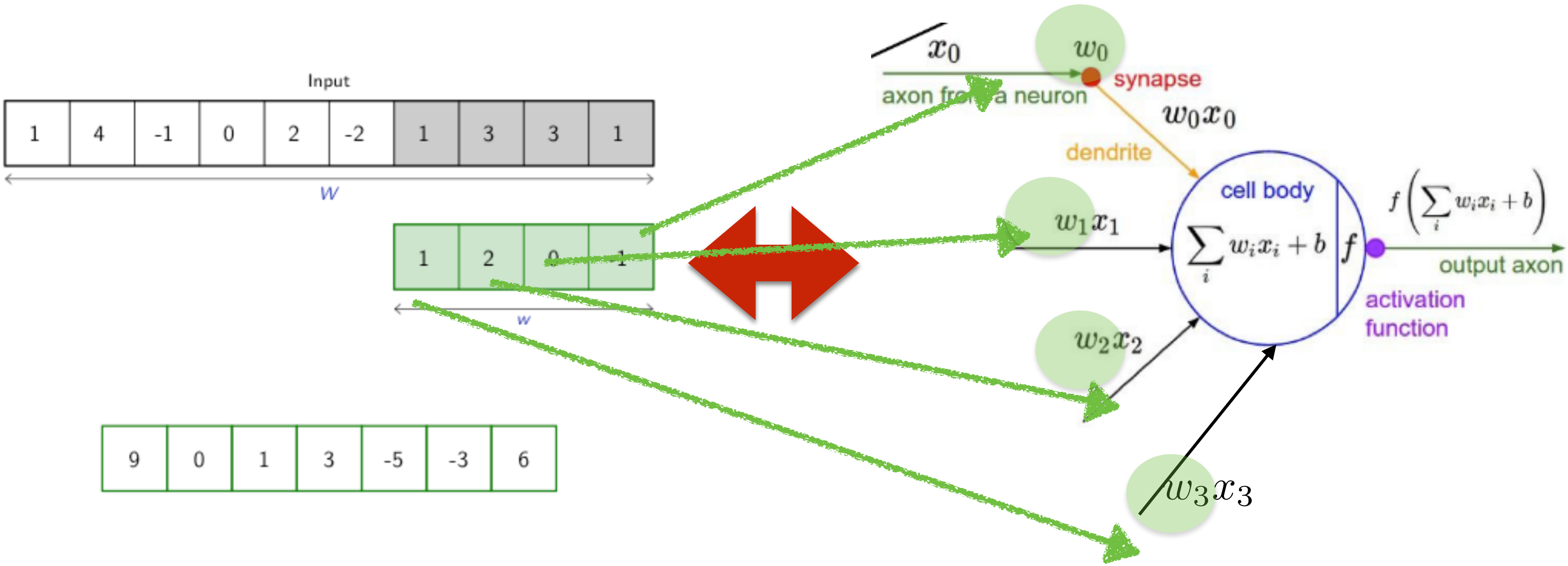




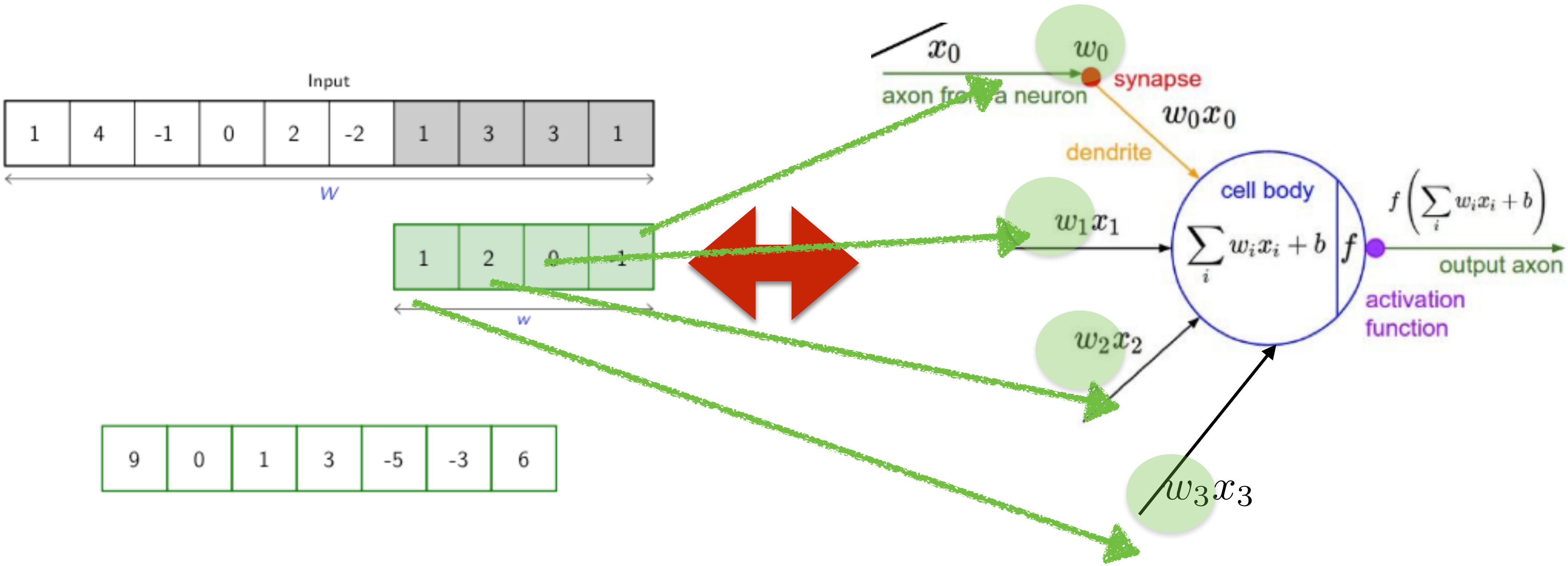
THE CONVOLUTION BUILDING BLOCK OPERATION (BEFORE ACTIVATION) IS EQUIVALENT TO A NEURON WITH AS MANY INPUTS AS KERNEL ELEMENTS AND WEIGHTS EQUAL TO THE KERNEL



THE CONVOLUTION BUILDING BLOCK OPERATION (BEFORE ACTIVATION) IS EQUIVALENT TO A NEURON WITH AS MANY INPUTS AS KERNEL ELEMENTS AND WEIGHTS EQUAL TO THE KERNEL



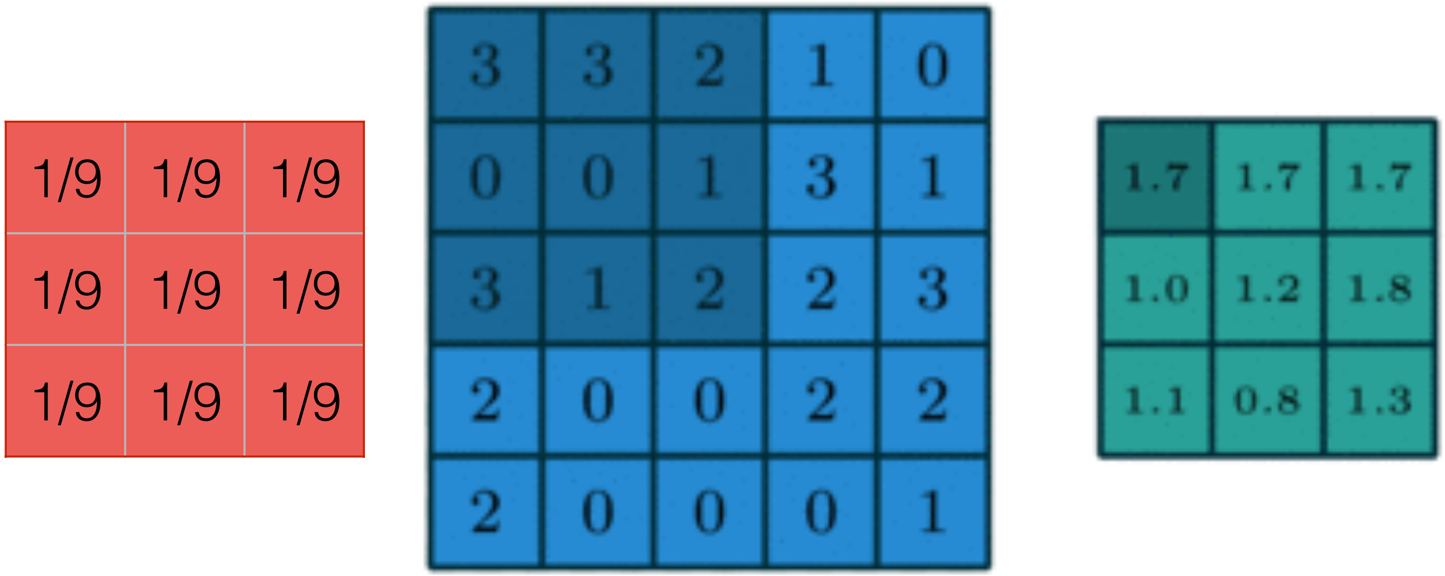
THE CONVOLUTION BUILDING BLOCK OPERATION (BEFORE ACTIVATION) IS EQUIVALENT TO A NEURON WITH AS MANY INPUTS AS KERNEL ELEMENTS AND WEIGHTS EQUAL TO THE KERNEL



WITH THE ADVANTAGE THAT THE SAME WEIGHTS ARE APPLIED TO ALL THE SIGNAL: <u>TRANSLATION INVARIANCE</u>

2-D CONVOLUTION

SAME IDEA, BUT THE KERNEL IS NOW 2D

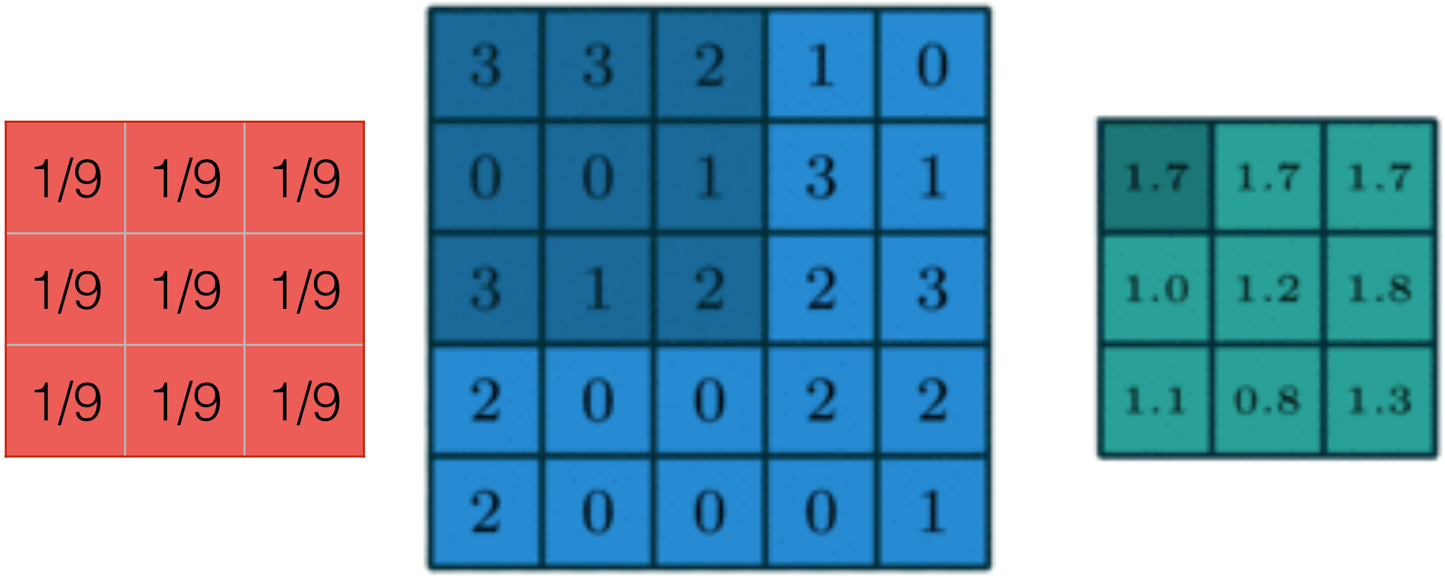


KERNEL **INPUT (IMAGE)** **OUTPUT**

Credit: animations from https://github.com/vdumoulin/conv_arithmetic

2-D CONVOLUTION

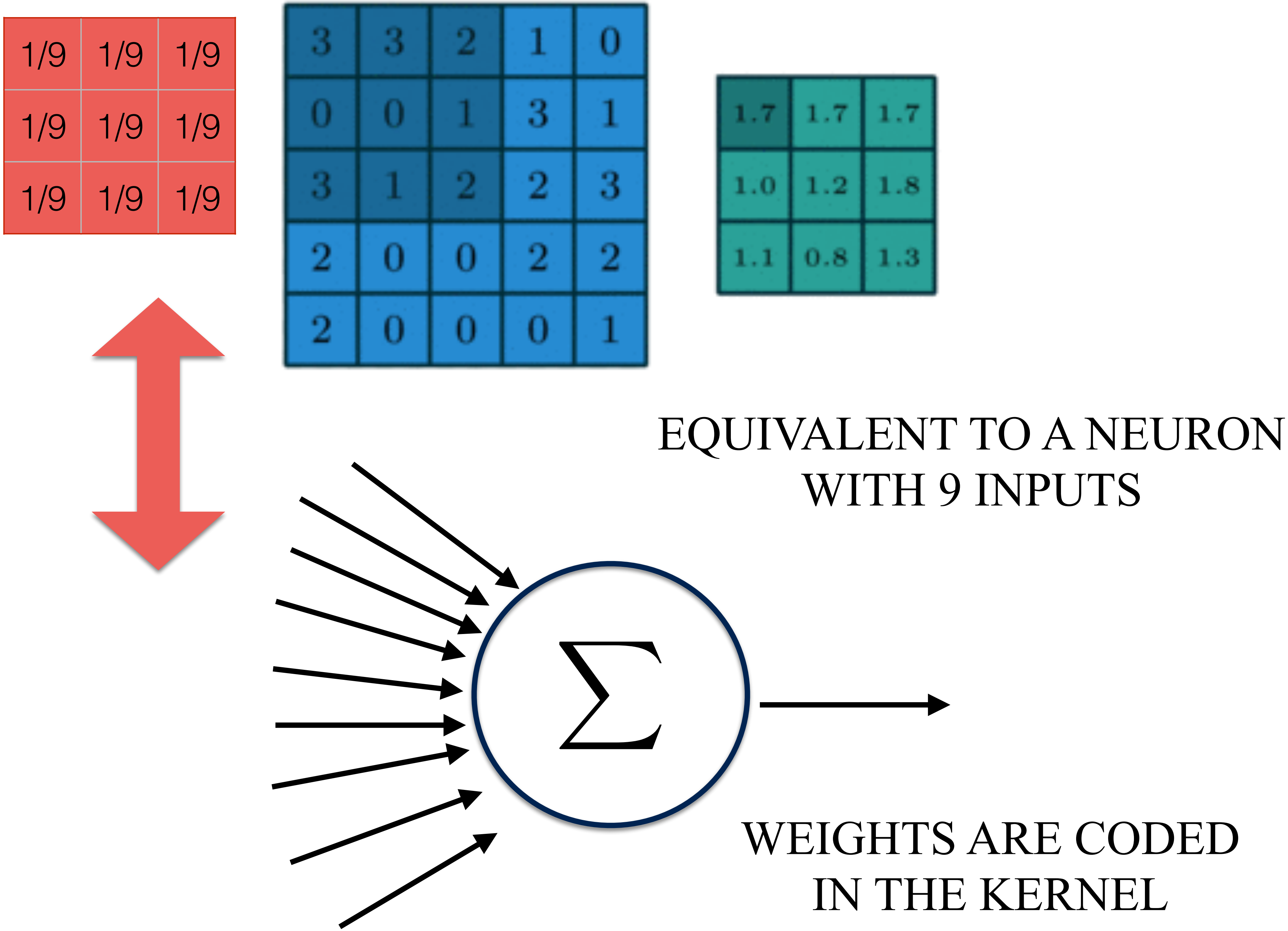
SAME IDEA, BUT THE KERNEL IS NOW 2D



IN THE EXAMPLE: EACH 3x3 REGION GENERATES AN OUTPUT

$$\boxed{Size_{output} = Size_{input} - Size_{kernel} + 1}$$

Credit: animations from https://github.com/vdumoulin/conv_arithmetic

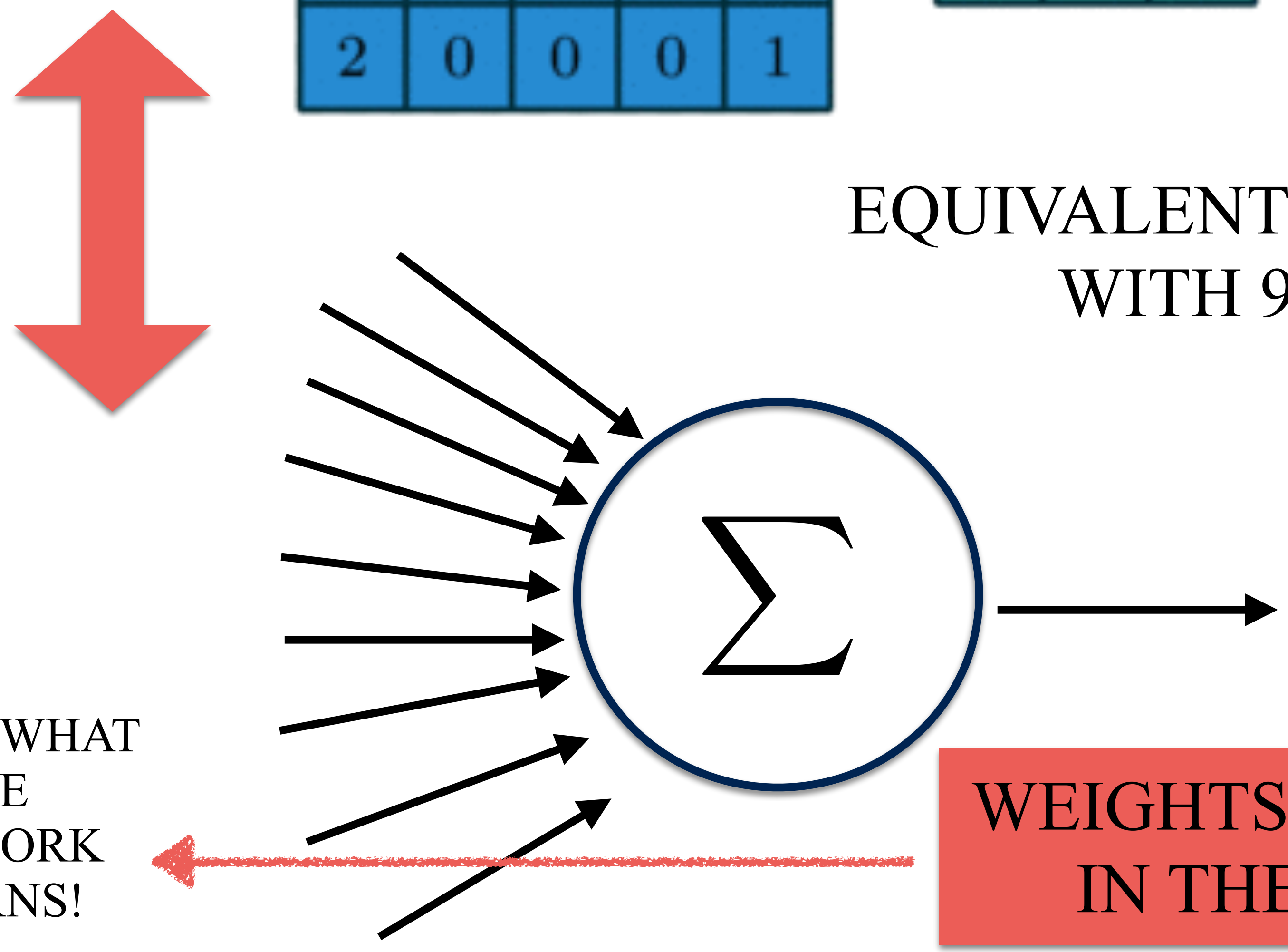

1/9 1/9 1/9
1/9 1/9 1/9
1/9 1/9 1/9
3 3 2 1 0
0 0 1 3 1
3 1 2 2 3
2 0 0 2 2
2 0 0 0 1
1.7 1.7 1.7
1.0 1.2 1.8
1.1 0.8 1.3
EQUIVALENT TO A NEURON
WITH 9 INPUTS
Σ
WEIGHTS ARE CODED
IN THE KERNEL

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

1.7	1.7	1.7
1.0	1.2	1.8
1.1	0.8	1.3

EQUIVALENT TO A NEURON WITH 9 INPUTS



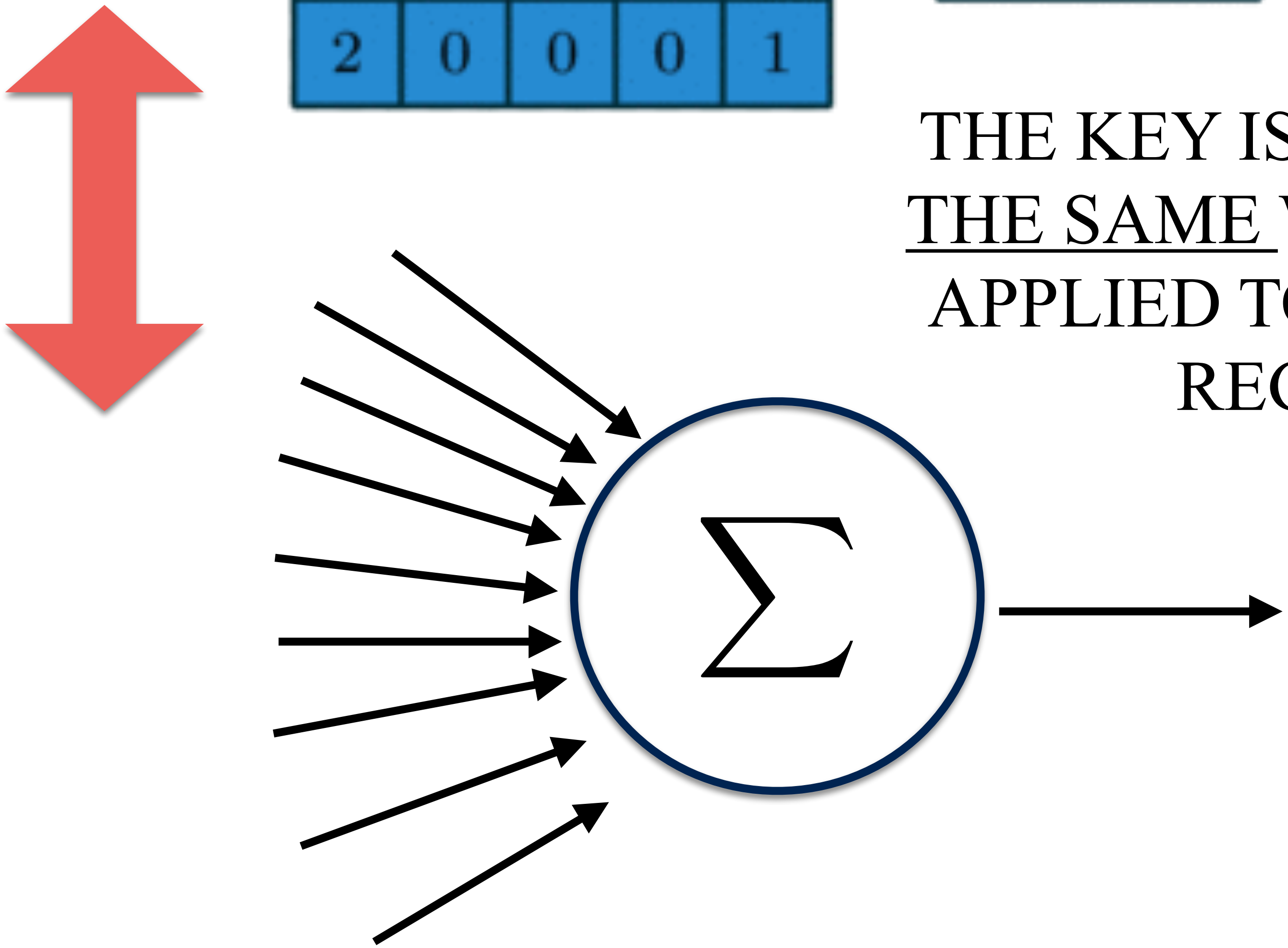
THIS IS WHAT THE NETWORK LEARNS!

WEIGHTS ARE CODED IN THE KERNEL

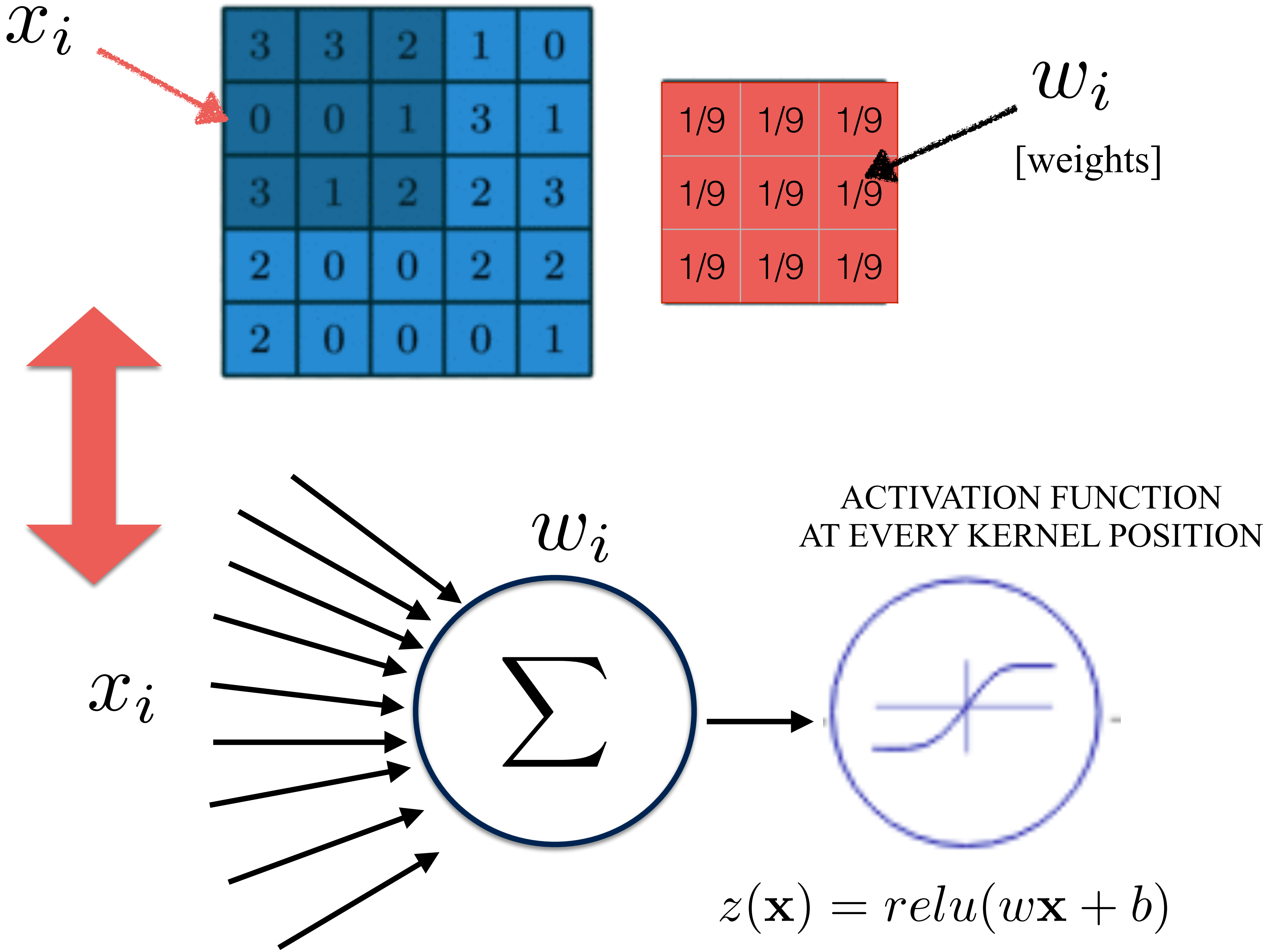
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

1.7	1.7	1.7
1.0	1.2	1.8
1.1	0.8	1.3

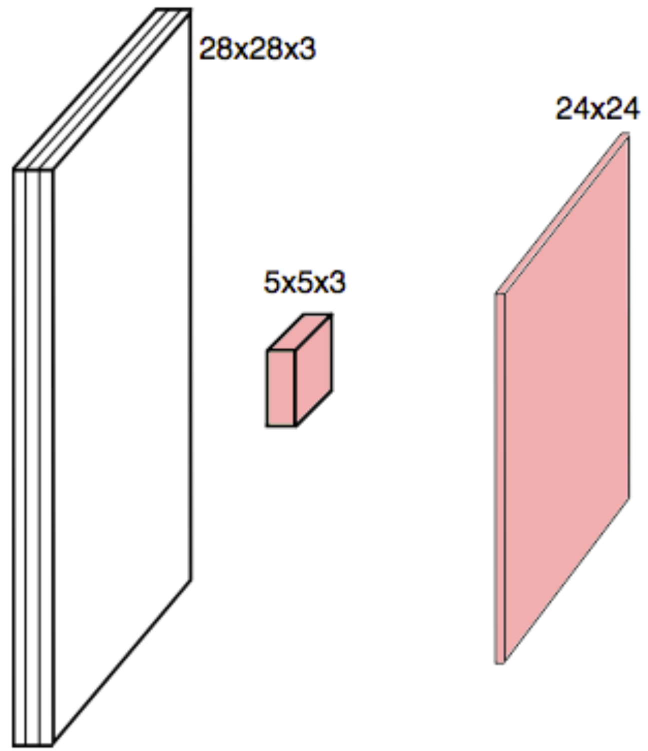


THE KEY IS AGAIN THAT THE SAME WEIGHTS ARE APPLIED TO ALL IMAGE REGIONS

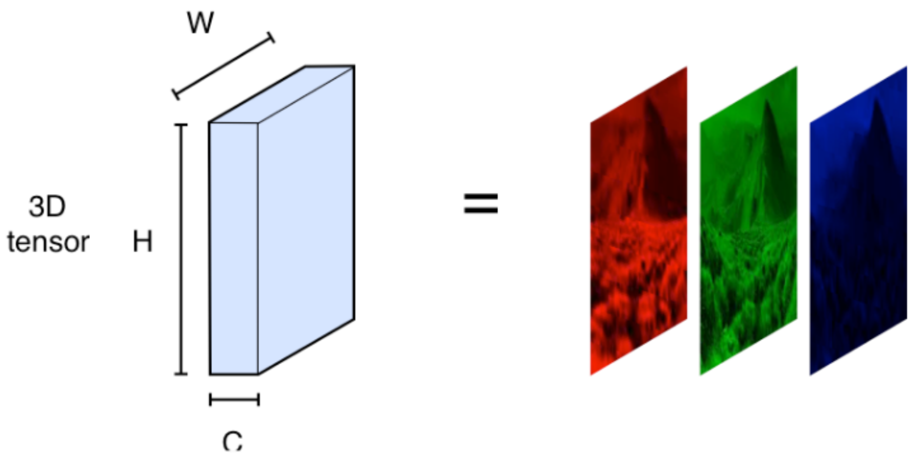


x_i
3 3 2 1 0
0 0 1 3 1
3 1 2 2 3
2 0 0 2 2
2 0 0 0 1
1/9 1/9 1/9
1/9 1/9 1/9
1/9 1/9 1/9
w_i
[weights]
w_i
ACTIVATION FUNCTION
AT EVERY KERNEL POSITION
x_i
Σ
$z(\mathbf{x}) = relu(w\mathbf{x} + b)$

CONVOLUTIONS CAN ALSO BE COMPUTED ACROSS CHANNELS (OR COLORS)

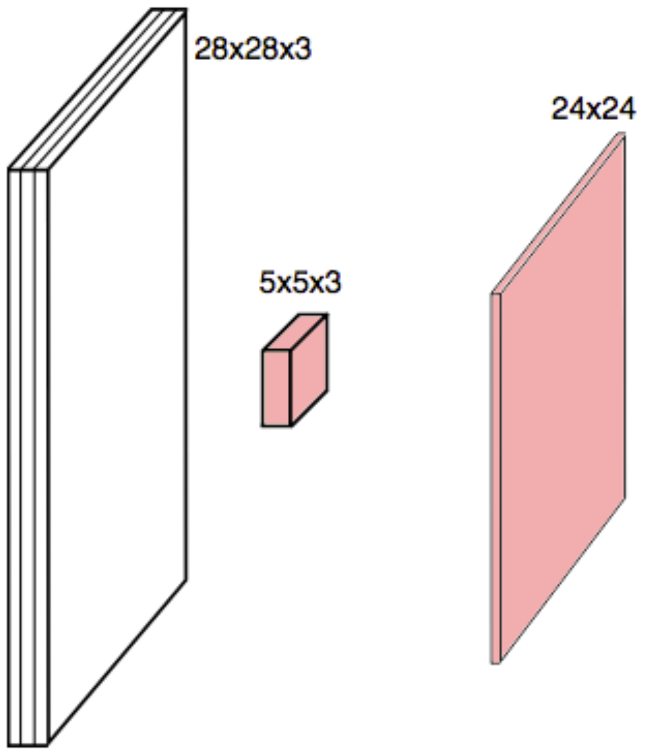


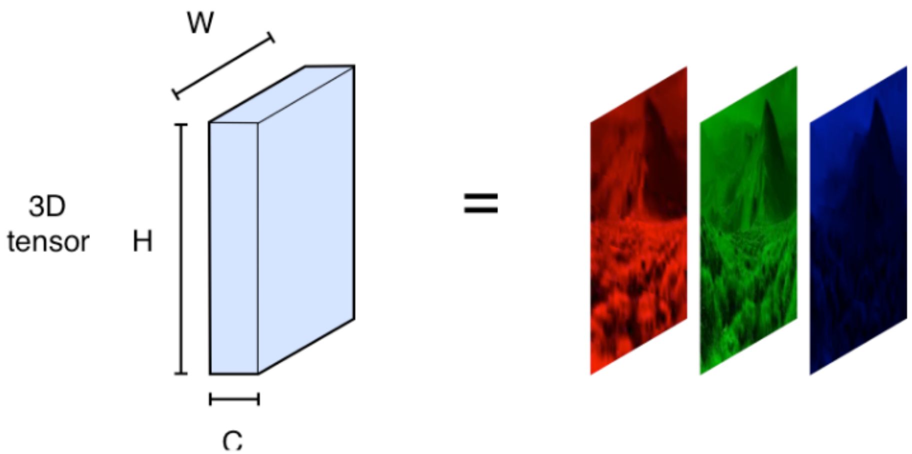
A COLOR IMAGE IS A TENSOR
OF SIZE height x width x channels



CONVOLUTIONS CAN ALSO BE COMPUTED ACROSS CHANNELS (OR COLORS)

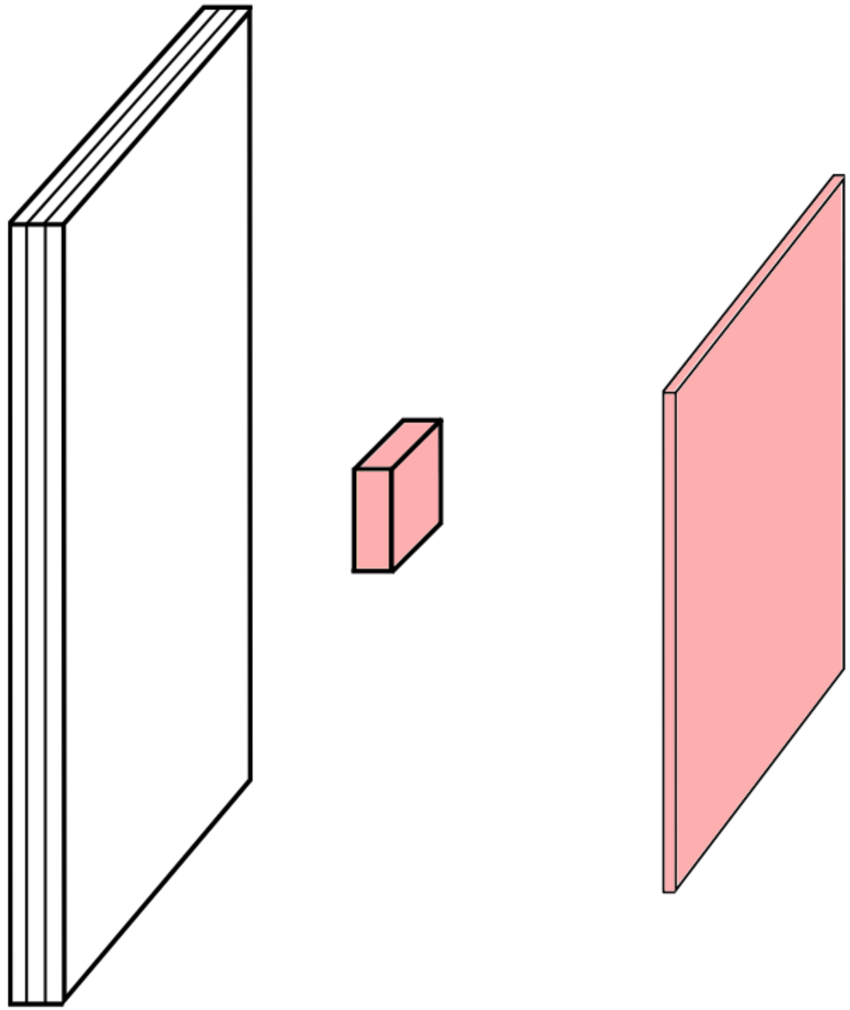
A COLOR IMAGE IS A TENSOR
OF SIZE height x width x channels





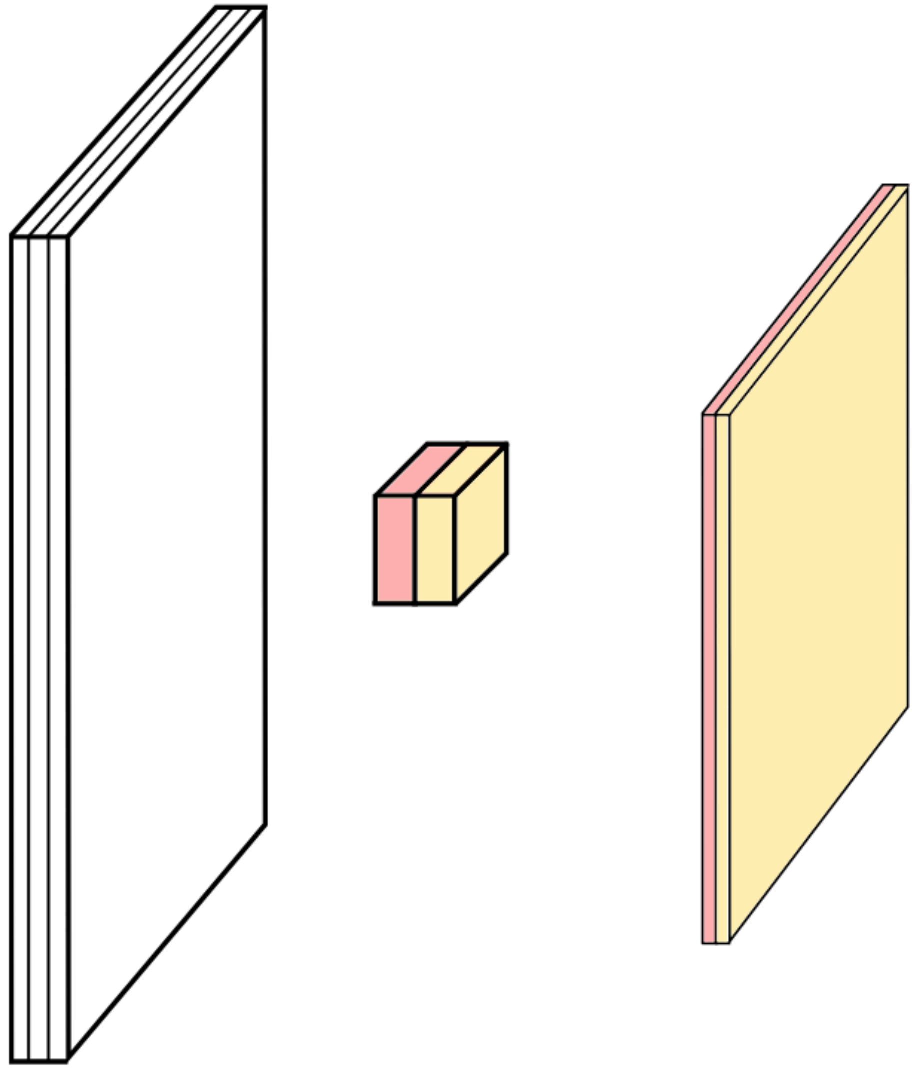
<u>THEN THE KERNEL HAS ALSO 3 CHANNELS</u>

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



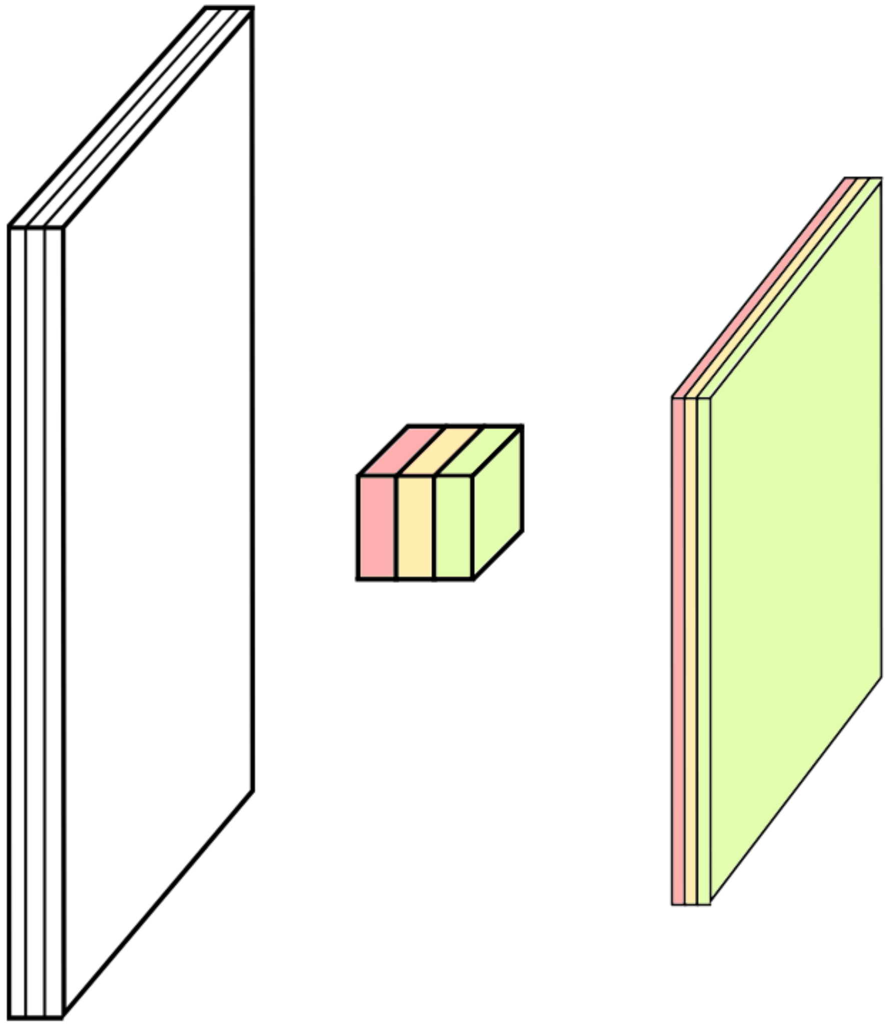
credit

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



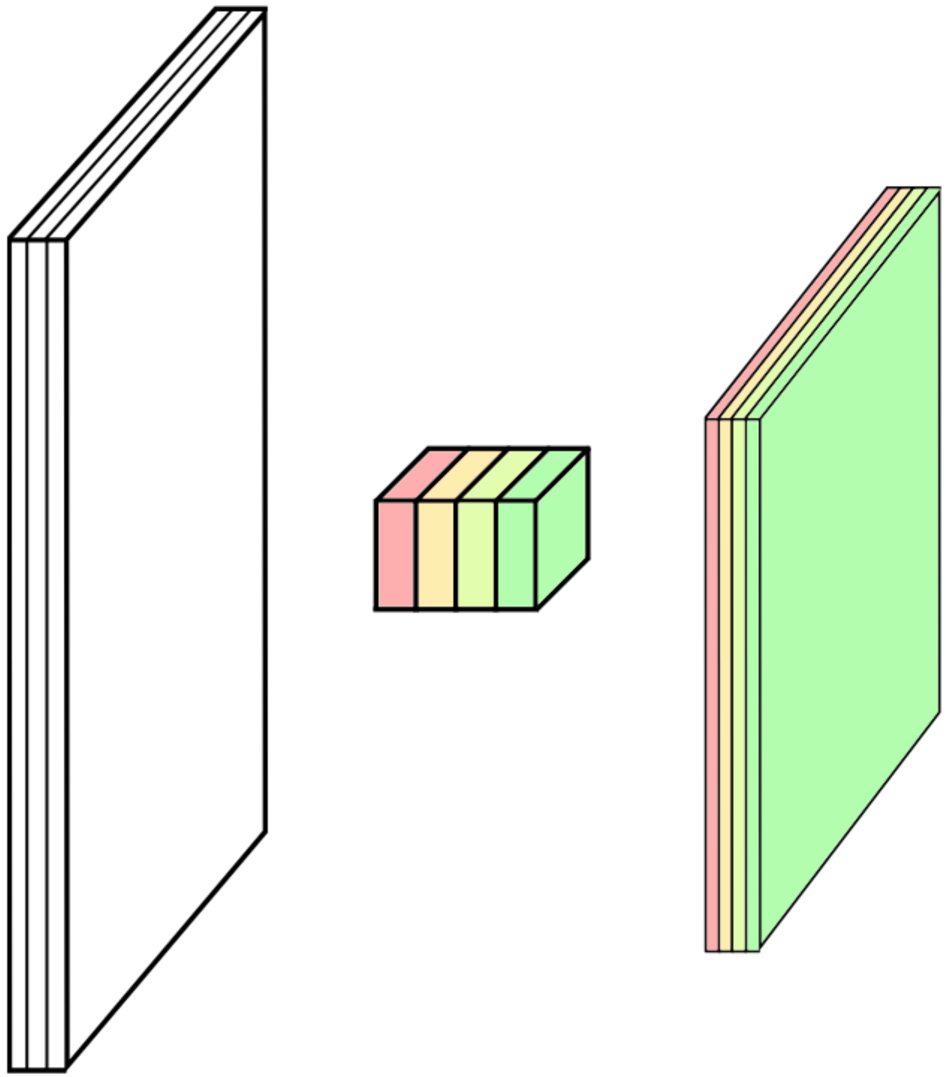
credit

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



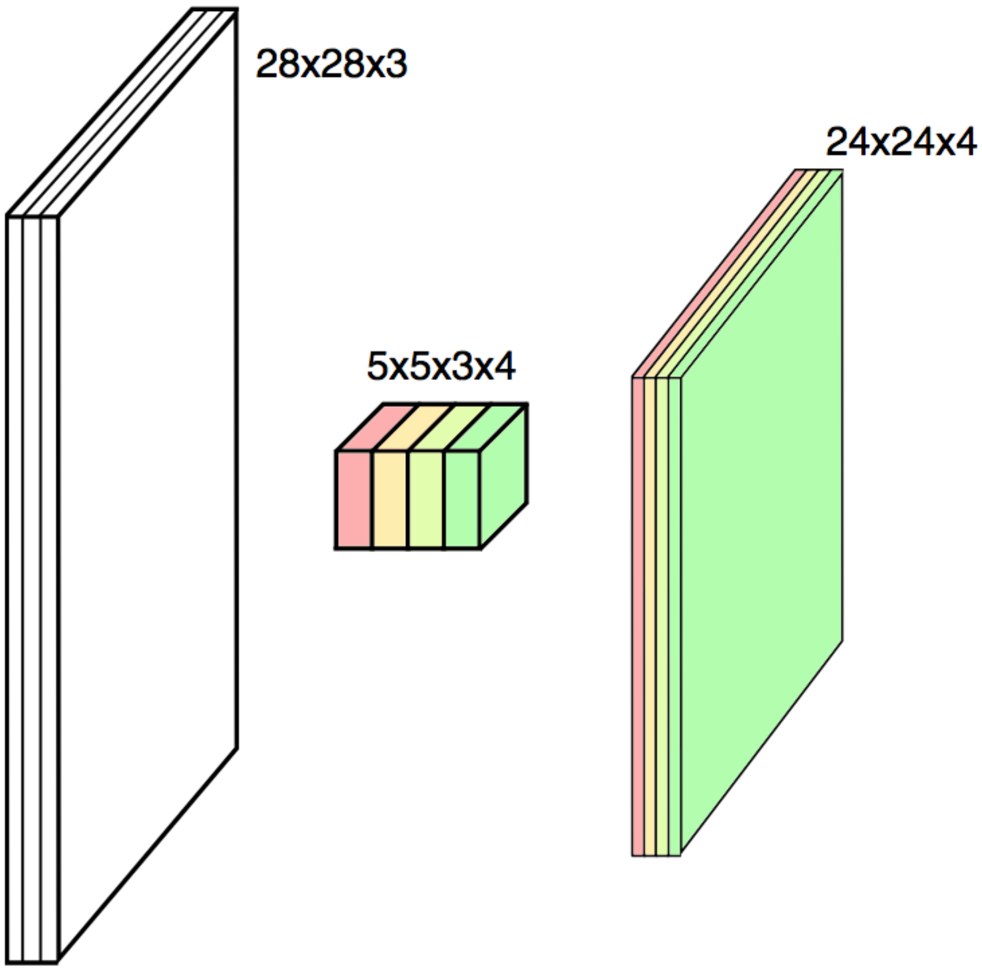
credit

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



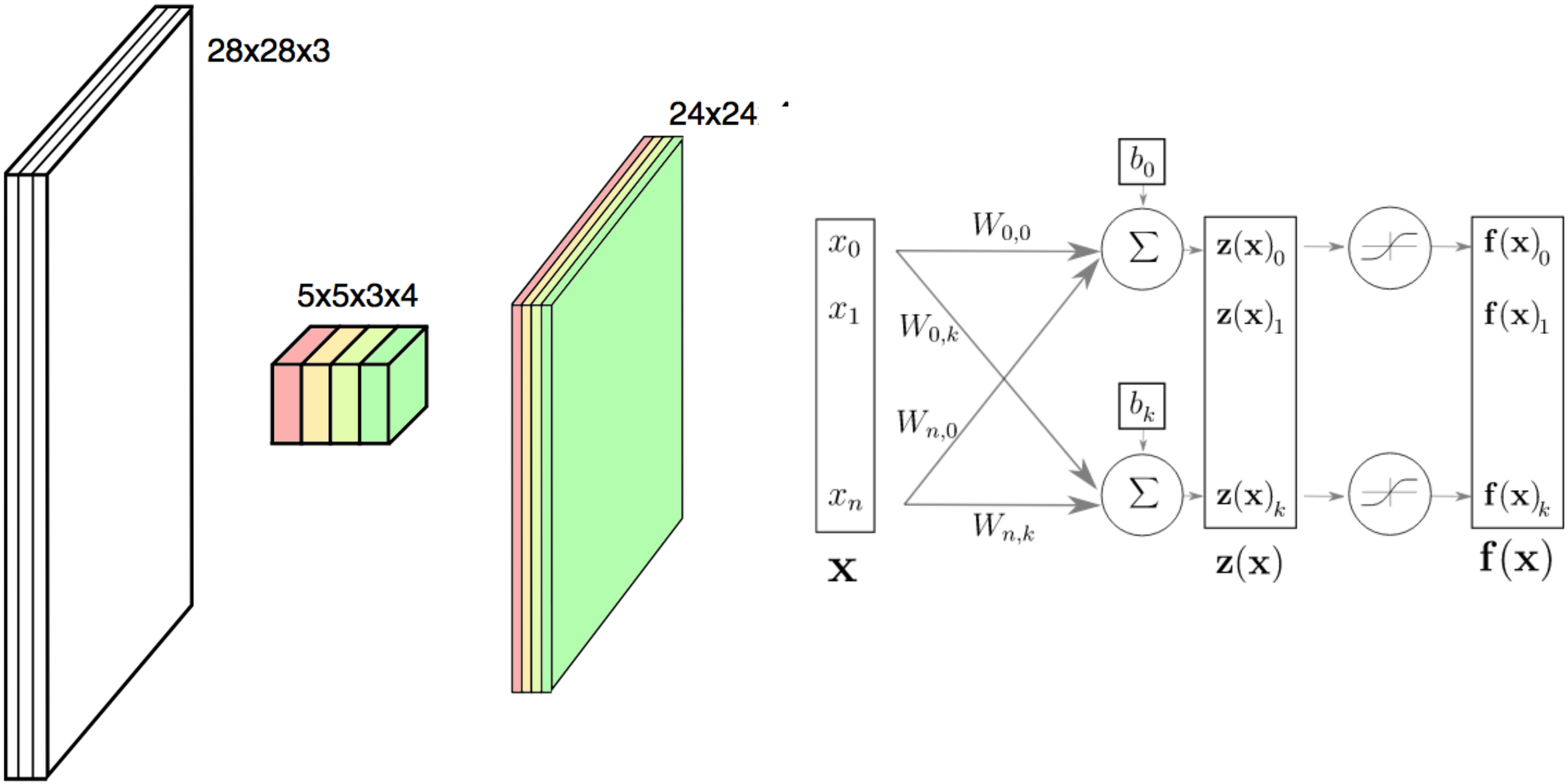
credit

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



credit

MULTIPLE CONVOLUTIONS WITH DIFFERENT KERNELS CAN BE PERFORMED



credit

IN KERAS…

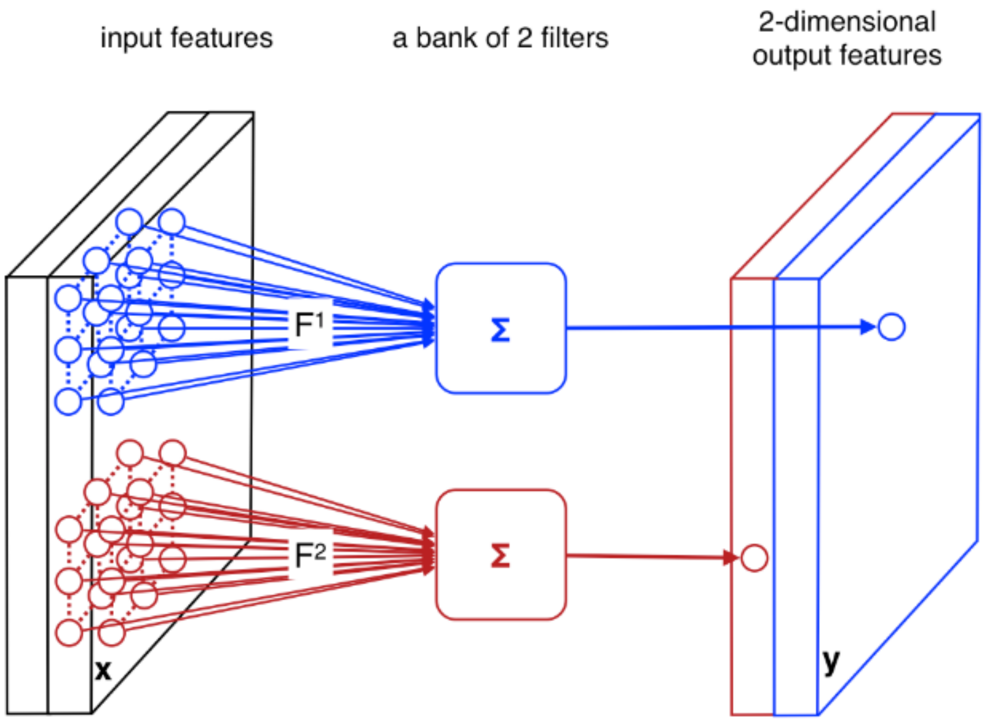
```
model = Sequential()

model.add(Convolution2D(4,5,5, activation="relu"))
```

sizeX, sizeY

depth

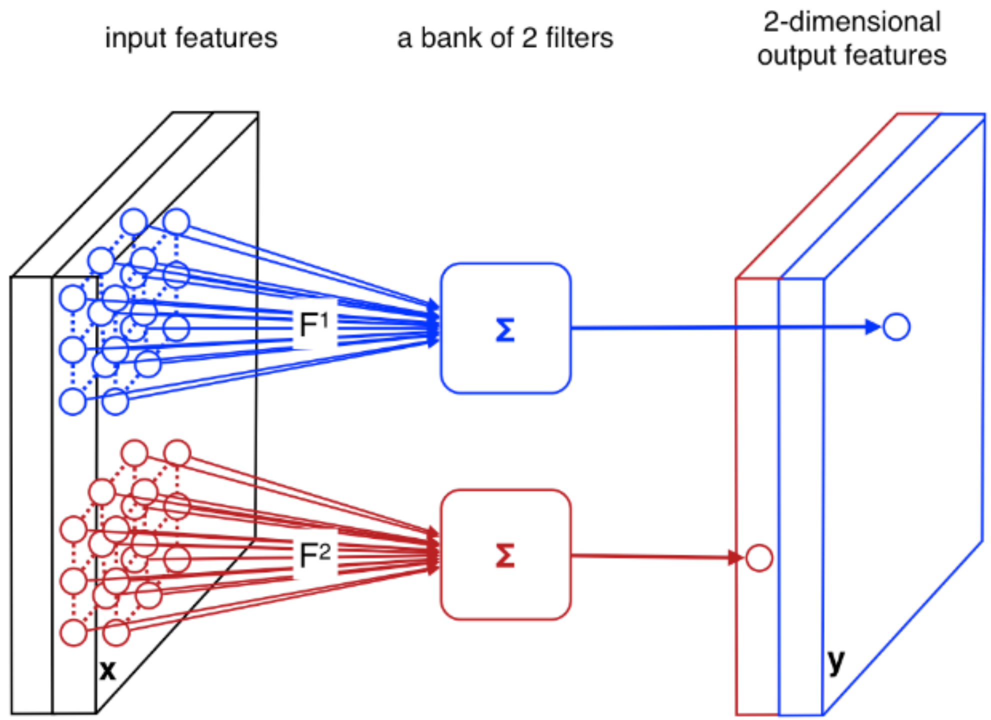
SINCE CONVOLUTIONS OUTPUT ONE SCALAR, THEY CAN BE SEEN AS AN INDIVIDUAL NEURON WITH A RECEPTIVE FIELD LIMITED TO THE KERNEL DIMENSIONS



Credit

SINCE CONVOLUTIONS OUTPUT ONE SCALAR< THEY CAN BE SEEN AS AN INDIVIDUAL NEURON WITH A RECEPTIVE FIELD LIMITED TO THE KERNEL DIMENSIONS

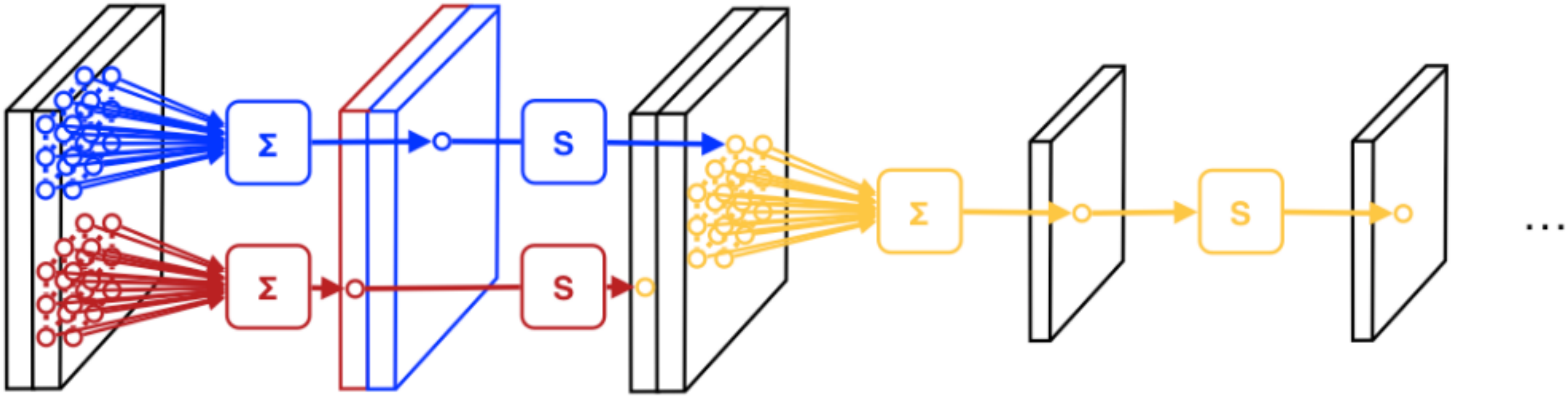
THE SAME NEURON IS FIRED WITH DIFFERENT AREAS FROM THE INPUT



Credit

DOWNSAMPLING

DOWNSAMPLING IS APPLIED TO REDUCE THE OVERALL SIZE OF TENSORS



POOLING

CONVOLUTIONS ARE OFTEN FOLLOWED BY AN OPERATION OF DOWNSAMPLING [POOLING]

VERY SIMPLE OPERATION - ONLY ONE OUT OF EVERY N PIXELS ARE KEPT

OFTEN MATCHED WITH AN INCREASE OF THE FEATURE CHANNELS

TYPES OF POOLING

<u>SUM POOLING</u>

$$y = \sum x_{uv}$$

<u>SQUARE SUM POOLING</u>

$$y = \sqrt{\sum x_{uv}^2}$$

<u>MAX POOLING</u>

$$y = max(x_{uv})$$

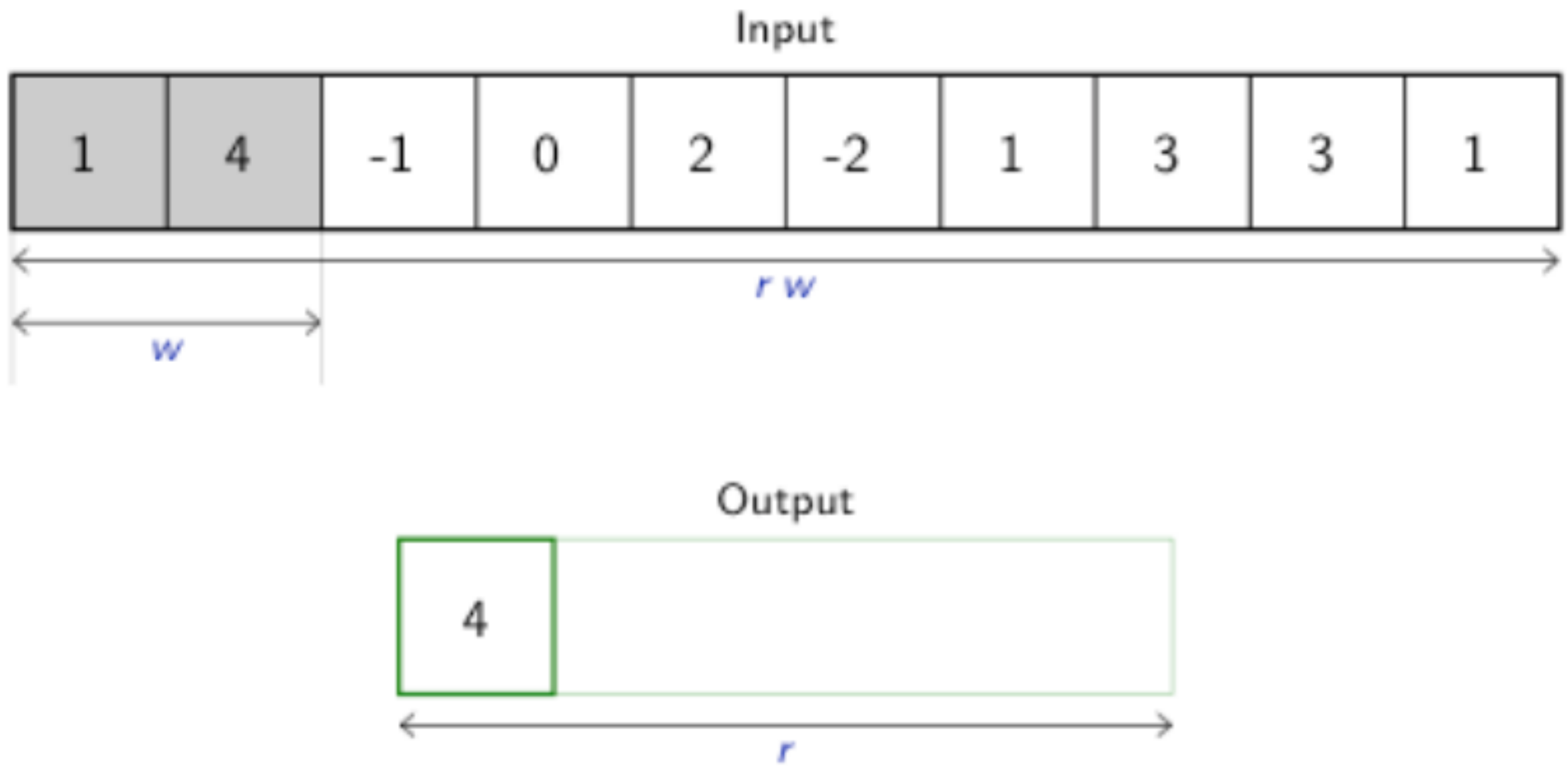
TYPES OF POOLING

<u>SUM POOLING</u> $y = \sum x_{uv}$

<u>SQUARE SUM POOLING</u> $y = \sqrt{\sum x_{uv}^2}$

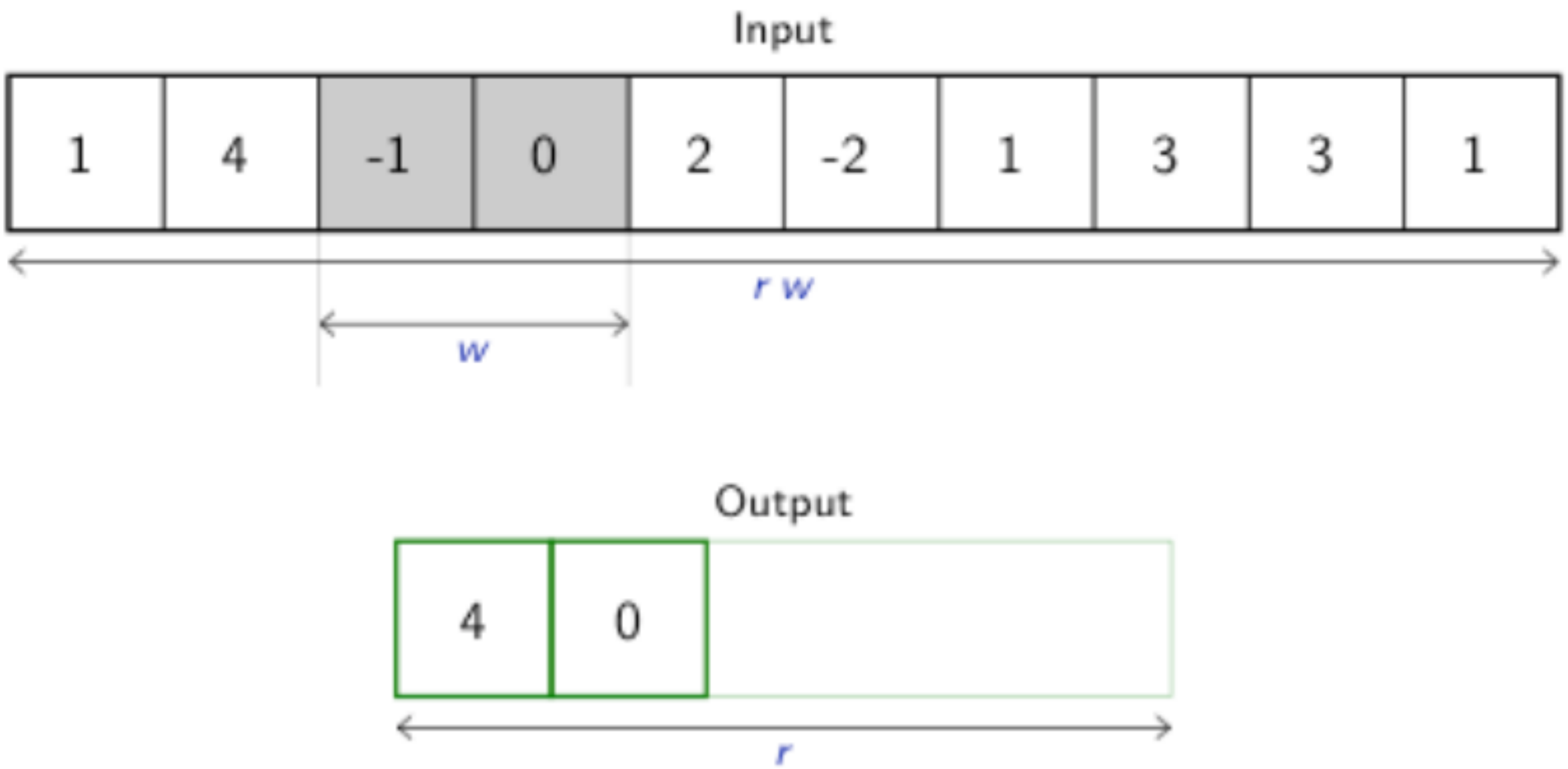
<u>MAX POOLING</u> $y = max(x_{uv})$

MAX POOLING 1D



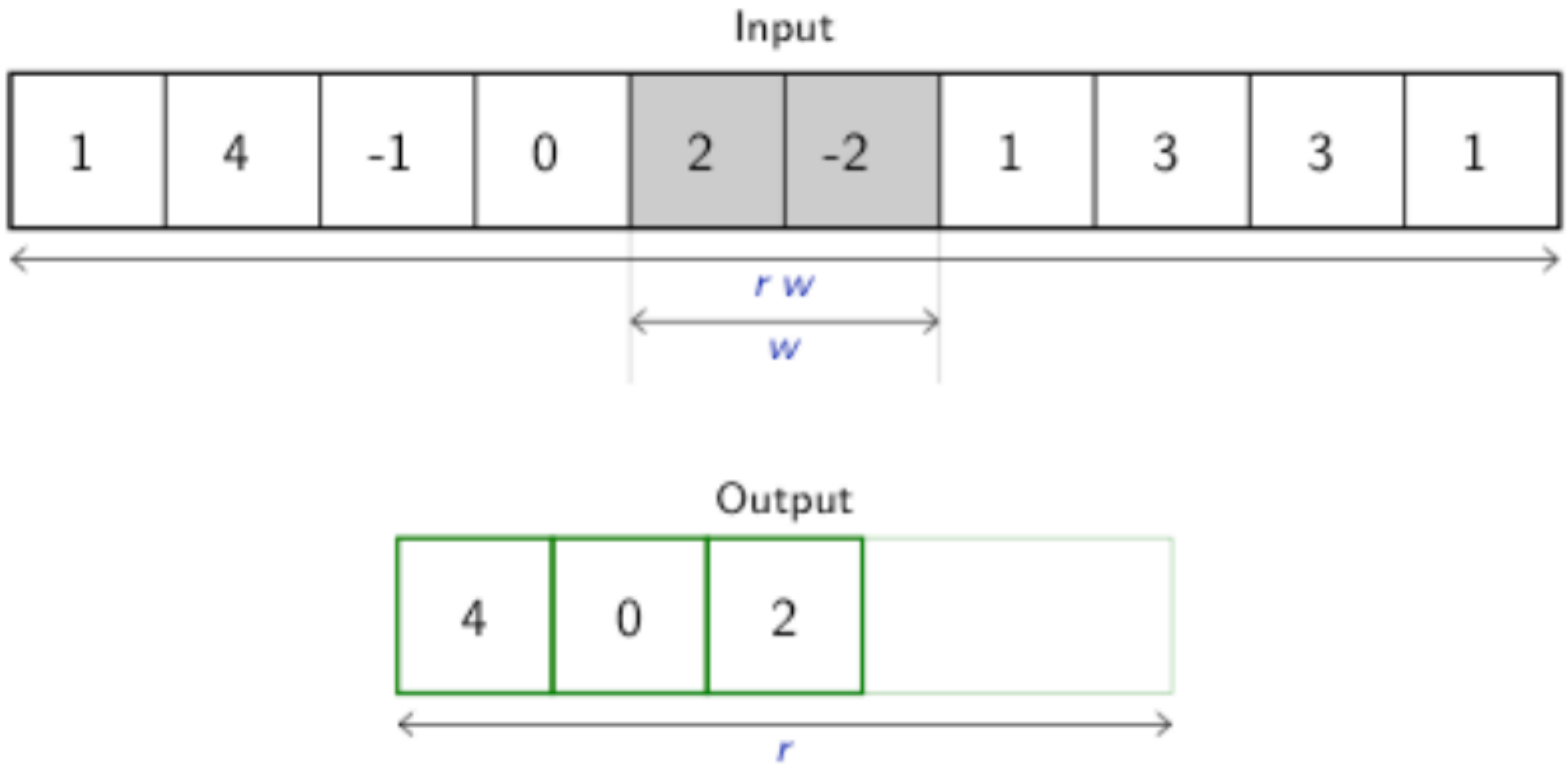
Credit: F. Fleuret

MAX POOLING 1D



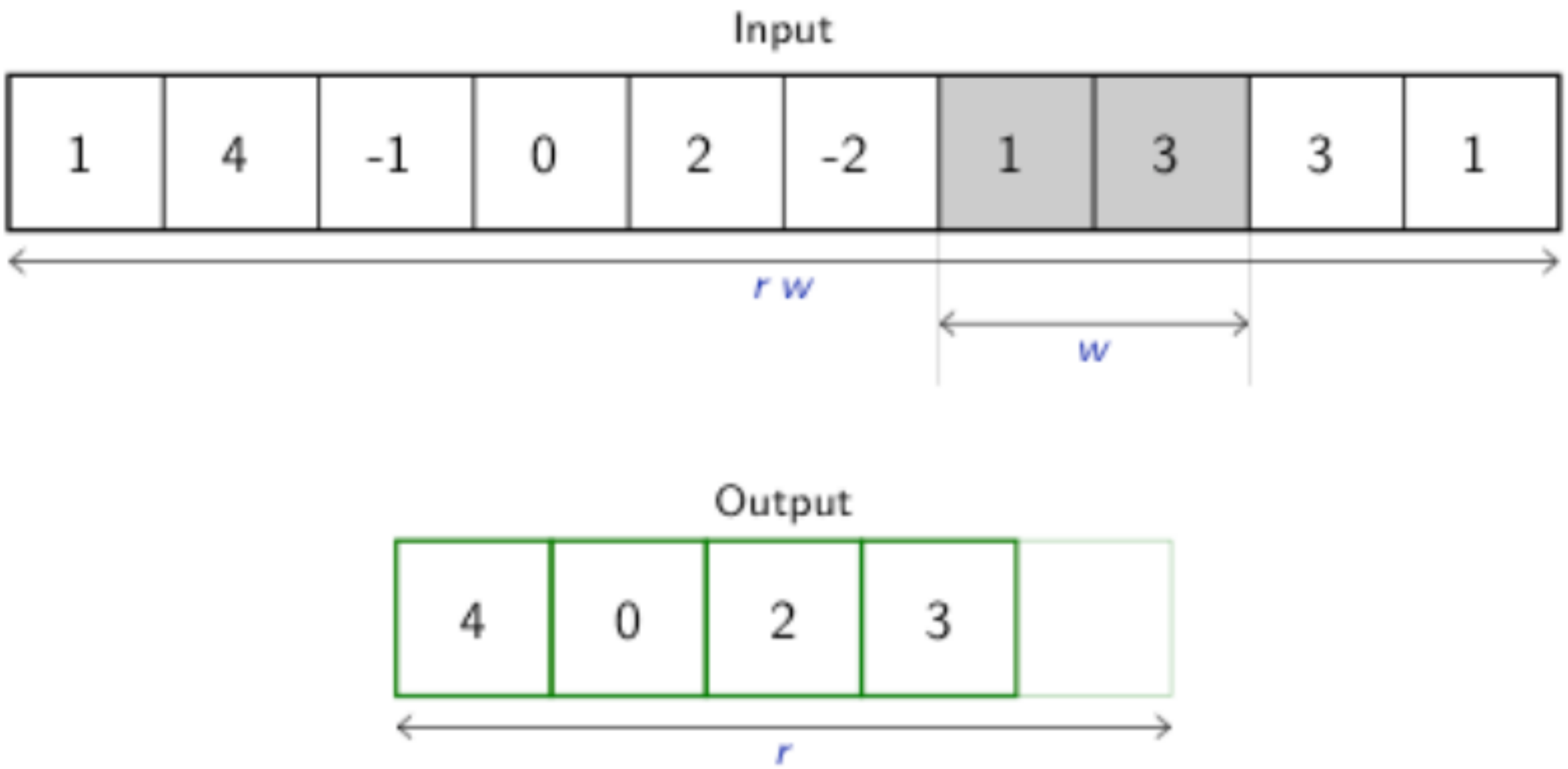
Credit: F. Fleuret

MAX POOLING 1D



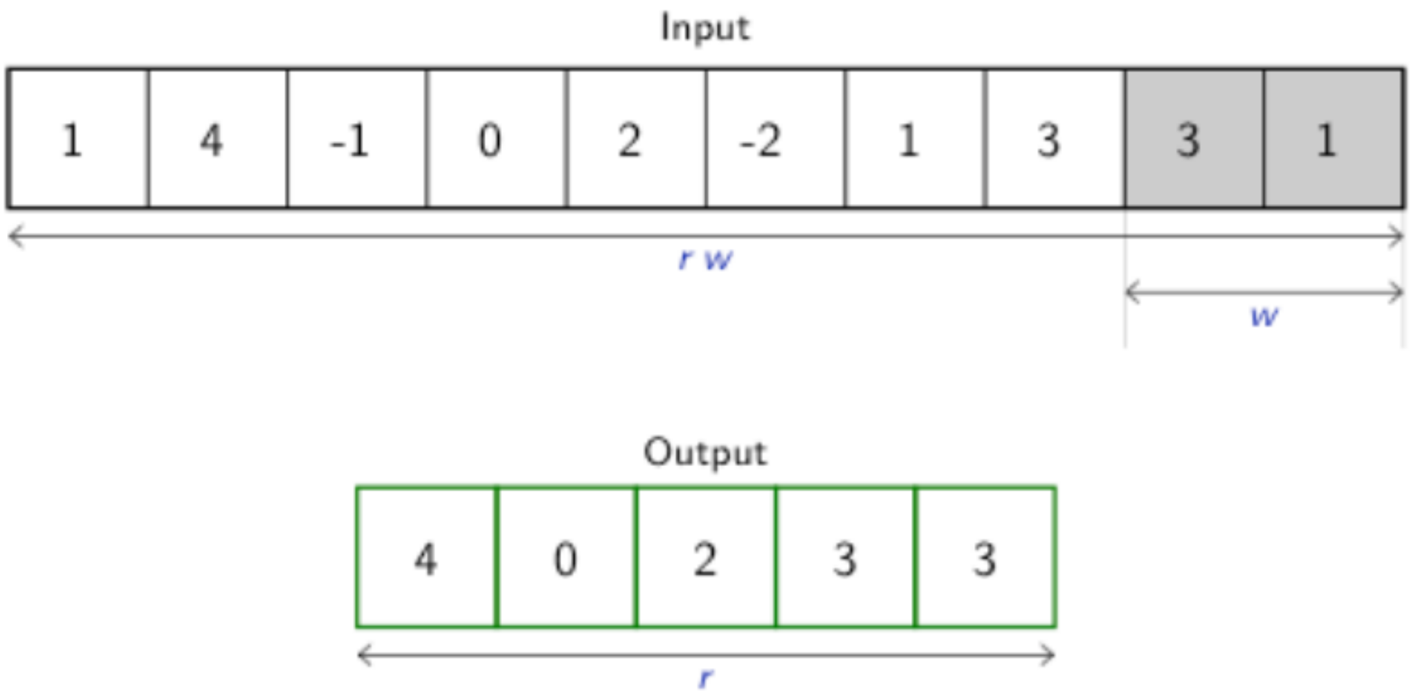
Credit: F. Fleuret

MAX POOLING 1D



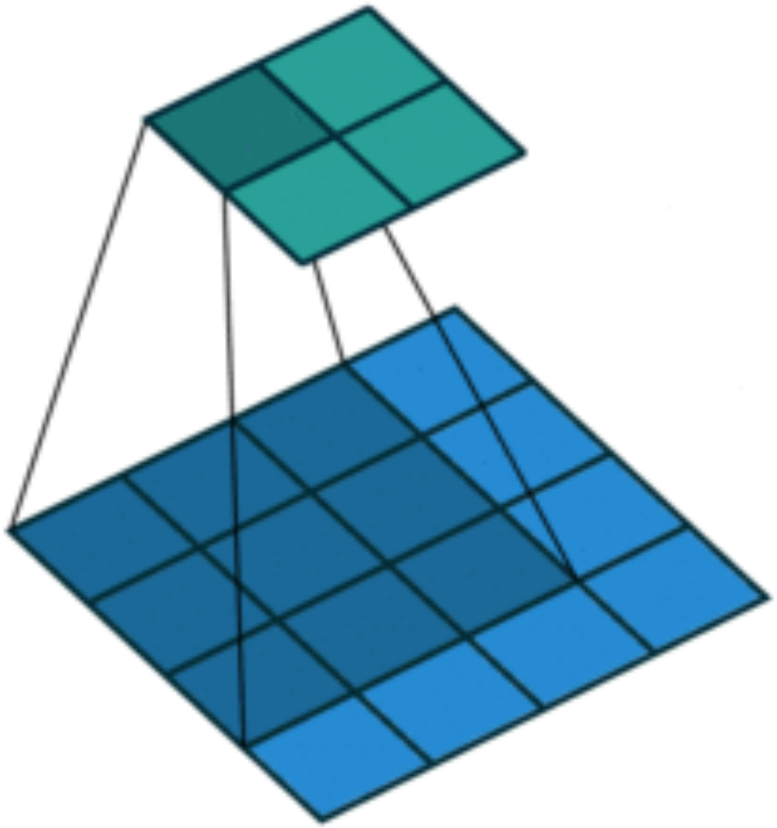
Credit: F. Fleuret

MAX POOLING 1D

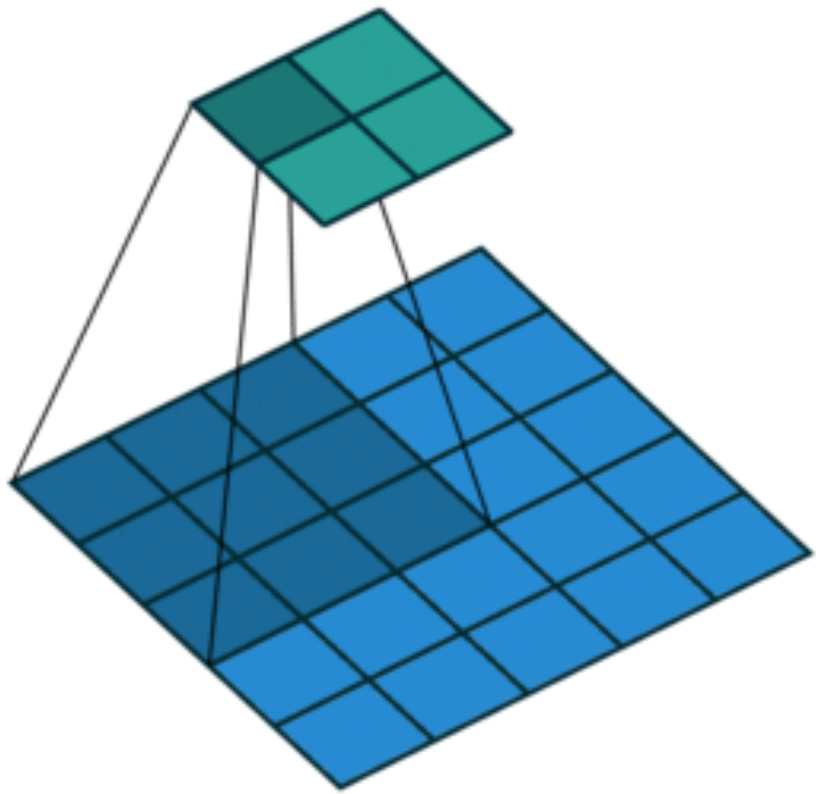


Credit: F. Fleuret

OPTIONS: **STRIDES**

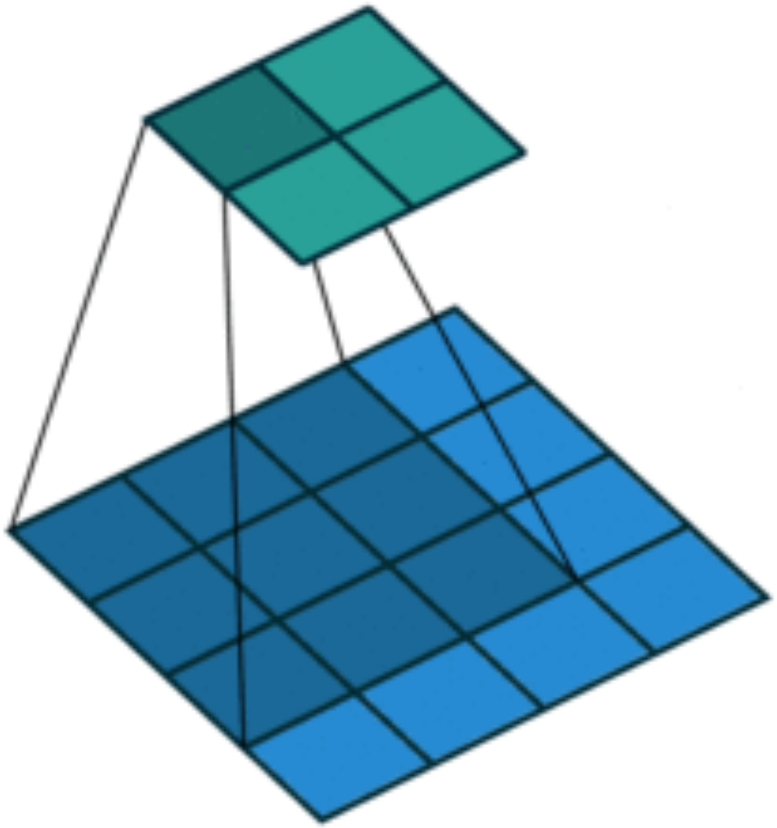


NO STRIDES

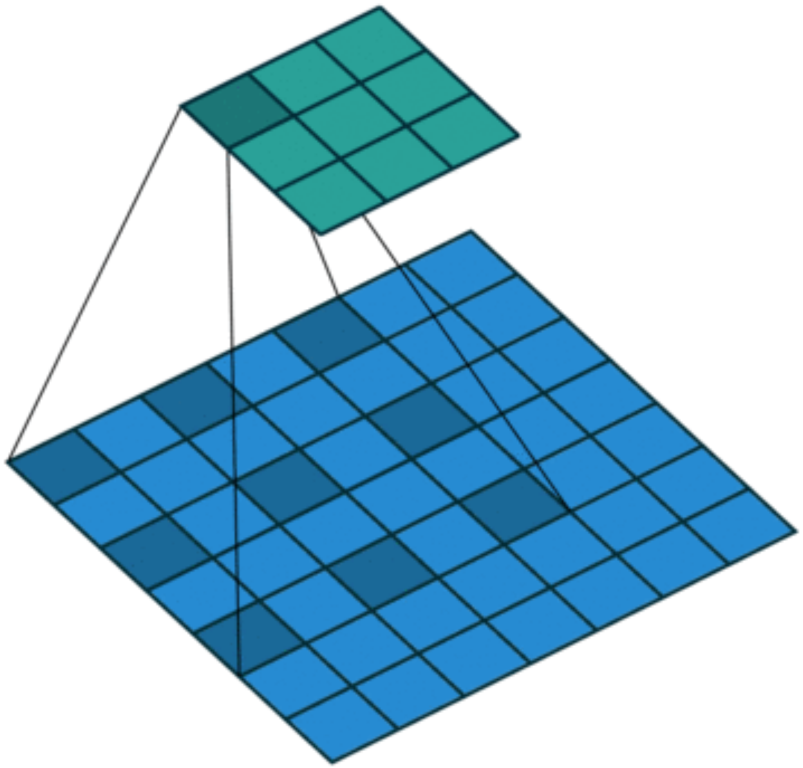


STRIDES

OPTIONS: **DILATION**

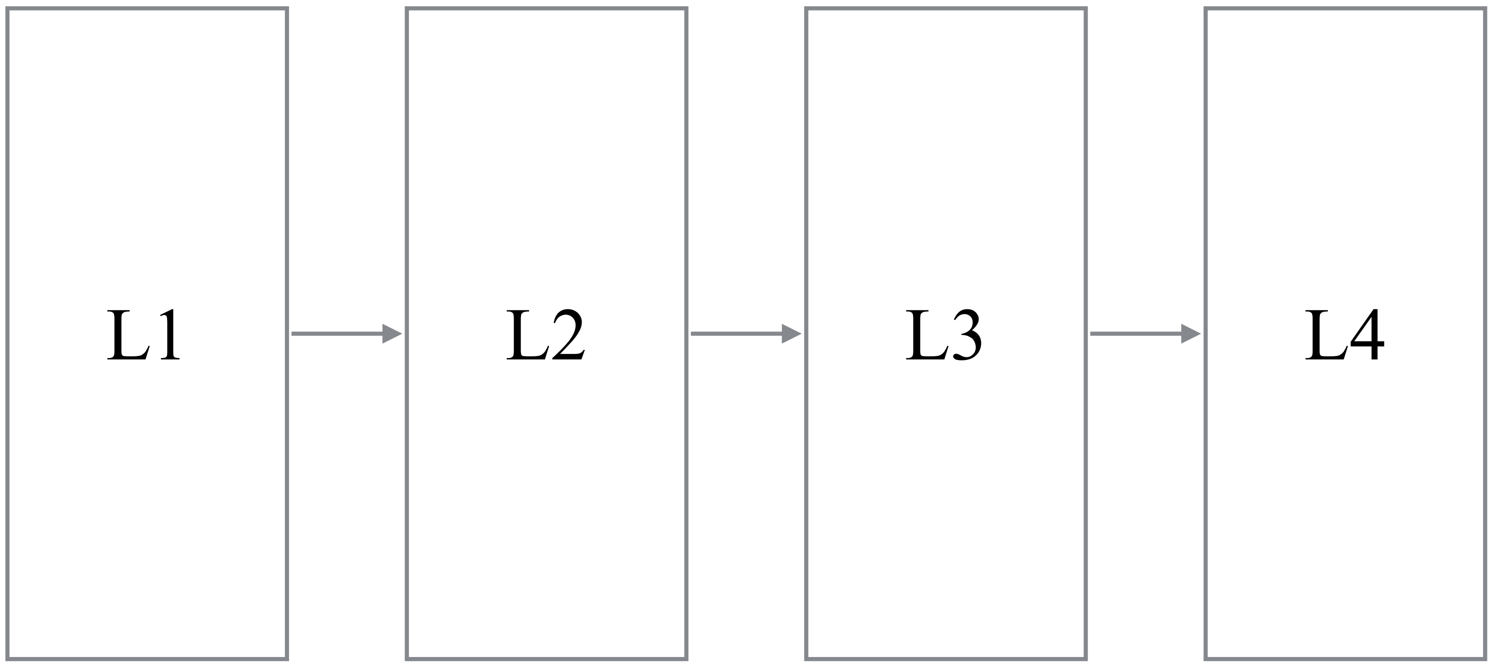


NO STRIDES



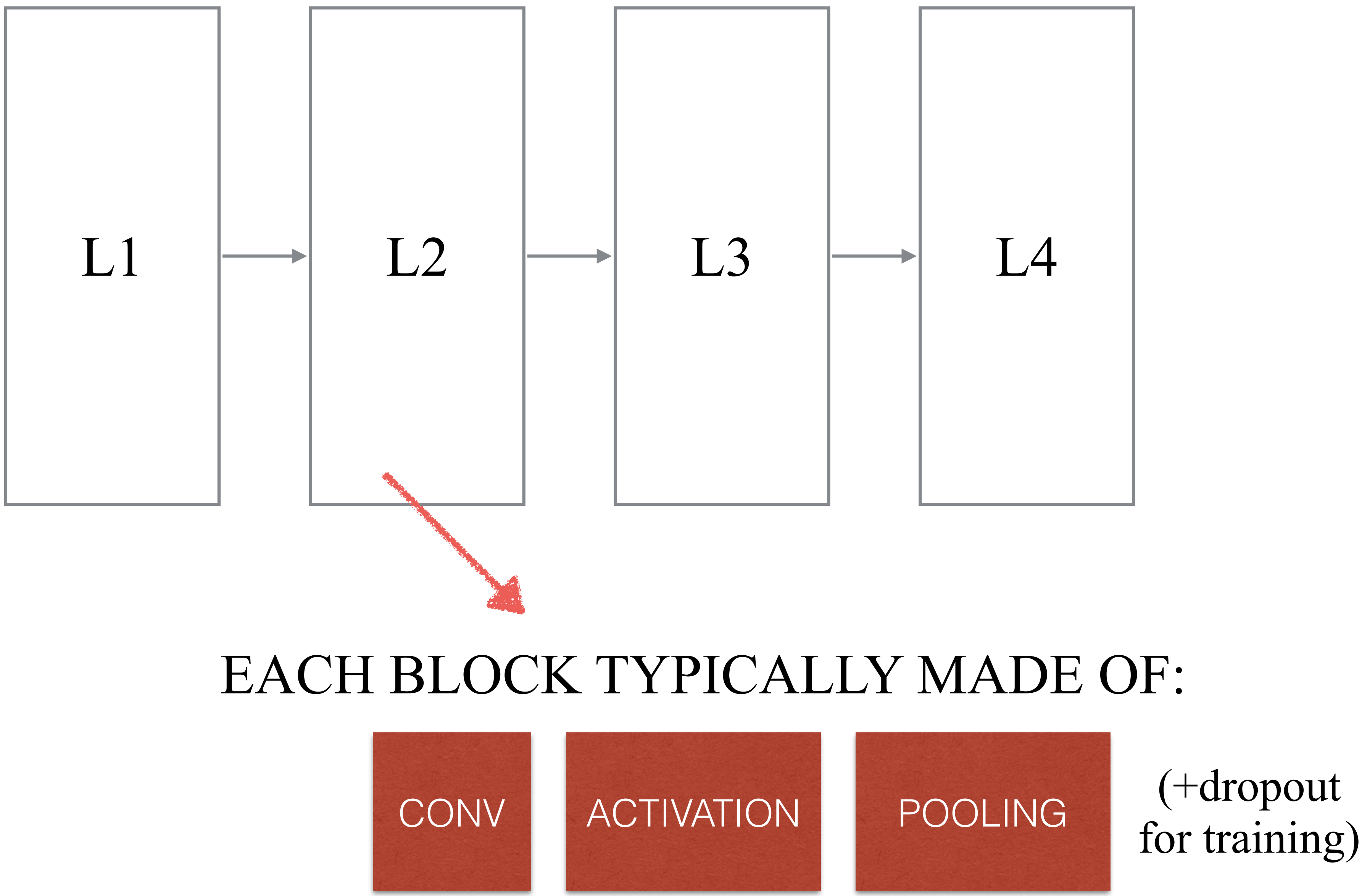
DILATION

CONVNET OR CNN

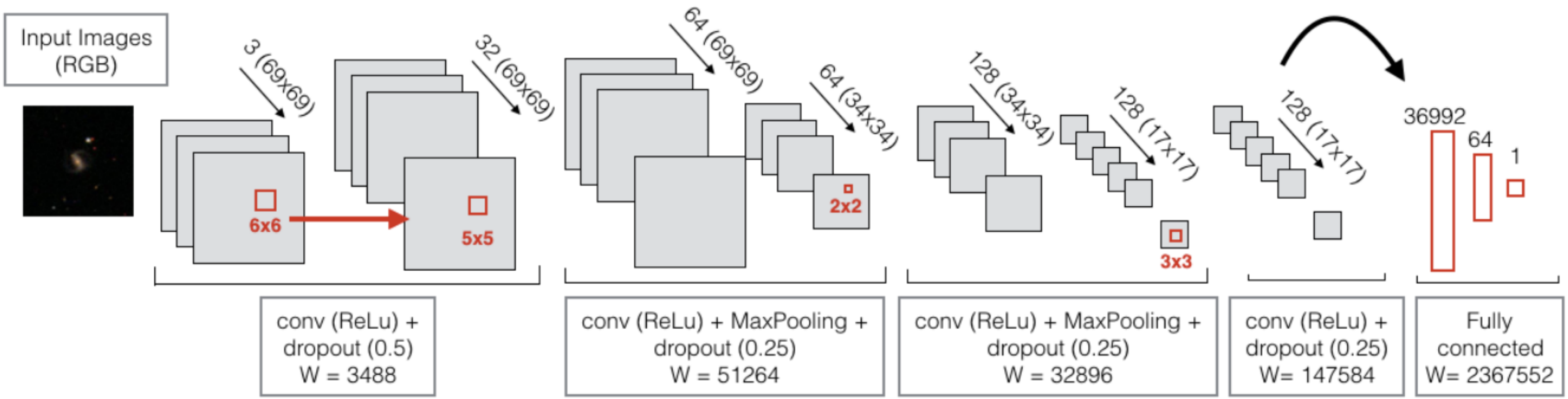


A CONCATENATION OF MULTIPLE CONVOLUTIONAL BLOCKS

CONVNET OR CNN



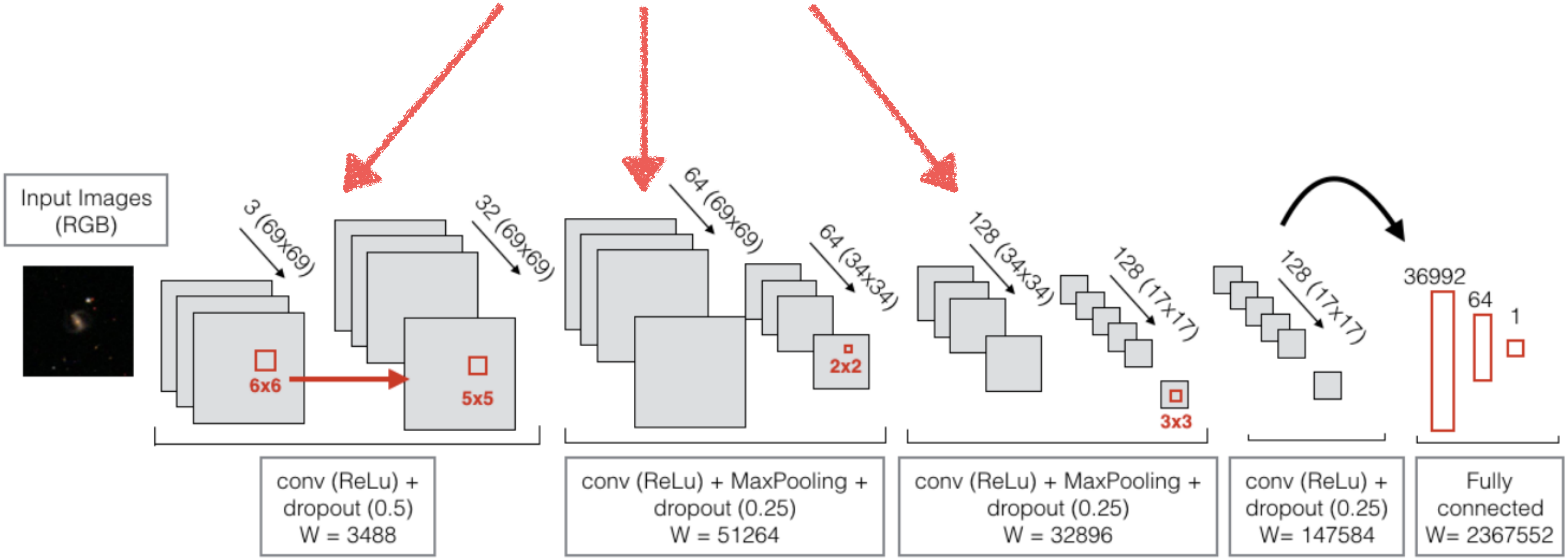
EXAMPLE OF VERY SIMPLE CNN



Dominguez-Sanchez+18

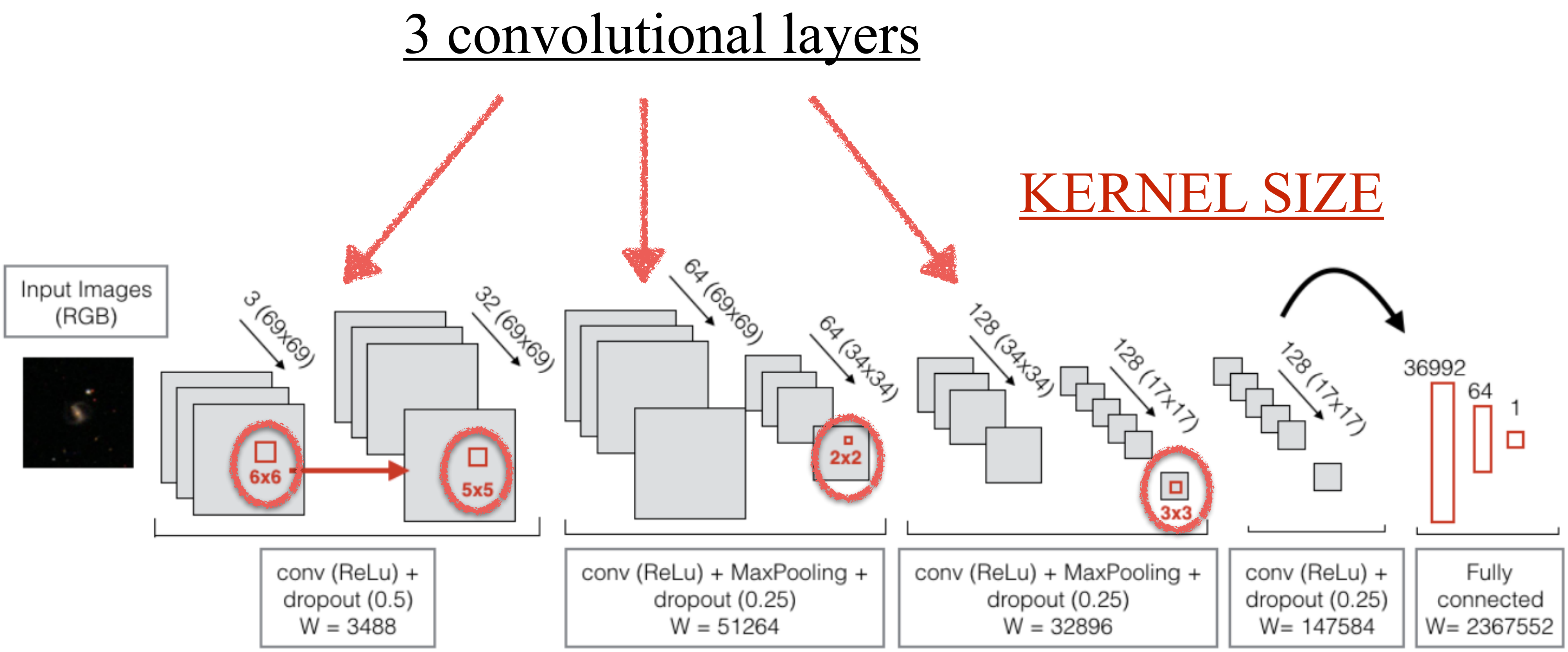
EXAMPLE OF VERY SIMPLE CNN

3 convolutional layers



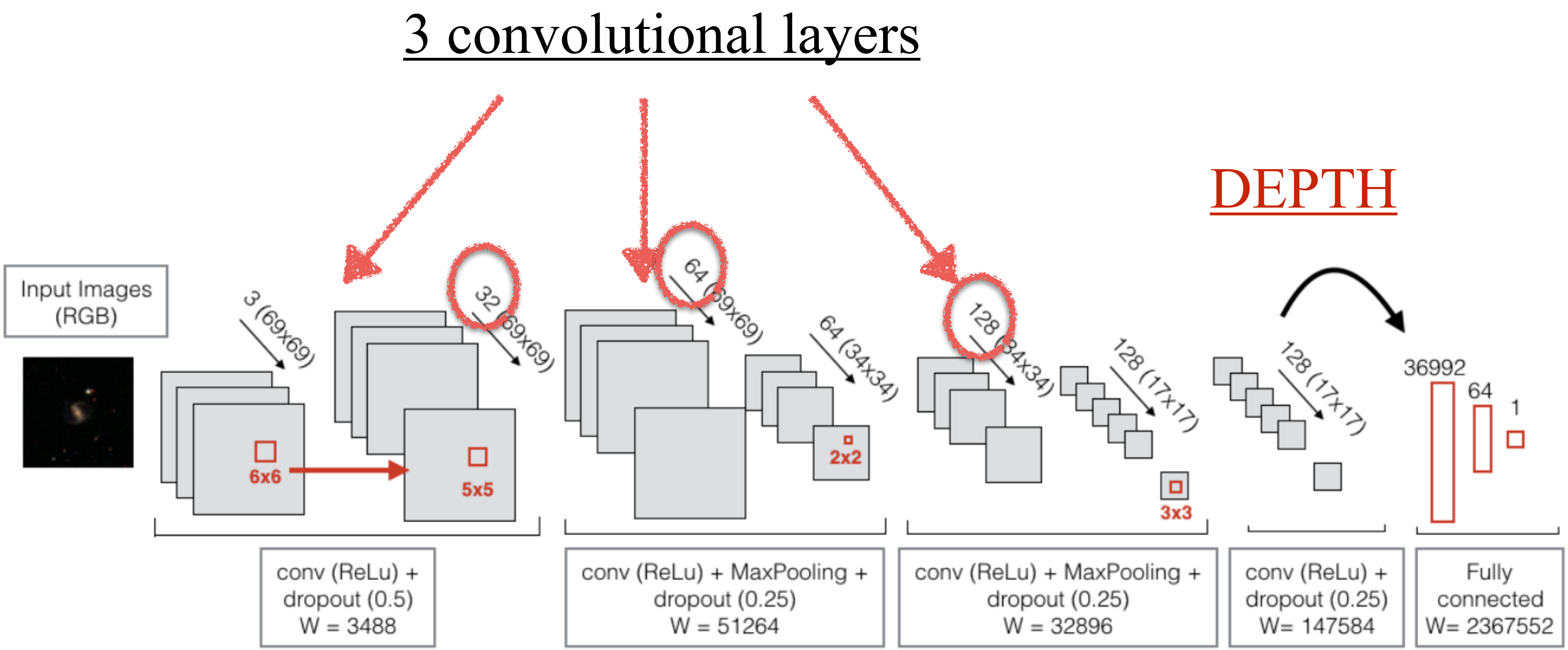
Dominguez-Sanchez+18

EXAMPLE OF VERY SIMPLE CNN



Dominguez-Sanchez+18

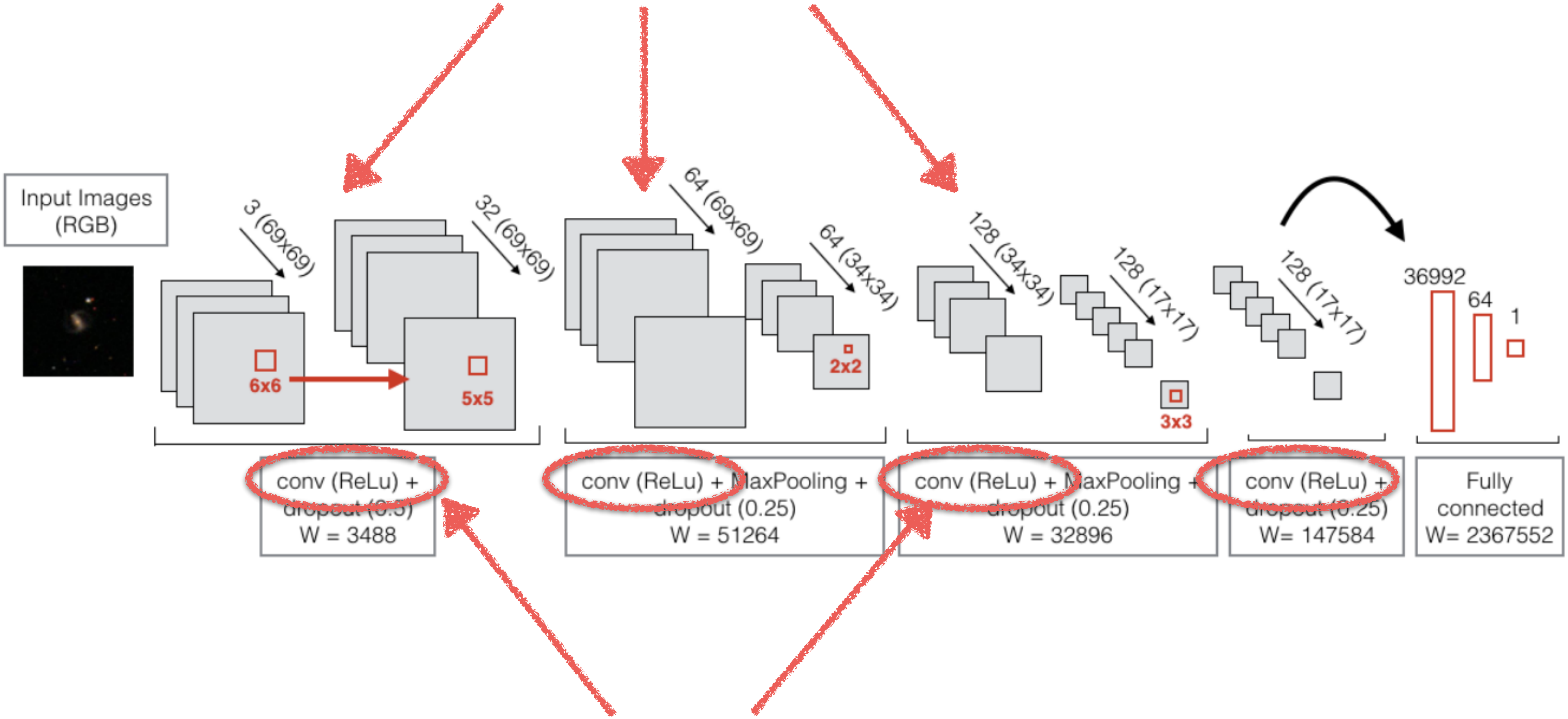
EXAMPLE OF VERY SIMPLE CNN



Dominguez-Sanchez+18

EXAMPLE OF VERY SIMPLE CNN

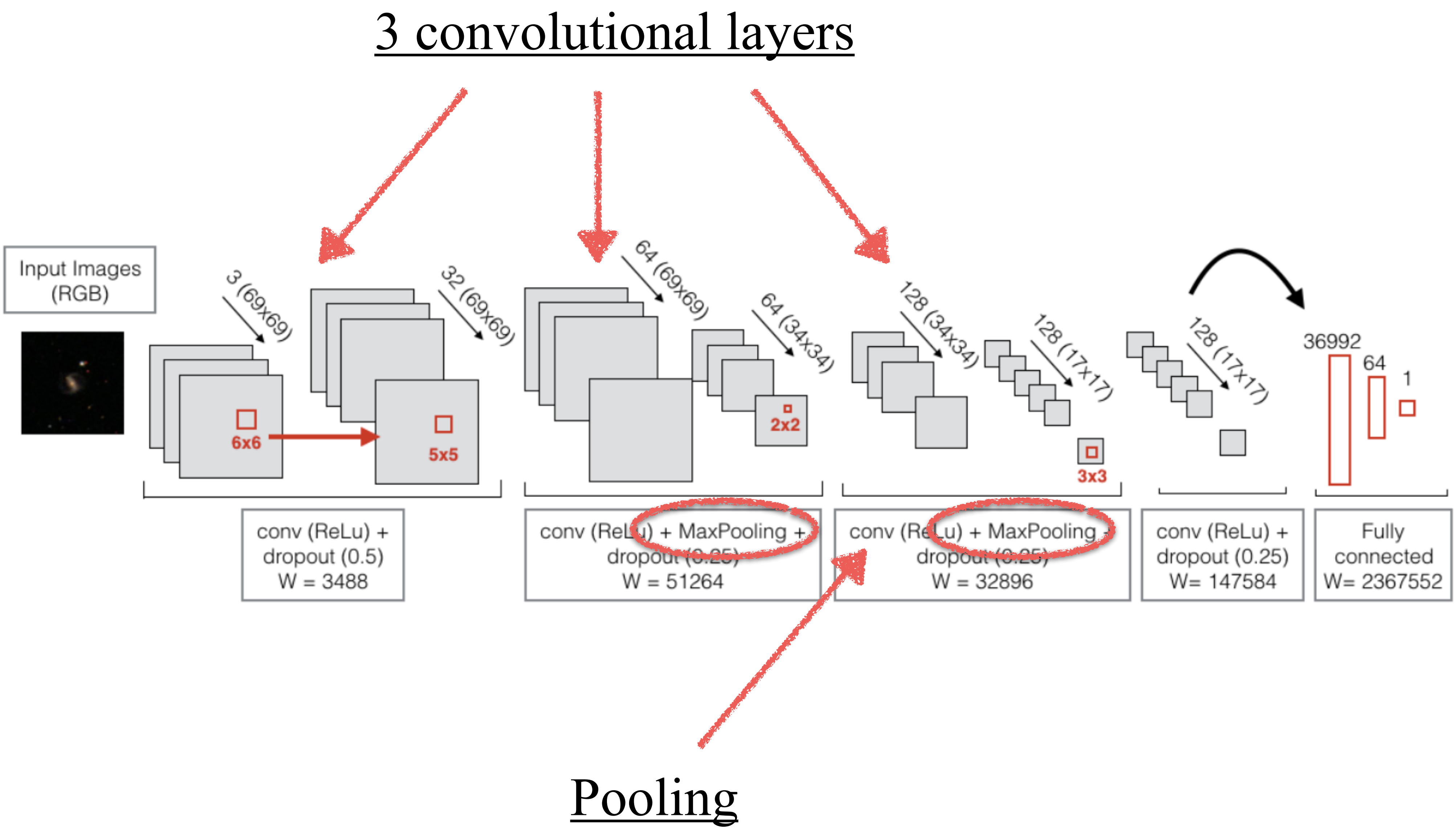
3 convolutional layers



ReLu activation

Dominguez-Sanchez+18

EXAMPLE OF VERY SIMPLE CNN

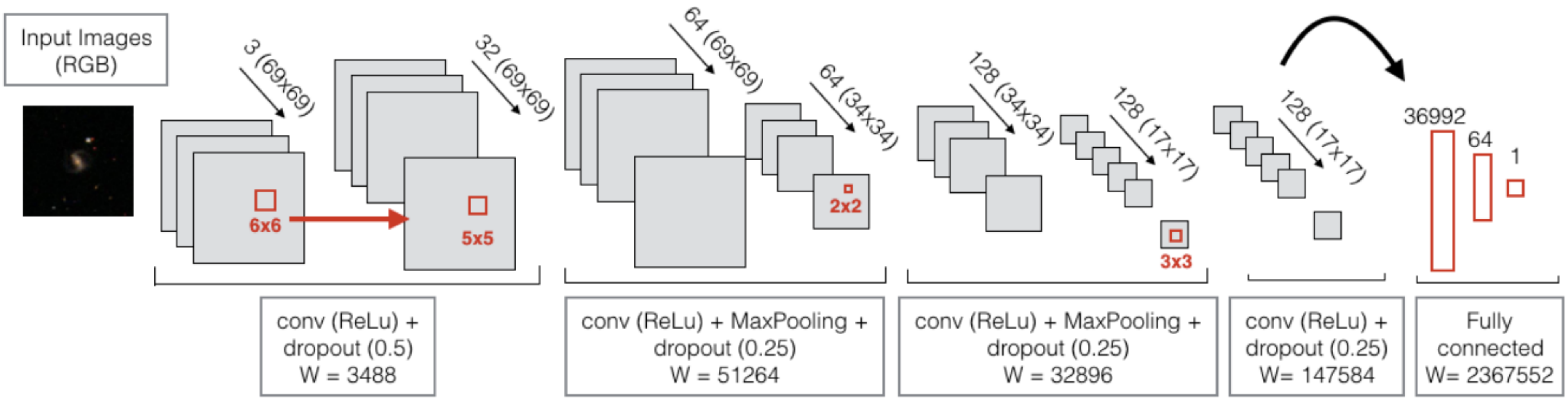


Dominguez-Sanchez+18

EXAMPLE OF VERY SIMPLE CNN

OVERALL:

- decrease of tensor size
- increase of depth



Dominguez-Sanchez+18

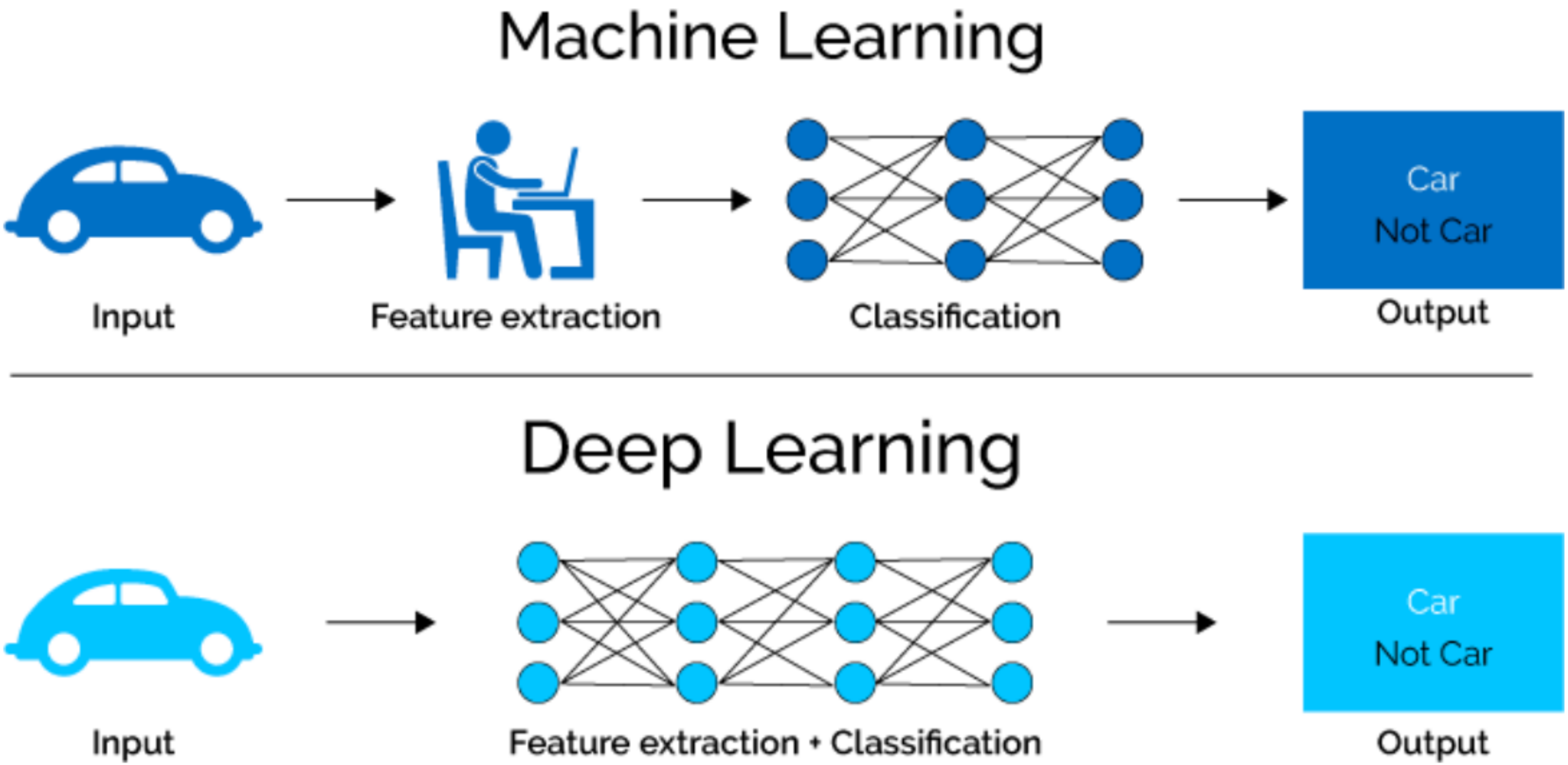
DEEP LEARNING

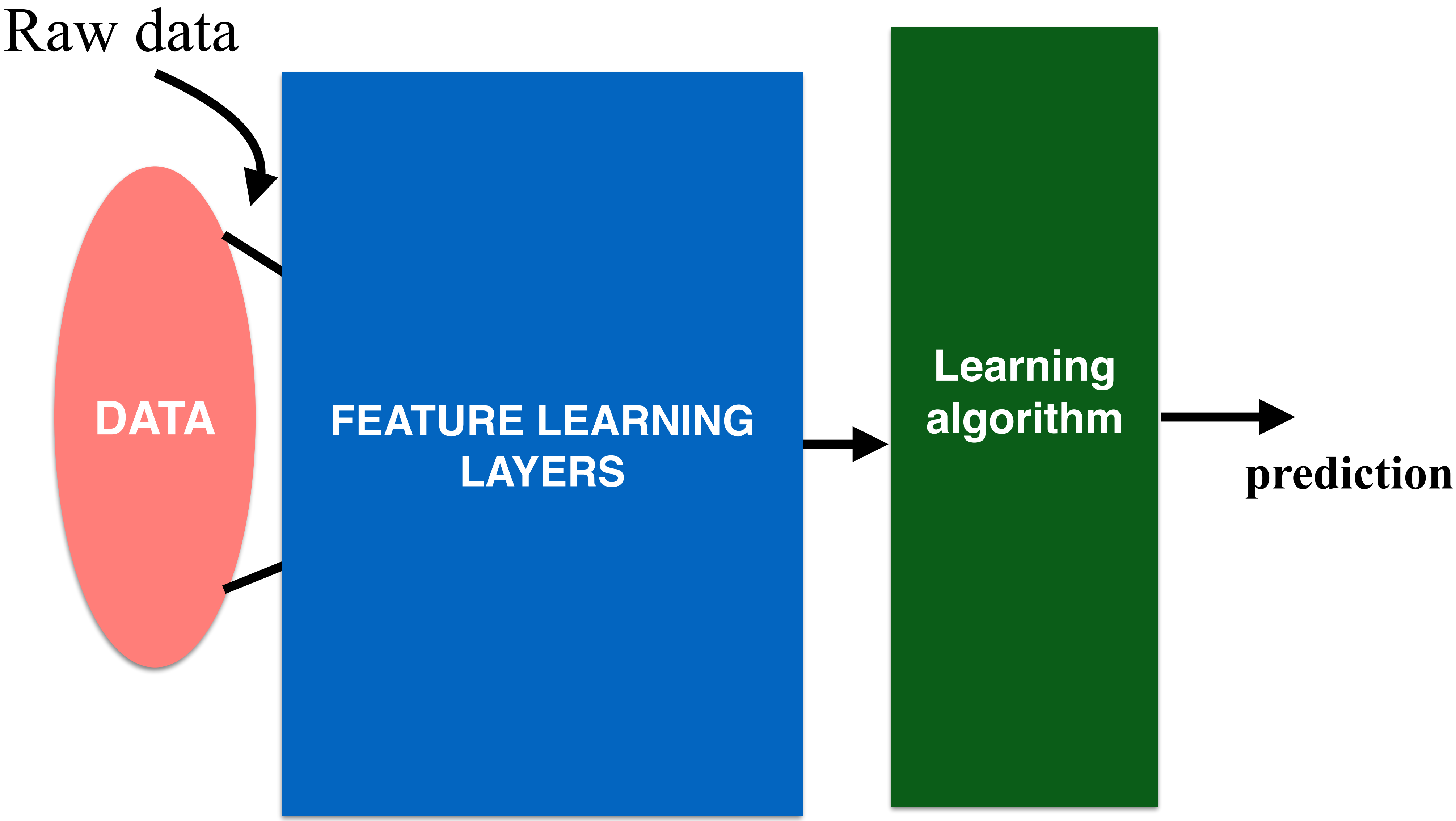
$$p_w(\theta|X)$$

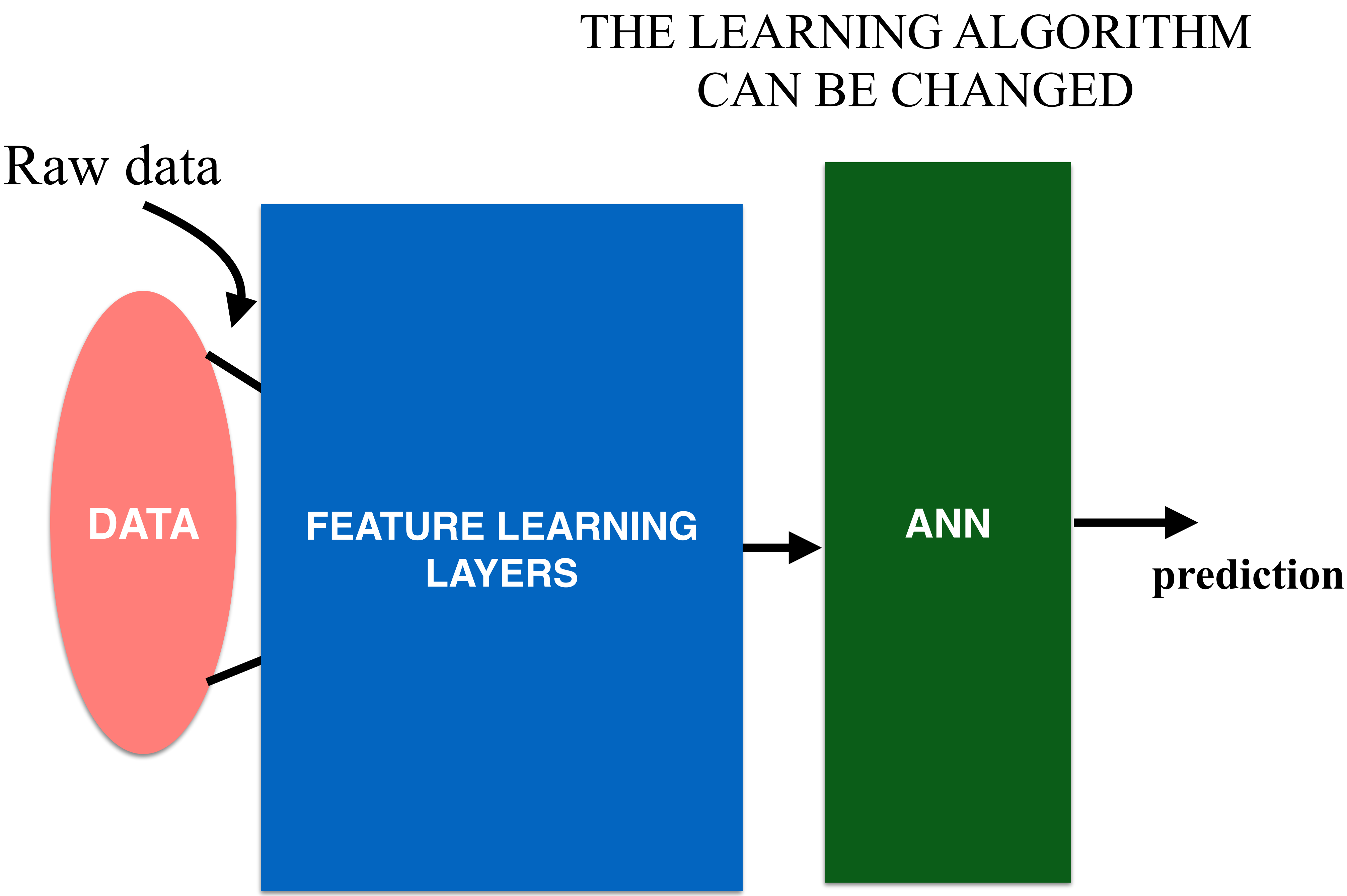
LET THE NETWORK FIGURE THIS OUT ("<u>unsupervised feature extraction</u>")

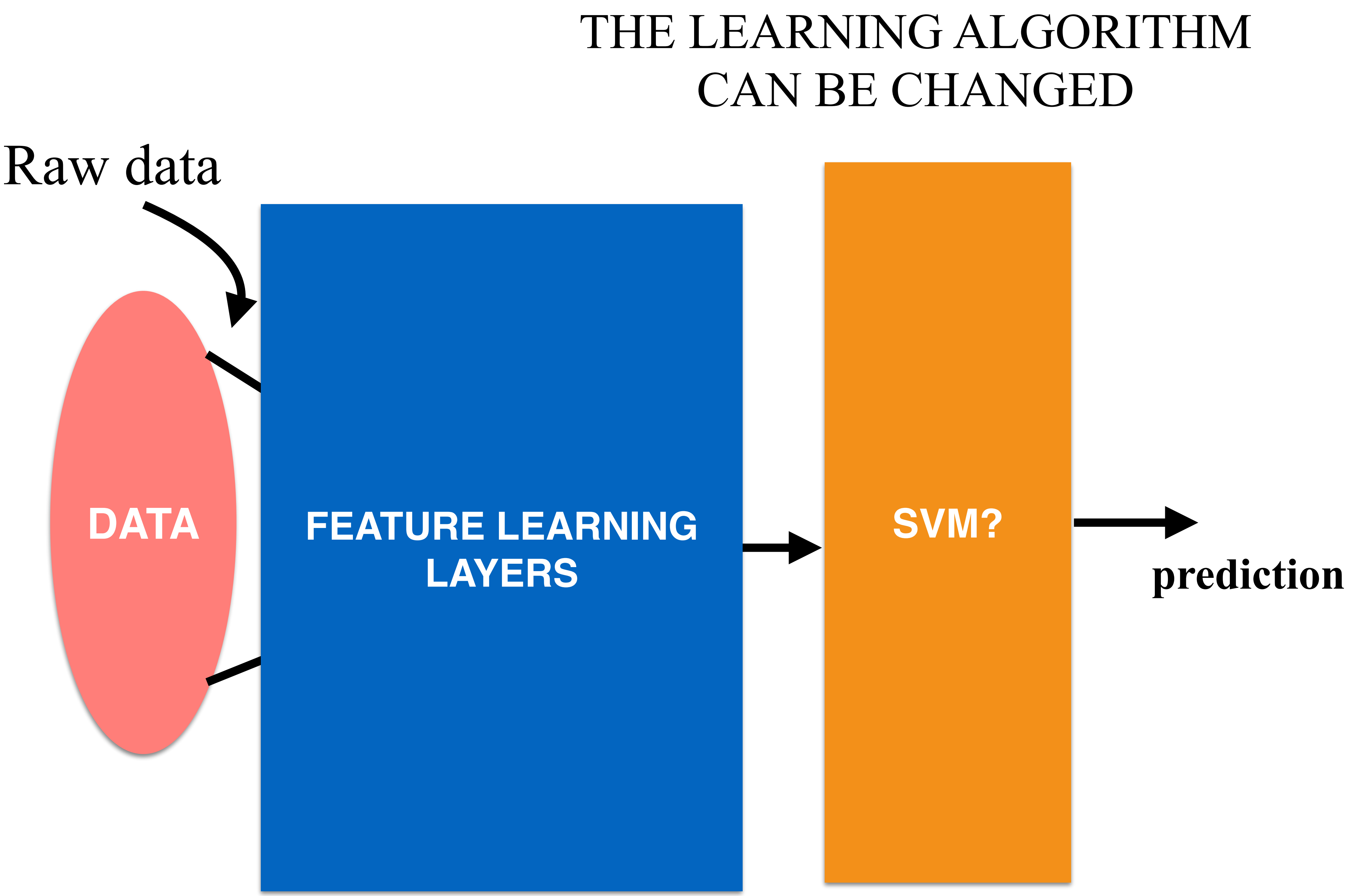
LET'S GO A STEP FORWARD INTO LOOSING CONTROL….

THIS IS A CHANGE OF PARADIGM!

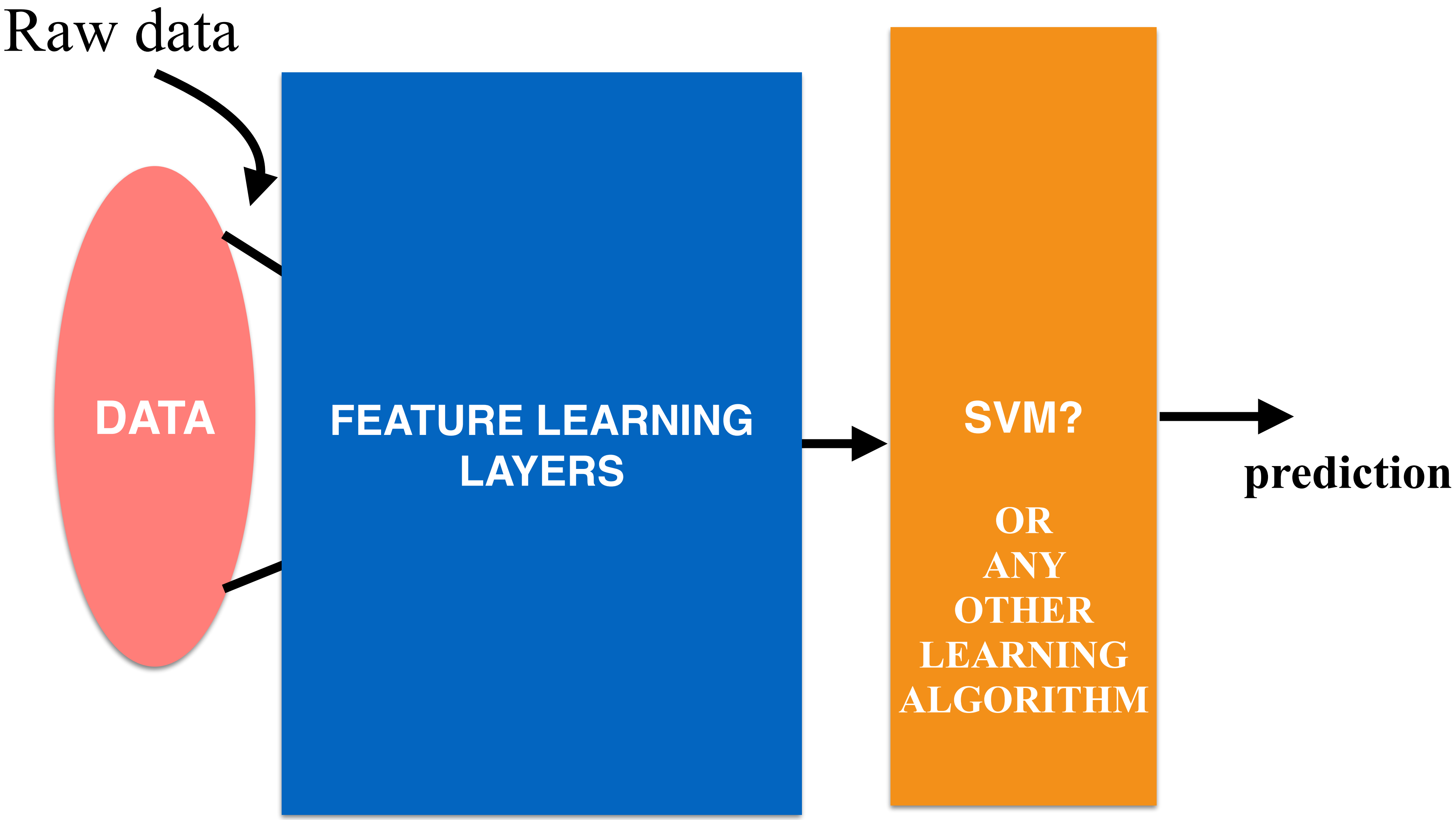



Raw data
DATA
FEATURE LEARNING LAYERS
Learning algorithm
prediction

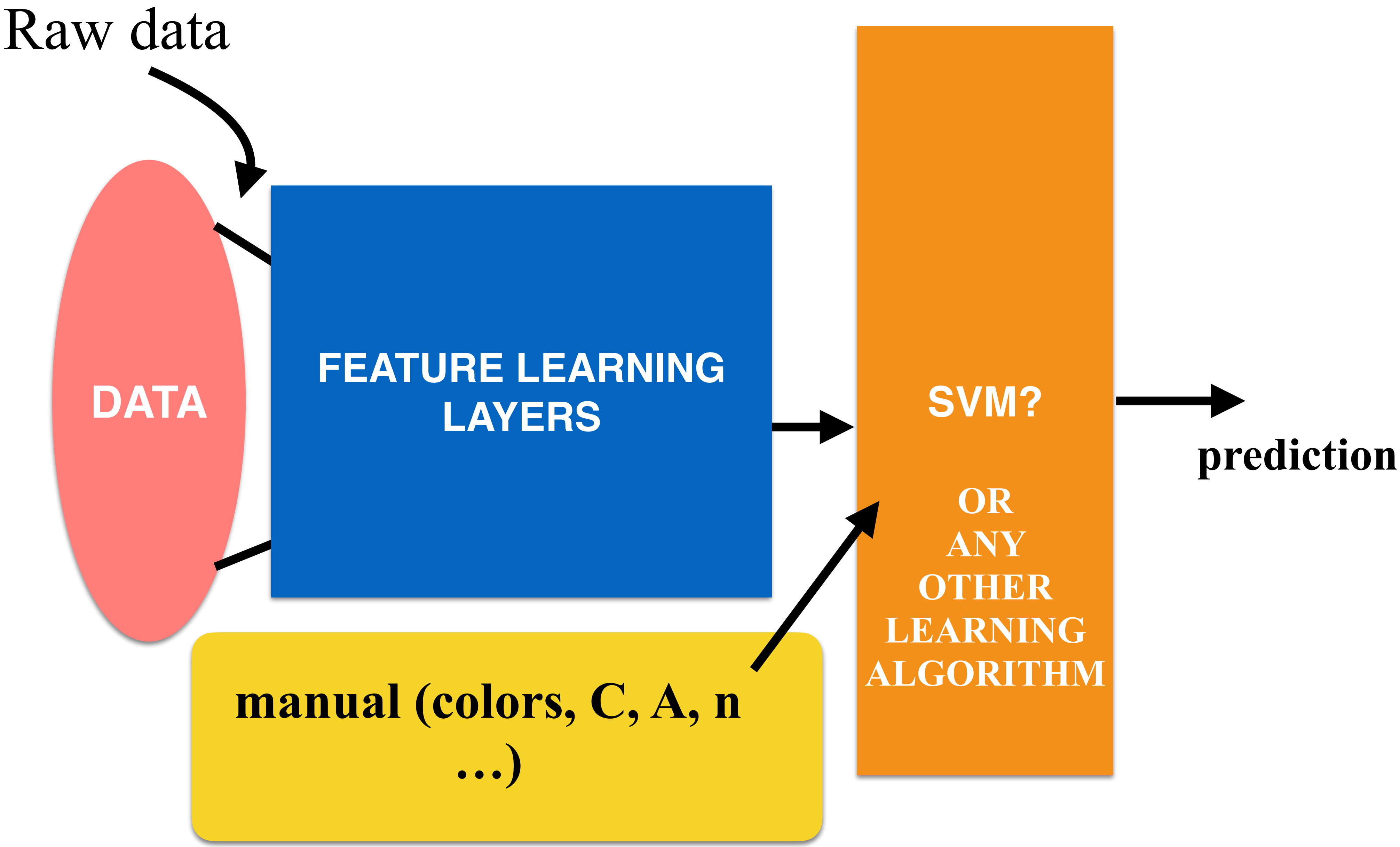

THE LEARNING ALGORITHM
CAN BE CHANGED
Raw data
DATA
FEATURE LEARNING
LAYERS
ANN
prediction

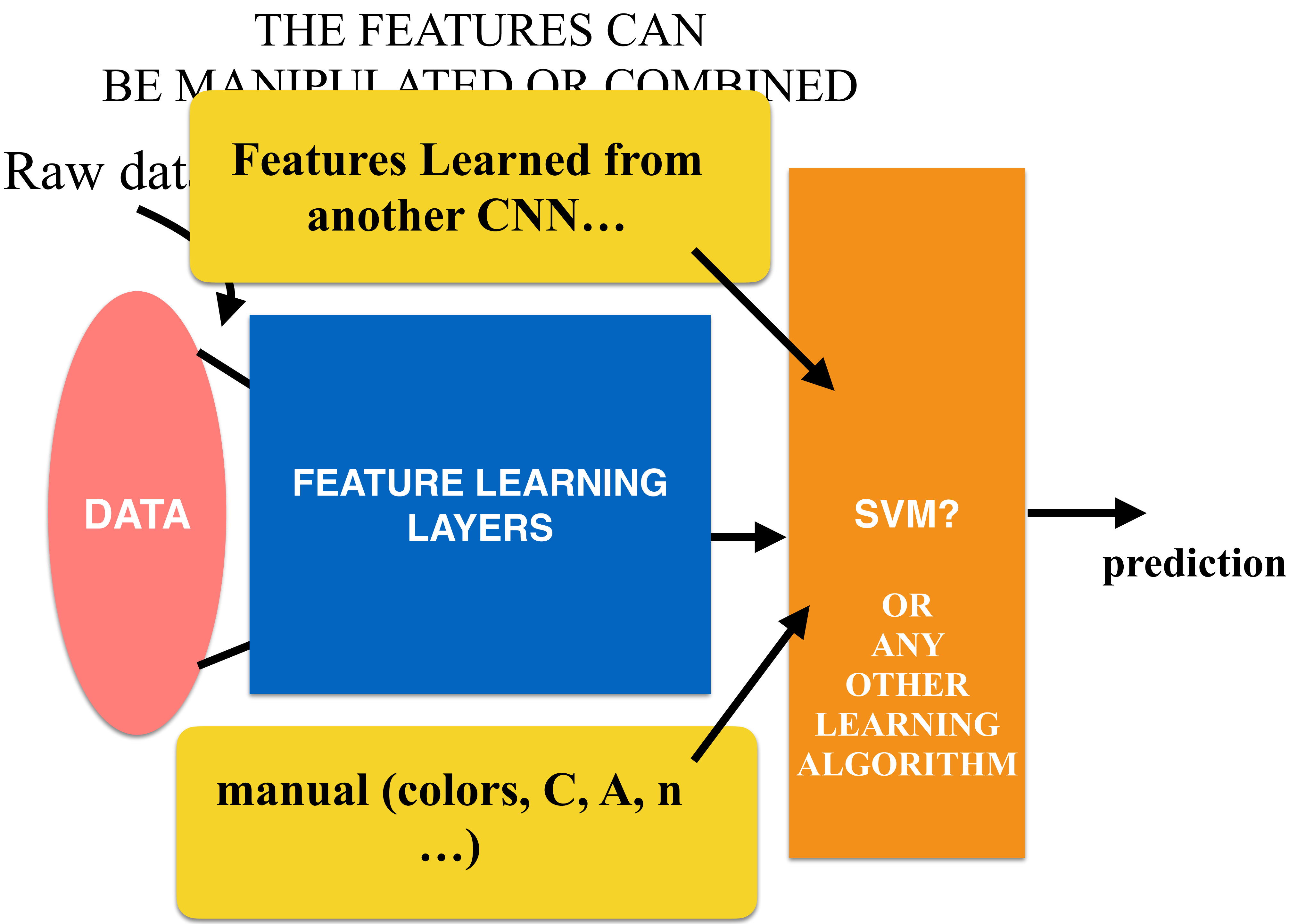

THE LEARNING ALGORITHM
CAN BE CHANGED
Raw data
DATA
FEATURE LEARNING
LAYERS
SVM?
prediction

THE LEARNING ALGORITHM CAN BE CHANGED

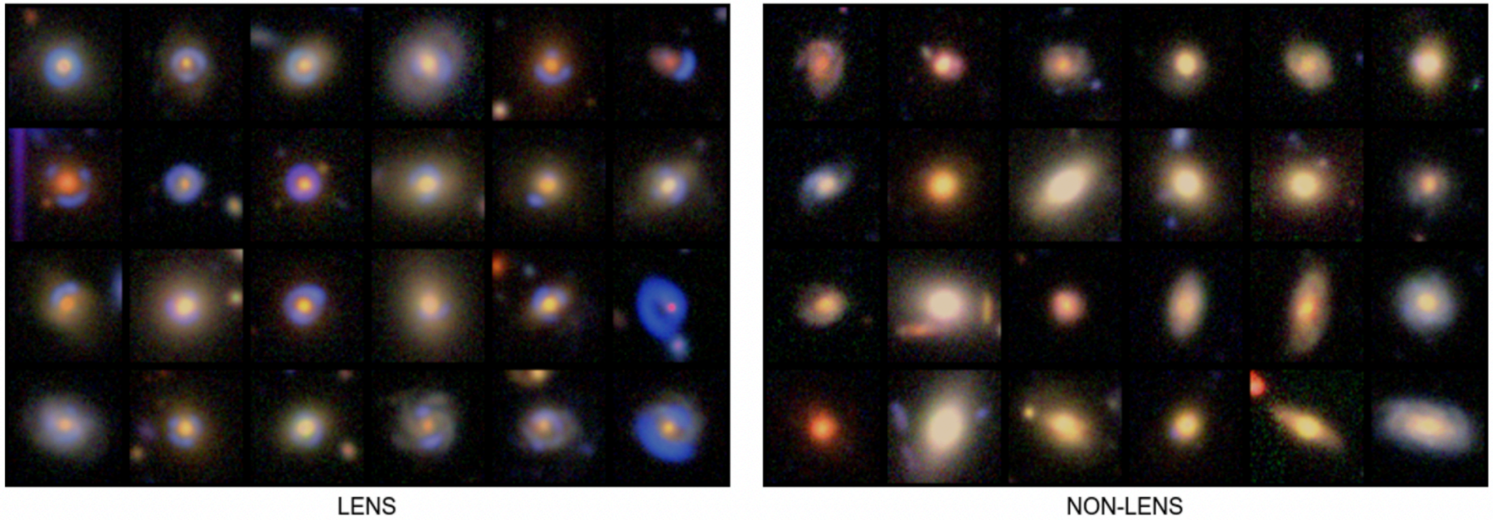


THE FEATURES CAN
BE MANIPULATED OR COMBINED




THE FEATURES CAN
BE MANIPULATED OR COMBINED
Raw dat
Features Learned from another CNN…
DATA
FEATURE LEARNING LAYERS
SVM?
OR ANY OTHER LEARNING ALGORITHM
prediction
manual (colors, C, A, n …)

1. Classification

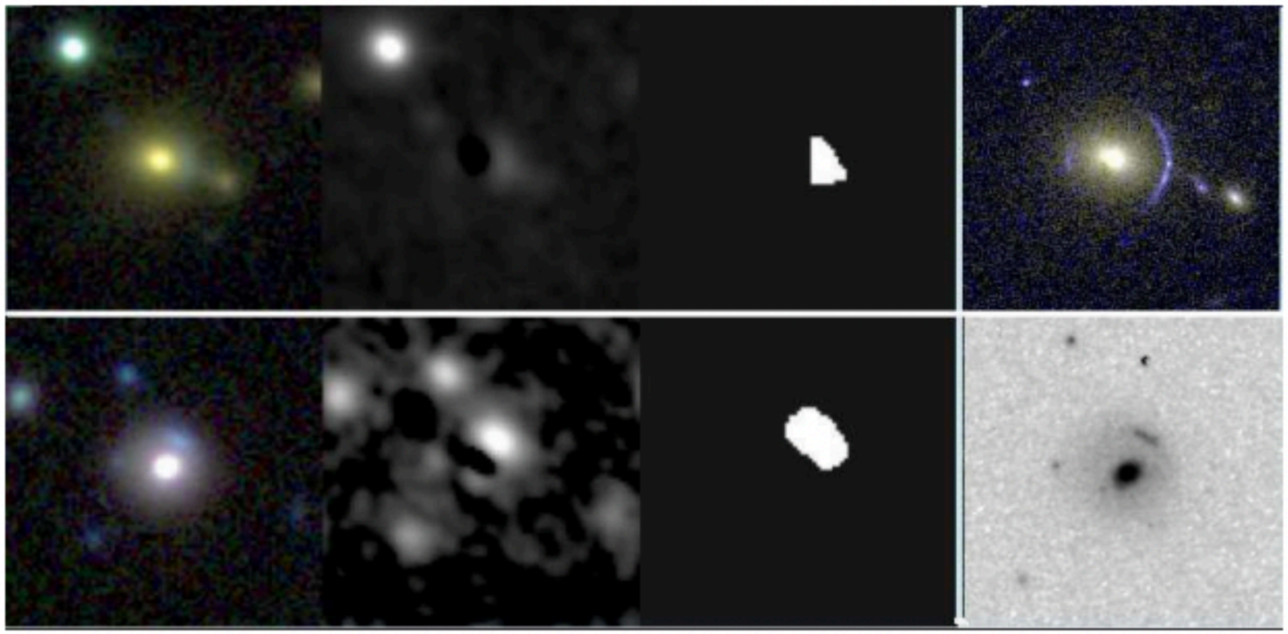


Detection of Strong Lenses
Valuable information of
Dark Matter properties

Future surveys will increase the samples by orders of magnitude.

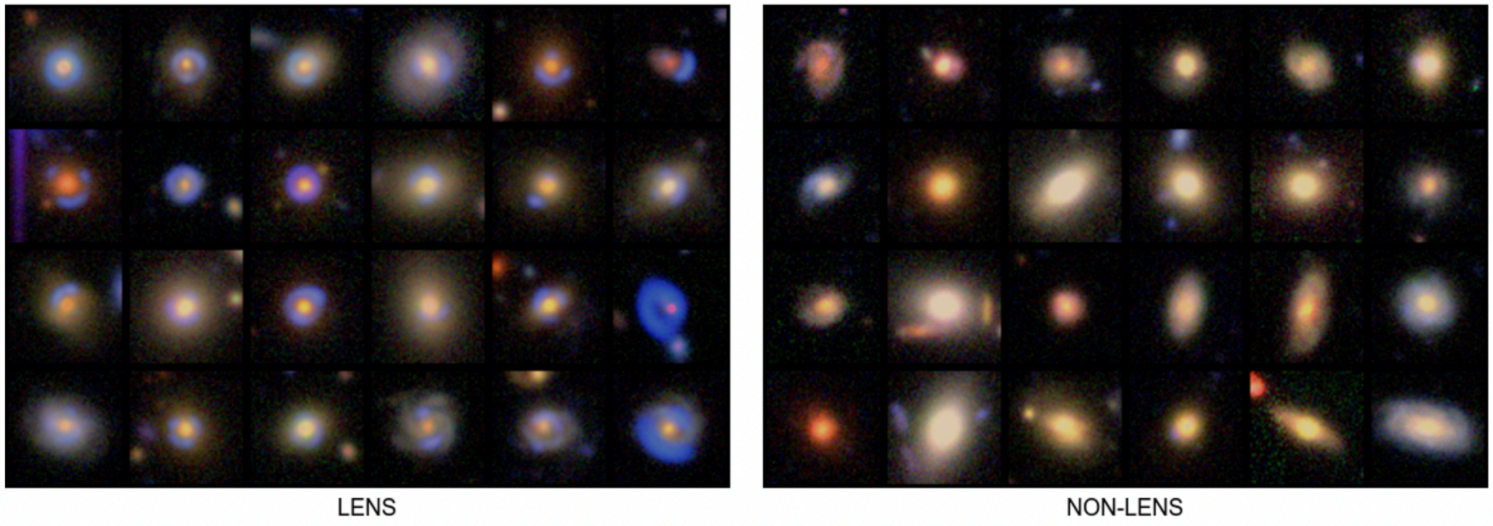
Jacobs+17

"Pre Deep Learning" Approach

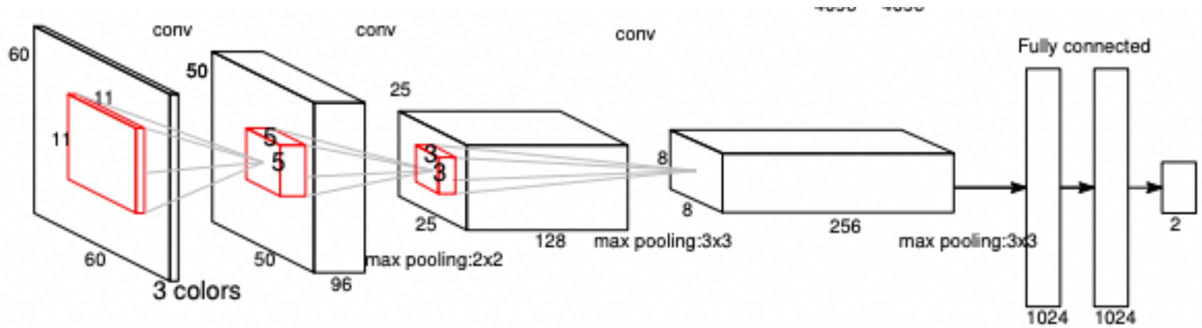


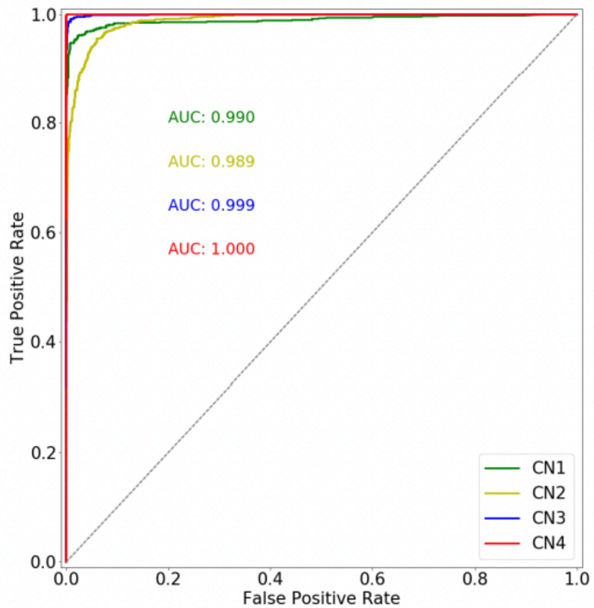
Gavazzi+17

1. Classification



Jacobs+17





It illustrates the change of paradigm from an algorithmic centric focus to a purely data driven approach to data

Dieleman+15, Huertas-Company+15, Aniyan+17, Charnock+17, Gieseke+17, Jacobs+17, Petrillo+17, Schawinski+17, Alhassan+18, Dominguez-Sanchez+18, George+18, Hinners+18, Lukic+18, Moss+18, Razzano+18, Schaefer+18, Allen+19, Burke+19, Carrasco-Davis+19, Chatterjee+19, Davies+19, Dominguez-Sanchez+19, Fusse+19, Glaser+19, Ishida+19, Jacobs+19, Katebi+19, Lanusse+19Liu+19, Lukic+19, Metcalf+19, Muthukrishna+19, Petrillo+19, Reiman+19, Boucaud+20, Chiani+20, Ghosh+20, Gomez+20, Hausen+20, Hlozek+20, Hosseinzadeh+20. Huang+20, Li+20, Moller+20, Paillassa+20, Tadaki+20, Vargas dos Santos+20, Walmsley+20, Wei+20, Allam+21, Arcelin+21, Becker+21, Bretonniere+21, Burhanudin+21, Davison+21, Donoso-Oliva+21, Jia+21, Lauritsen+21, Ono+21, Ruan+21, Sadegho+21, Tang+21, Tanoglidis+21, Vojtekova+21, Dhar+22Hausen+22, Orwat+22, Pimentel+22, Rezaei+22, Samudre+22, Shen+22, Walmsley+22

1. Classification

CNN based classifications reach unprecedented accuracy

Name	type	AUROC	TPR_0	TPR_{10}	short description
CMU-DeepLens-Resnet-ground3	Ground-Based	0.98	0.09	0.45	CNN
CMU-DeepLens-Resnet-Voting	Ground-Based	0.98	0.02	0.10	CNN
LASTRO EPFL	Ground-Based	0.97	0.07	0.11	CNN
CAS Swinburne Melb	Ground-Based	0.96	0.02	0.08	CNN
AstrOmatic	Ground-Based	0.96	0.00	0.01	CNN
Manchester SVM	Ground-Based	0.93	0.22	0.35	SVM / Gabor
Manchester2	Ground-Based	0.89	0.00	0.01	Human Inspection
ALL-star	Ground-Based	0.84	0.01	0.02	edges/gradiants and Logistic Reg.
CAST	Ground-Based	0.83	0.00	0.00	CNN / SVM
YattaLensLite	Ground-Based	0.82	0.00	0.00	SExtractor
LASTRO EPFL	Space-Based	0.93	0.00	0.08	CNN
CMU-DeepLens-Resnet	Space-Based	0.92	0.22	0.29	CNN
GAMOCLASS	Space-Based	0.92	0.07	0.36	CNN
CMU-DeepLens-Resnet-Voting	Space-Based	0.91	0.00	0.01	CNN
AstrOmatic	Space-Based	0.91	0.00	0.01	CNN
CMU-DeepLens-Resnet-aug	Space-Based	0.91	0.00	0.00	CNN
Kapteyn Resnet	Space-Based	0.82	0.00	0.00	CNN
CAST	Space-Based	0.81	0.07	0.12	CNN
Manchester1	Space-Based	0.81	0.01	0.17	Human Inspection
Manchester SVM	Space-Based	0.81	0.03	0.08	SVM / Gabor
NeuralNet2	Space-Based	0.76	0.00	0.00	CNN / wavelets
YattaLensLite	Space-Based	0.76	0.00	0.00	Arcs / SExtractor
All-now	Space-Based	0.73	0.05	0.07	edges/gradiants and Logistic Reg.
GAHEC IRAP	Space-Based	0.66	0.00	0.01	arc finder

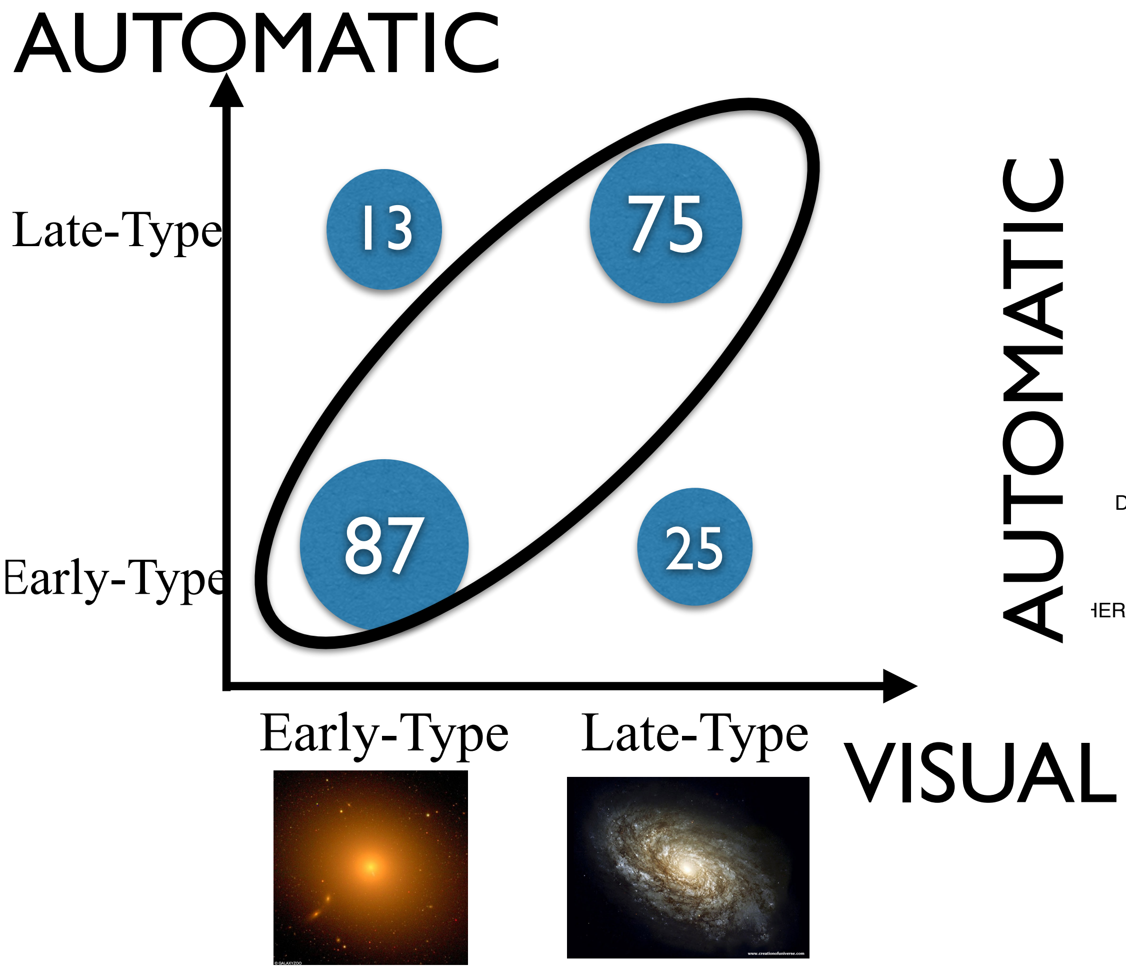
Metcalf+19

Dieleman+15, Huertas-Company+15, Aniyan+17, Charnock+17, Gieseke+17, Jacobs+17, Petrillo+17, Schawinski+17, Alhassan+18, Dominguez-Sanchez+18, George+18, Hinners+18, Lukic+18, Moss+18, Razzano+18, Schaefer+18, Allen+19, Burke+19, Carrasco-Davis+19, Chatterjee+19, Davies+19, Dominguez-Sanchez+19, Fusse+19, Glaser+19, Ishida+19, Jacobs+19, Katebi+19, Lanusse+19Liu+19, Lukic+19, Metcalf+19, Muthukrishna+19, Petrillo+19, Reiman+19, Boucaud+20, Chiani+20, Ghosh+20, Gomez+20, Hausen+20, Hlozek+20, Hosseinzadeh+20. Huang+20, Li+20, Moller+20, Paillassa+20, Tadaki+20, Vargas dos Santos+20, Walmsley+20, Wei+20, Allam+21, Arcelin+21, Becker+21, Bretonniere+21, Burhanudin+21, Davison+21, Donoso-Oliva+21, Jia+21, Lauritsen+21, Ono+21, Ruan+21, Sadegho+21, Tang+21, Tanoglidis+21, Vojtekova+21, Dhar+22Hausen+22, Orwat+22, Pimentel+22, Rezaei+22, Samudre+22, Shen+22, Walmsley+22

ALSO FOR GALAXY MORPHOLOGY

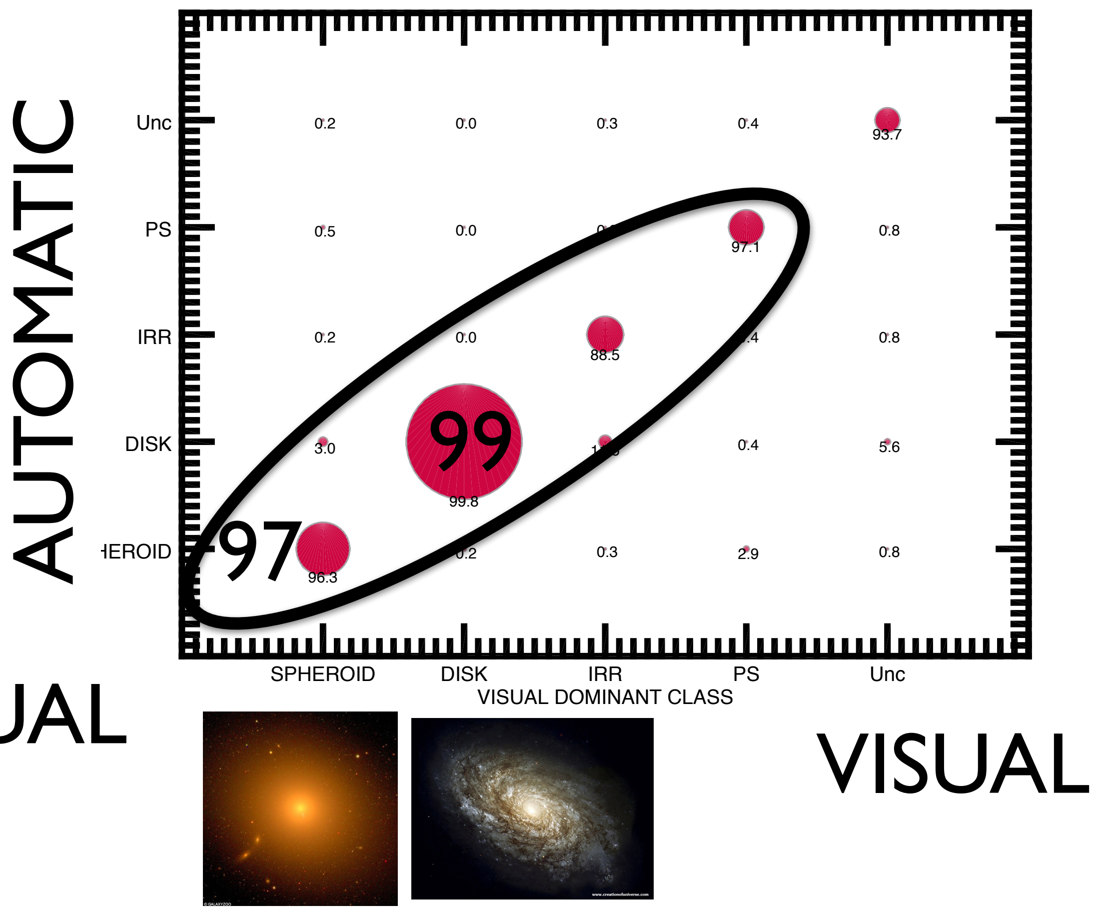
SVMs

[HUERTAS-COMPANY+14]



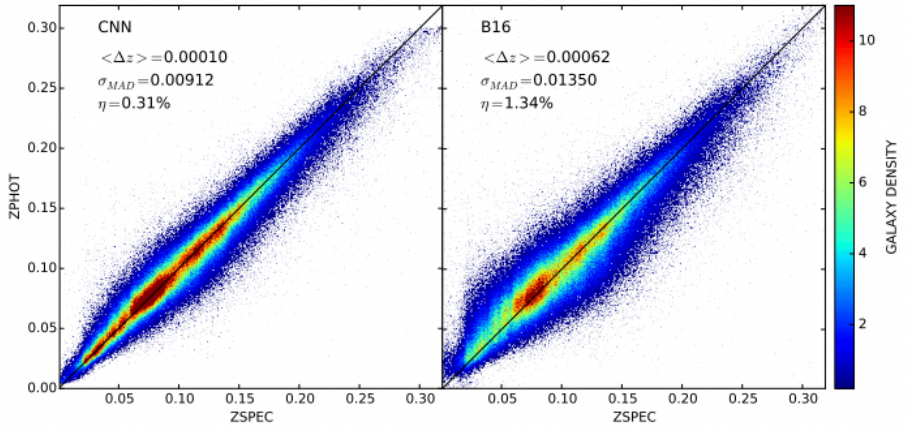
CNNs

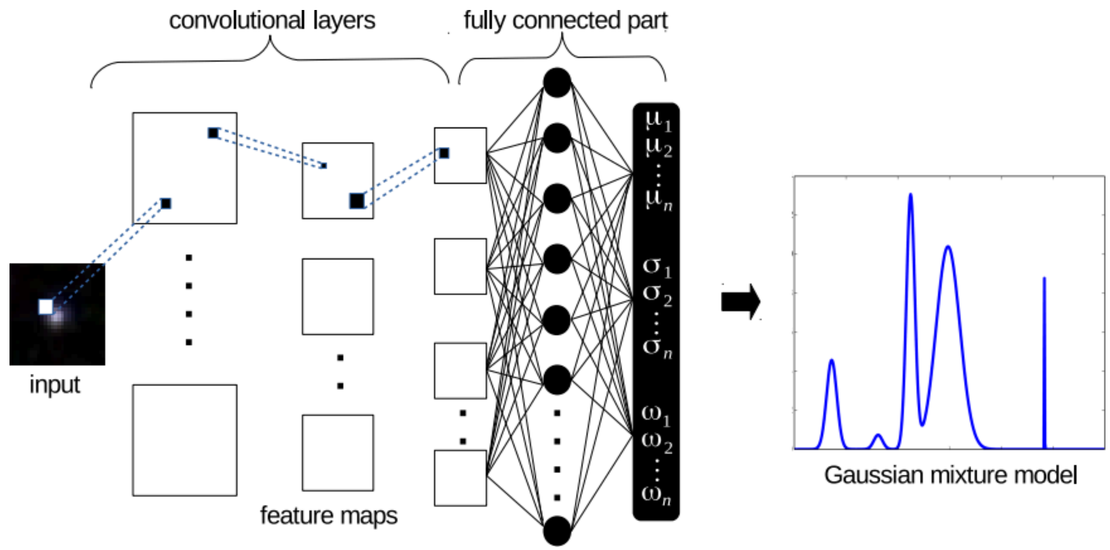
[HUERTAS-COMPANY+15b]



Photometric Redshifts

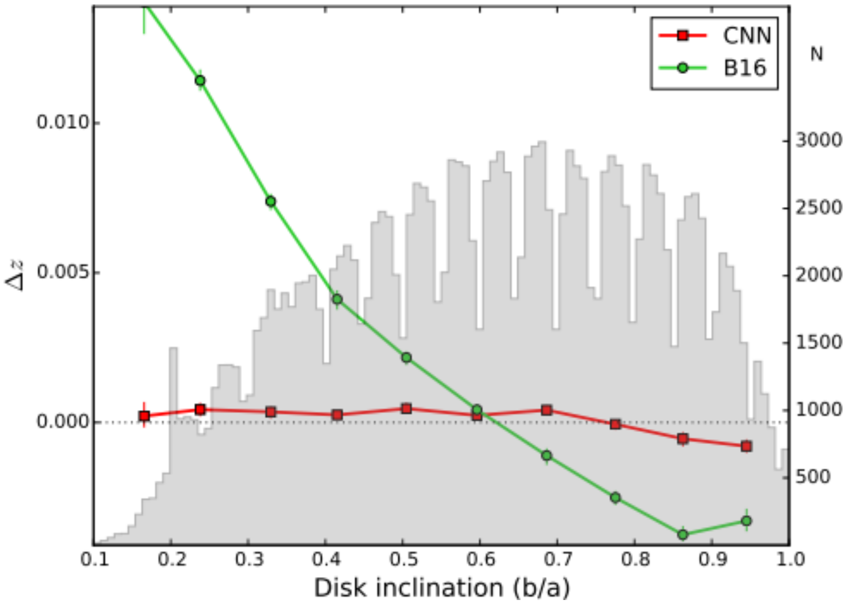
Deep Learning **Classical approach**





Disanto+18

Geometric Effects are automatically considered (beyond photometry)

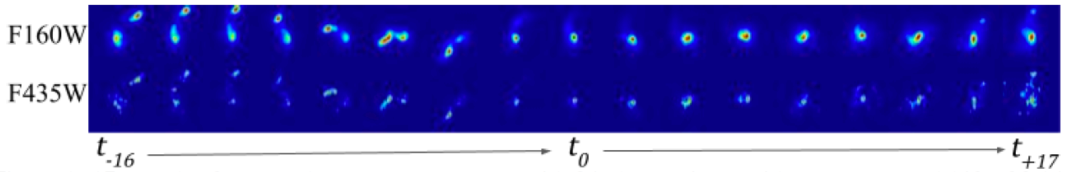


Pasquet+18

Uncertainty quantification through Mixture Density Networks

Hazevah+17, Perreault-Levasseur+17, Morningstar+18, Stark+18, Tuccillo+18, Bom+19, Lovell+19, Madireddy+19, Pearson+19, Simet+19, Pasquet+19, Wu+19, Menou+19, Aragon-Calvo+20, Chianese+20, Stahl+20, Surana+20, Shuntov+20, Cabayol-Garcia+20, Campagne+20, Buck+21, Grover+21, Hayat+21, Li+21, Maresca+21, Qiu+21, Rhea+21, Schuldt+21, Tohill+21, Yao-Yu+21, Dey+21, Lee+21, Zhou+21, Henghes+21, Ansari+21

Mergers of Galaxies

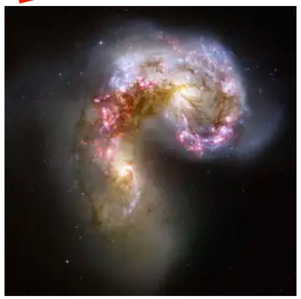


Merger of galaxies sequence from cosmological simulations

?

?

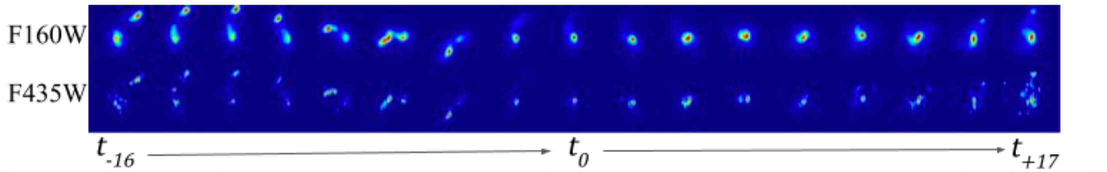
Merger stage?



Neural Networks to find relations between observables and physical processes

Ntampaka+16, Huertas-Company+18, Schawinski+18, Armitage+19, Bottrell+19, Calderon+19, Diaz+19, Ho+19, Pearson+19, Cai+20, Ferreira+20, Hansen+20, Kodi Ramanah+20, Wu+20, Yan+20, Gupta+20, de Andres+21, Bickley+21, Ciprijanovic+21, Ho+21, Koppula+21, von Marttens+21, Shao+21, Villanueva-Domingo+21, Whithney+21, Holwerda+21

Mergers of Galaxies

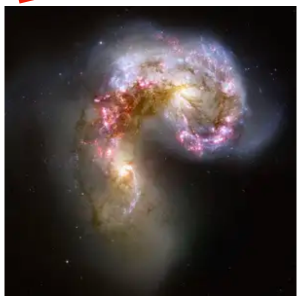


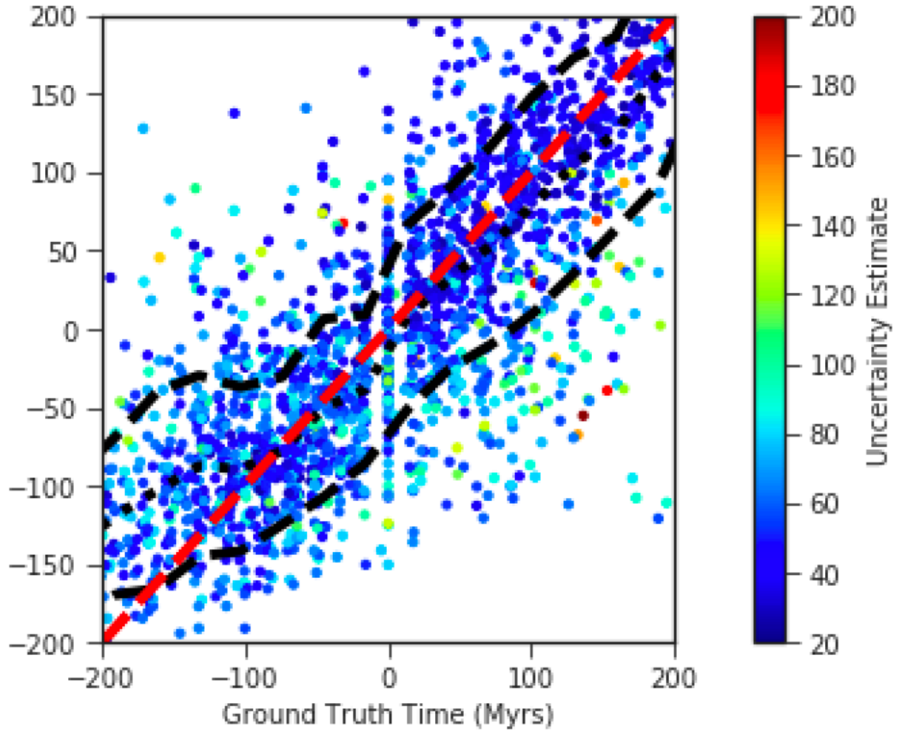
Merger of galaxies sequence from cosmological simulations

?

?

Merger stage?



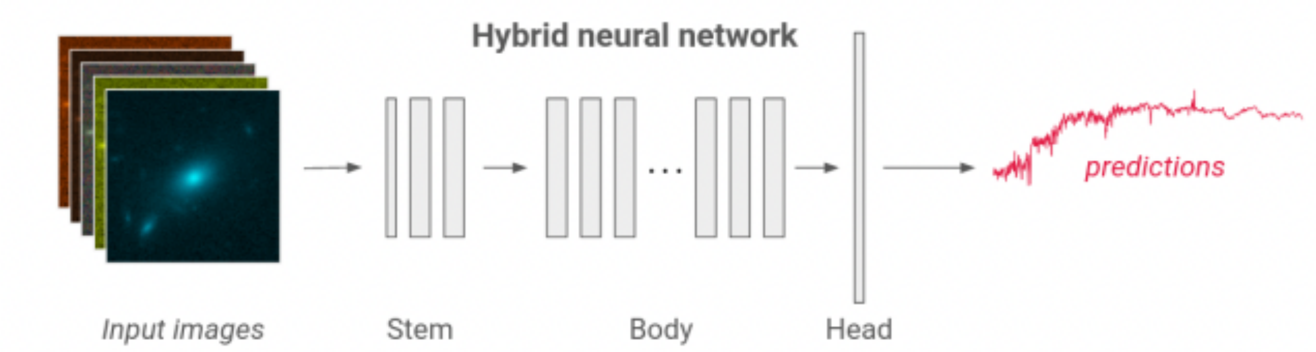


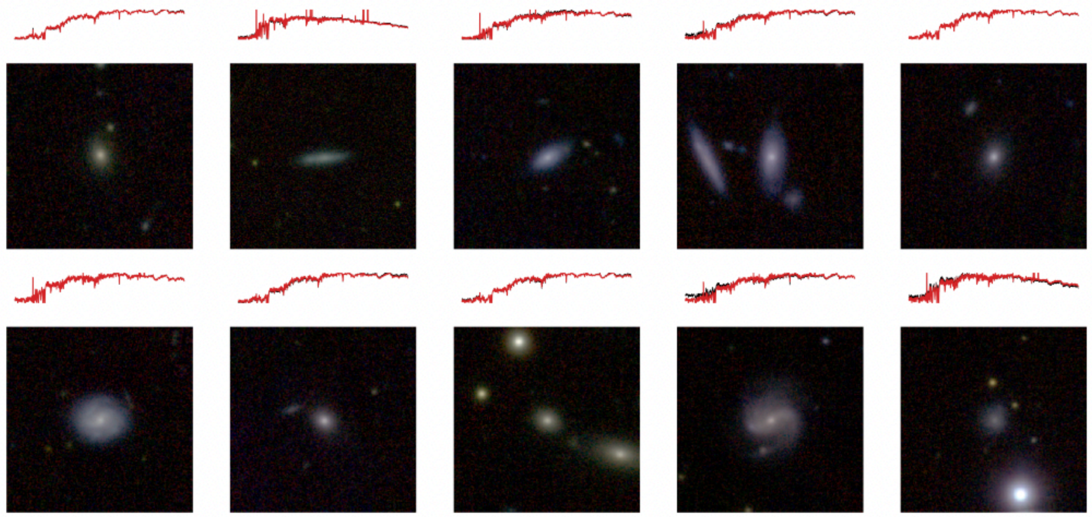
Koppula+21

Neural Networks to find relations between observables and physical processes

Ntampaka+16, Huertas-Company+18, Schawinski+18, Armitage+19, Bottrell+19, Calderon+19, Diaz+19, Ho+19, Pearson+19, Cai+20, Ferreira+20, Hansen+20, Kodi Ramanah+20, Wu+20, Yan+20, Gupta+20, de Andres+21, Bickley+21, Ciprijanovic+21, Ho+21, Koppula+21, von Marttens+21, Shao+21, Villanueva-Domingo+21, Whithney+21, Holwerda+21

Virtual Observatory



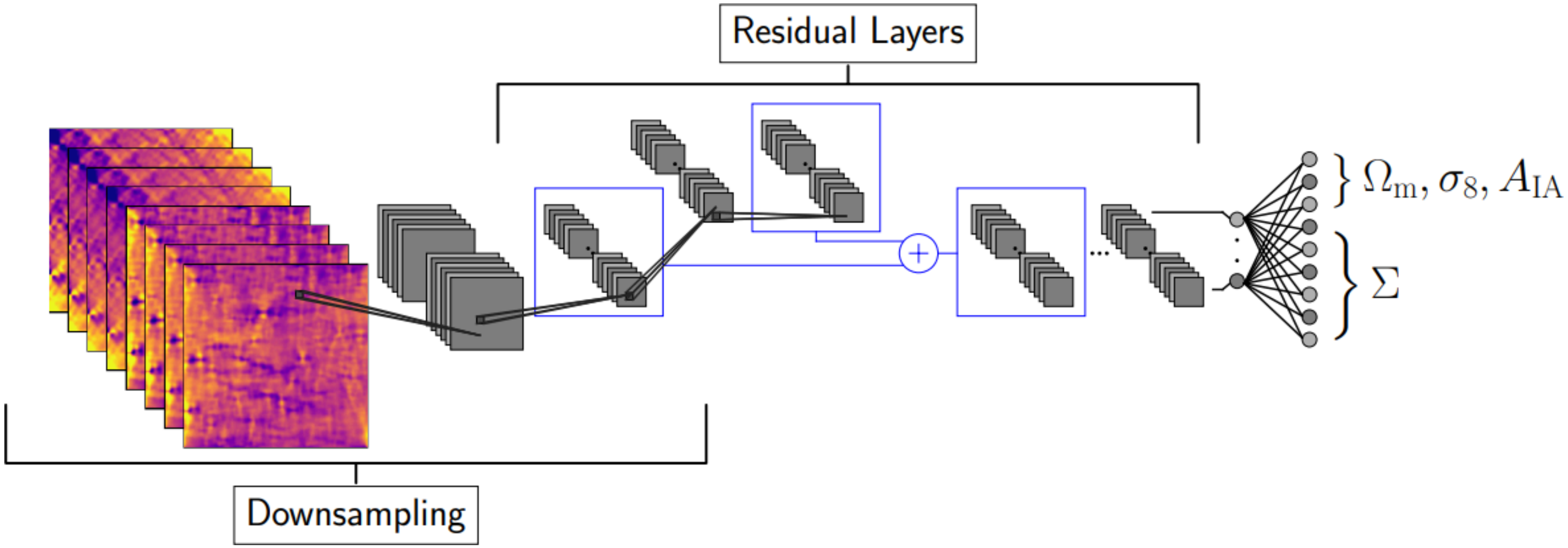


Neural Networks to learn complex mapping between observables

Wu+21

Ntampaka+16, Huertas-Company+18, Schawinski+18, Armitage+19, Bottrell+19, Calderon+19, Diaz+19, Ho+19, Pearson+19, Cai+20, Ferreira+20, Hansen+20, Kodi Ramanah+20, Wu+20, Yan+20, Gupta+20, de Andres+21, Bickley+21, Ciprijanovic+21, Ho+21, Koppula+21, von Marttens+21, Shao+21, Villanueva-Domingo+21, Whithney+21, Holwerda+21

Deep Learning for Cosmological Inference



Motivation: Generalize comparison of observations with theory, beyond basic summary statistics

Neural Networks are used as efficient feature extractors

Adding additional invariances

DATA AUGMENTATION

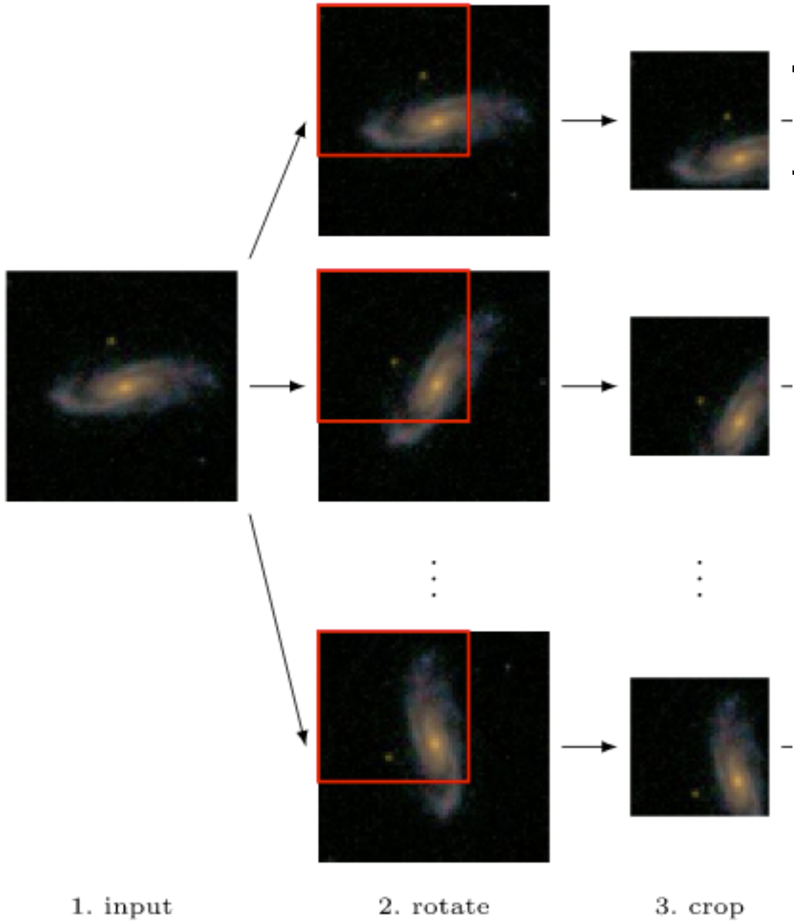
ANOTHER WAY TO REDUCE OVER-FITTING IS TO "AUGMENT" THE SIZE OF THE DATASET AVAILABLE FOR TRAINING

FOR MANY APPLICATIONS THE CLASSIFICATION SHOULD BE INDEPENDENT TO:

- TRANSALTIONS
- ROTATIONS
- SCALINGS

ETC…

DATA TATION



Dieleman+15

FOR MANY APPLICATIONS THE CLASSIFICATION SHOULD BE INDEPENDENT TO:

- TRANSALTIONS
- ROTATIONS
- SCALINGS

ETC…

DATA AUGMENTATION

FOR MANY APPLICATIONS THE CLASSIFICATION SHOULD BE INDEPENDENT TO:

- TRANSALTIONS
- ROTATIONS
- SCALINGS

ETC…