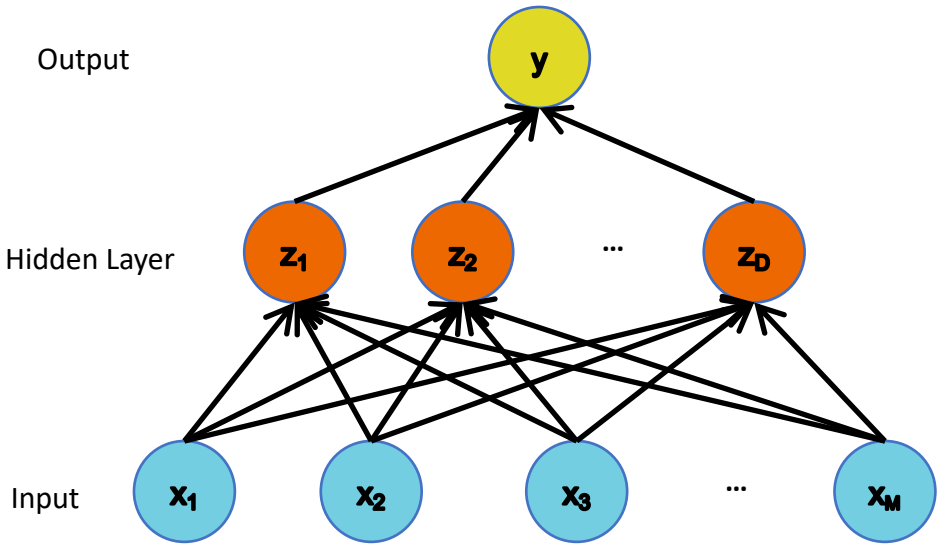
Convolutional Neural Networks (CNN)

Qi Yan
Research Assistant Professor
Division of Pulmonary Medicine
Department of Pediatrics
Children's Hospital of Pittsburgh of UPMC
University of Pittsburgh

Neural Networks

Architectures



(F) **Loss**
$J = \frac{1}{2}(y - y^*)^2$

(E) **Output (sigmoid)**
$y = \frac{1}{1+\exp(-b)}$

(D) **Output (linear)**
$b = \sum_{j=0}^{D} \beta_j z_j$

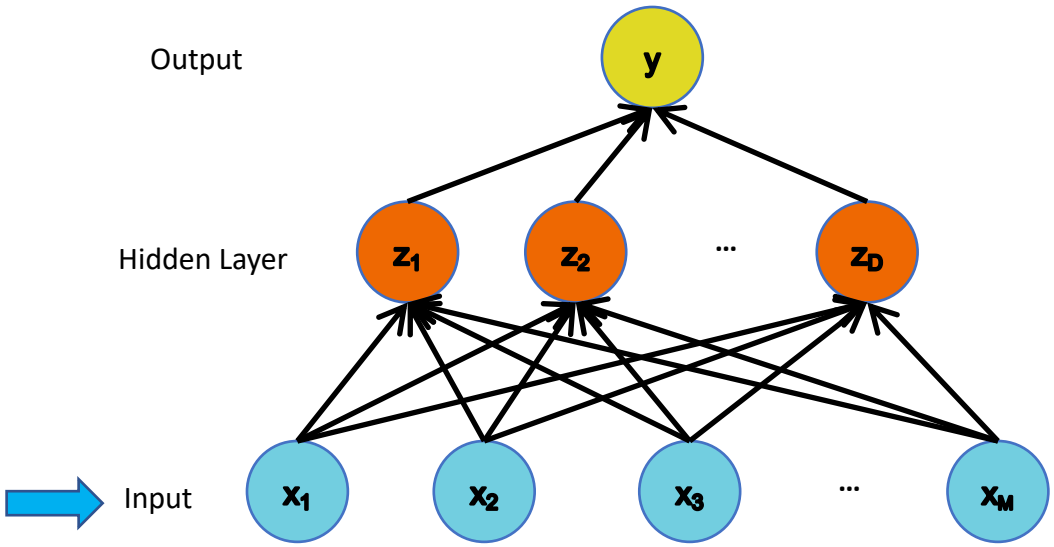
(C) **Hidden (sigmoid)**
$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$

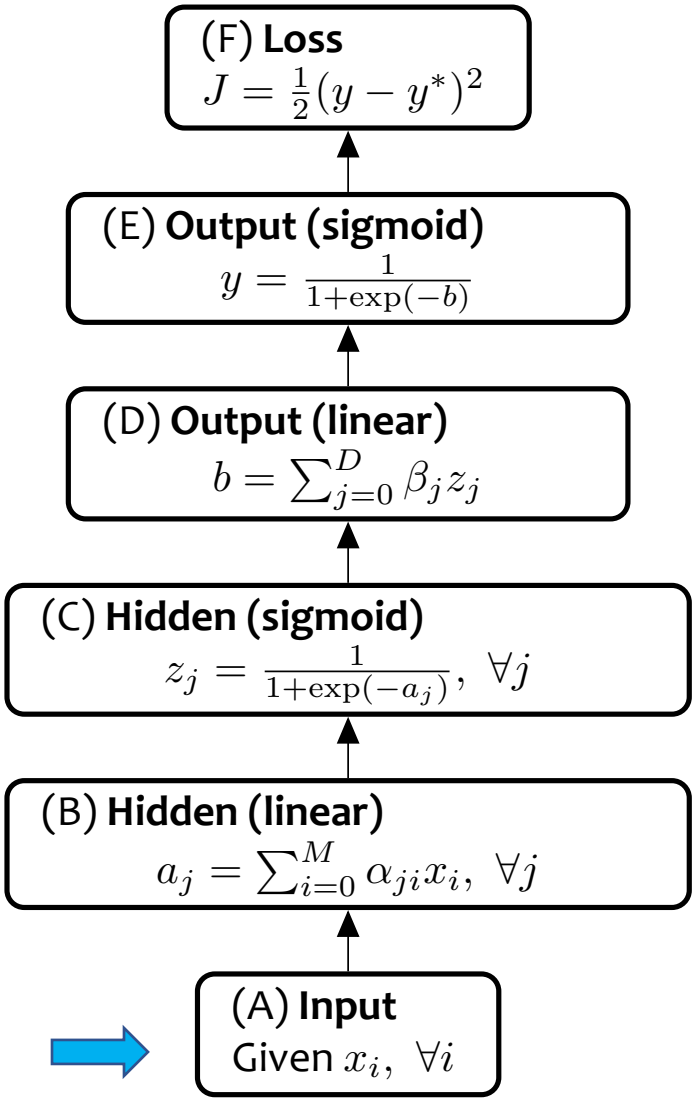
(B) **Hidden (linear)**
$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$

(A) **Input**
Given $x_i, \ \forall i$

Neural Networks

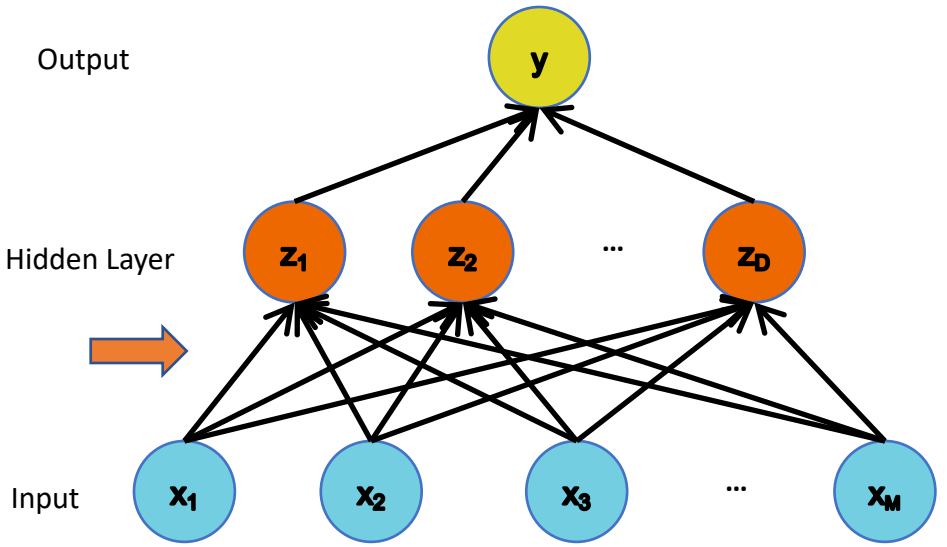
Architectures





Neural Networks

Architectures



(F) **Loss**

$$J = \frac{1}{2}(y - y^*)^2$$

(E) **Output (sigmoid)**

$$y = \frac{1}{1+\exp(-b)}$$

(D) **Output (linear)**

$$b = \sum_{j=0}^{D} \beta_j z_j$$

(C) **Hidden (sigmoid)**

$$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$$

(B) **Hidden (linear)**

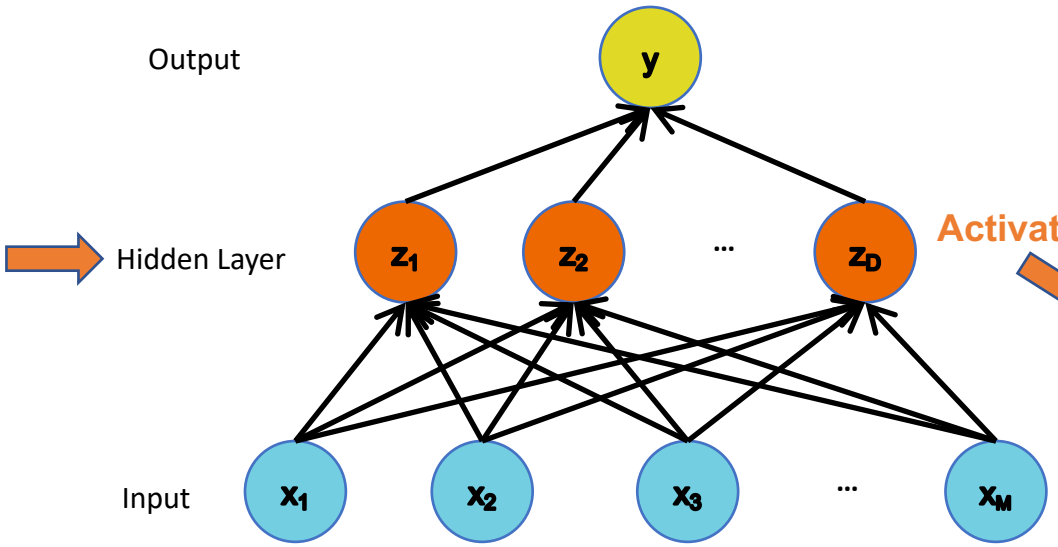
$$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$$

(A) **Input**

Given $x_i, \ \forall i$

Neural Networks

Architectures



Activation

(F) **Loss**
$J = \frac{1}{2}(y - y^*)^2$

(E) **Output (sigmoid)**
$y = \frac{1}{1+\exp(-b)}$

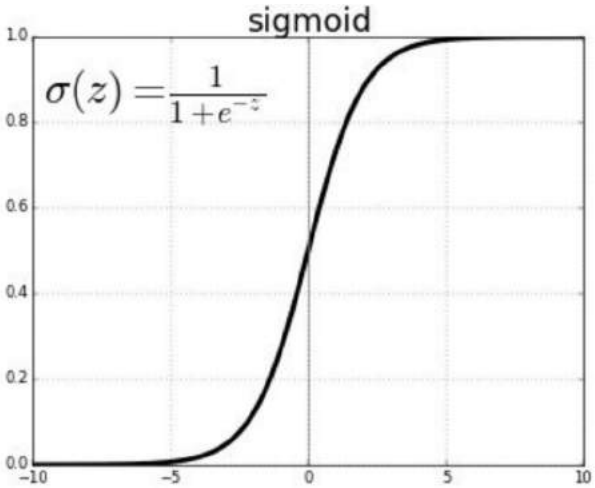
(D) **Output (linear)**
$b = \sum_{j=0}^{D} \beta_j z_j$

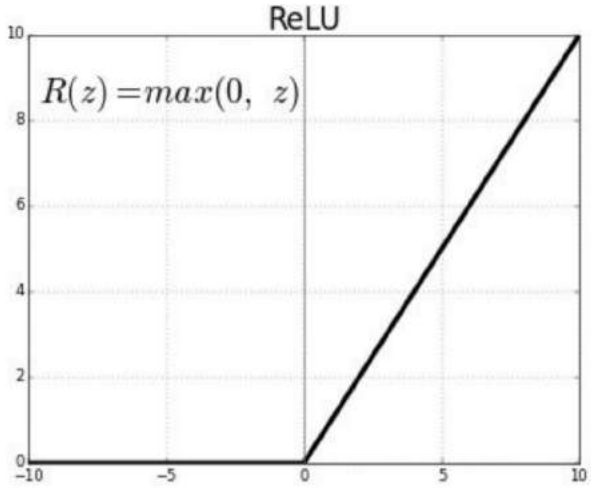
(C) **Hidden (sigmoid)**
$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$

(B) **Hidden (linear)**
$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$

(A) **Input**
Given $x_i, \ \forall i$

Activation function



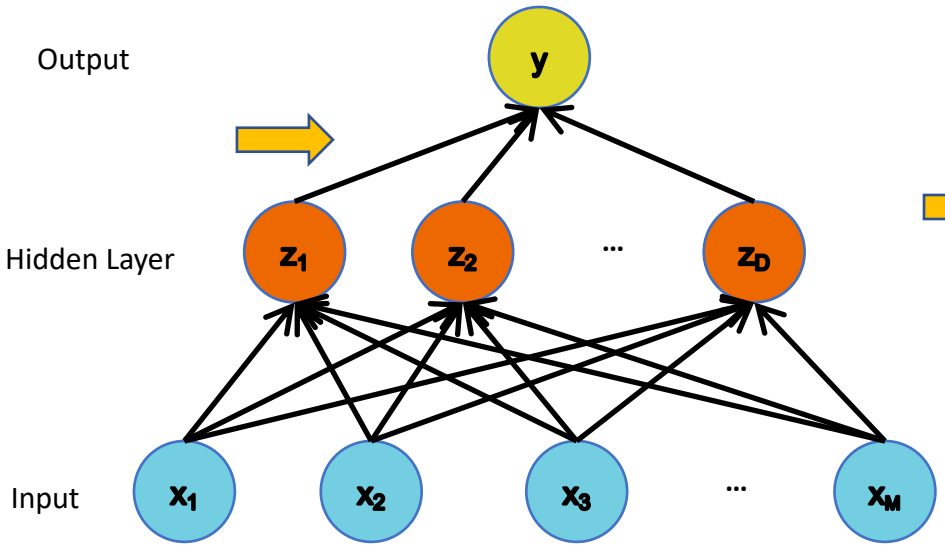


- Used to increase non-linearity of the network without affecting receptive fields of conv layers
- Prefer ReLU, results in faster training
- LeakyReLU addresses the vanishing gradient problem

Other types:
Leaky ReLU, Randomized Leaky ReLU, Parameterized ReLU Exponential Linear Units (ELU), Scaled Exponential Linear Units Tanh, hardtanh, softtanh, softsign, softmax, softplus...

Neural Networks

Architectures



(F) **Loss**
$J = \frac{1}{2}(y - y^*)^2$

(E) **Output (sigmoid)**
$y = \frac{1}{1+\exp(-b)}$

(D) **Output (linear)**
$b = \sum_{j=0}^{D} \beta_j z_j$

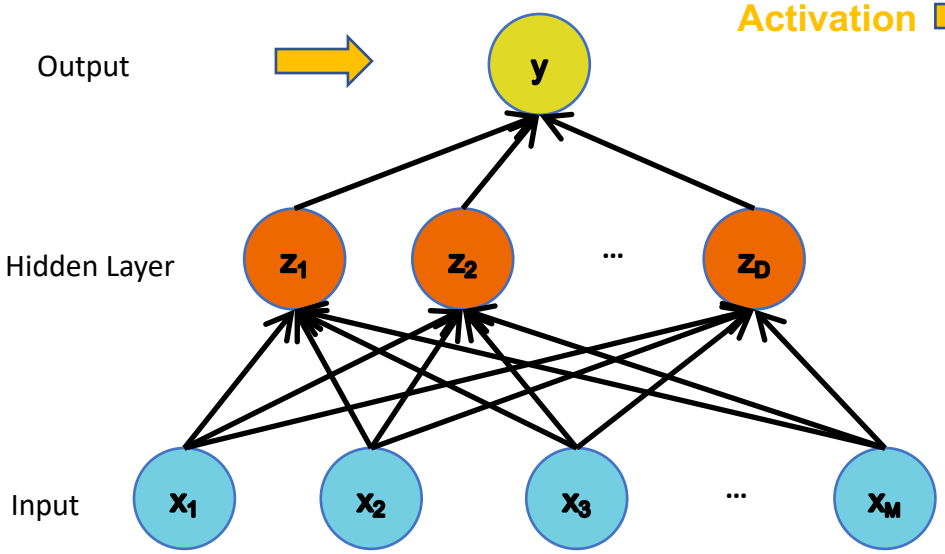
(C) **Hidden (sigmoid)**
$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$

(B) **Hidden (linear)**
$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$

(A) **Input**
Given $x_i, \ \forall i$

Neural Networks

Architectures



Activation

(F) **Loss**
$J = \frac{1}{2}(y - y^*)^2$

(E) **Output (sigmoid)**
$y = \frac{1}{1+\exp(-b)}$

(D) **Output (linear)**
$b = \sum_{j=0}^{D} \beta_j z_j$

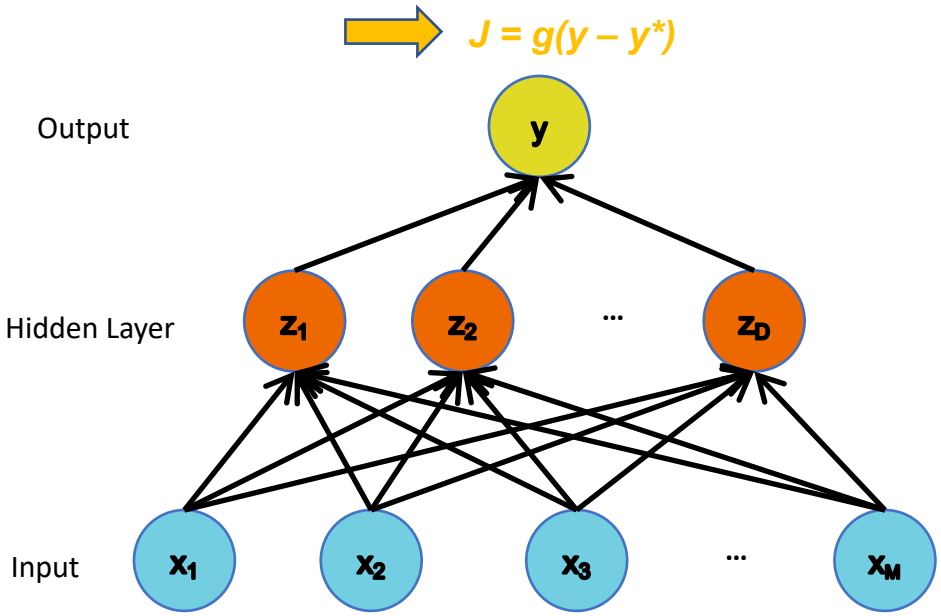
(C) **Hidden (sigmoid)**
$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$

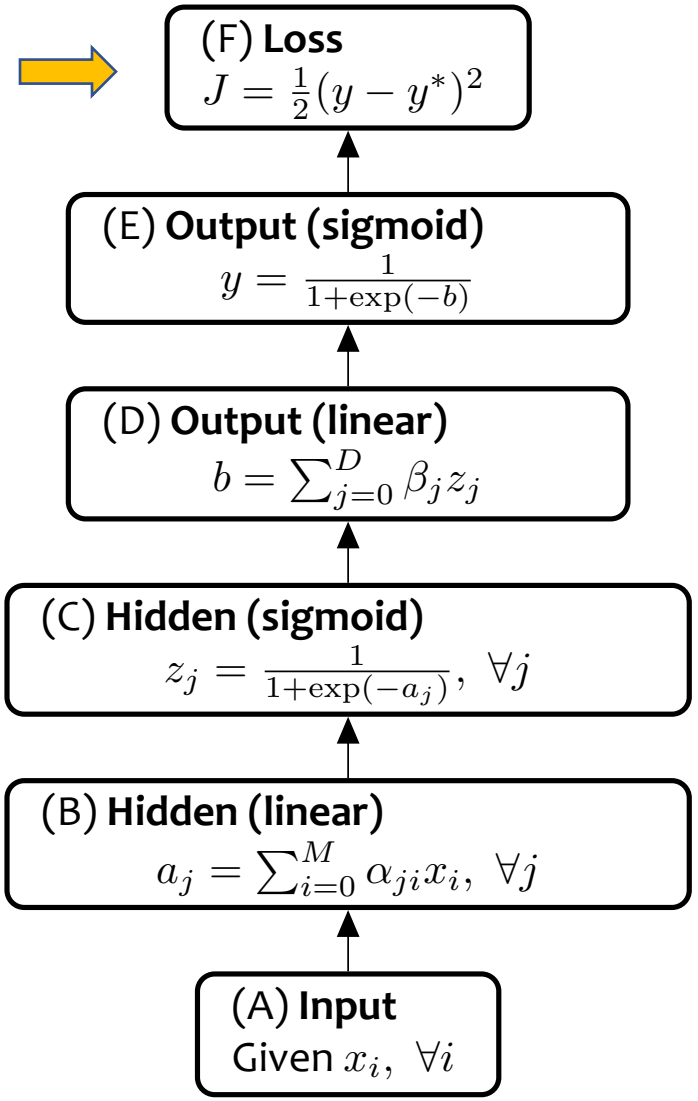
(B) **Hidden (linear)**
$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$

(A) **Input**
Given $x_i, \ \forall i$

Neural Networks

Architectures





Loss function

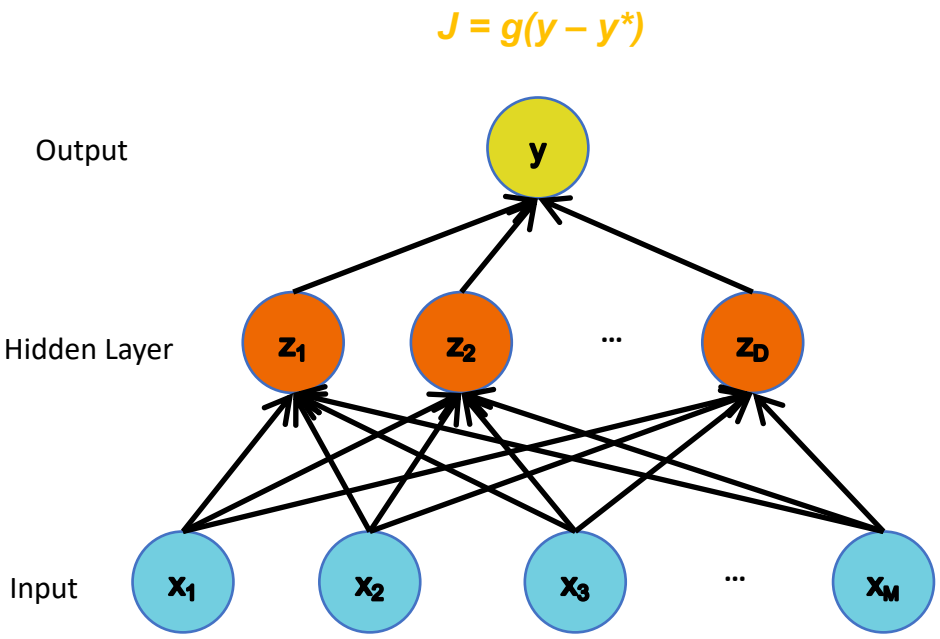
$$H(p, q) = -\sum_{x} p(x) \log q(x)$$

Binary case $\quad -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$

General case $\quad -\sum_{i} p_i \log q_i$

- L1, L2 loss
- Cross-Entropy loss (works well for classification, e.g., image classification)
- Hinge Loss
- Huber Loss, more resilient to outliers with smooth gradient
- Minimum Squared Error (works well for regression task, e.g., Behavioral Cloning)

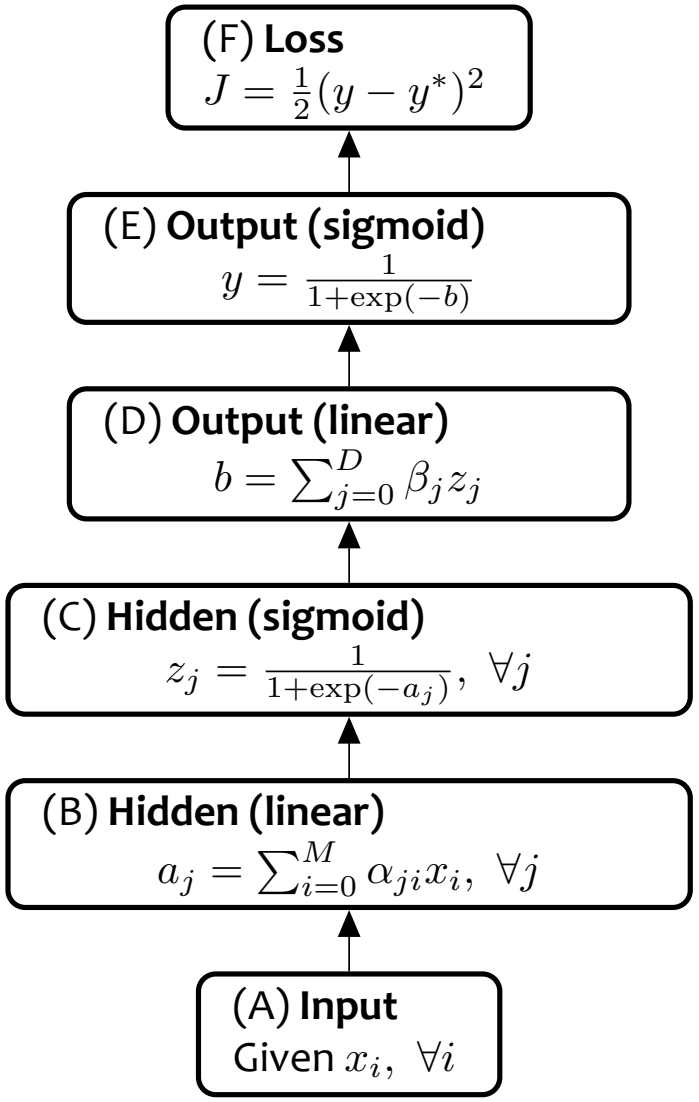
Gradient descent



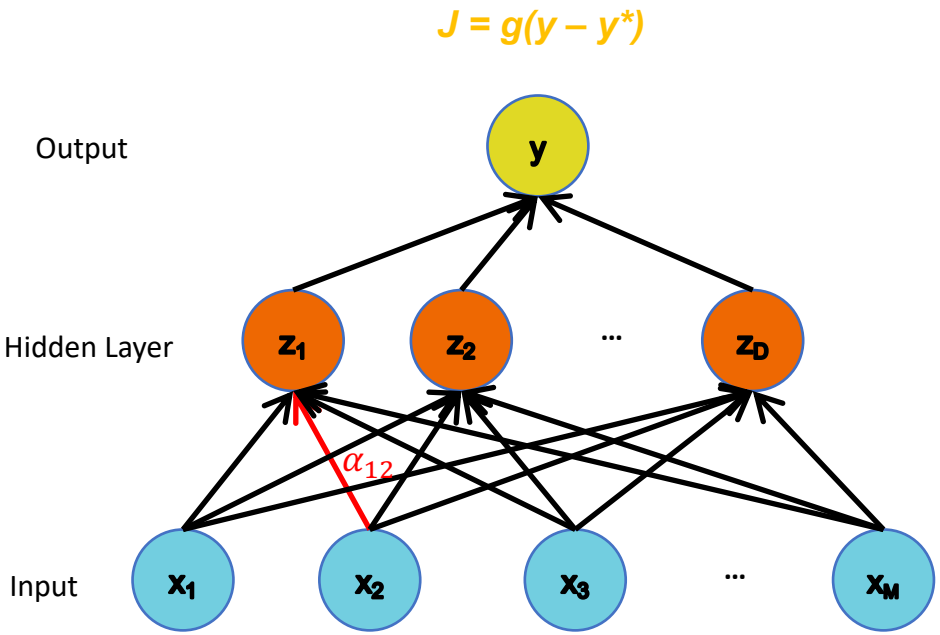
We want to start with random parameters and make our parameters better and better gradually as an iterative manner. Gradient descent is:

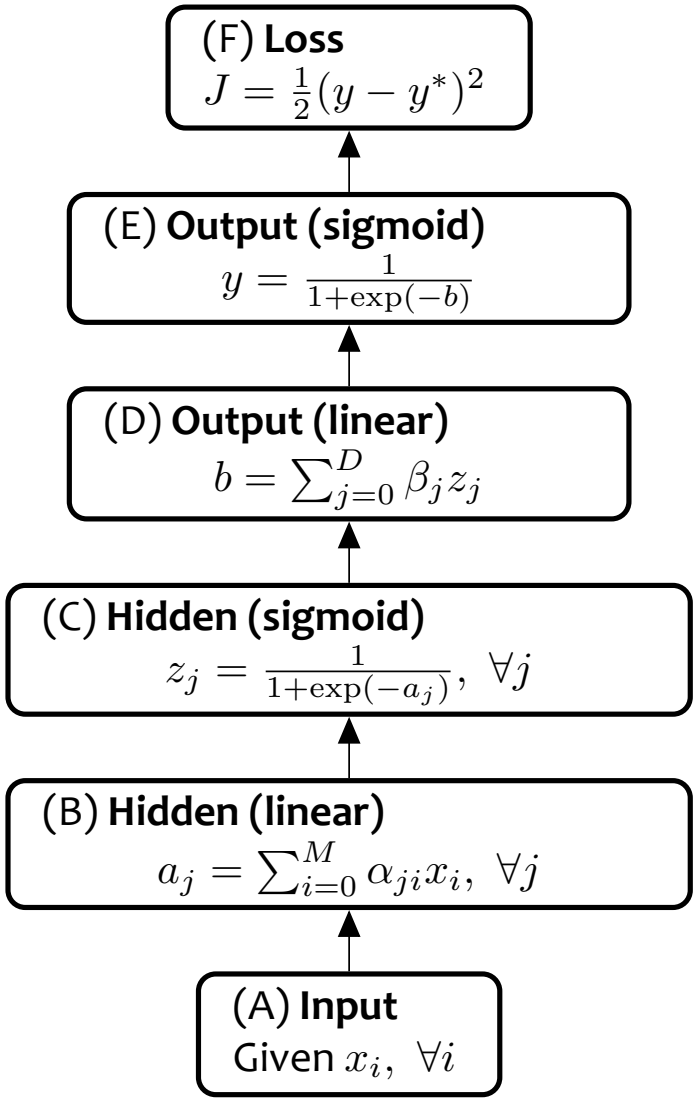
$$W^{(t+1)} = W^{(t)} - \eta \frac{dJ}{dW^{(t)}}$$

Tricky to calculate



Gradient descent



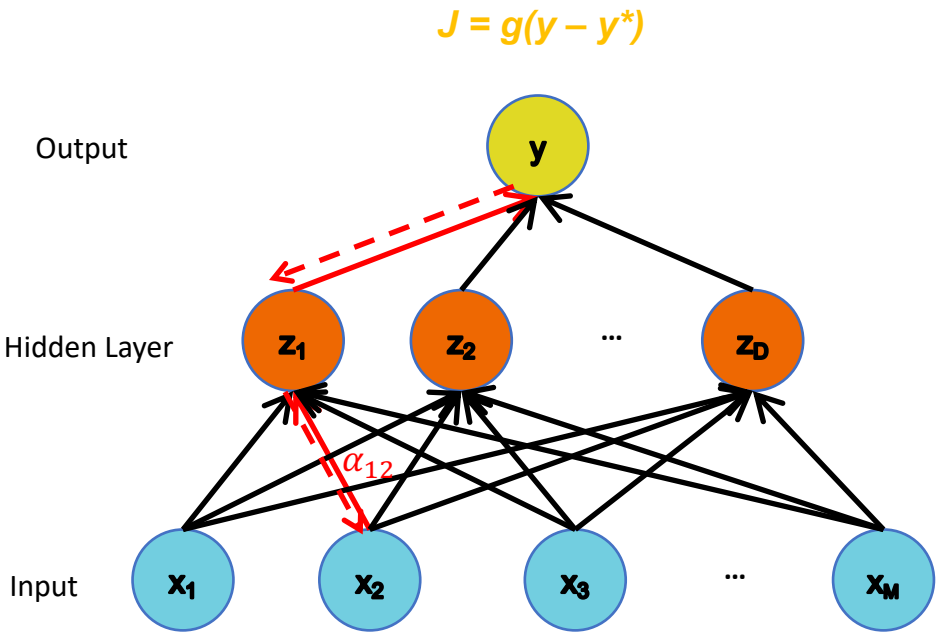


We want to start with random parameters and make our parameters better and better gradually as an iterative manner. Gradient descent is:

$$W^{(t+1)} = W^{(t)} - \eta \frac{dJ}{dW^{(t)}}$$

For example: $\alpha_{12}^{(t+1)} = \alpha_{12}^{(t)} - \eta \frac{dJ}{\alpha_{12}^{(t)}}$

Back propagation



$$\alpha_{12}^{(t+1)} = \alpha_{12}^{(t)} - \eta \frac{dJ}{d\alpha_{12}^{(t)}}$$

Chain rule ⇨ $$\frac{dJ}{d\alpha_{12}^{(t)}} = \left(\frac{dJ}{dz_1^{(t)}}\right)\left(\frac{dz_1^{(t)}}{d\alpha_{12}^{(t)}}\right)$$

(F) **Loss**
$J = \frac{1}{2}(y - y^*)^2$

(E) **Output (sigmoid)**
$y = \frac{1}{1+\exp(-b)}$

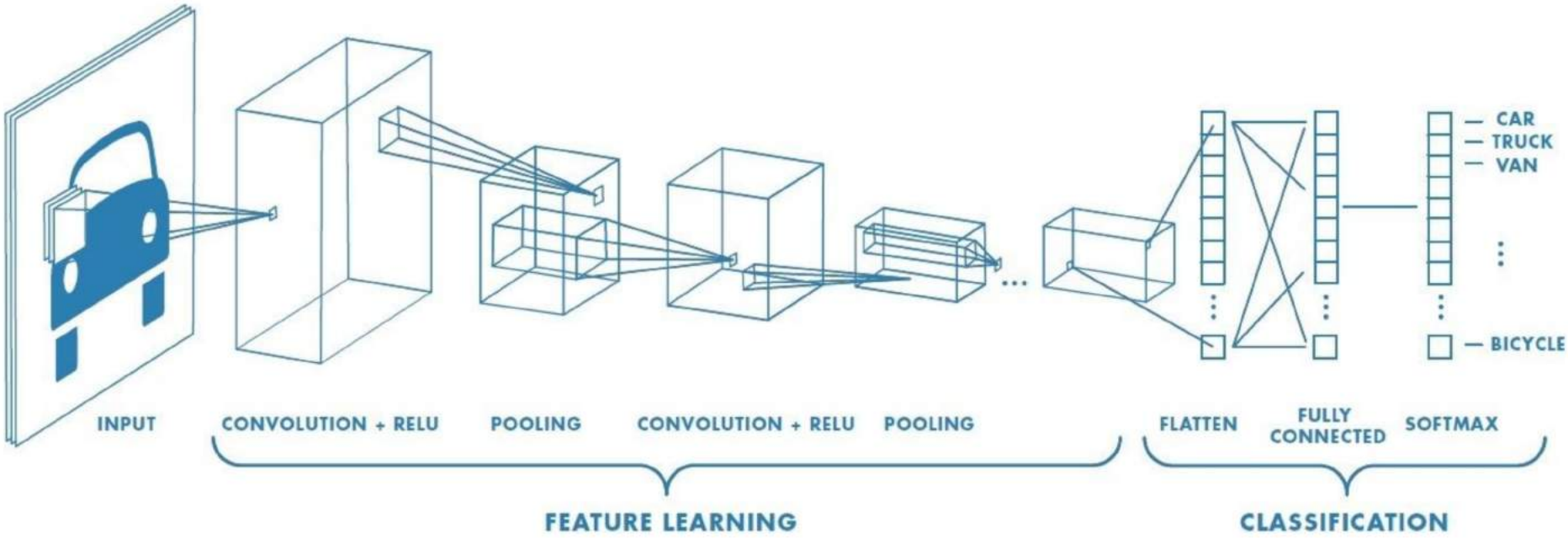
(D) **Output (linear)**
$b = \sum_{j=0}^{D} \beta_j z_j$

(C) **Hidden (sigmoid)**
$z_j = \frac{1}{1+\exp(-a_j)}, \ \forall j$

(B) **Hidden (linear)**
$a_j = \sum_{i=0}^{M} \alpha_{ji} x_i, \ \forall j$

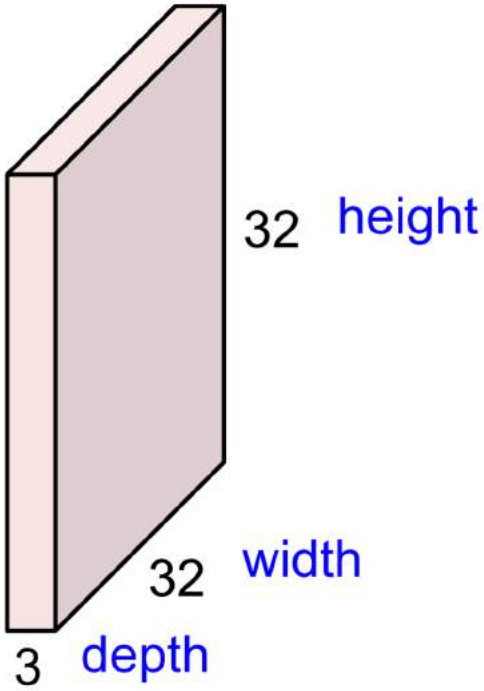
(A) **Input**
Given $x_i, \ \forall i$

Convolutional Neural Networks



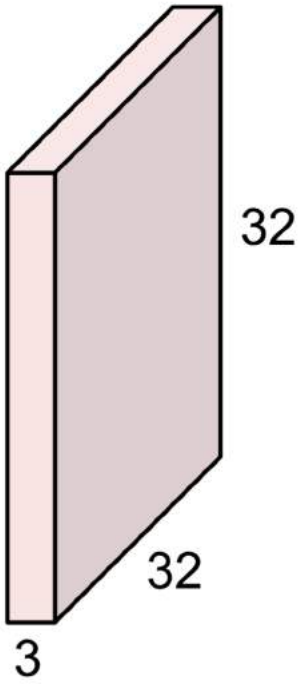
Convolutional layer

32x32x3 image -> preserve spatial structure

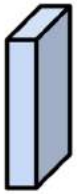


Convolutional layer

32x32x3 image

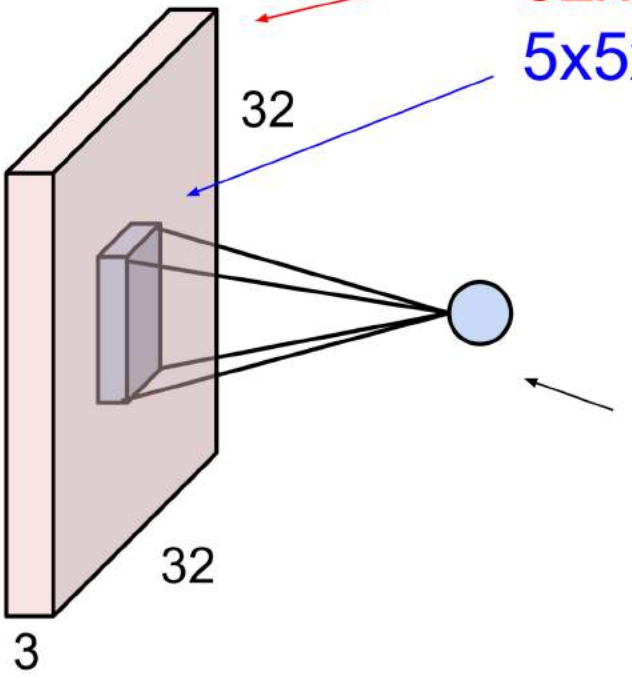


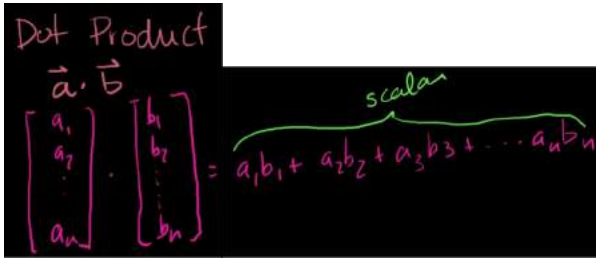
5x5x3 filter



Convolve the filter with the image
i.e. “slide over the image spatially,
computing dot products”

Convolutional layer

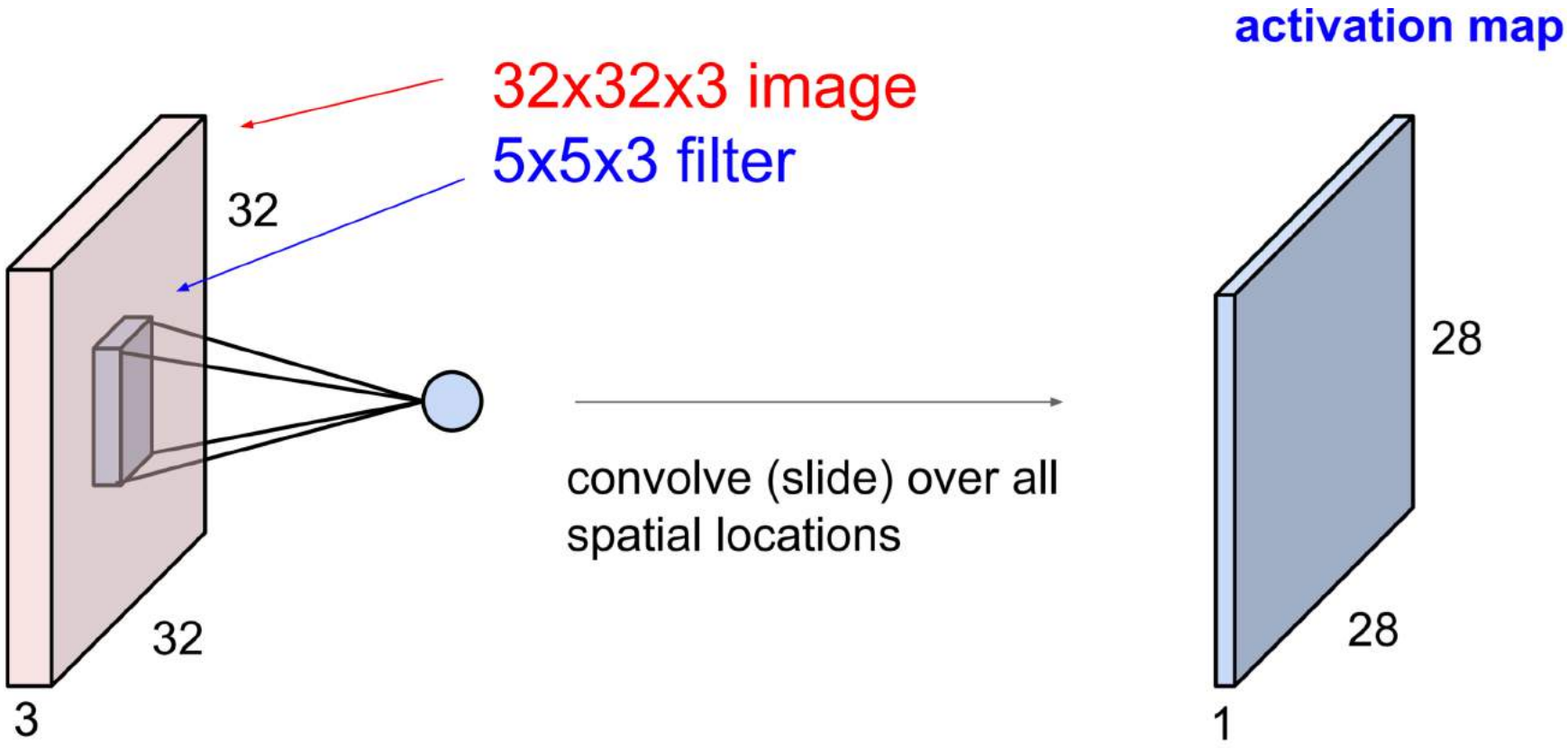




1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

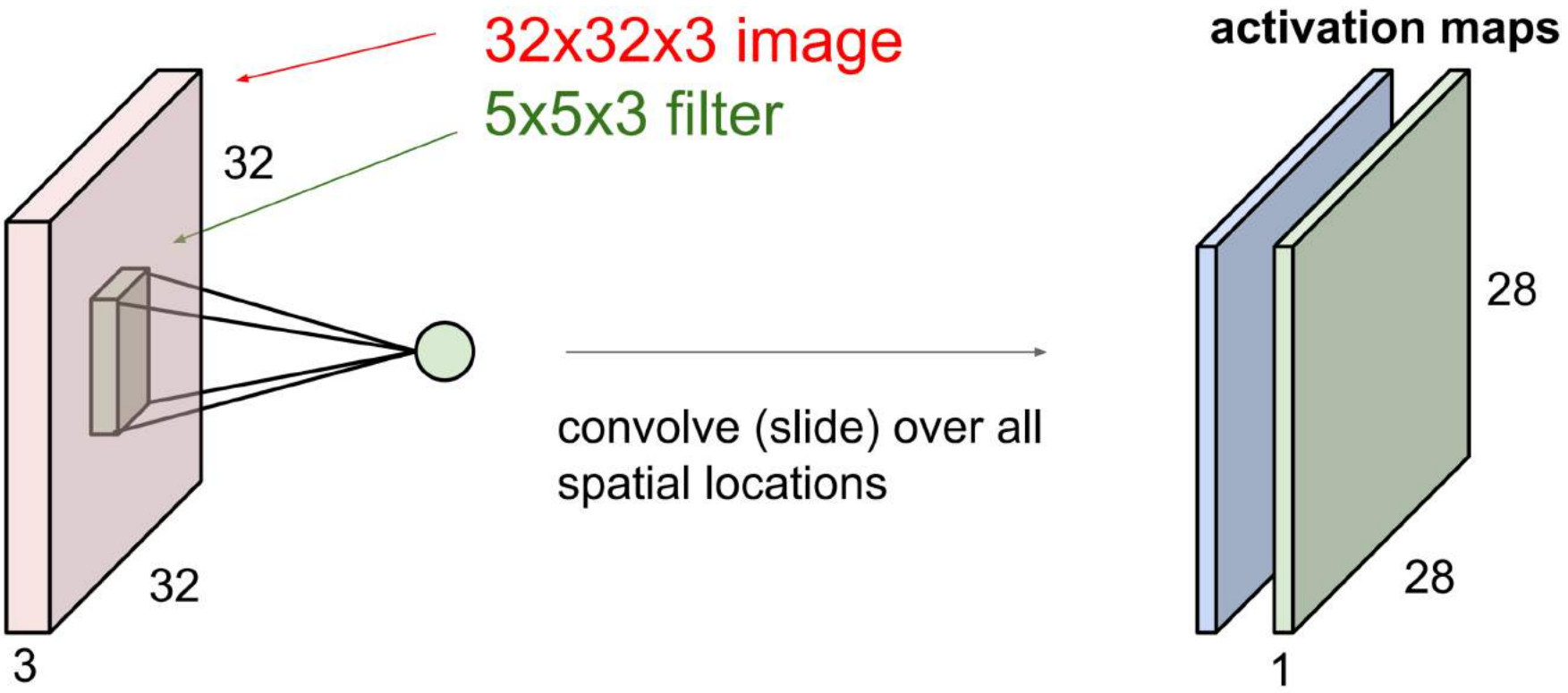
$$w^T x + b$$

Convolutional layer



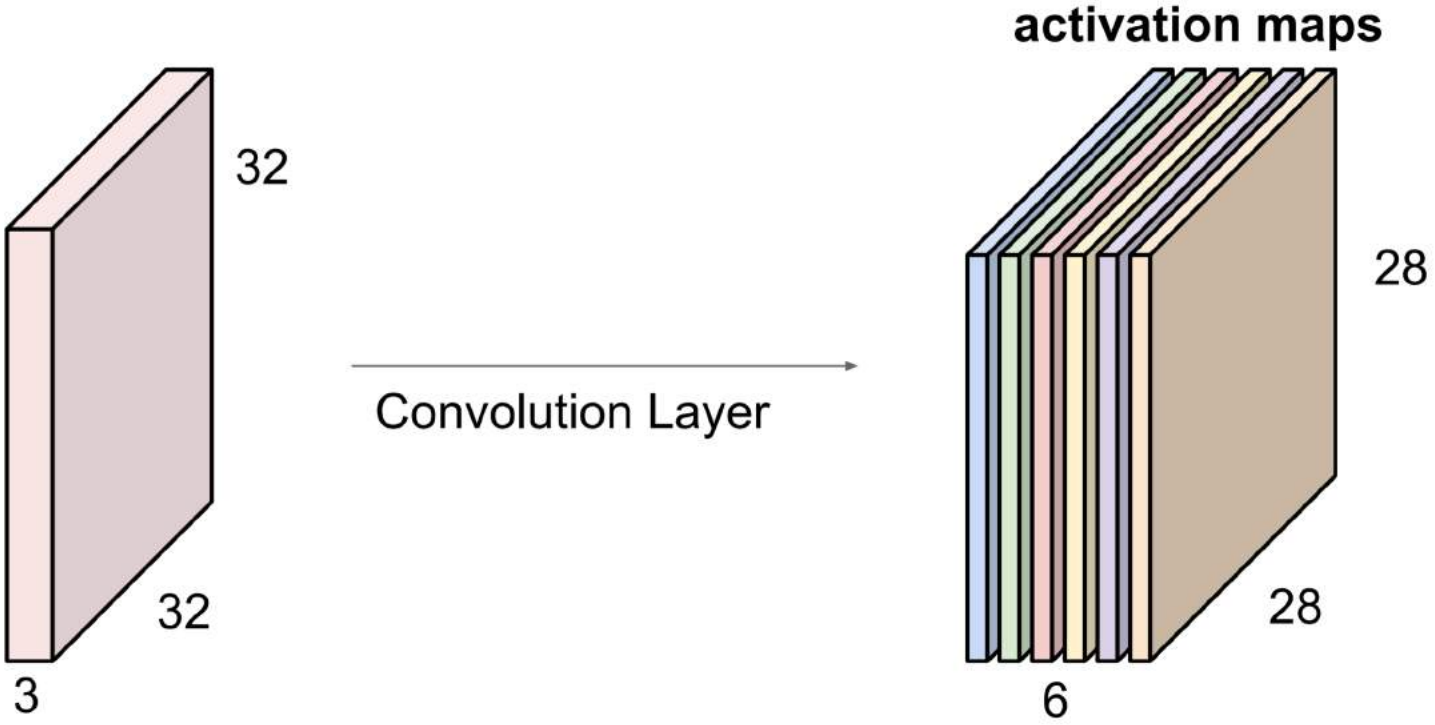
Convolutional layer

consider a second, green filter



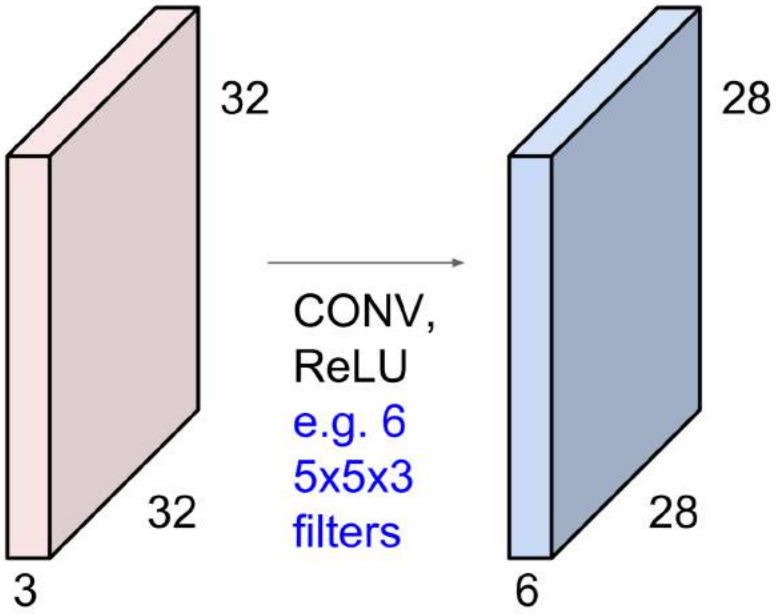
Convolutional layer

For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:

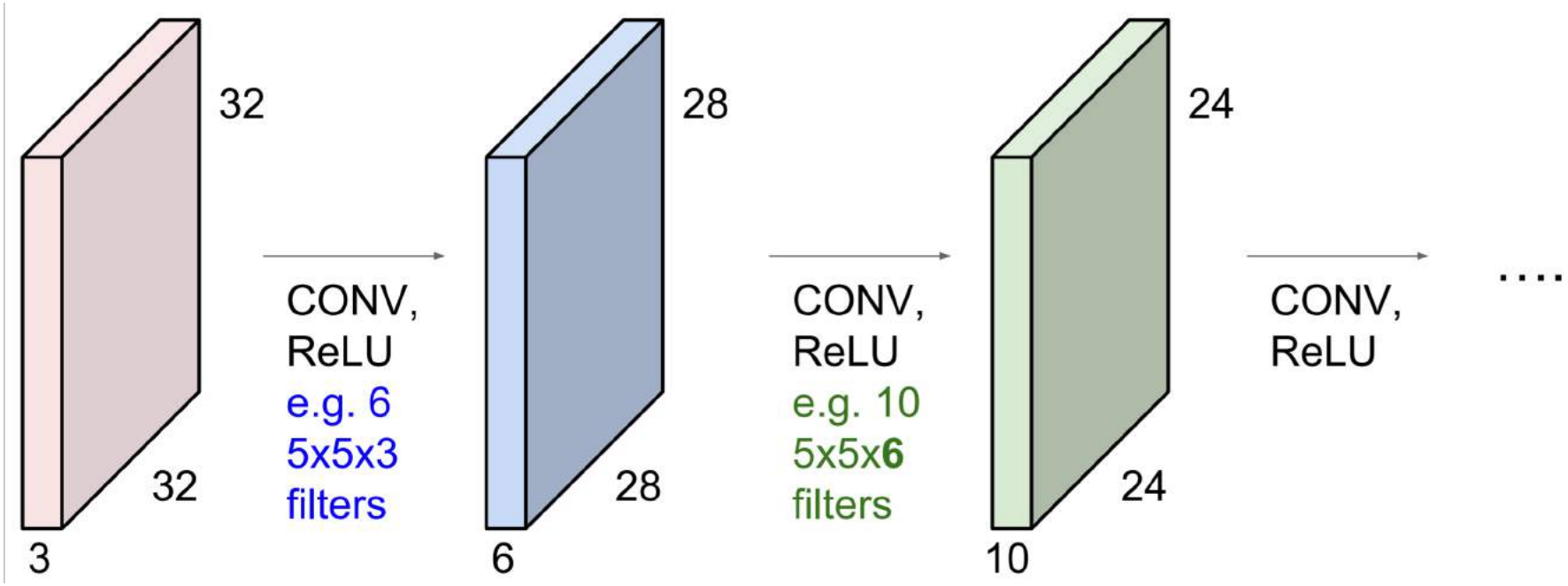


We stack these up to get a "new image" of size 28x28x6!

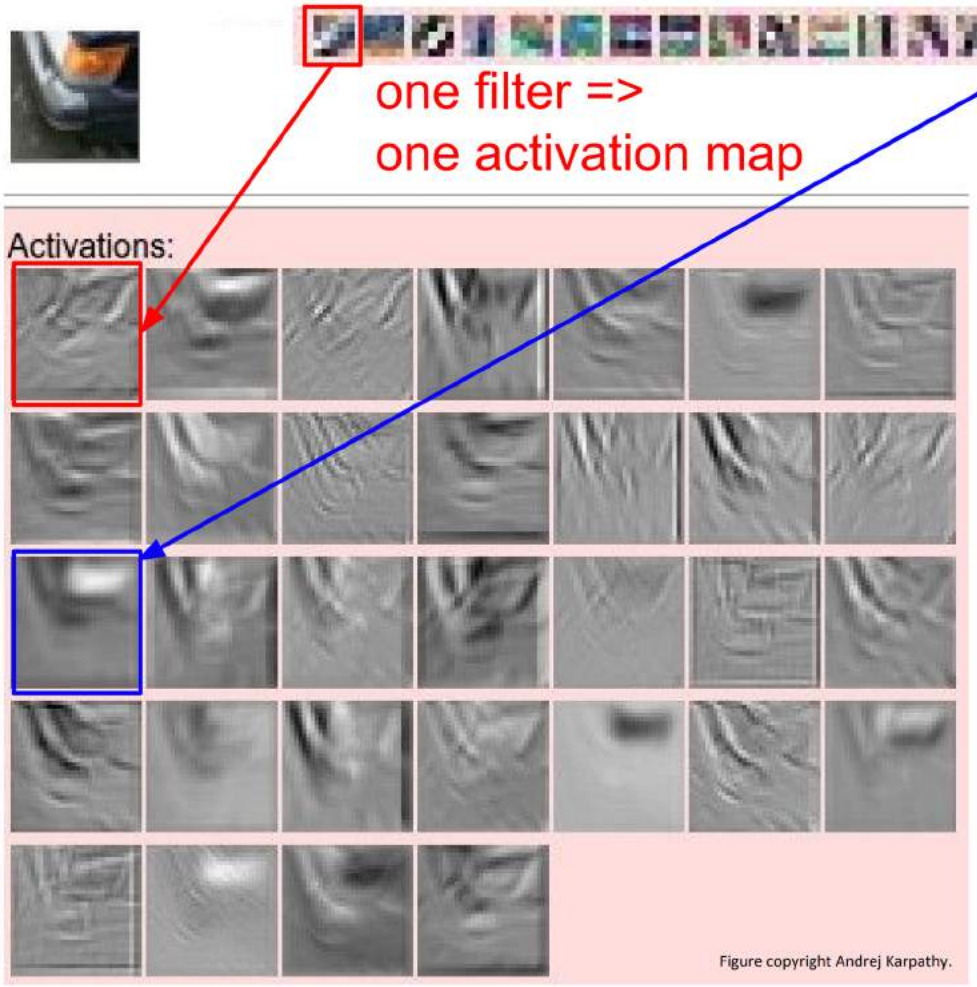
Convolutional layer



Convolutional layer



Convolutional layer



example 5x5 filters
(32 total)

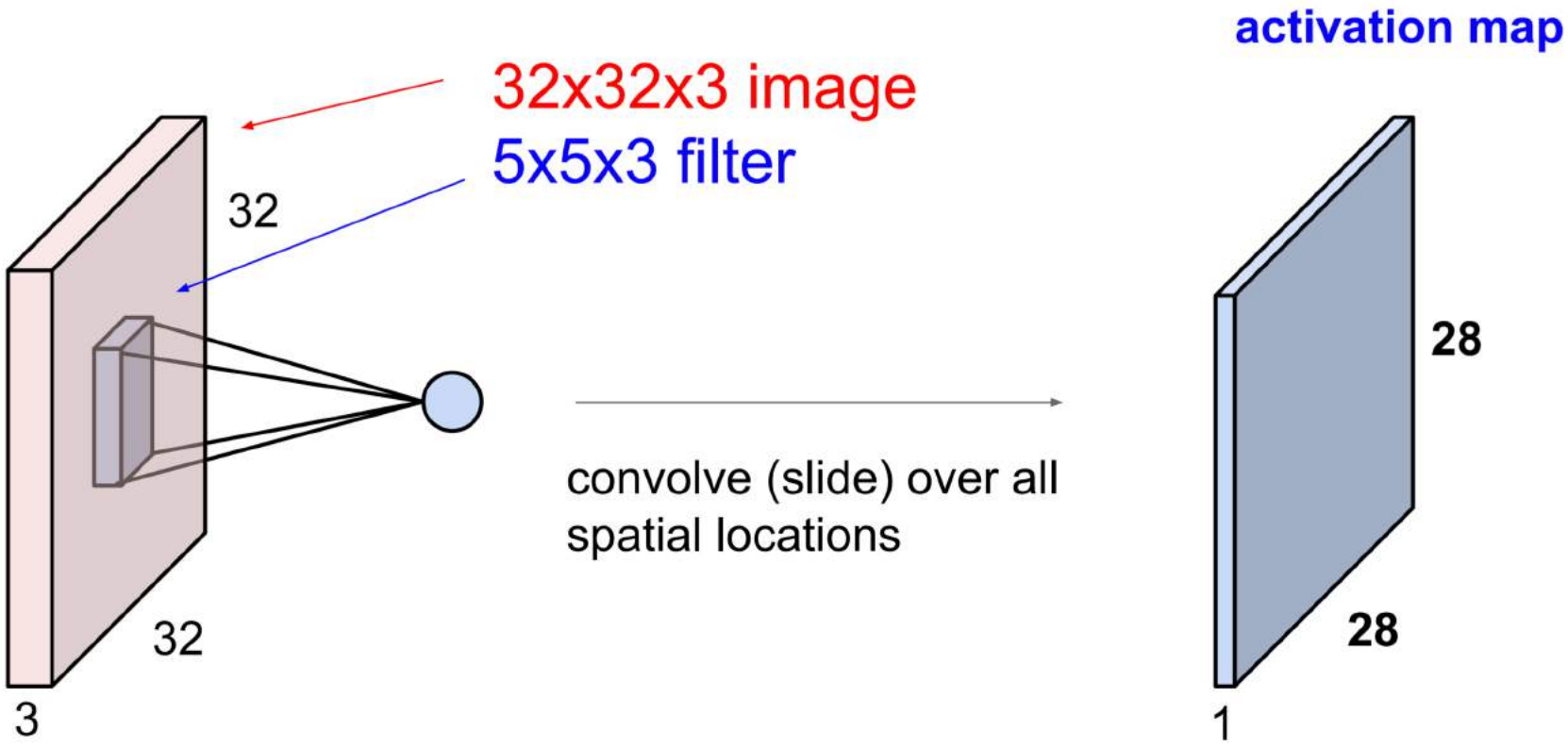
We call the layer convolutional because it is related to convolution of two signals:

$$f[x,y] * g[x,y] = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} f[n_1,n_2] \cdot g[x-n_1, y-n_2]$$

elementwise multiplication and sum of a filter and the signal (image)

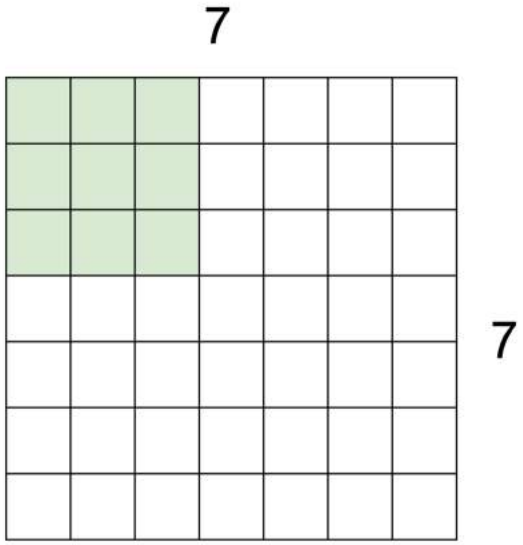
Convolutional layer

A closer look at spatial dimensions:



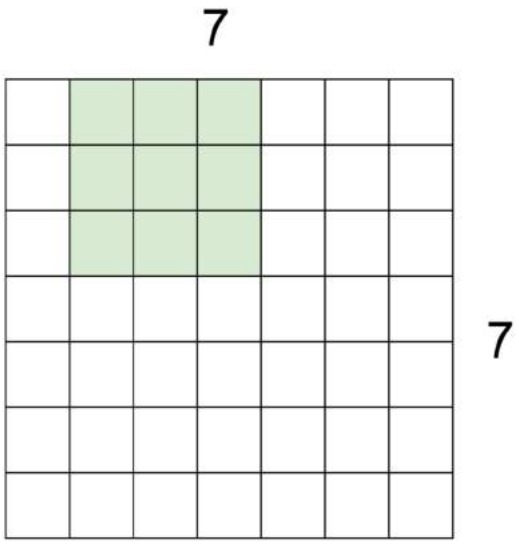
Convolutional layer

A closer look at spatial dimensions:



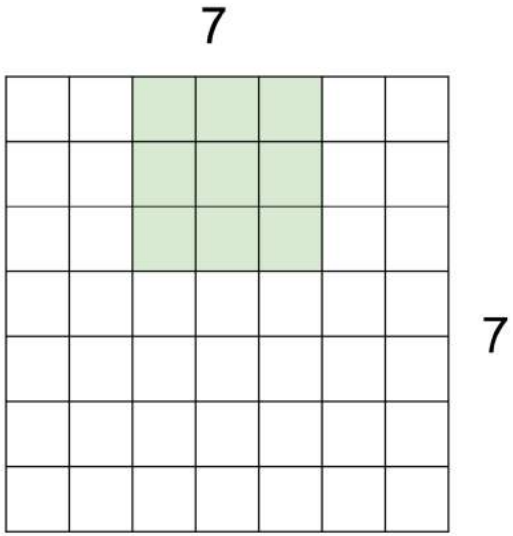
7x7 input (spatially)
assume 3x3 filter

Convolutional layer



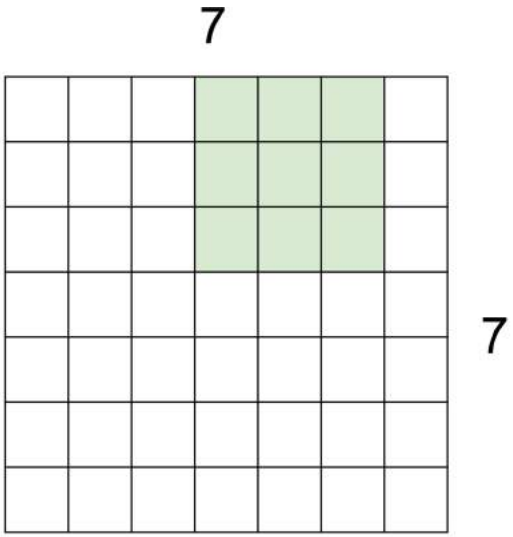
7x7 input (spatially)
assume 3x3 filter

Convolutional layer



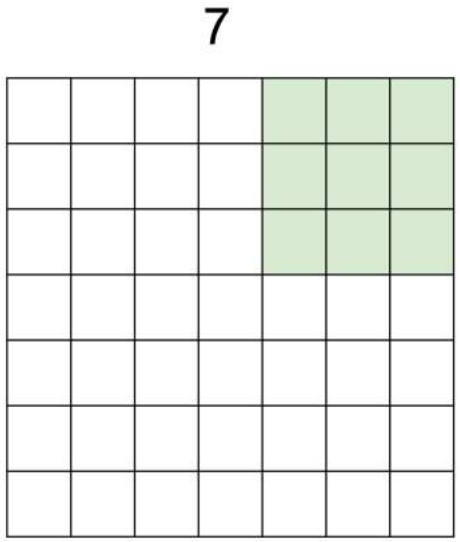
7x7 input (spatially)
assume 3x3 filter

Convolutional layer



7x7 input (spatially)
assume 3x3 filter

Convolutional layer

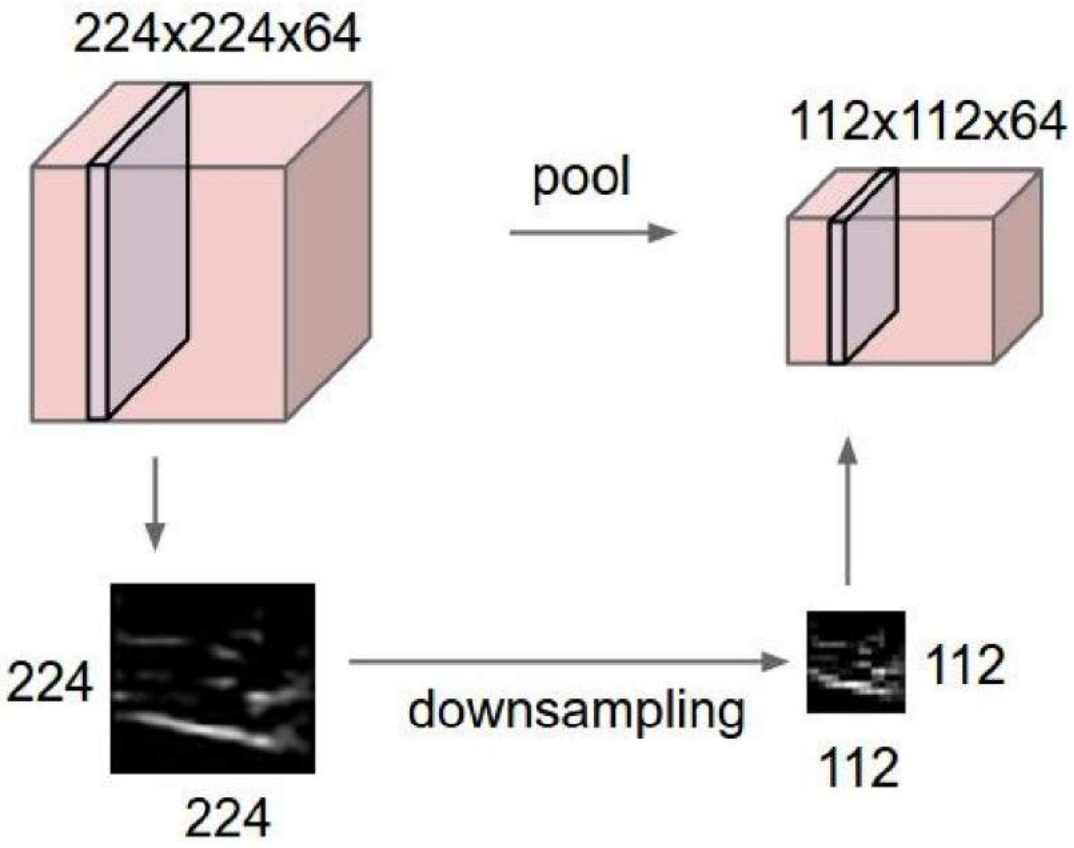


7x7 input (spatially)
assume 3x3 filter

=> 5x5 output

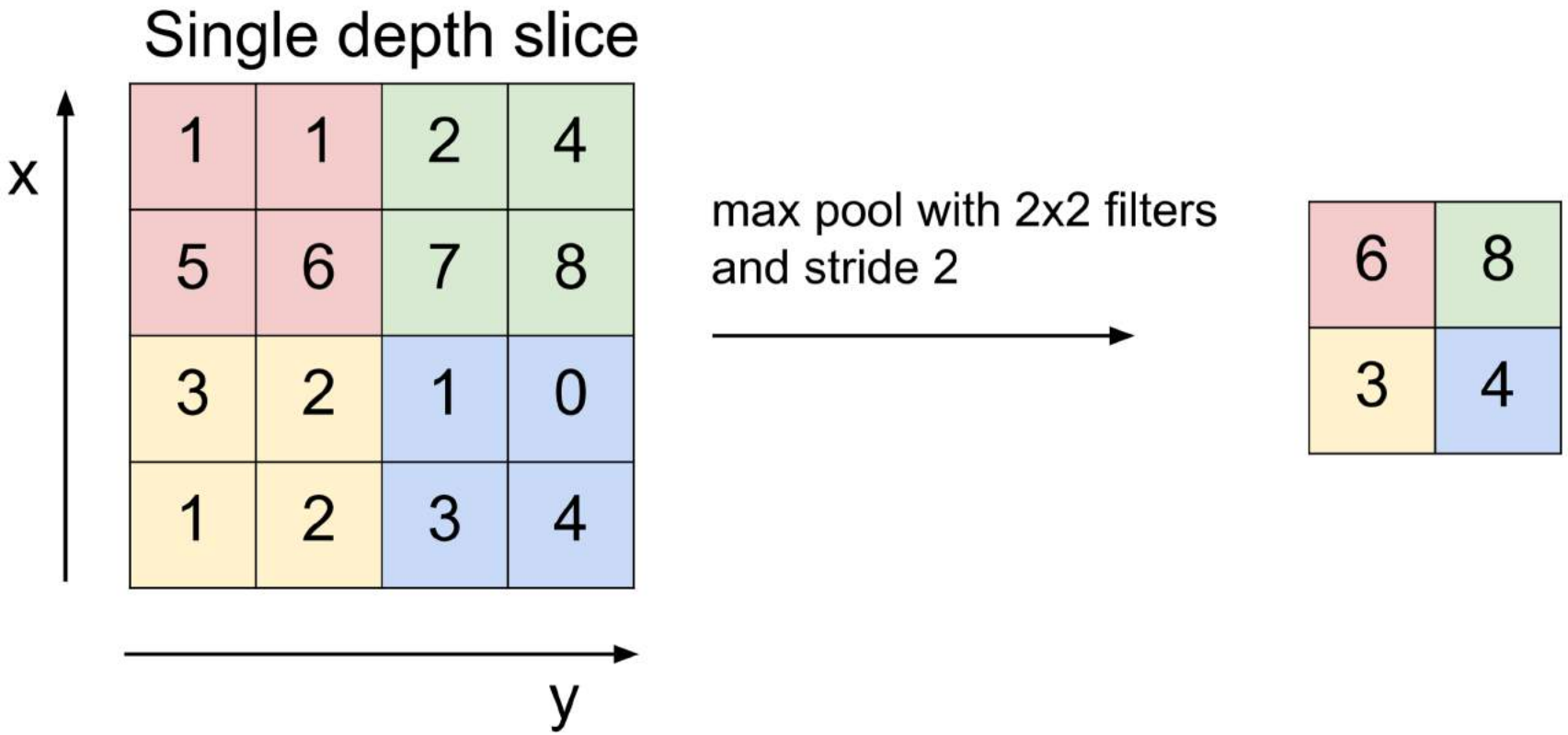
Pooling layer

- makes the representations smaller and more manageable
- operates over each activation map independently:

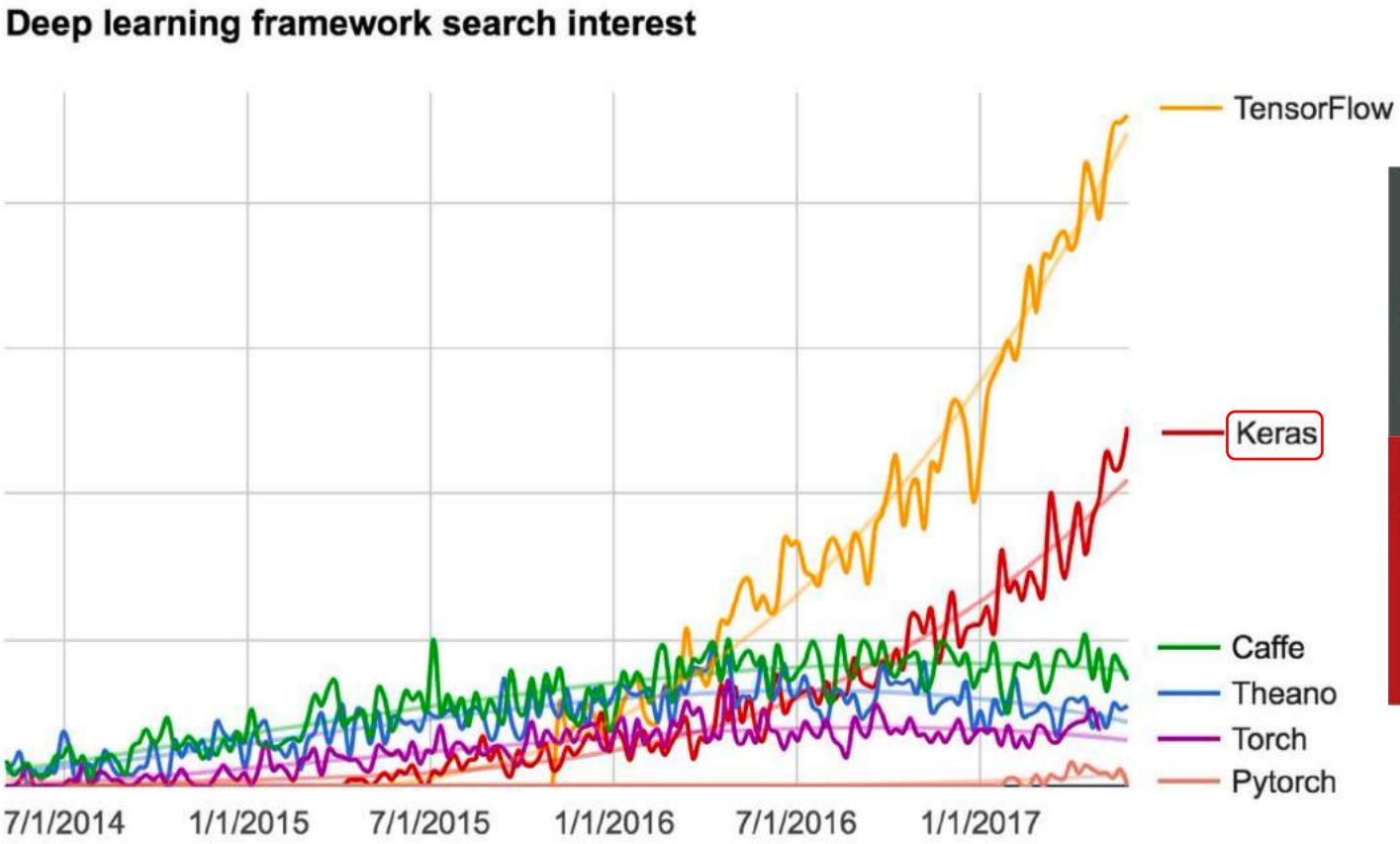


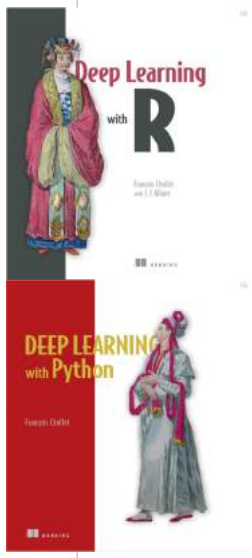
Pooling layer

MAX POOLING



Frameworks


Deep learning framework search interest
TensorFlow
Keras
Caffe
Theano
Torch
Pytorch
7/1/2014
1/1/2015
7/1/2015
1/1/2016
7/1/2016
1/1/2017


Deep Learning
with
R
DEEP LEARNING
with Python

Real study: AMD

nature machine intelligence

ARTICLES

https://doi.org/10.1038/s42256-020-0154-9

Deep-learning-based prediction of late age-related macular degeneration progression

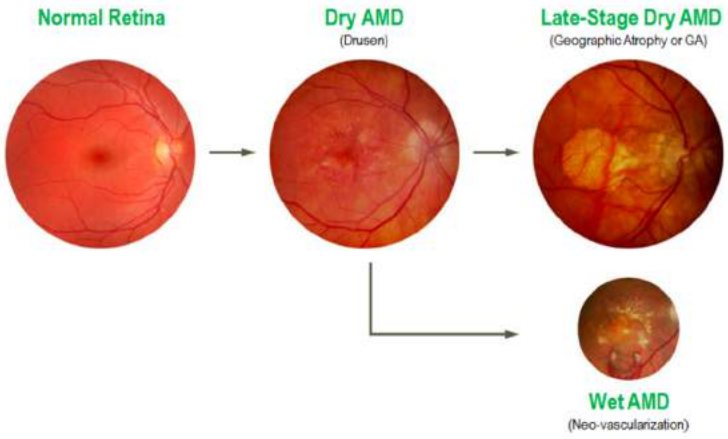
Qi Yan[1,7] ✉, Daniel E. Weeks[2,3], Hongyi Xin[1], Anand Swaroop[4], Emily Y. Chew[5], Heng Huang[6], Ying Ding[3,7] ✉ and Wei Chen[1,2,3,7] ✉

Both genetic and environmental factors influence the etiology of age-related macular degeneration (AMD), a leading cause of blindness. AMD severity is primarily measured by images of the fundus of the retina and recently developed machine learning methods can successfully predict AMD progression using image data. However, none of these methods have used both genetic and image data for predicting AMD progression. Here we used both genotypes and fundus images to predict whether an eye had progressed to late AMD with a modified deep convolutional neural network. In total, we used 31,262 fundus images and 52 AMD-associated genetic variants from 1,351 subjects from the Age-Related Eye Disease Study, which provided disease severity phenotypes and fundus images available at baseline and follow-up visits over a period of 12 years. Our results showed that fundus images coupled with genotypes could predict late AMD progression with an averaged area-under-the-curve value of 0.85 (95% confidence interval 0.83–0.86). The results using fundus images alone showed an averaged area under the receiver operating characteristic curve value of 0.81 (95% confidence interval 0.80–0.83). We implemented our model in a cloud-based application for individual risk assessment.

Real study: AMD

Background

- **A**ge-related **M**acular **D**egeneration (AMD) is a heritable neurodegenerative disease and a leading cause of blindness in the elderly population in the United States.
- AMD severity is mainly diagnosed by color fundus images and recent studies have shown the success of machine learning methods in predicting AMD progression using image data.
- In this study, we jointly used genotypes and fundus images to dynamically predict an eye as having progressed to late AMD with a modified deep convolutional neural network (CNN).
- Study Population: Caucasian patients from AREDS (Age-Related Eye Disease study) including genotyping data, longitudinal color fundus photographs, and disease severity assessment over a period of 12 years.



a The view from a healthy eye

b The view from an eye suffering AMD

https://www.clarityeye.net/age-related-macular-degeneration/

Real study: AMD

National Eye Institute (NEI) Age-Related Eye Disease Study (AREDS)

dbGaP Study Accession: phs000001.v3.p1

Study Description

The Age-Related Eye Disease Study (AREDS) was initially designed as a long-term multi-center, prospective study of the clinical course of age-related macular degeneration (AMD) and age-related cataract. In addition to collecting natural history data, AREDS included a clinical trial of high-dose vitamin and mineral supplements for AMD and a clinical trial of high-dose vitamin supplements for cataract. AREDS participants were **55 to 80 years of age** at enrollment and had to be free of any illness or condition that would make long-term follow-up or compliance with study medications unlikely or difficult. On the basis of fundus photographs graded by a central reading center, best-corrected visual acuity and ophthalmologic evaluations, **4,757 participants** were enrolled in one of several AMD categories, including persons with no AMD.

- AREDS participants were followed on the clinical trials for a **median time of 6.5 years**. Subsequent to the conclusion of the clinical trials, participants were followed for **an additional 5 years** and natural history data were collected.
- Blood samples were also collected from **3,700+ AREDS participants for genetic research**. However, not all of the 3,700+ AREDS participants who submitted a blood sample currently have DNA available.
- In November 2010, over **72,000 high quality fundus and lens photographs of 595 AREDS participants** were made available in the AREDS dbGaP.
- In February 2014 over **134,500 high-quality fundus photographs of 4613 AREDS participants** were added to the existing AREDS dbGaP resource.

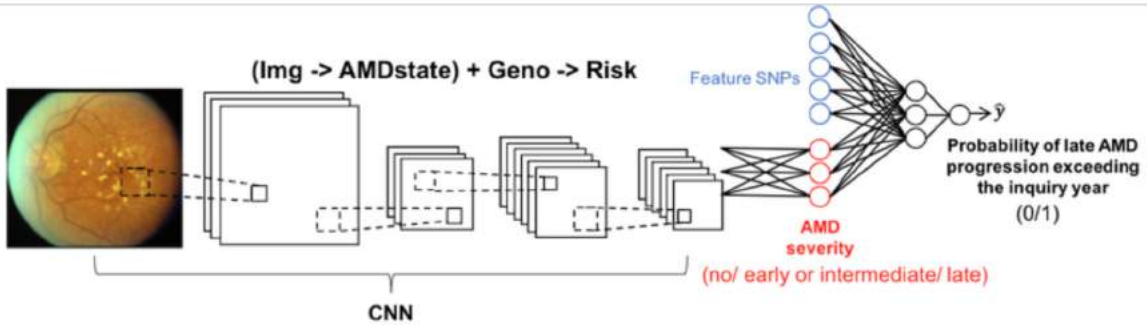
Real study: AMD

http://www.pitt.edu/~qiy17/amdprediction.html

Late AMD Fundus Image Prediction

Predictor

Fundus image + Genotypes -> Current AMD severity + Advanced AMD progression chance



Year 2 | Year 3 | Year 4 | Year 5 | Year 6 | Year 7

Fundus Image (required, image must be png, less than 1Mb): Select Image

Genotype File (optional, 52 SNPs): Select File

RUN RESET

Note: The models were trained using only Caucasians with age above 55 years.

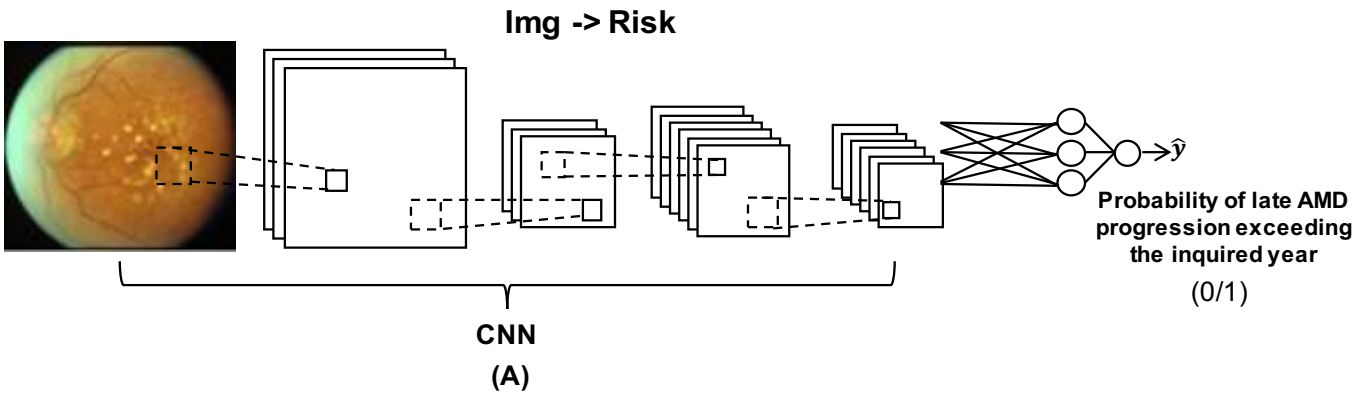
Disclaimer: This website is made available to give you a general understanding of fundus images and genetics on AMD risk, not to provide specific clinical advice.

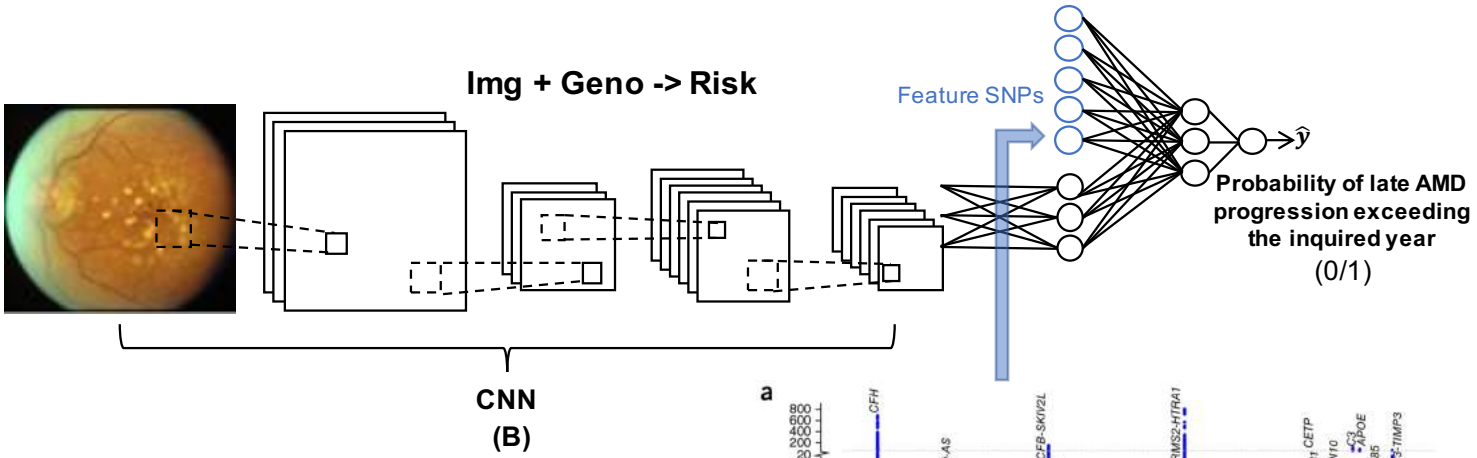
Real study: AMD

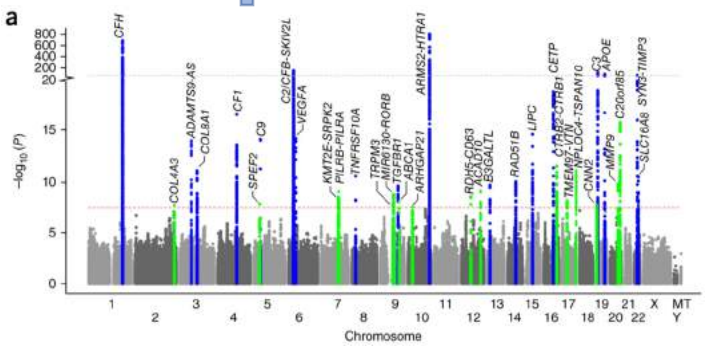
Table 1 | Characteristics of the participants

	AREDS	Training	Test
Subject-level variables	1,351 subjects	1,223 subjects	128 subjects
Baseline age, year (mean ± s.d.)	68.8 ± 5.0	68.8 ± 5.0	68.5 ± 4.8
Female (*N*, %)	750 (55.5)	682 (55.8)	68 (53.1)
Follow-up time, (mean ± s.d.)	10.3 ± 1.6	10.2 ± 1.7	10.9 ± 1.0
Baseline smoking status (*N*, %)			
Never smoked	626 (46.3)	566 (46.3)	60 (46.9)
Former smoker	634 (46.9)	576 (47.1)	58 (45.3)
Current smoker	91 (6.7)	81 (6.6)	10 (7.8)
Eye-level variables	2,678 eyes	2,422 eyes	256 eyes
Baseline AMD severity score at eye-level			
Mean ± s.d.	3.9 ± 3.2	4.0 ± 3.2	3.9 ± 3.2
1–3 (*N*, %)	1,442 (53.8)	1,310 (54.1)	132 (51.6)
4–6 (*N*, %)	600 (22.5)	528 (21.8)	72 (28.1)
7–8 (*N*, %)	636 (23.7)	584 (24.1)	52 (20.3)
Progressed eyes with baseline severity			
1–3 (*N*, %)	50 (3.5)	48 (3.7)	2 (1.5)
4–6 (*N*, %)	300 (50.0)	260 (49.2)	40 (55.6)
7–8 (*N*, %)	585 (92.0)	537 (92.0)	48 (92.3)
Observation-level variables			
Number of fundus images used for prediction with progression cutoff			
2 years	27,499	24,654	2,845
3 years	25,862	23,170	2,692
4 years	24,287	21,709	2,578
5 years	22,435	20,041	2,394
6 years	20,240	18,118	2,122
7 years	18,066	16,172	1,894

Real study: AMD

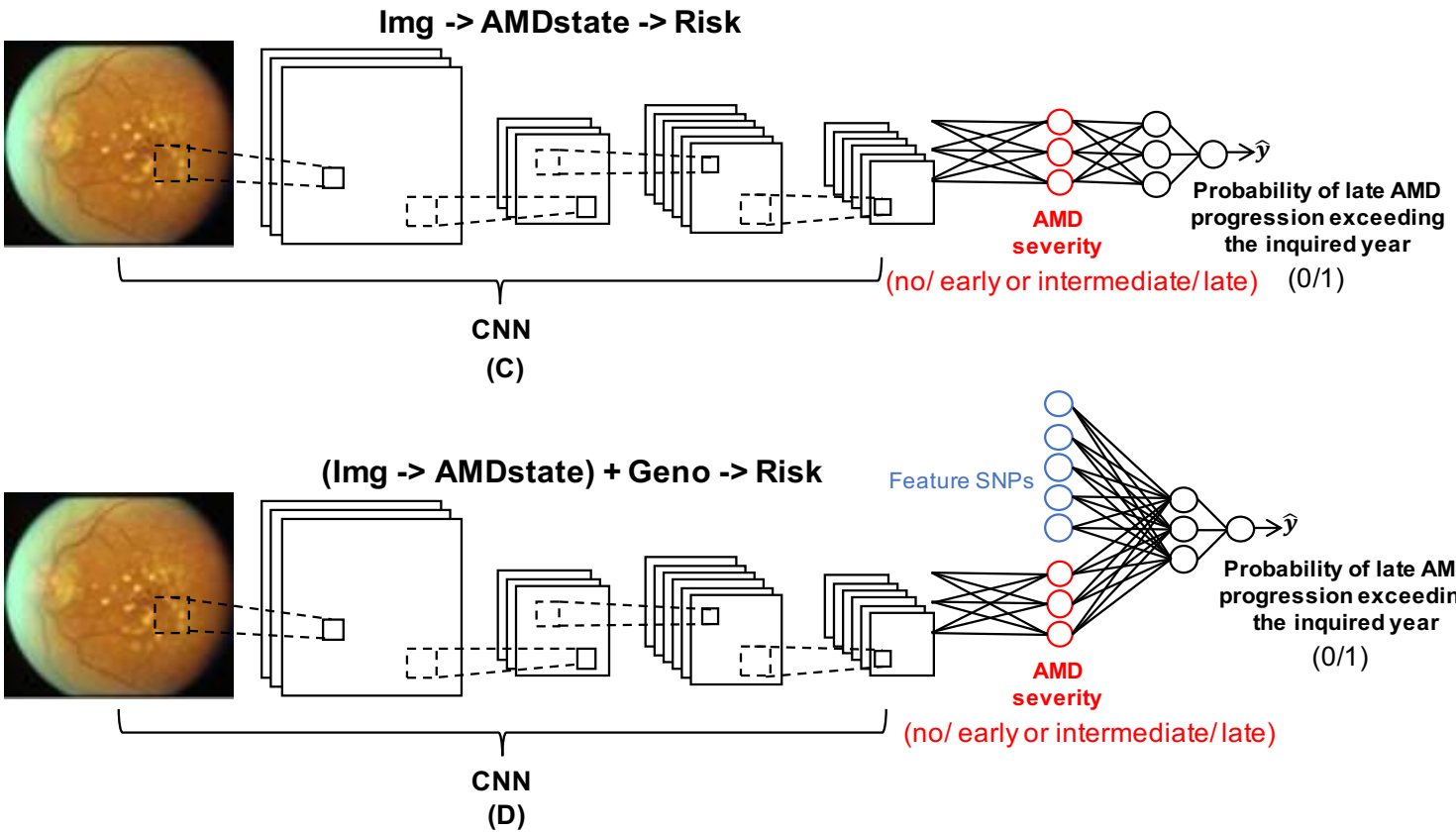






Nat Genet. 2016 Feb;48(2):134-43. doi: 10.1038/ng.3448

Real study: AMD



Real study: AMD

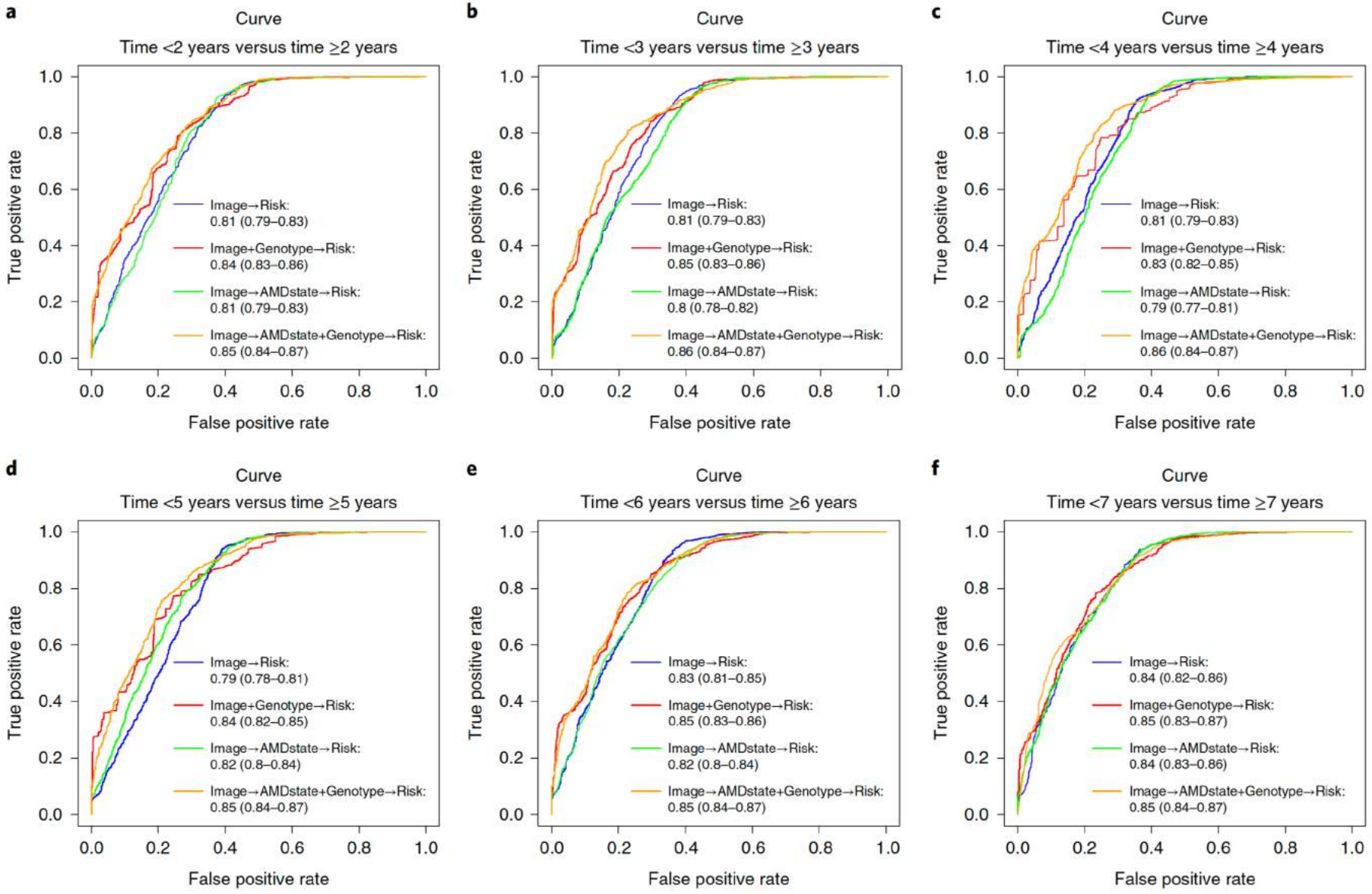


Fig. 1 | Receiver operator characteristic curves of the prediction of late AMD progression time exceeding the inquired years for four models.

Real study: AMD

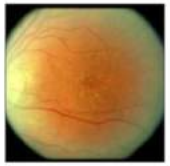
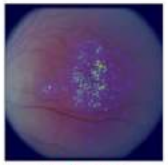
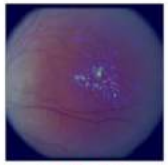
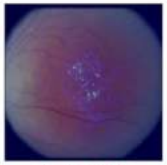
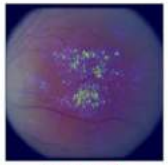
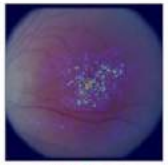
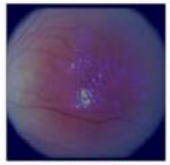
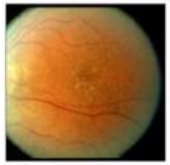
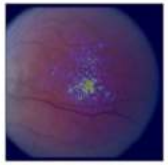
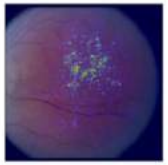
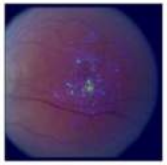
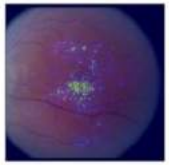
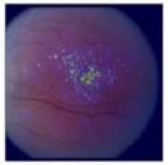
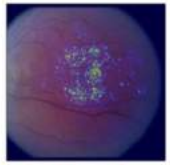
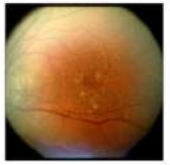
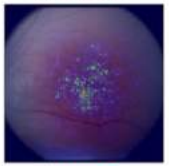
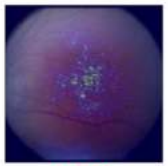
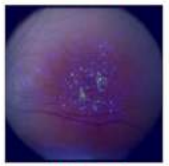
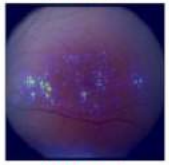
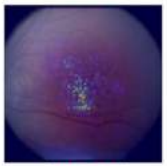
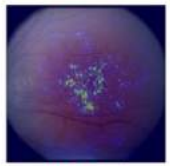
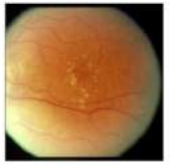
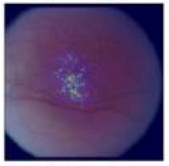
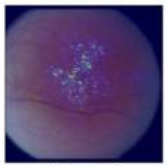
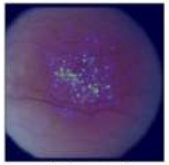
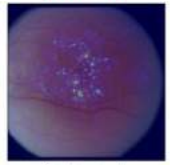
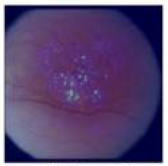
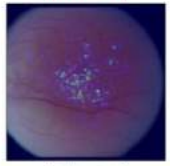
Visit year	Time left to censored time (4.8 years)	Original images	Saliency maps (true label/predicted probability)					
			0: <2 years 1: ≥2 years	0: <3 years 1: ≥3 years	0: <4 years 1: ≥4 years	0: <5 years 1: ≥5 years	0: <6 years 1: ≥6 years	0: <7 years 1: ≥7 years
		(Youden index)	(0.69)	(0.61)	(0.67)	(0.50)	(0.52)	(0.42)
0	4.8		 (1/0.95)	 (1/0.95)	 (1/0.63)	 (0/0.27)	 (0/0.01)	 (0/0.10)
2	2.8		 (1/0.88)	 (0/0.98)	 (0/0.46)	 (0/0.31)	 (0/0.09)	 (0/0.19)
4	0.8		 (0/0.58)	 (0/0.80)	 (0/0.17)	 (0/0.06)	 (0/0.00)	 (0/0.06)
5.9	0		 (0/0.03)	 (0/0.10)	 (0/0.05)	 (0/0.05)	 (0/0.00)	 (0/0.00)

Fig. 2 | Saliency maps for left eye of subject 1 over 5.9 years. This subject progressed to late AMD after 4.8 years of follow-up.

Real study: AMD

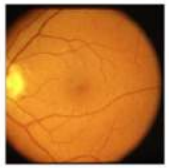
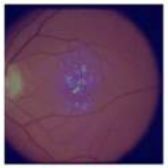
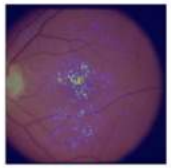
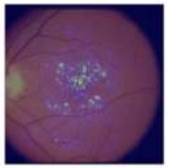
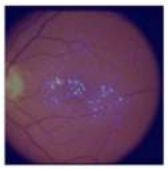
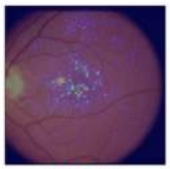
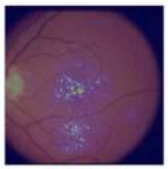
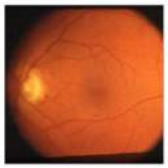
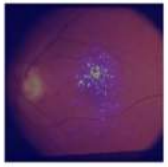
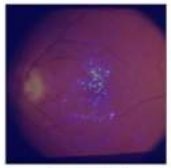
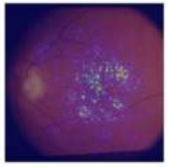
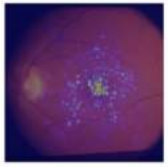
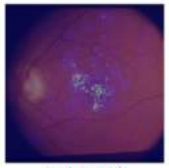
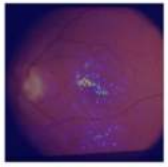
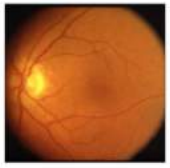
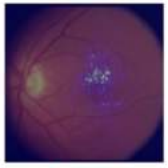
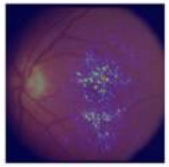
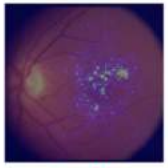
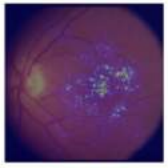
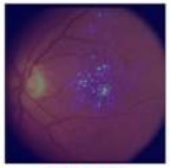
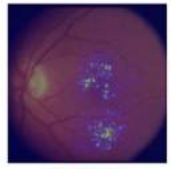
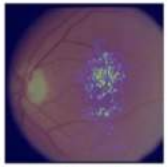
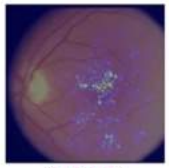
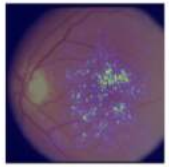
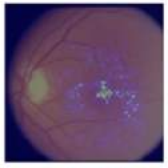
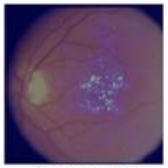
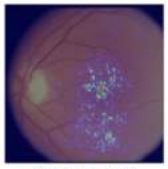
Visit year	Time left to censored time (11.1 years)	Original images	Saliency maps (true label/predicted probability)					
			0: <2 years 1: ≥2 years	0: <3 years 1: ≥3 years	0: <4 years 1: ≥4 years	0: <5 years 1: ≥5 years	0: <6 years 1: ≥6 years	0: <7 years 1: ≥7 years
		(Youden index)	(0.69)	(0.61)	(0.67)	(0.50)	(0.52)	(0.42)
0	11.1		 (1/0.99)	 (1/0.99)	 (1/1.00)	 (1/0.99)	 (1/1.00)	 (1/1.00)
1.9	9.2		 (1/1.00)	 (1/0.99)	 (1/0.99)	 (1/0.99)	 (1/1.00)	 (1/1.00)
3.8	7.3		 (1/0.99)	 (1/1.00)	 (1/0.99)	 (1/0.99)	 (1/1.00)	 (1/1.00)
5.8	5.3		 (1/0.99)	 (1/1.00)	 (1/0.99)	 (1/0.99)	 (NA/1.00)	 (NA/1.00)

Fig. 3 | Saliency maps for left eye of subject 2 over the first 5.8 years. This subject was censored after 11.1 years of follow-up.

Real study: AMD

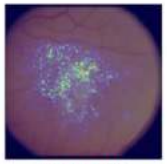
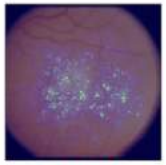
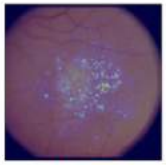
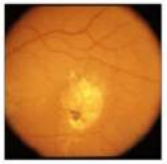
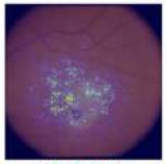
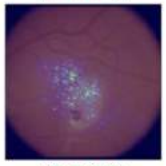
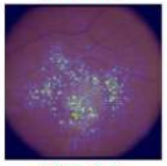
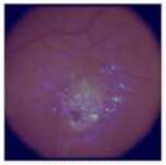
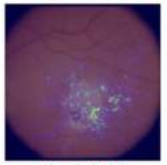
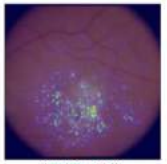
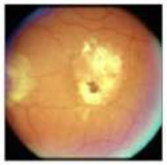
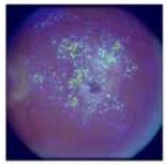
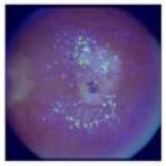
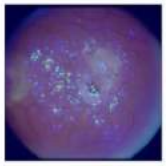
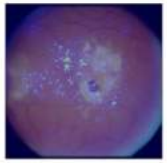
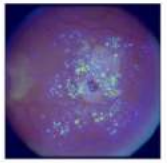
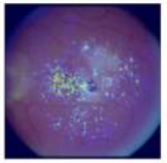
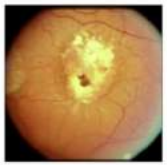
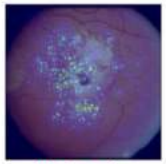
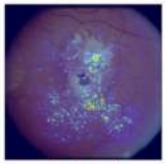
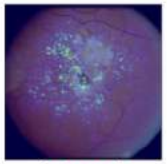
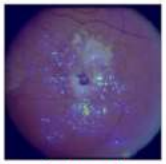
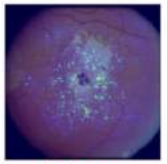
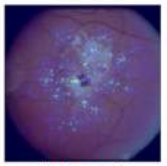
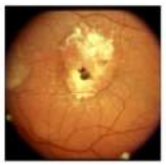
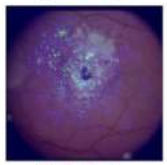
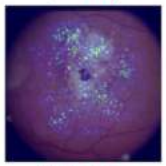
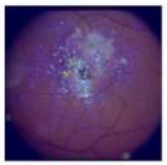
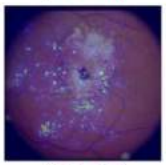
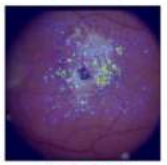
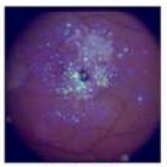
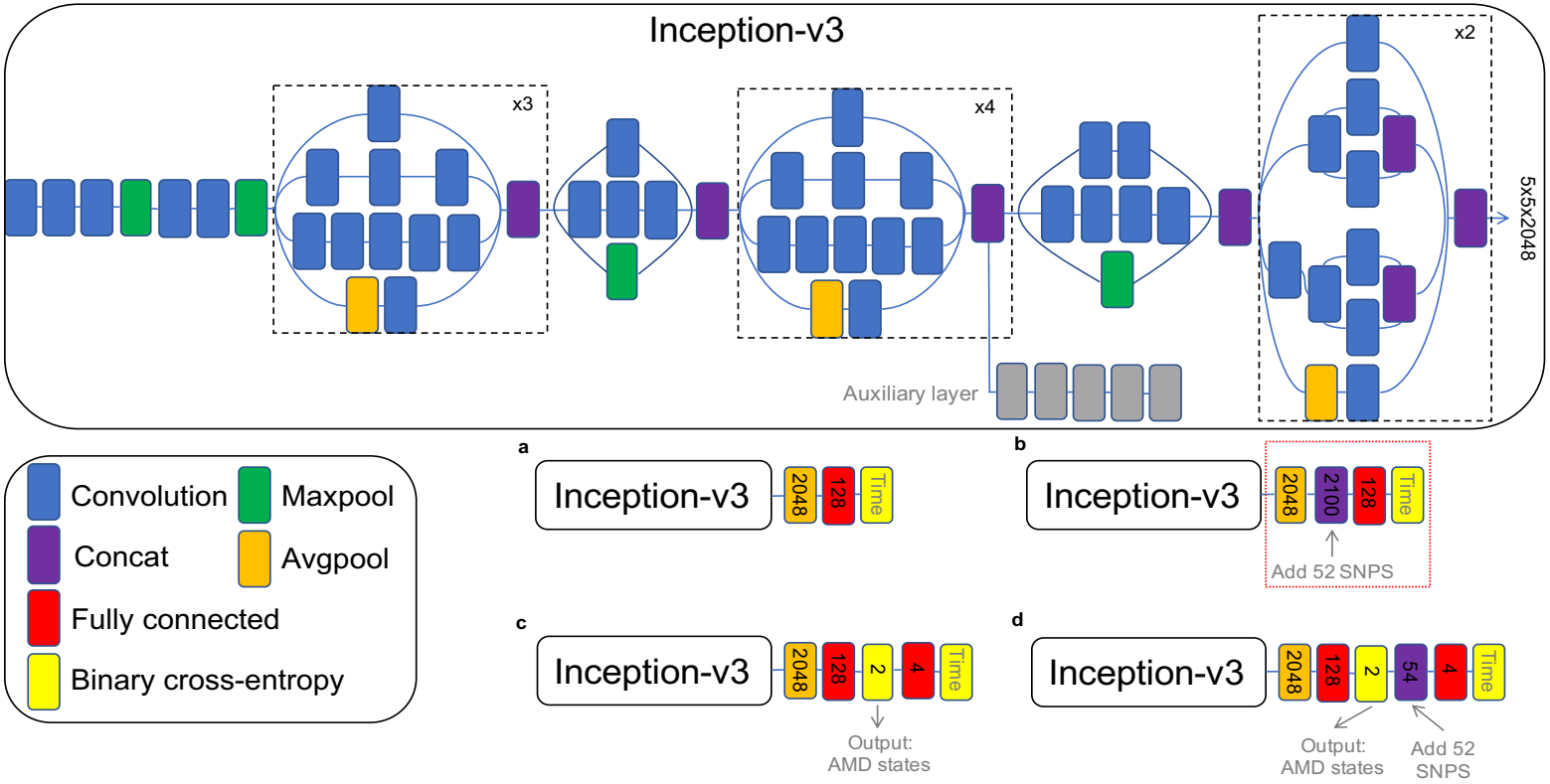
Visit year	Time left to late AMD progression (0 years)	Original images	Saliency maps (true label/predicted probability)					
			0: <2 years 1: ≥2 years	0: <3 years 1: ≥3 years	0: <4 years 1: ≥4 years	0: <5 years 1: ≥5 years	0: <6 years 1: ≥6 years	0: <7 years 1: ≥7 years
		(Youden index)	(0.69)	(0.61)	(0.67)	(0.50)	(0.52)	(0.42)
0	0		 (0/0.00)	 (0/0.01)	 (0/0.07)	 (0/0.03)	 (0/0.01)	 (0/0.06)
1.9	0		 (0/0.01)	 (0/0.02)	 (0/0.04)	 (0/0.05)	 (0/0.00)	 (0/0.02)
7.8	0		 (0/0.00)	 (0/0.01)	 (0/0.00)	 (0/0.00)	 (0/0.00)	 (0/0.00)
10	0		 (0/0.01)	 (0/0.01)	 (0/0.01)	 (0/0.00)	 (0/0.00)	 (0/0.00)
12	0		 (0/0.01)	 (0/0.00)	 (0/0.01)	 (0/0.02)	 (0/0.00)	 (0/0.00)

Fig. 4 | Saliency maps for left eye of subject 3 over 12 years. This subject developed late AMD before enrollment.

Real study: AMD



Other options:

AlexNet, GoogLeNet, VGG, ResNet, Inception-ResNet-V2 …

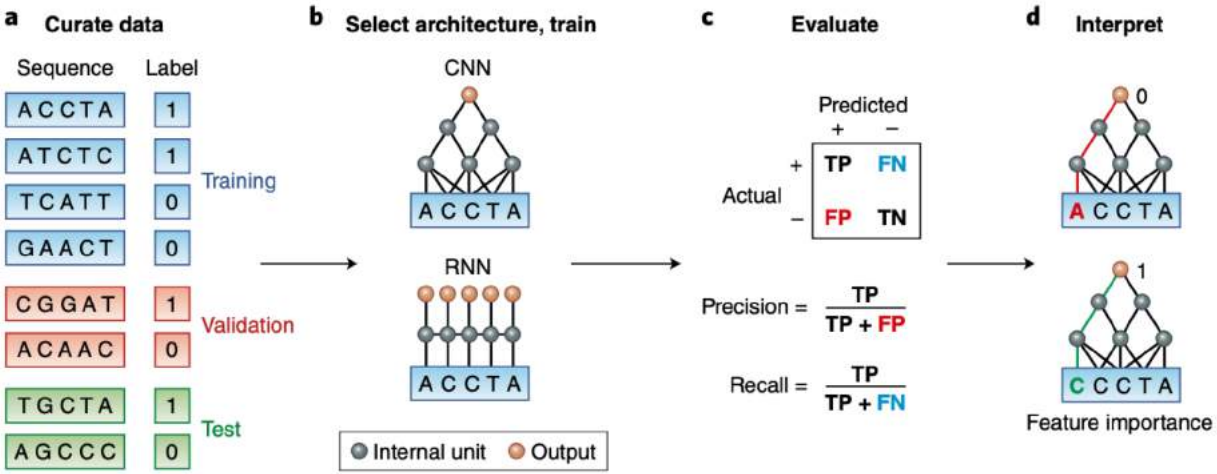
Possibilities

PERSPECTIVE
https://doi.org/10.1038/s41588-018-0295-5
nature genetics

A primer on deep learning in genomics

James Zou[1,2,3]*, Mikael Huss[4,5], Abubakar Abid[3], Pejman Mohammadi[6,7], Ali Torkamani[6,7] and Amalio Telenti[6,7]*

Deep learning methods are a class of machine learning techniques capable of identifying highly complex patterns in large data-sets. Here, we provide a perspective and primer on deep learning applications for genome analysis. We discuss successful applications in the fields of regulatory genomics, variant calling and pathogenicity scores. We include general guidance for how to effectively use deep learning methods as well as a practical guide to tools and resources. This primer is accompanied by an interactive online tutorial.



Deep learning workflow in genomics. a, A dataset should be randomly split into training, validation and test sets. The positive and negative examples should be balanced for potential confounders (for example, sequence content and location) so that the predictor learns salient features rather than confounders. b, The appropriate architecture is selected and trained on the basis of domain knowledge. For example, CNNs capture translation invariance, and RNNs capture more flexible spatial interactions. c, True positive (TP), false positive (FP), false negative (FN) and true negative (TN) rates are evaluated. When there are more negative than positive examples, precision and recall are often considered. d, The learned model is interpreted by computing how changing each nucleotide in the input affects the prediction.

Nat Genet. 2019 Jan;51(1):12-18. doi: 10.1038/s41588-018-0295-5.

Possibilities

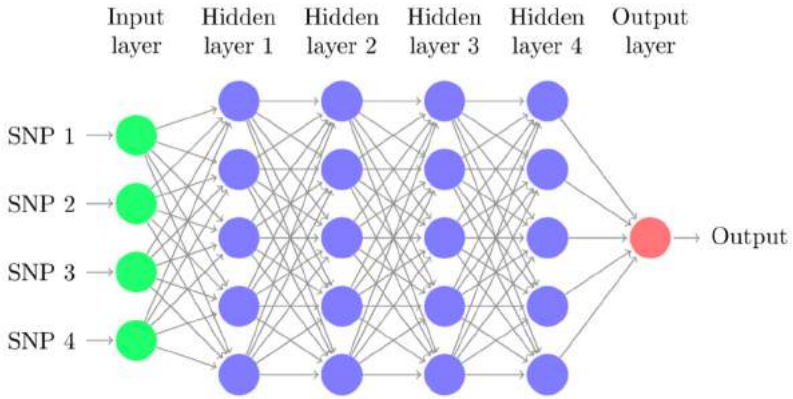
genes

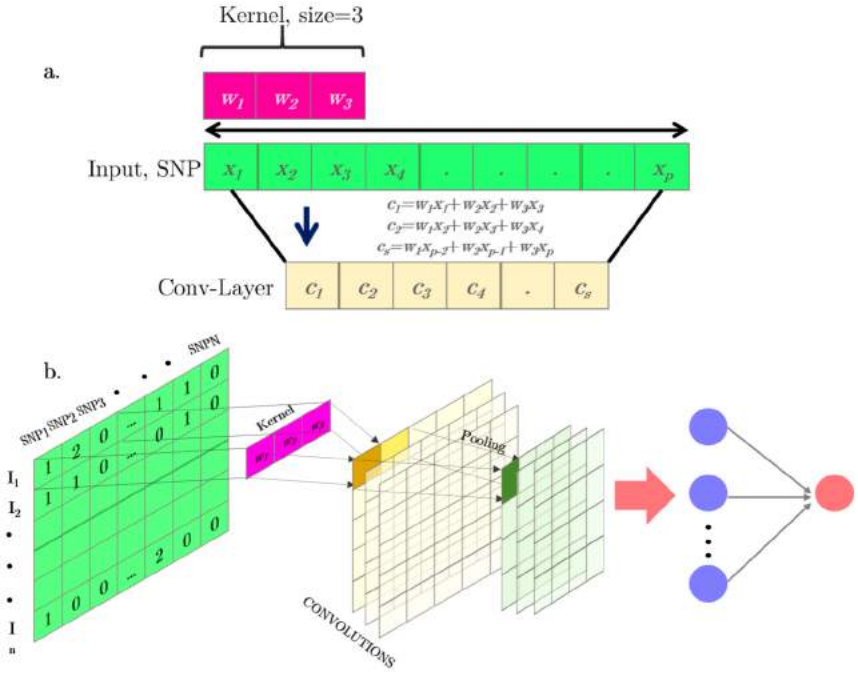
MDPI

Review

A Guide on Deep Learning for Complex Trait Genomic Prediction

Miguel Pérez-Enciso [1,2,*] and Laura M. Zingaretti [2]





Genes (Basel). 2019 Jul 20;10(7). pii: E553. doi: 10.3390/genes10070553.