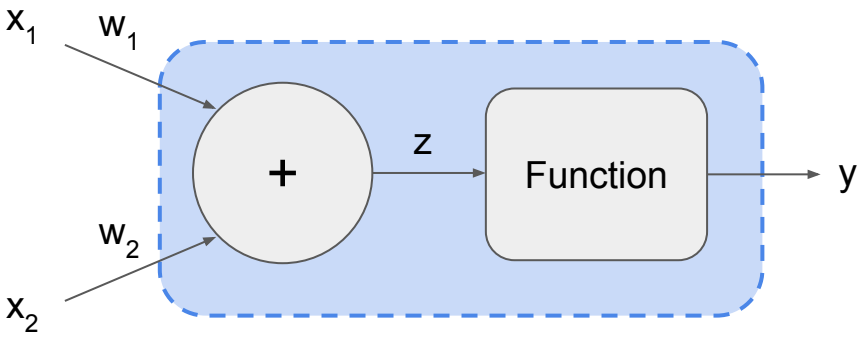
Neural Networks Introduction

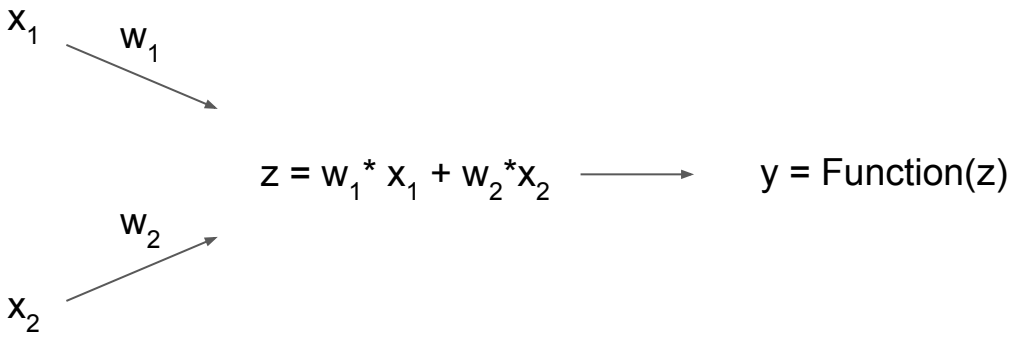
Gabriele Pozzati - gabriele.pozzati@scilifelab.se

Stockholm University

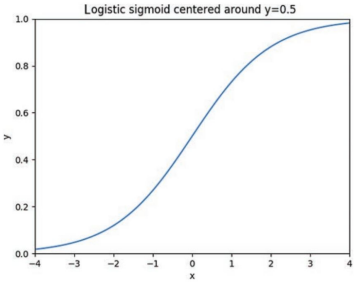
- Neuron

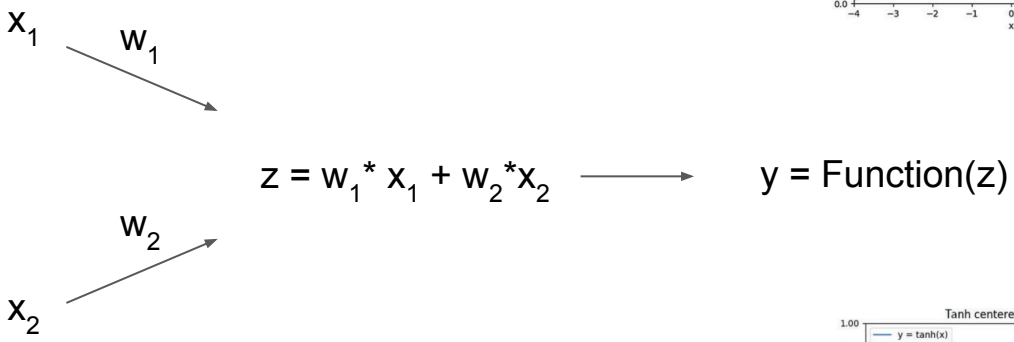


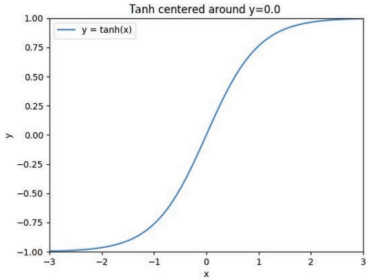
- Neuron



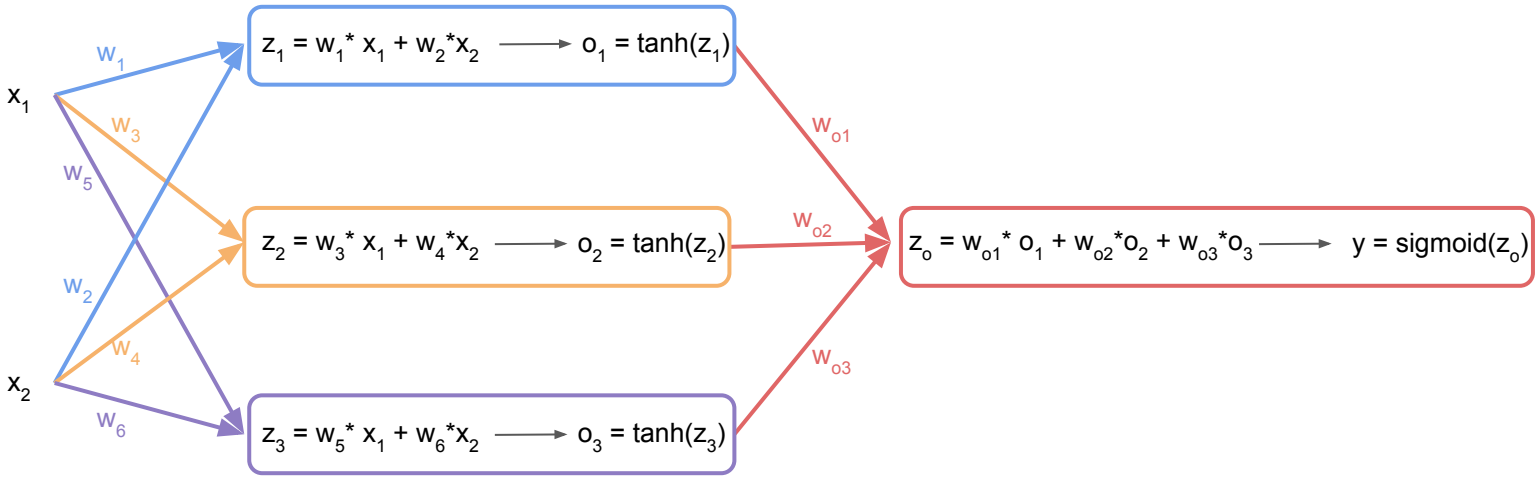
- Neuron


Logistic sigmoid centered around y=0.5
y
x

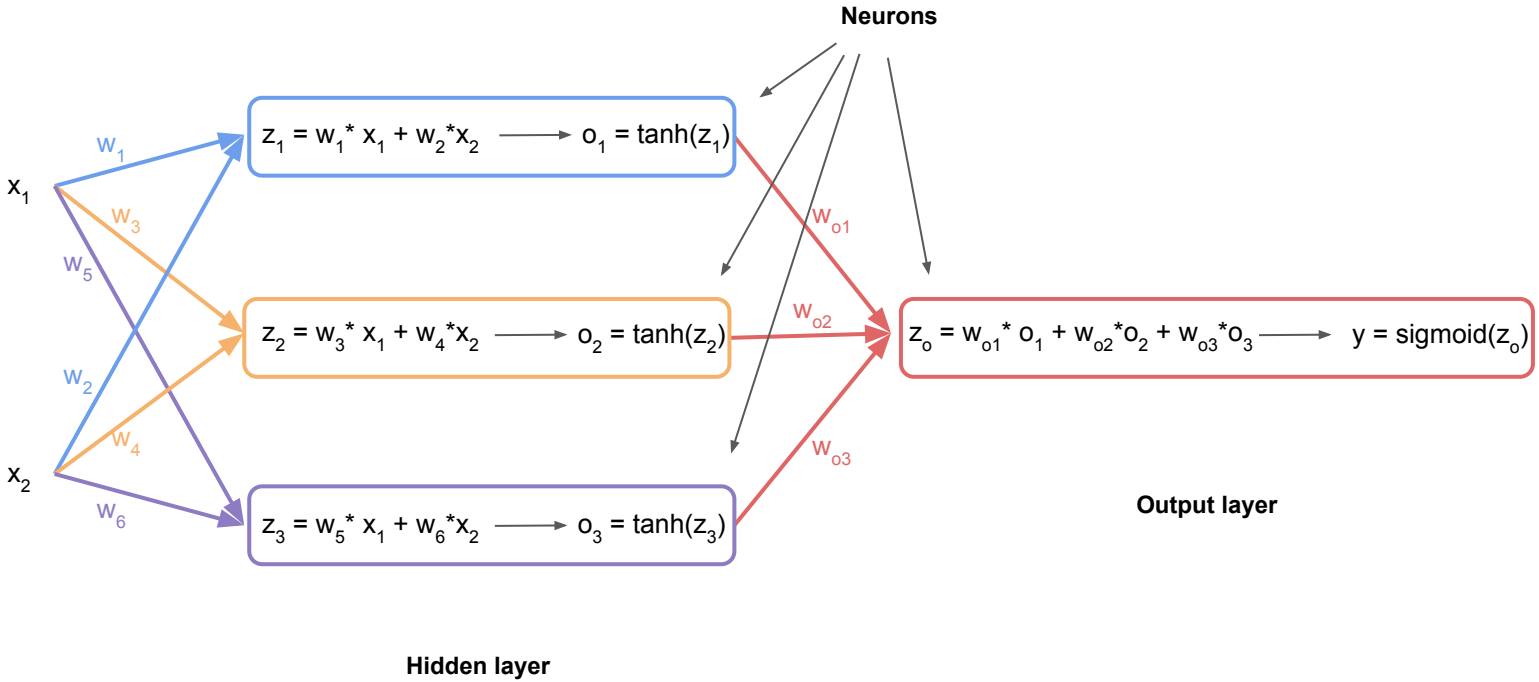

x1
w1
z = w1* x1 + w2*x2
y = Function(z)
w2
x2


Tanh centered around y=0.0
y = tanh(x)
y
x

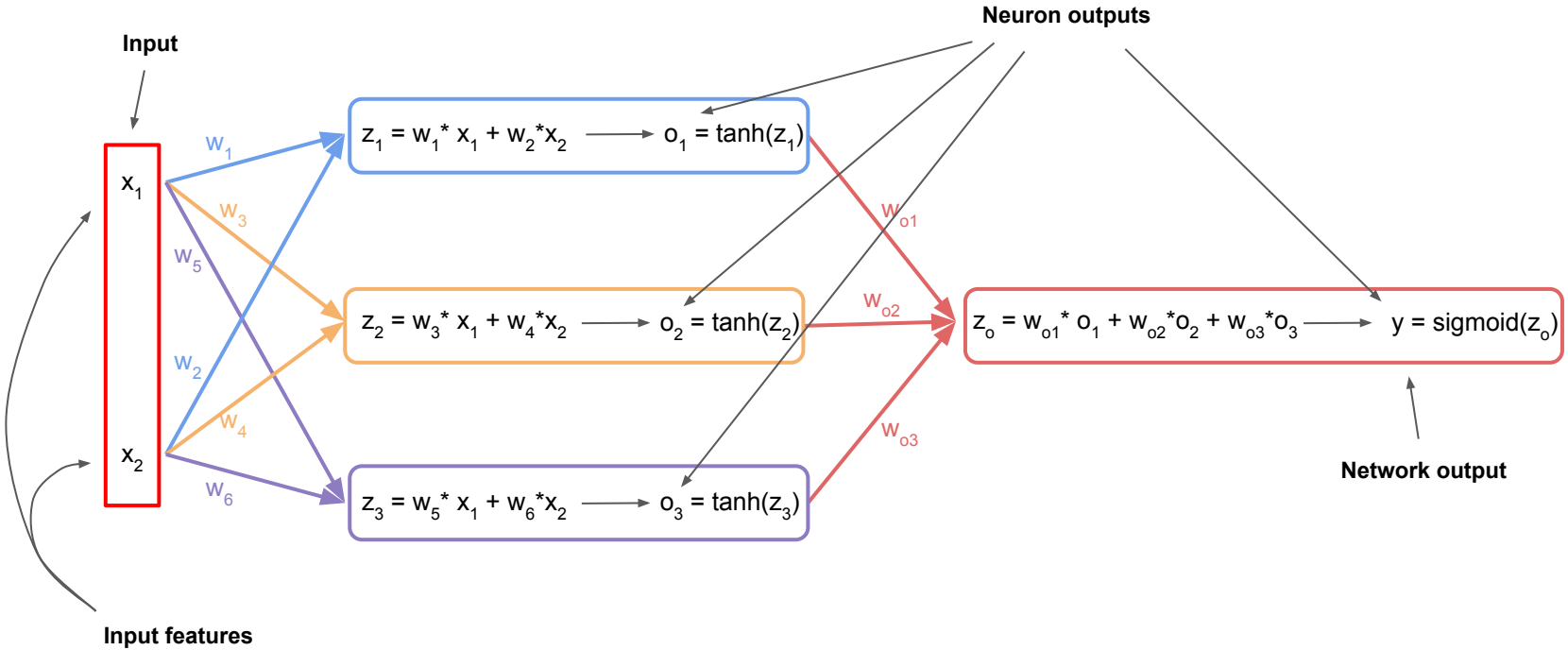
- Neural Network



- Neural Network



- Neural Network

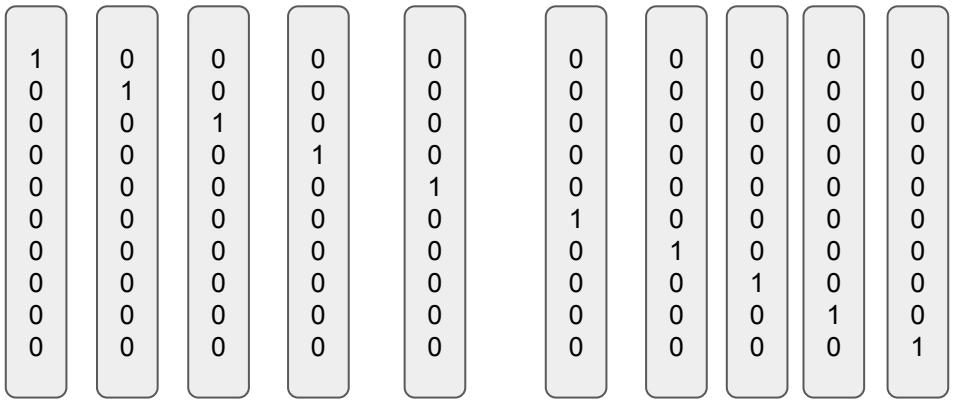


- Input types

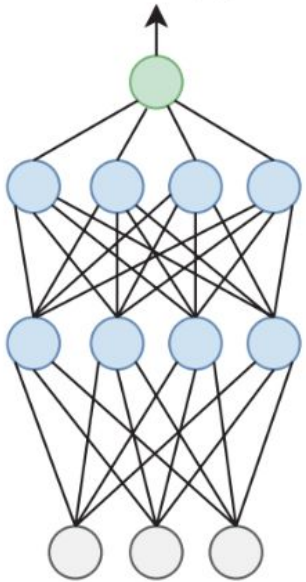
Letters/Text? ⟶ One-hot encoding

Input features: words from a set forming a sentence
Output labels: correct/incorrect sentence

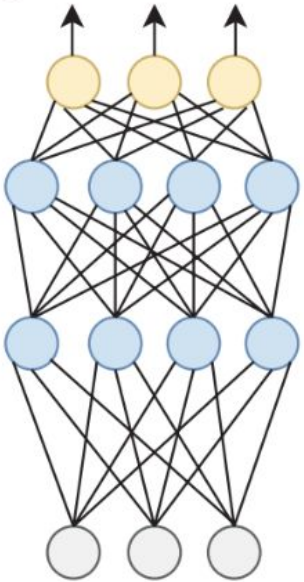
you, he, dog, cat, friend, have, is, are, a, and;



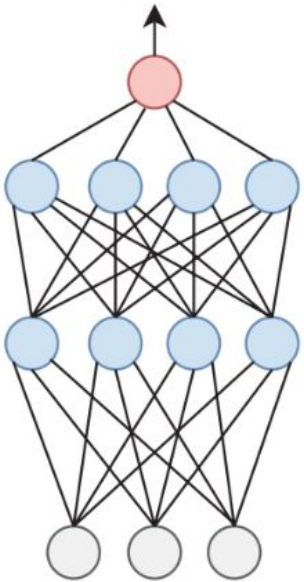
- Output types



Binary classification

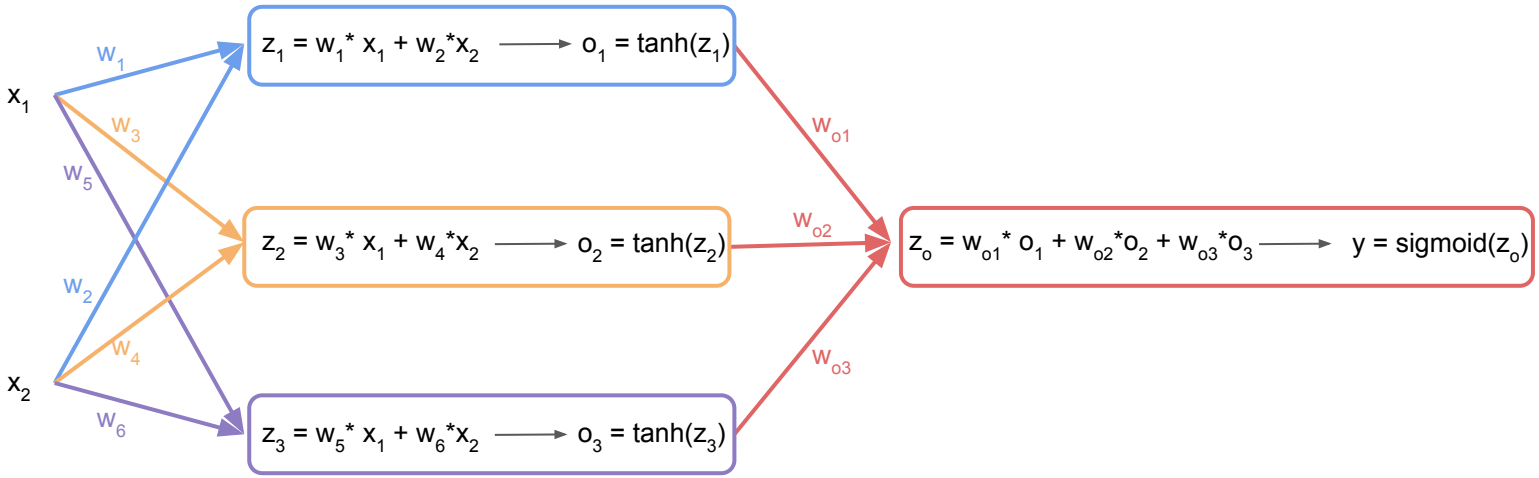


Multiclass classification



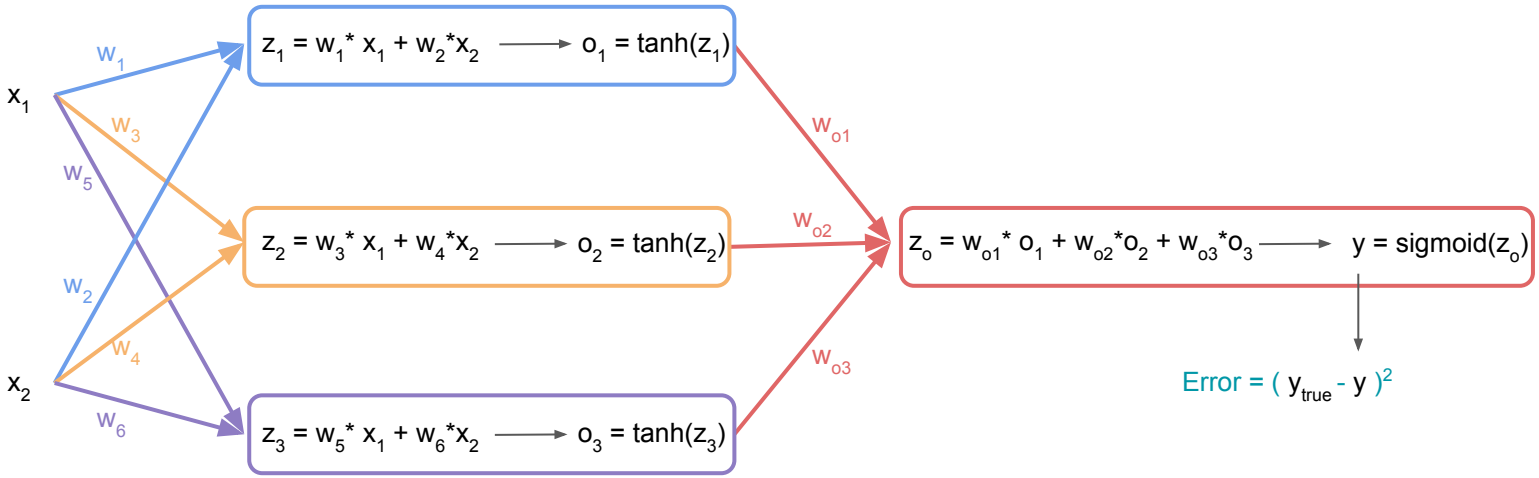
Regression

- Training



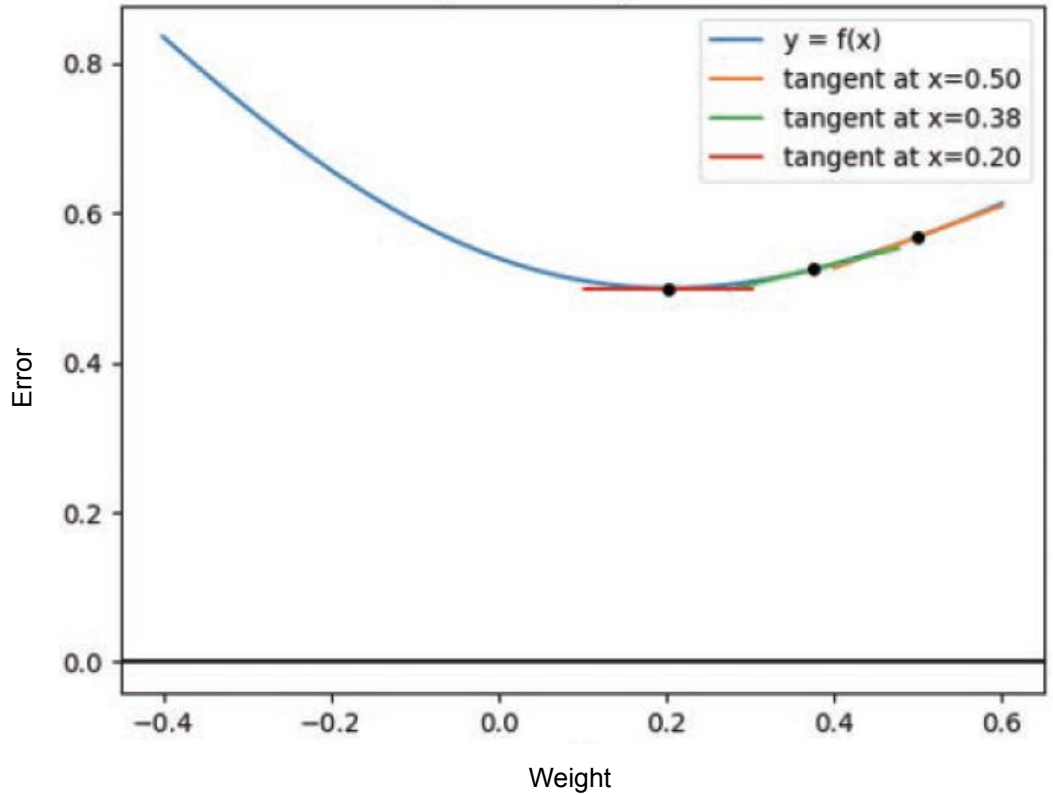
$$y = \text{sigmoid}(w_{o1} * \tanh(w_1 * x_1 + w_2 * x_2) + w_{o2} * \tanh(w_3 * x_1 + w_4 * x_2) + w_{o3} * \tanh(w_5 * x_1 + w_6 * x_2))$$

- Training



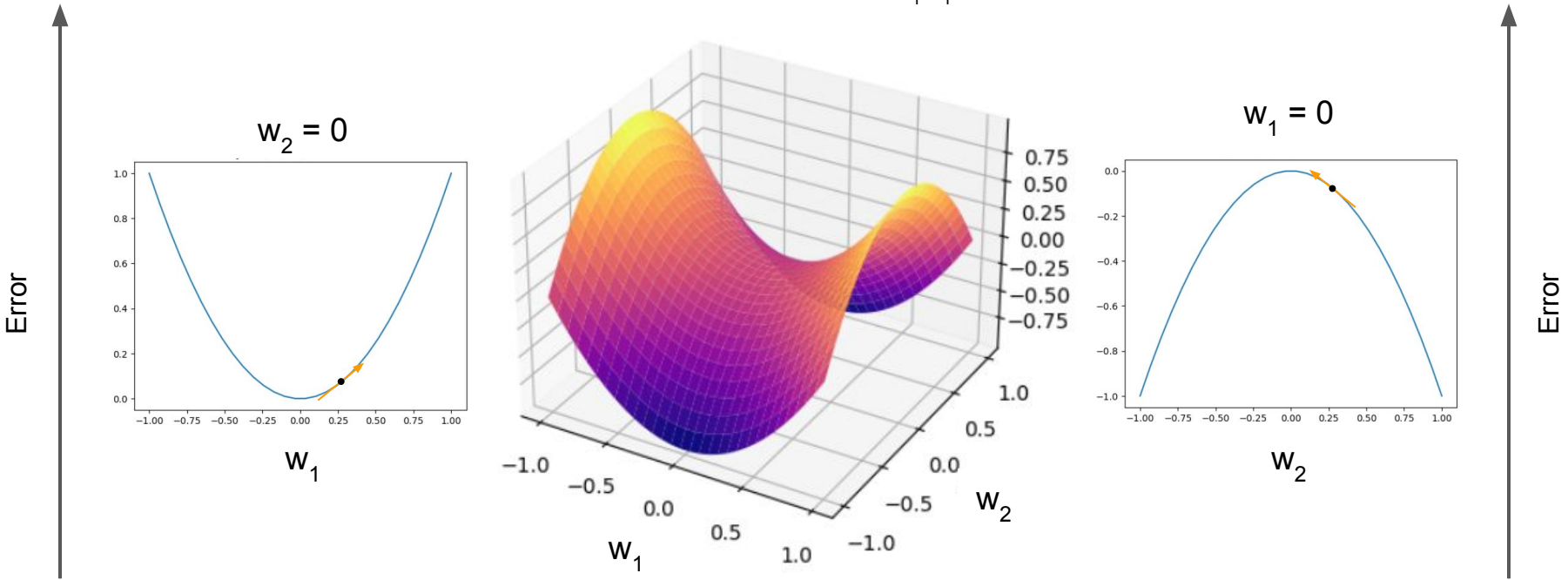
Error = $(y_{true}$ - sigmoid(w_{o1}* tanh(w_1* x_1 + w_2*x_2) + w_{o2}* tanh(w_3* x_1 + w_4*x_2) + w_{o3}* tanh(w_5* x_1 + w_6*x_2)))^2

- Derivatives

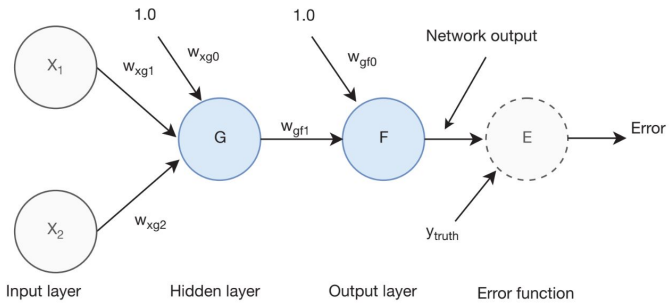


- Partial Derivatives

Error = Function(Activation($sum_i(w_i*...)$))

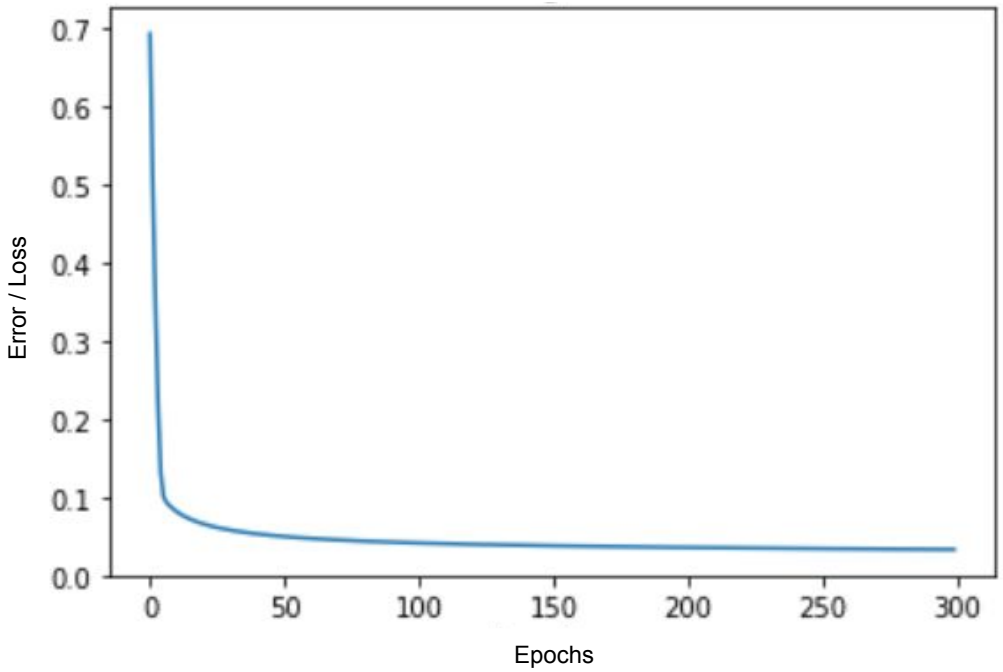


- Training procedure

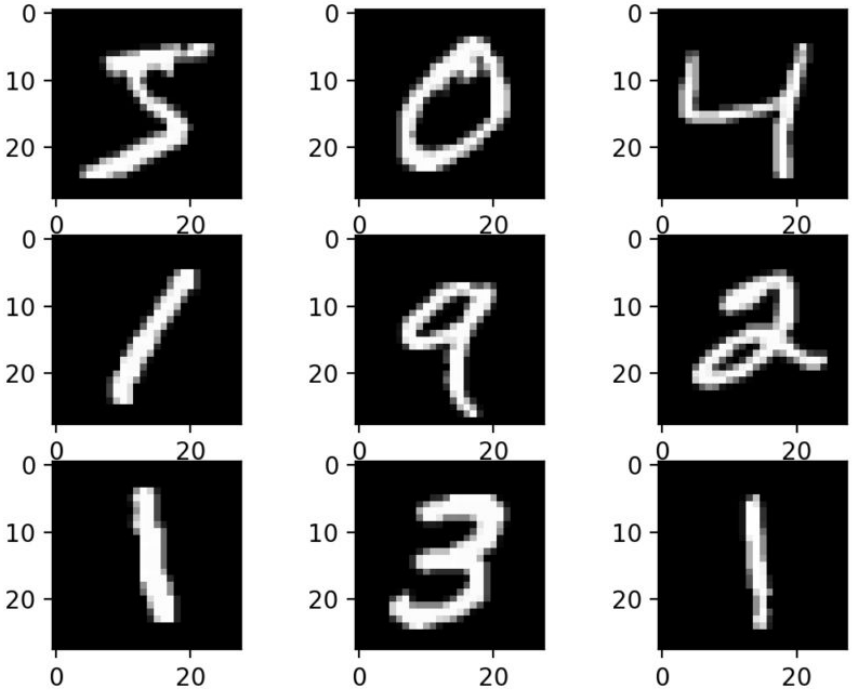


Training set

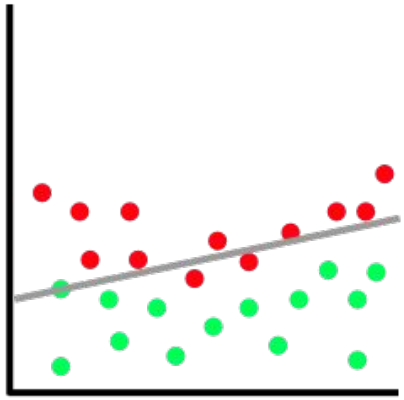
Sample	Feature 1	Feature 2	Ground Truth
1	0	1	1
2	1	-1	0
3	-1	1	1
4	0	0	0
5	1	0	1



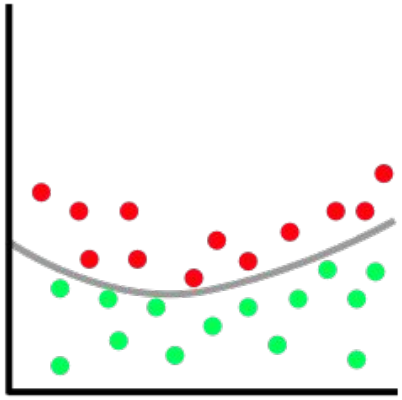
- Training procedure



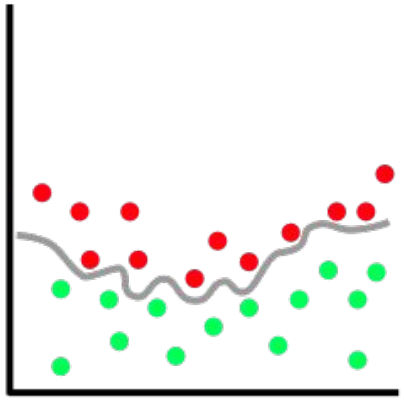
- Training procedure



Underfitting

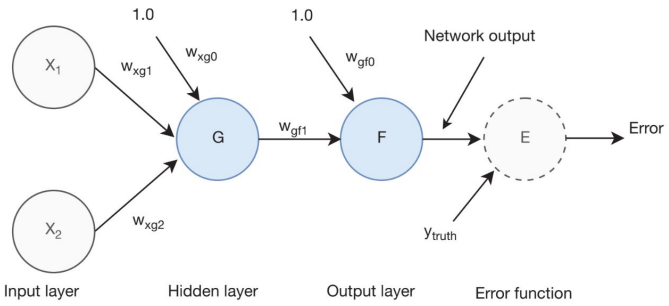


Balanced



Overfitting

- Training procedure

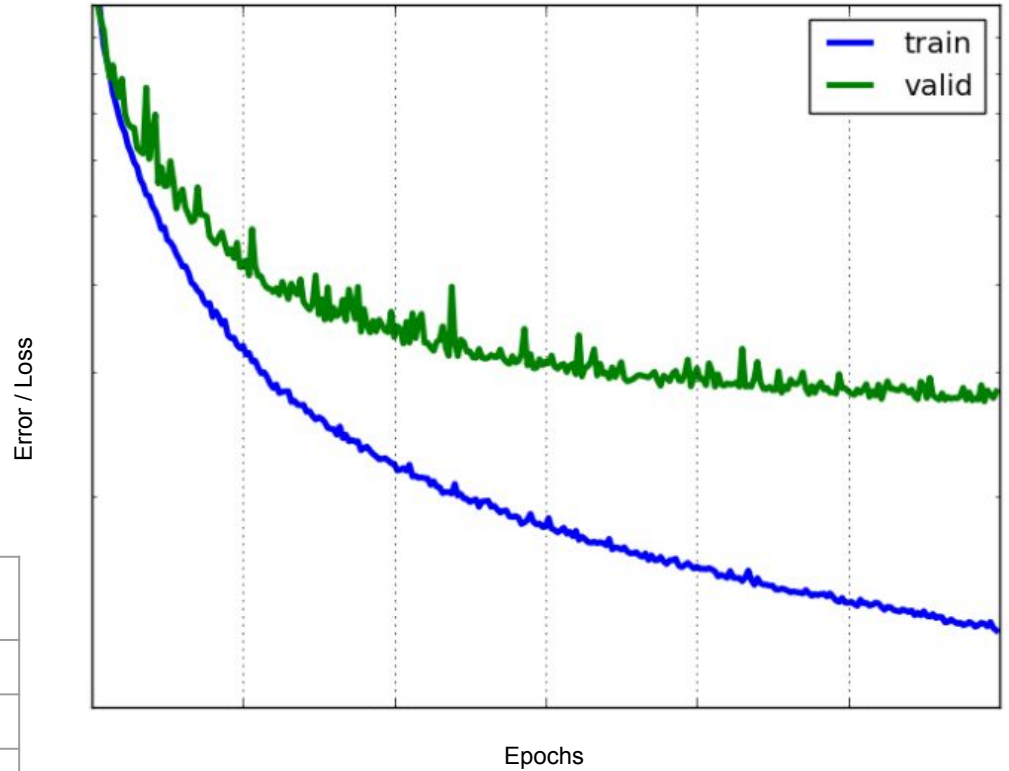


Training set

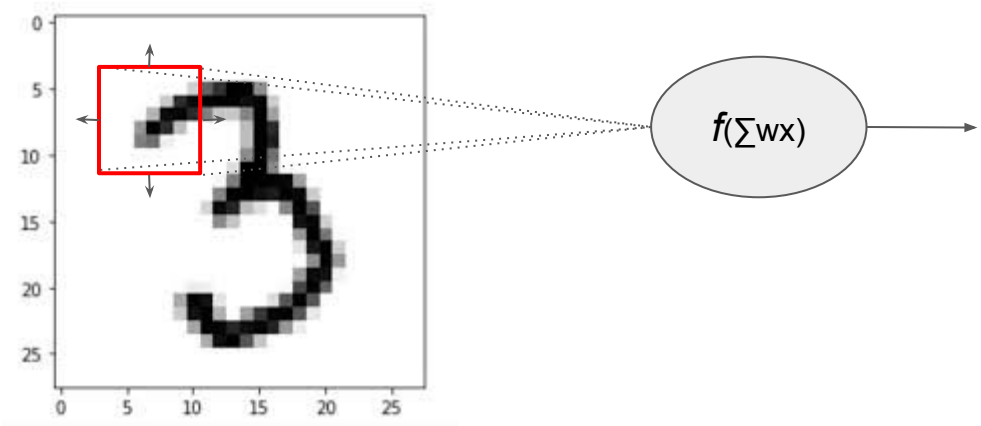
Id	Feat 1	Feat 2	GT	Pred
1	0.5	-0.6	1	1
2	0.1	0.4	0	1
3	-0.1	0.1	1	1
4	0.5	0.8	0	0
5	1	0.01	1	0

Validation set

Id	Feat 1	Feat 2	GT	Pred
1	0.1	-0.3	0	1
2	0.1	-0.1	0	0
3	-0.3	-0.7	1	1



- Dense vs. Convolutional Layers

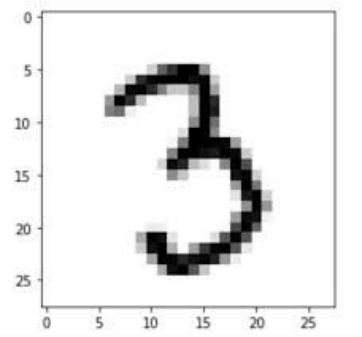


- Convolutional Layers

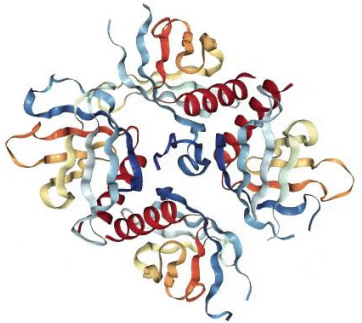
1D data

TACAGATCGGAA

2D data

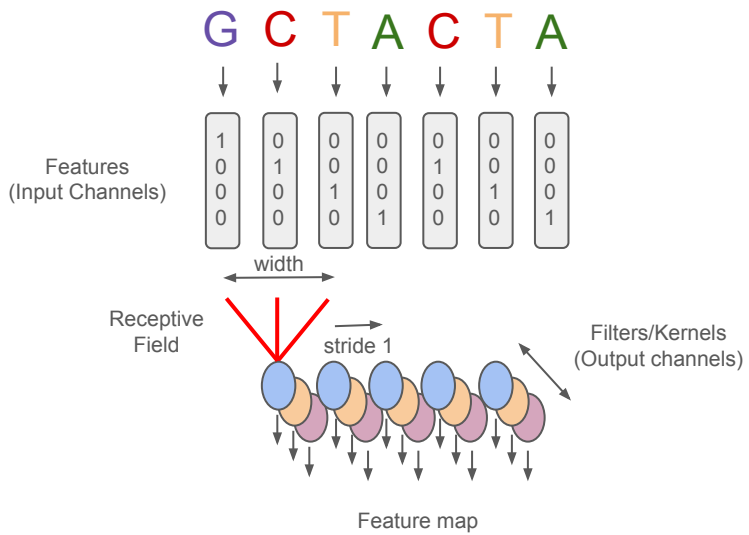


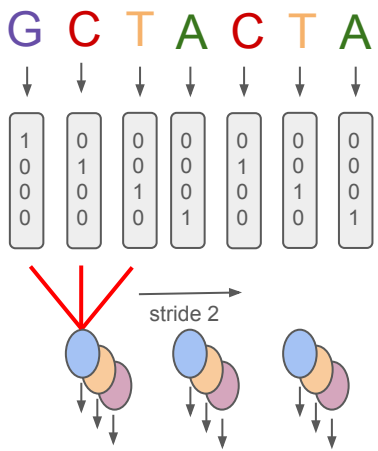
3D data



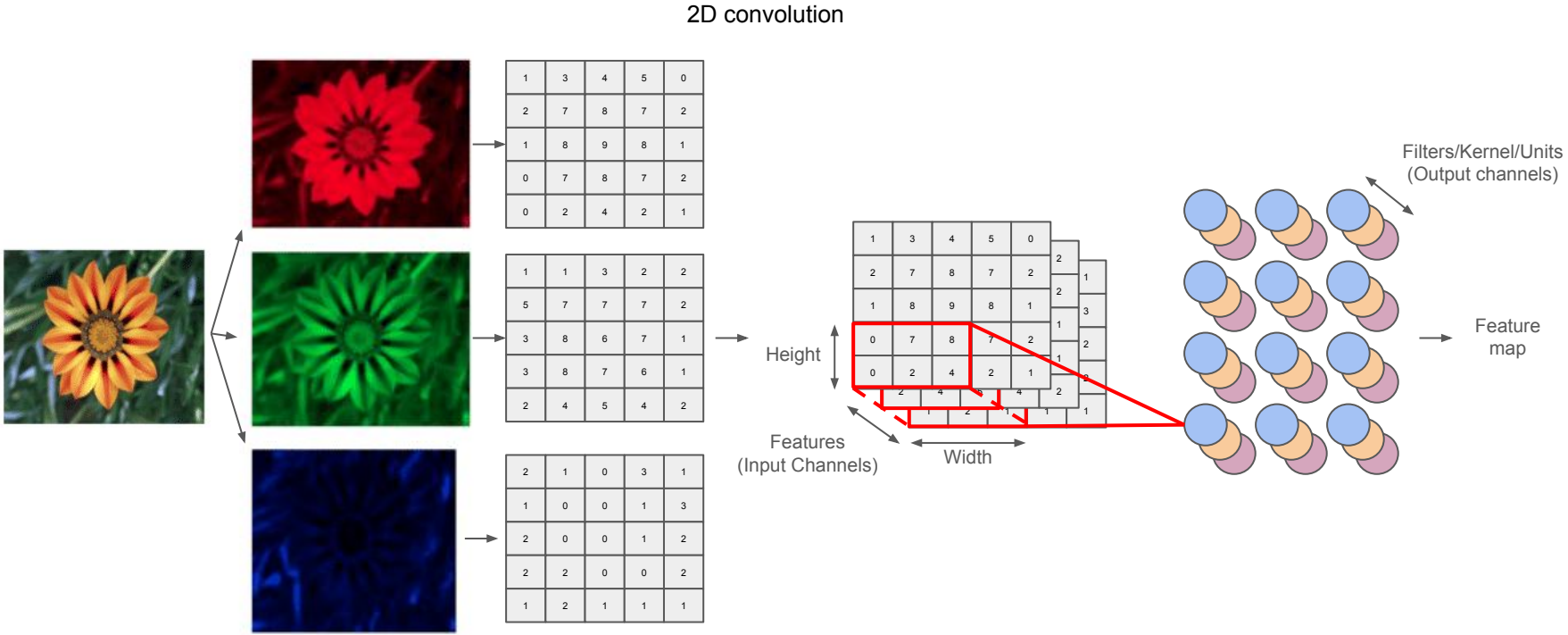
- Convolutional Layers

1D convolution

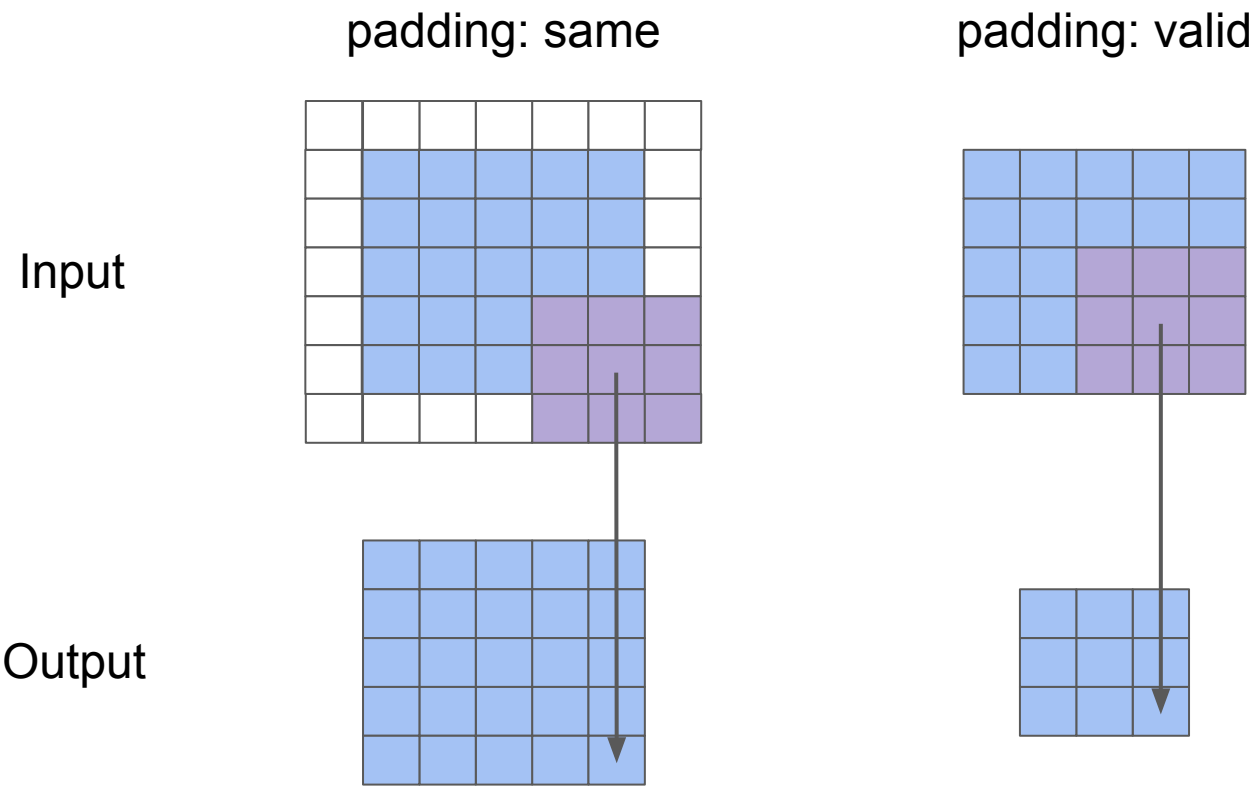




- Convolutional Layers



- Convolutional Layers



- Convolutional Layers