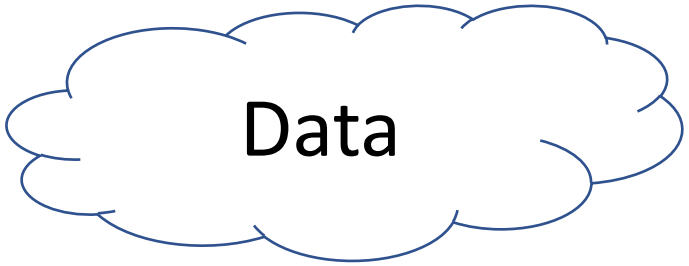
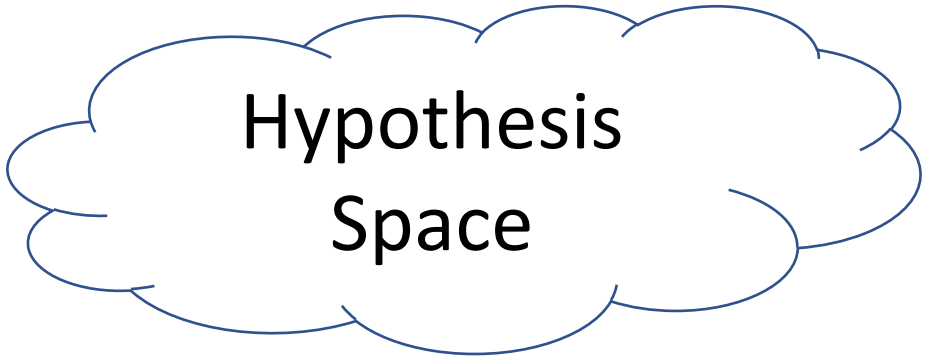
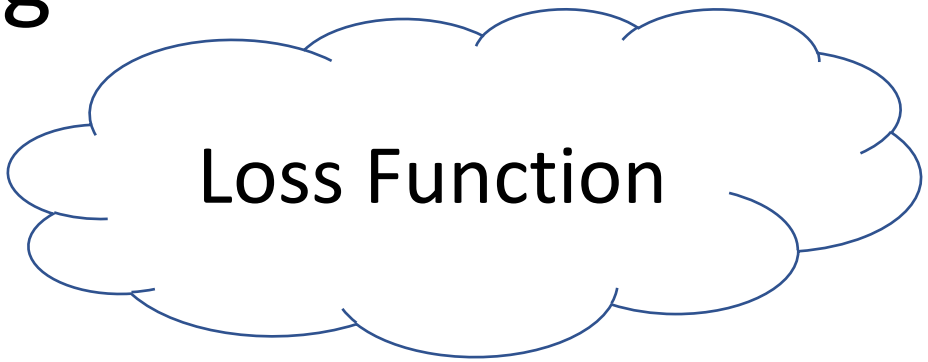


Three Components of Machine Learning

Alex Jung





Background Material

- course book: F. Chollet, “Deep Learning with Python”

https://aalto.finna.fi/Record/alli.833878

- A. Jung, “Machine Learning: Basic Principles” https://arxiv.org/abs/1805.05052

- I. Goodfellow, Y. Bengio and A. Courville, “Deep Learning”

https://www.deeplearningbook.org/

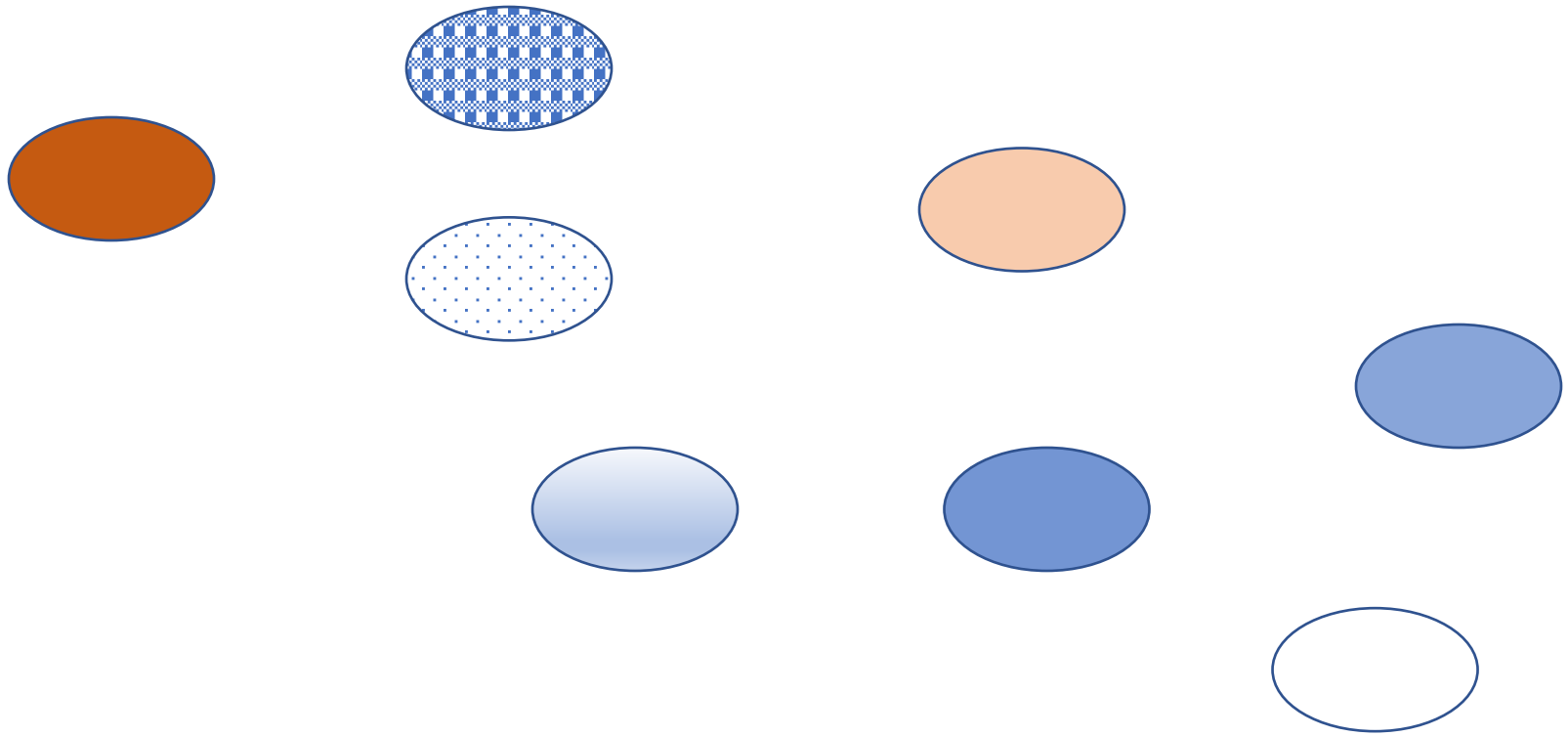
Corona ?

Three Main Components

- data (video, audio, text)
- model (network architecture)
- loss function (performance measure)

Data

Dataset = (Large) Set of “Data Points”



data points are different objects but of similar “type”

Dataset – “Cows”

Dataset – “Forests”

Dataset = “Days During Pandemic”

1/Mar/2020

13/Mar/2020

2/Mar/2020

22/Apr/2020

1/Apr/2020

Data Point = Atomic Unit of Information

- highly abstract concept
- data points can represent persons
- data points can represent random variables
- data points can represent machine learning problems

Features and Labels

- data points often have many different properties
- “features”: properties that can be measured/computed easily
- “labels”: properties that are “difficult” to compute
- labels are higher-level facts or quantities of interest
- labels can often be determined only in hindsight
- determining labels might require human (domain) experts

Corona ?

data point = “some human”

features = body temp., heart rate,…

label = 1 if Covid-19 infection,
0 otherwise

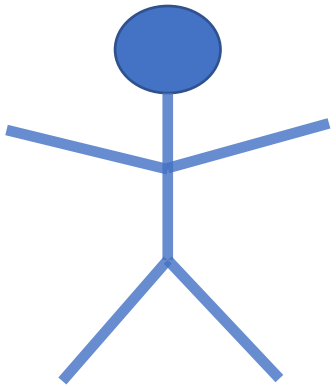
Data Point = “Some Photo”

features could include:

- red, green, blue intensities of pixels
- timestamp of photo
- location of photo shot
- identity of photographer

label:
hiking duration for the photographer to reach mountain peak

Data Point = “Some Person”



features:

- name
- healthcare records
- credit card transactions
- social media posts
- genetic fingerprint
- fingerprint
- travel history

label:

how likely will person need intensive care next week?

Data Point = “Some Dataset”

features:

- number of data points
- what type of features are used for data points
- what label is used for data points

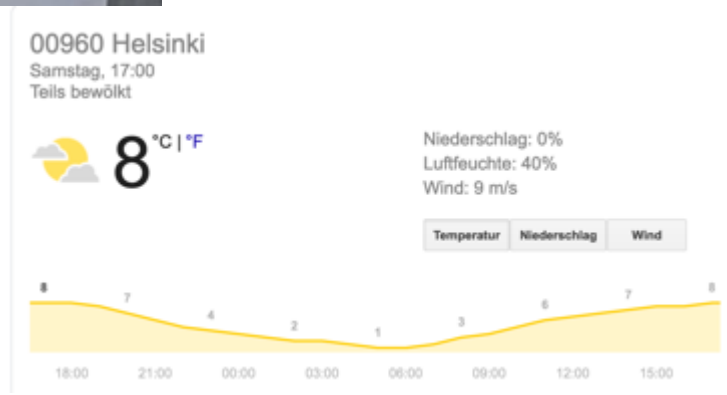
label:

can label value be predicted well from features?

Data Point = “Some Ski-Day Ahead”

features:

- snapshot in the morning
- morning temperature
- weather forecast



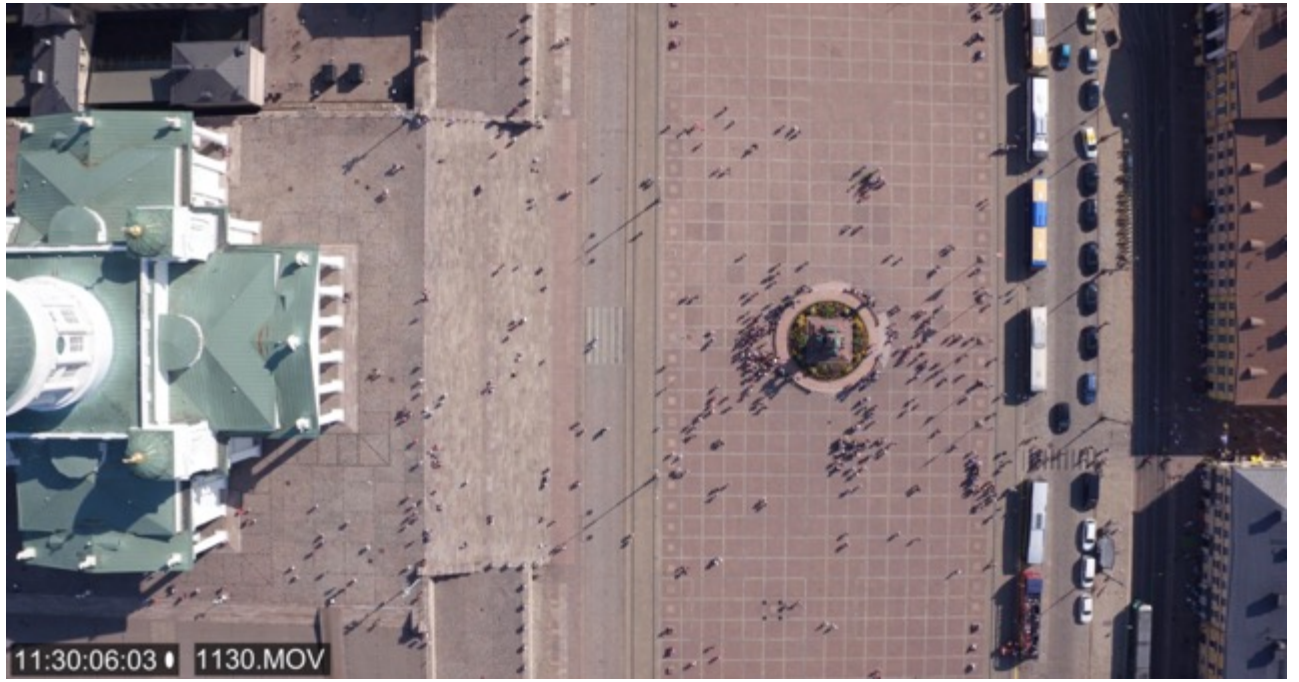
label:

- maximum daytime temperature (important for ski waxing)

Data Point = “Place in Helsinki”

features:

- coordinates of place
- city building maps
- current traffic statistics
- CET time
- drone video



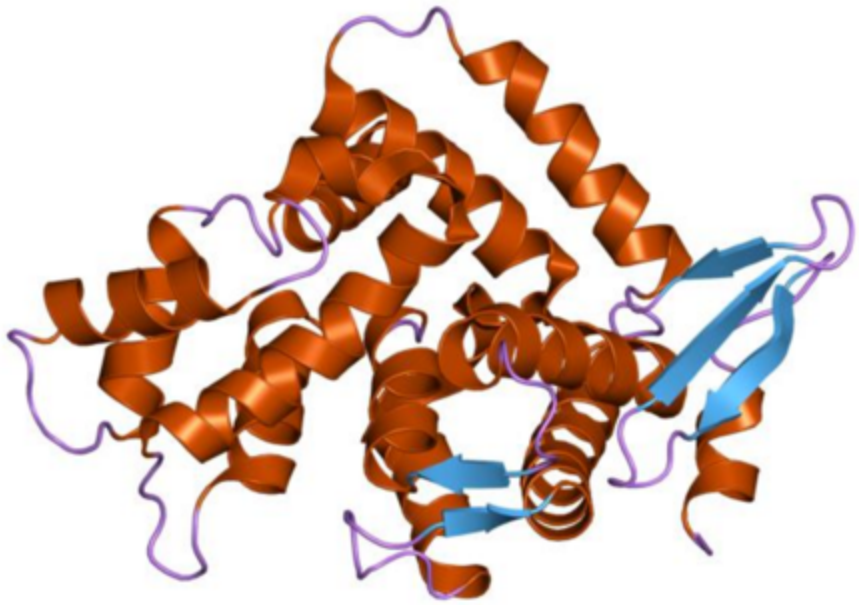
label:

are people keeping average distance of >= 1 m ?

Data Point = “Some Protein”

features:

- protein structure
- physical measurements
- scientific papers about this protein



label:

should this protein be considered for a Covid-19 vaccine?

Data Point = “Some Plant”

features:
- plant species
- RGB image
- multi-spectral image
- ambient temperature

label:
does the plant need more water?

Features and Labels Are Design Choices!

there is often a design freedom for defining/choosing features and labels

in some application the labels are higher-level facts that are defined by human experts who provide some labeled training examples

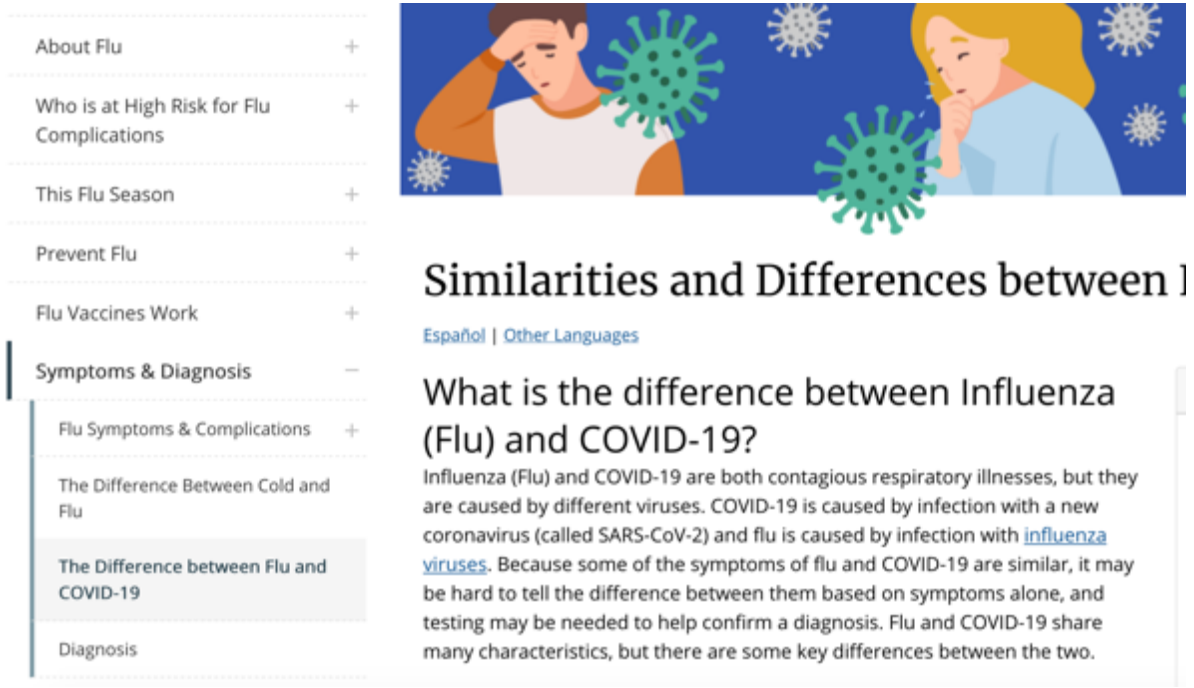
in other applications, labels are just subsets of features and we can get labeled data without any human labeling workers

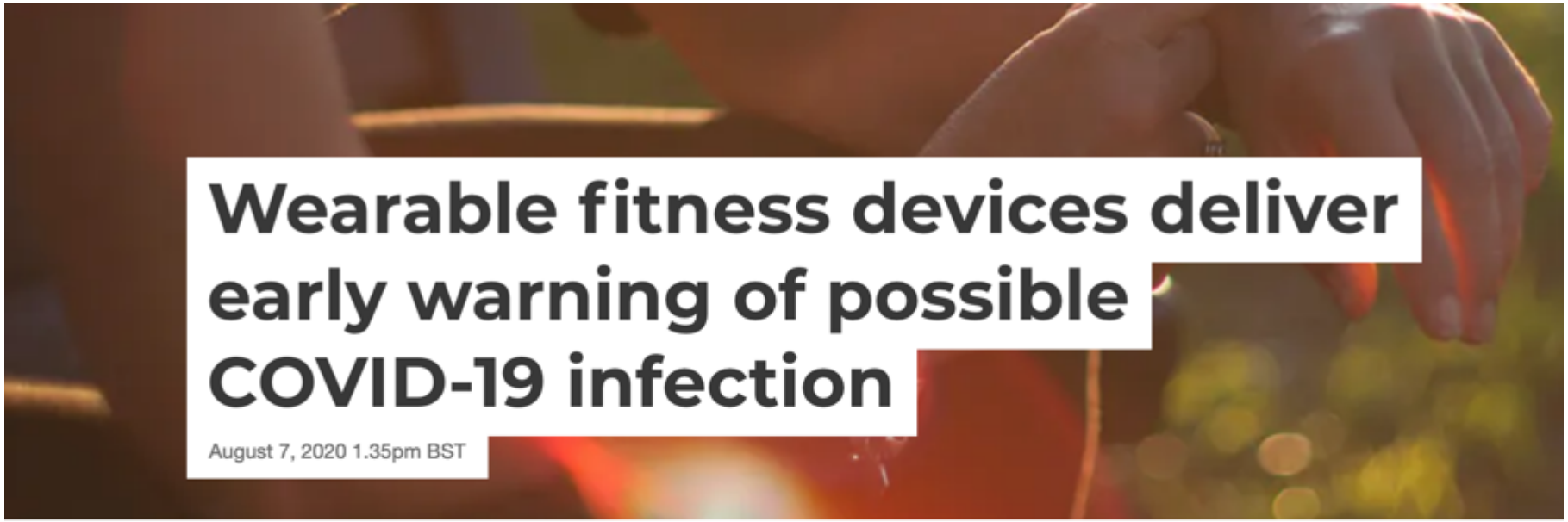
consider weather forecasts (data points = some day, features are short time history of temperature and label is 2 day ahead temperature

we get labeled data from historic recordings

Influenza or Covid-19

- data point = “some person”
- features = measurements by wearables
- label could be “Covid-19 Infection?”
- another label could b “Influenza?”
- different choice for label results in different ML problems
- related since both are contagious respiratory illnesses





Wearable fitness devices deliver early warning of possible COVID-19 infection

August 7, 2020 1.35pm BST

Fitness information from wearable devices can reveal when the body is fighting an infection. Nico De Pasquale Photography/Stone via Getty Images

Email
Twitter 26
Facebook 112
LinkedIn
Print

The difficulty many people have getting tested for SARS-CoV-2 and delays in receiving test results make early warning of possible COVID-19 infections all the more important, and data from wearable health and fitness devices shows promise for identifying who might have COVID-19.

https://theconversation.com/wearable-fitness-devices-deliver-early-warning-of-possible-covid-19-infection-143388

Label Encodings

- consider detection of animals from a webcam
- label values “wolf” vs. “bear”, vs. “lion”
- we can represent label values as y=0, y=1 or y=2
- another representation is “one-hot-encoding”

$$\begin{pmatrix}1\\0\\0\end{pmatrix} \qquad \begin{pmatrix}0\\1\\0\end{pmatrix} \qquad \begin{pmatrix}0\\0\\1\end{pmatrix}$$



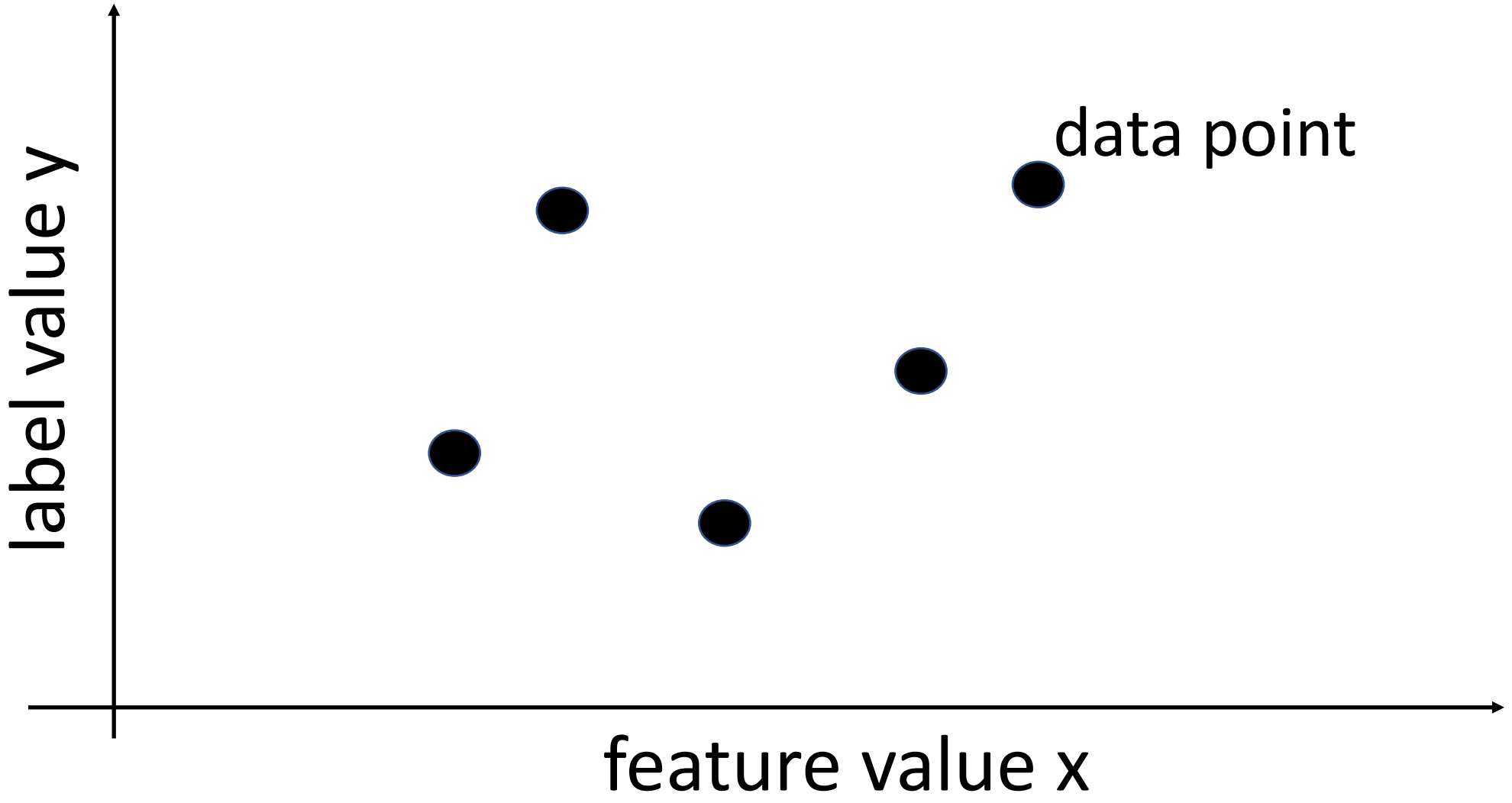
Supervised vs. Unsupervised

methods using labels which can only be defined with the help of humans are “supervised methods”

methods using labels that can be determined automatically are “unsupervised methods”

distinction between supervised and unsupervised methods is blurry

Scatterplot



Hypothesis Space

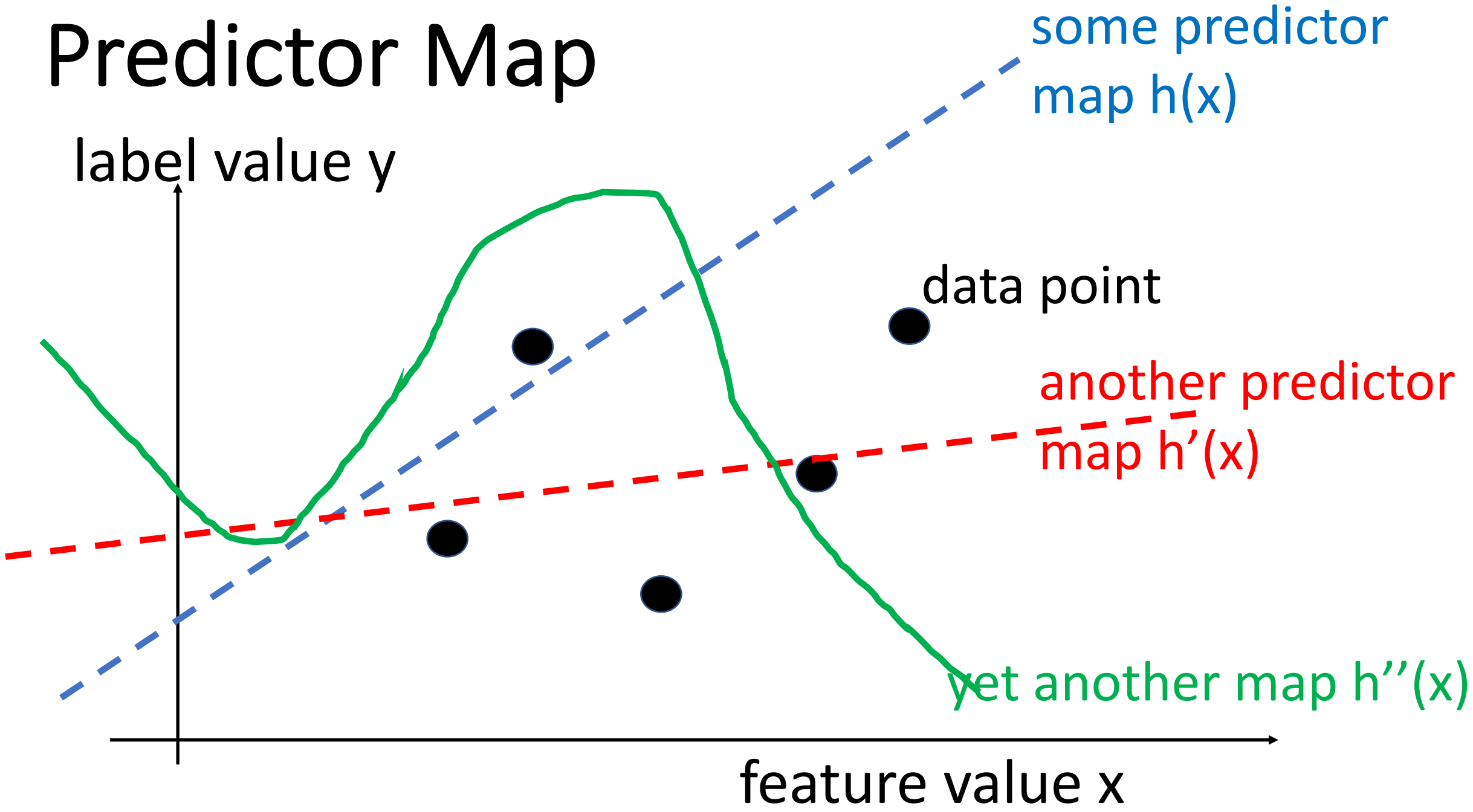
Predictor Map/Function

h(x)

$\hat{y} = 200\ minutes$
$to\ reach\ peak$

features x
of some data point

Predictor Map



Machine Learning

$\approx$

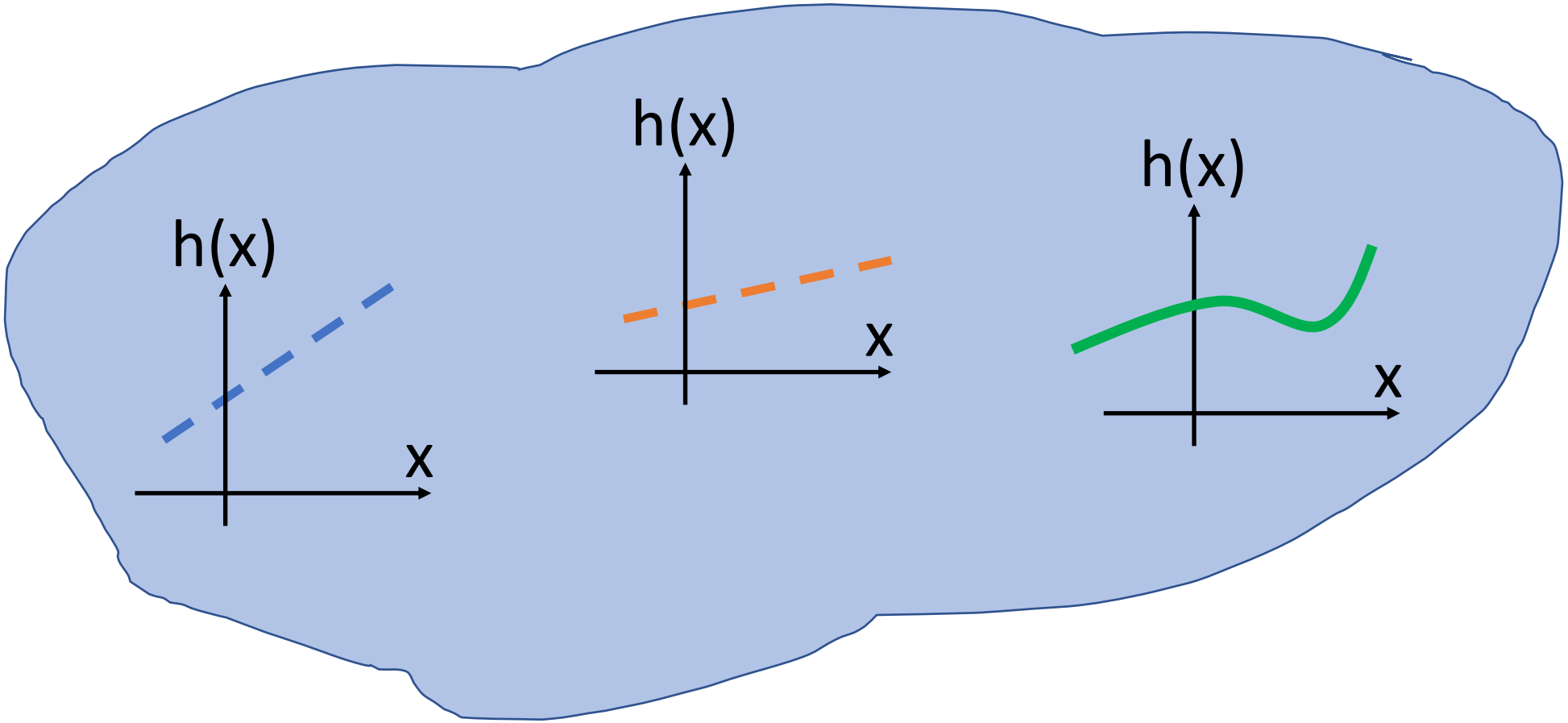
Find Good Predictor Map

consider data points with
single numeric feature x and label y

how many predictor maps h(x) are there ?

Have only finite resources!

a hypothesis space is a computationally tractable subset of predictor maps



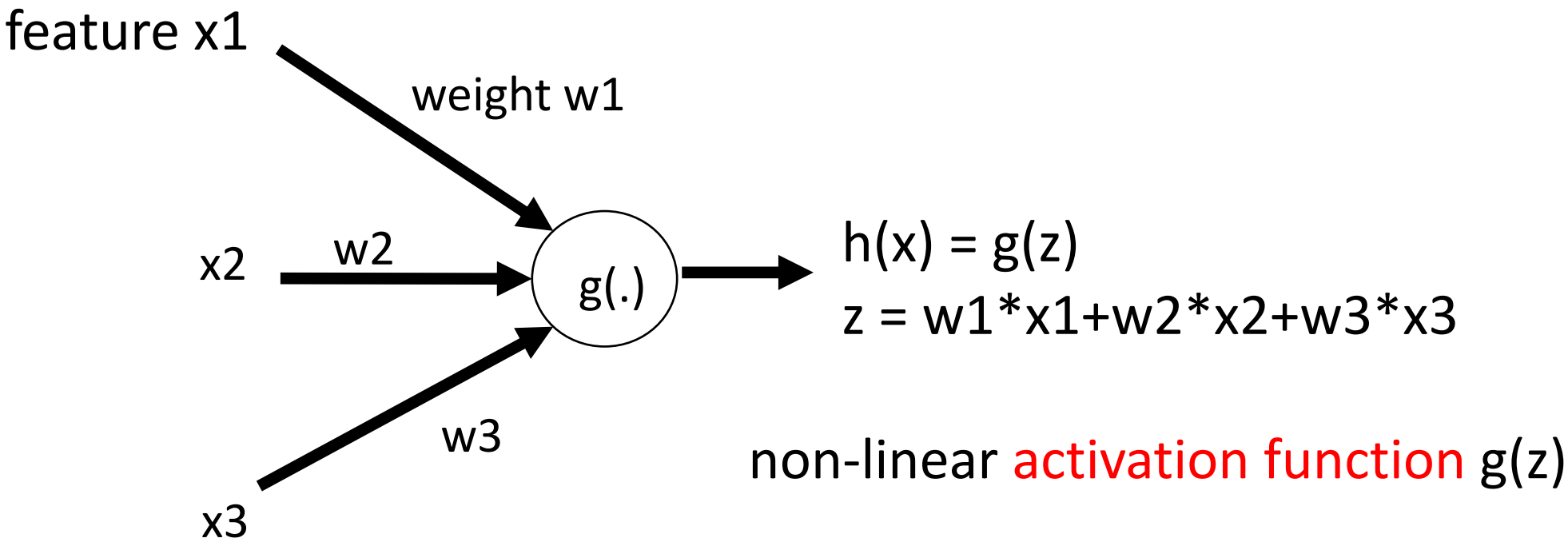
machine and deep learning Python libraries provide “fit()” function to search over (huge) hypothesis spaces

Machine Learning

- ML aim at finding/learning a good predictor h(x)
- predictor inputs features x and outputs predicted label
- predictor maps reading in millions of features
- must choose between many different maps
- deep learning uses special representation for maps

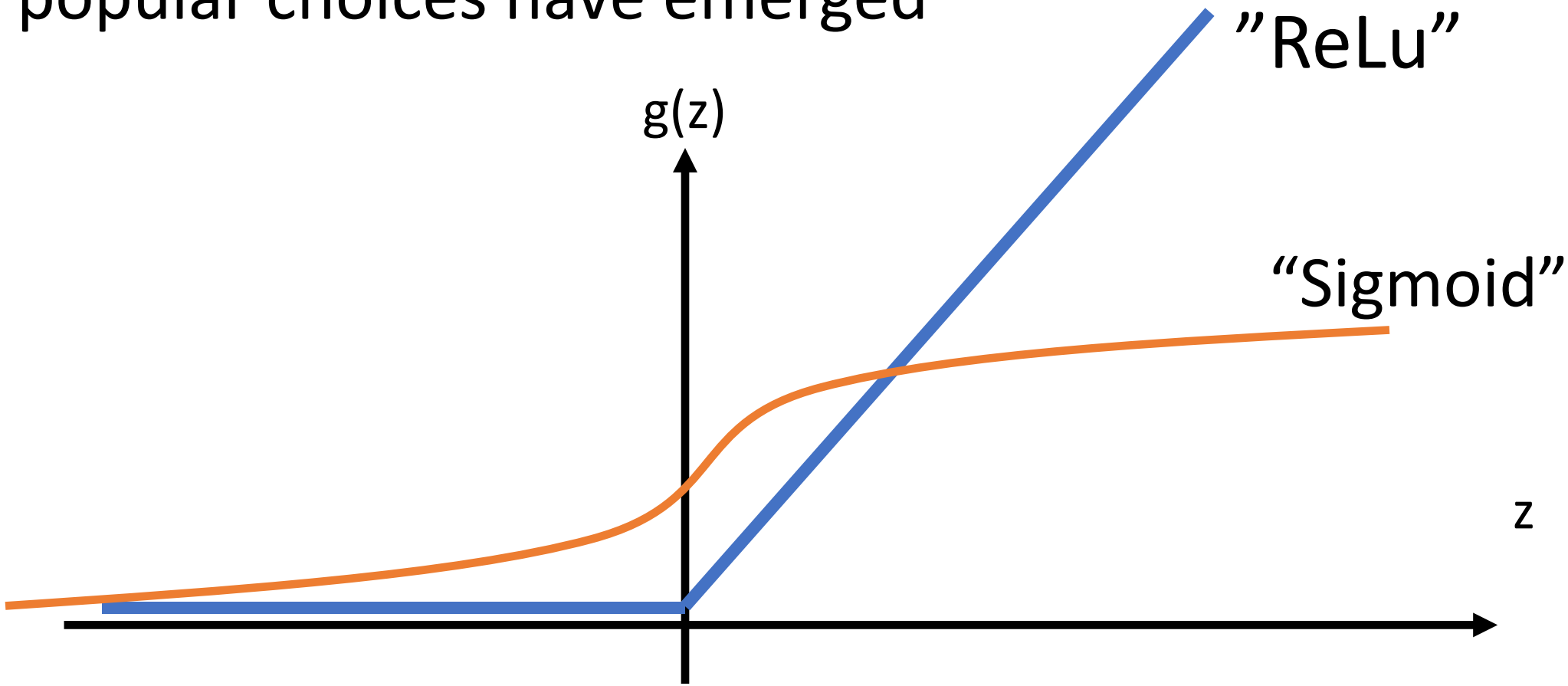
Artificial Neural Networks

- represent predictor map h(x) using network of neurons
- single neuron

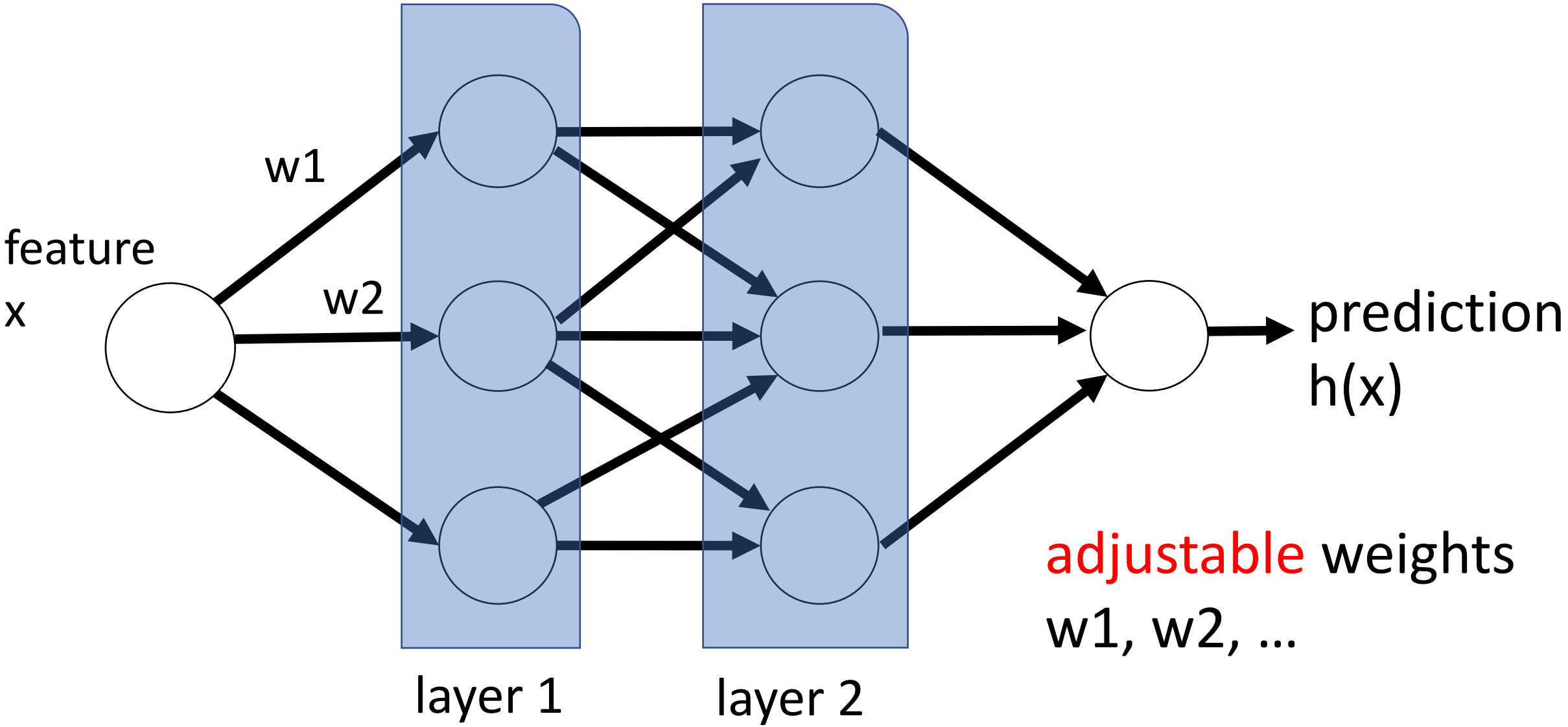


Activation Function

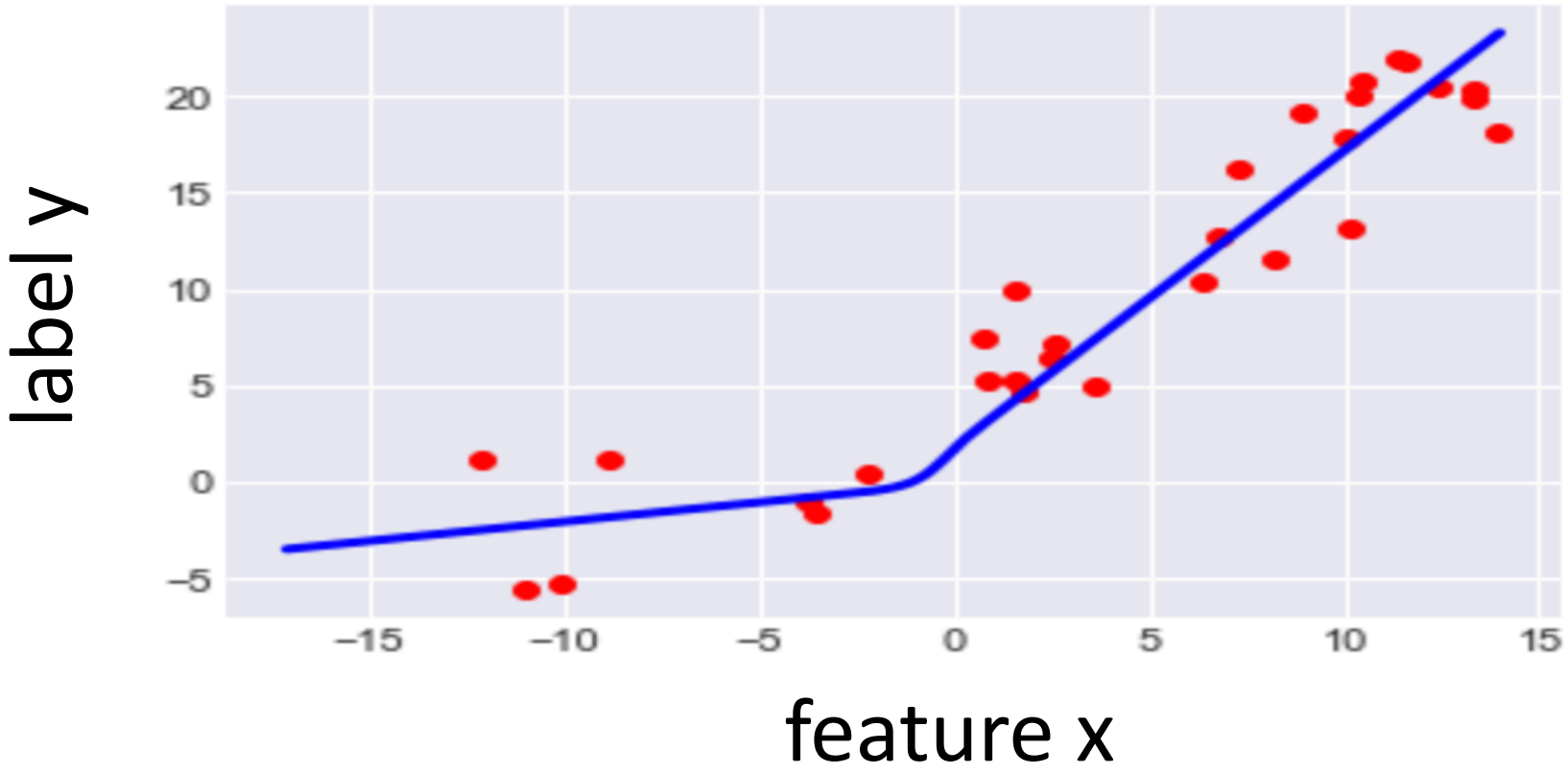
- few popular choices have emerged



(Deep) Neural Network=(Very) Non-Linear Function

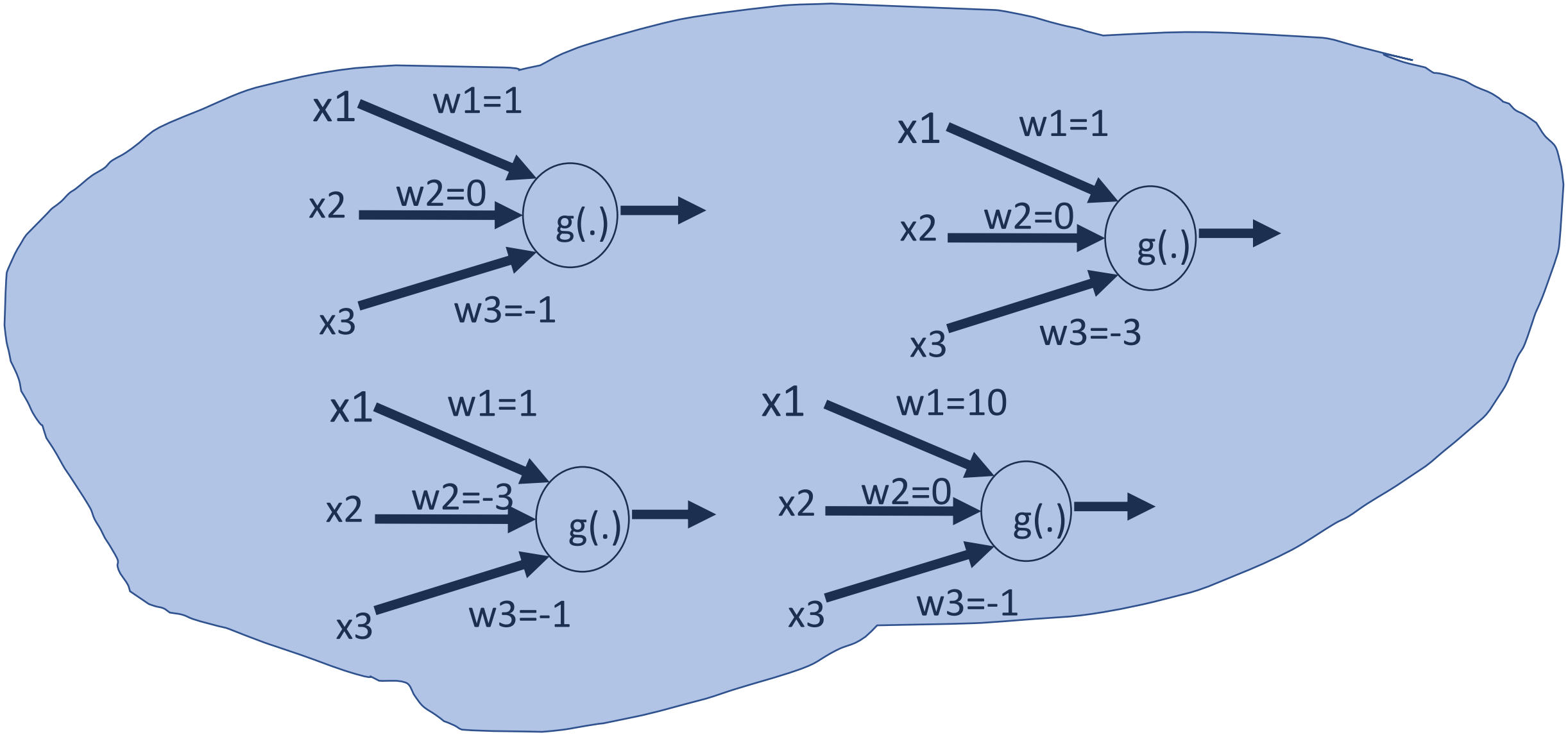


What is Deep Learning ?



deep learning methods fit non-linear maps to large data sets

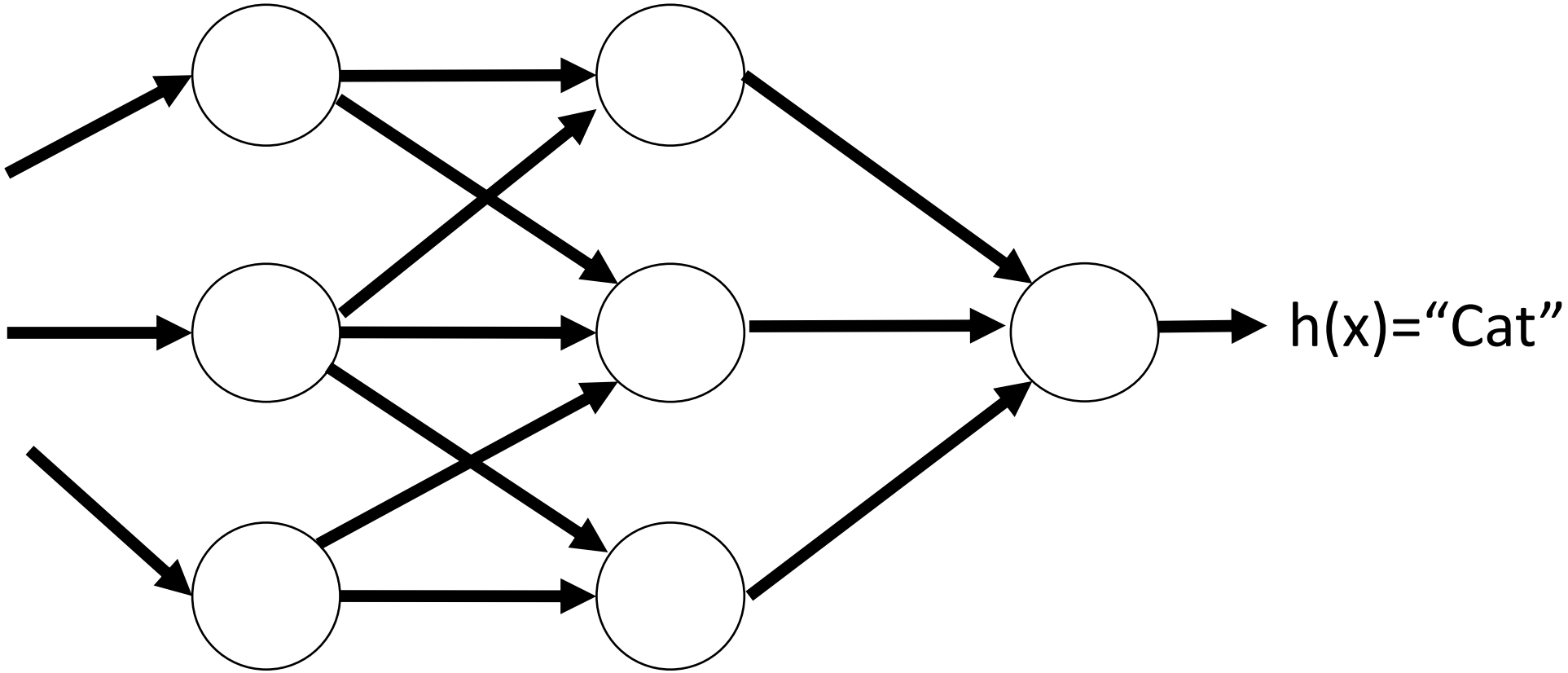
Hypothesis Space of ANN



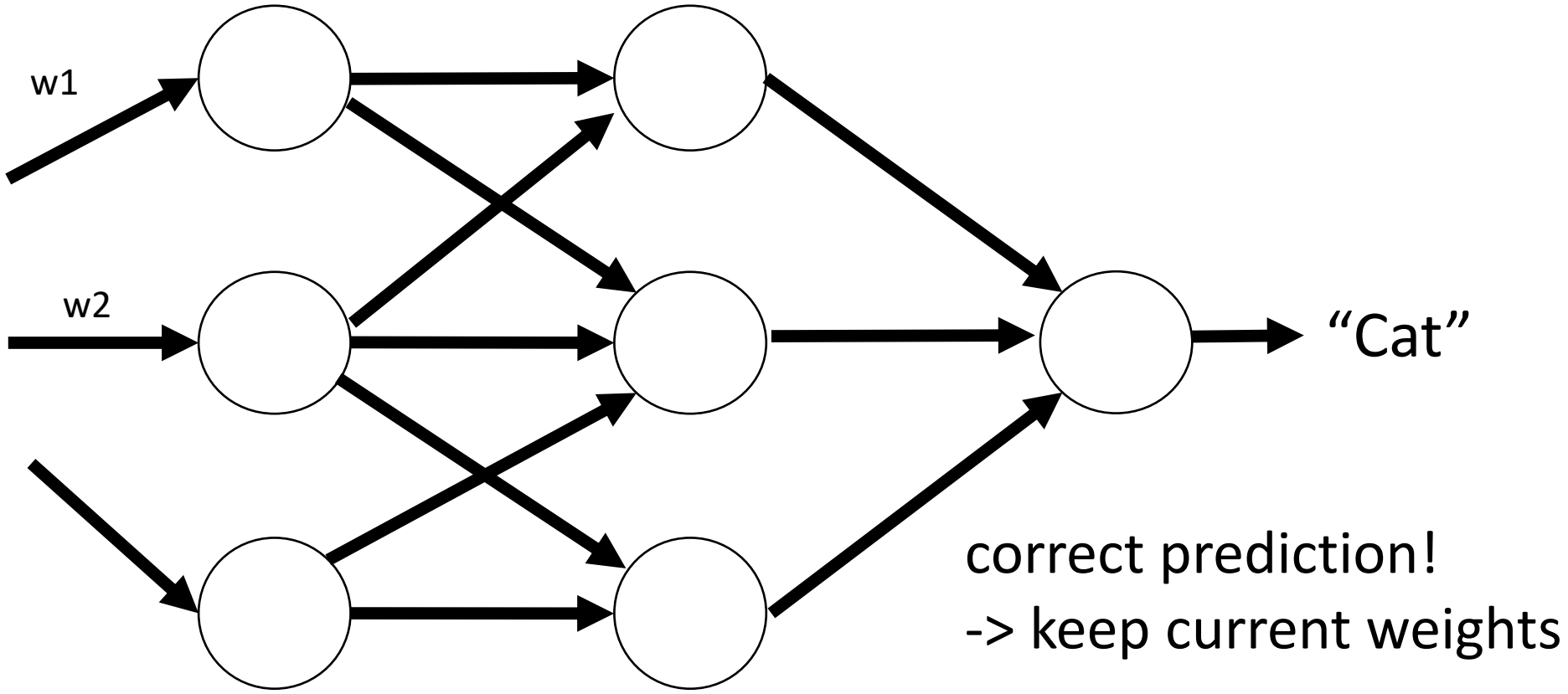
Loss Function

Evaluating Predictor (“Forward Pass”)

features x

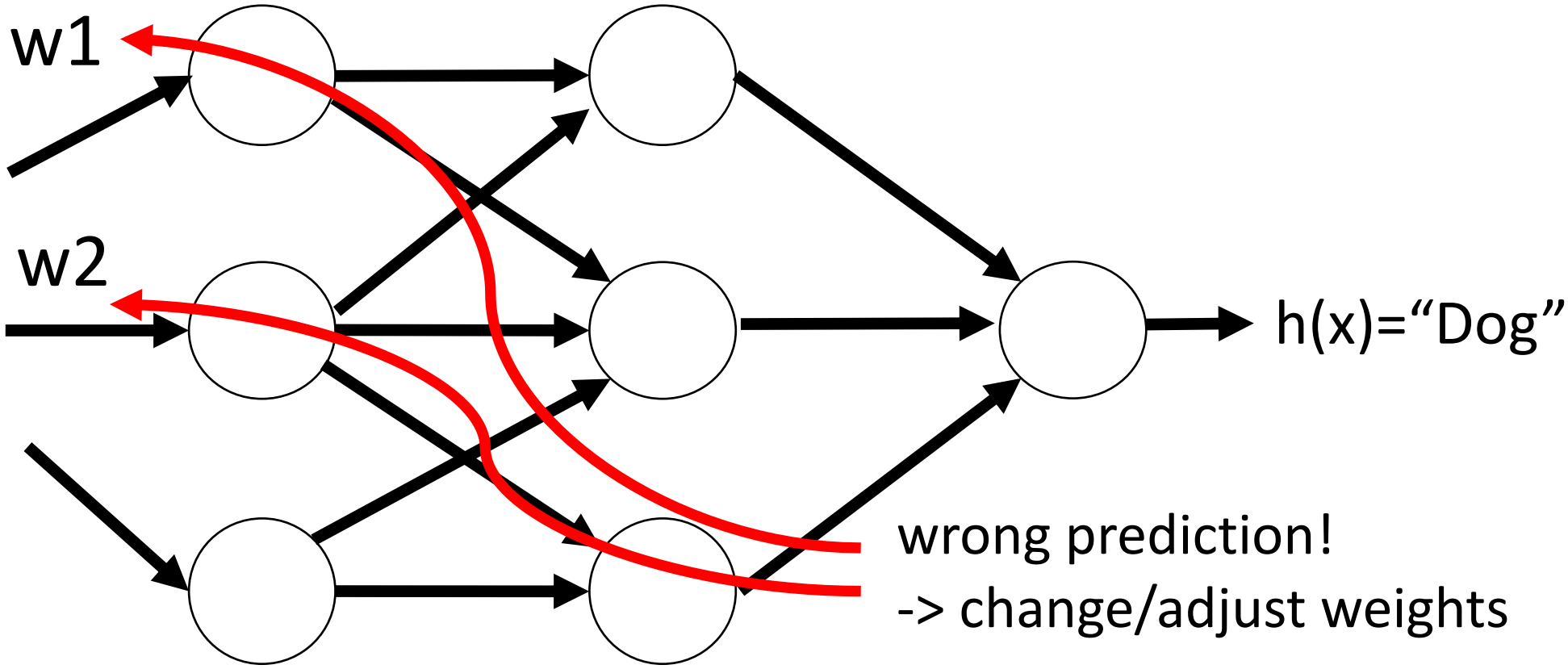


Deep Learning = Tune Weights



correct prediction!
-> keep current weights

Backward Pass (”Backpropagation”)



https://commons.wikimedia.org/wiki/File:Lion_d%27Afrique.jpg

learning is driven by making errors !

Loss Function

maps a pair of predictor map h (e.g. ANN) and data point (x,y) with features x and label y to some number

(h,(x,y)) -> “Loss” (denoted L(h,(x,y)))

loss function is design choice!

Some Popular Loss Functions

squared error loss (numeric labels):

$$L\big(h,(x,y)\big) = \ (y - h(x))^2$$

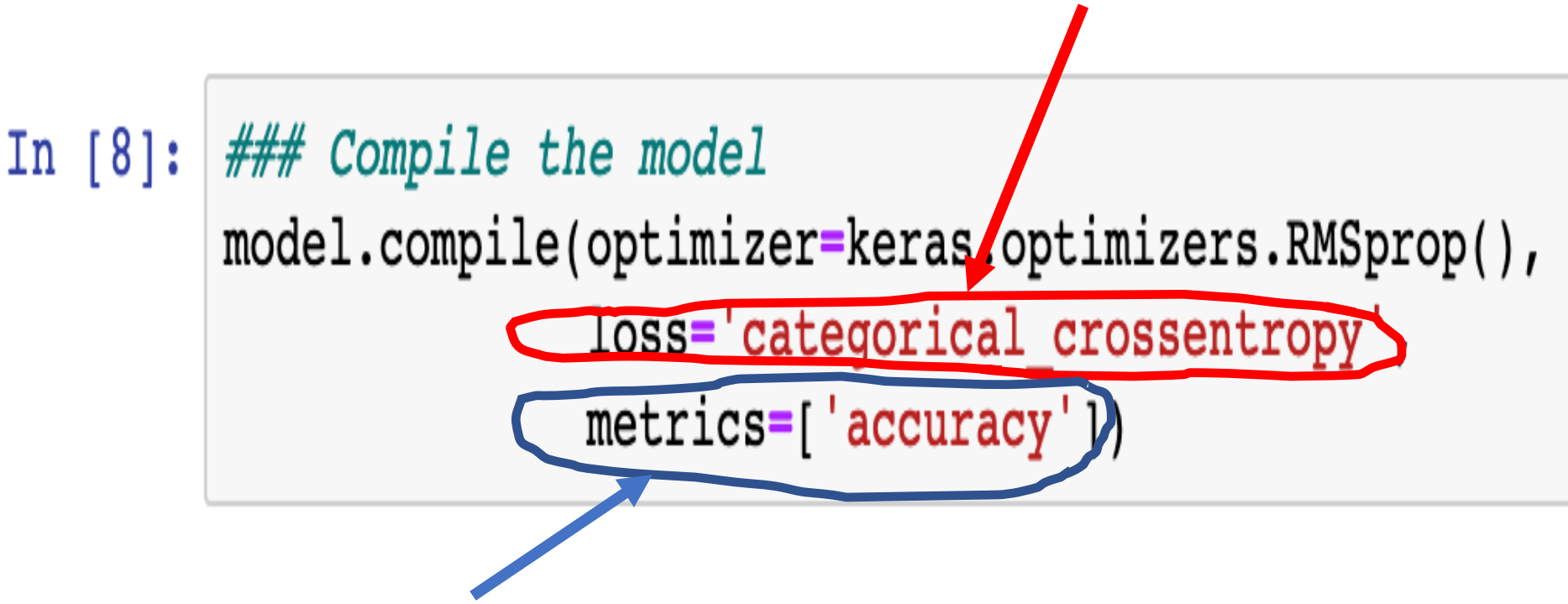
logistic loss (for binary labels, e.g., -1 and 1):

$$L\big(h,(x,y)\big) = \log_e (1+\exp(-yh(x)))$$

note that loss depends on (weights of) predictor map!

Chose Your Favorite Loss Function!

loss function used for adjusting weights

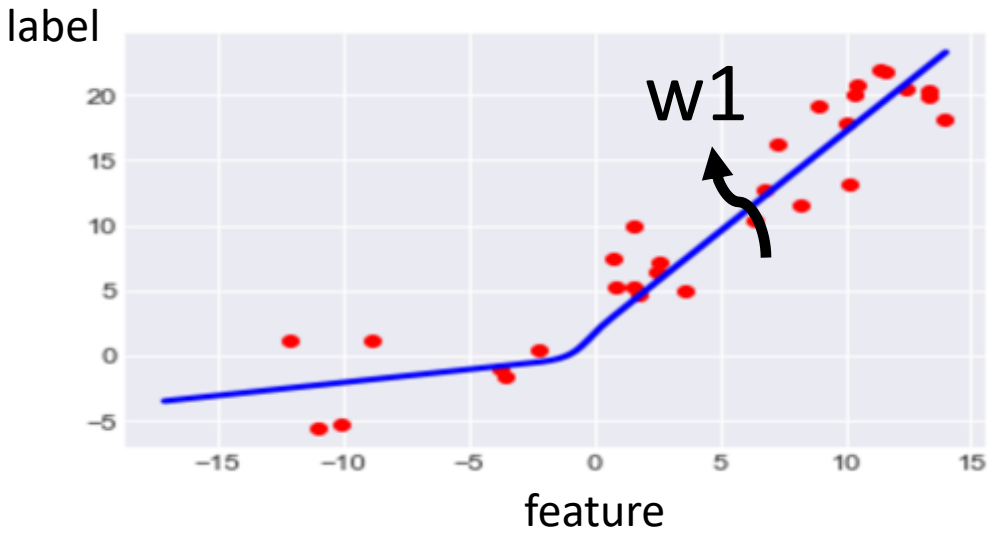


loss function used for final performance evaluation

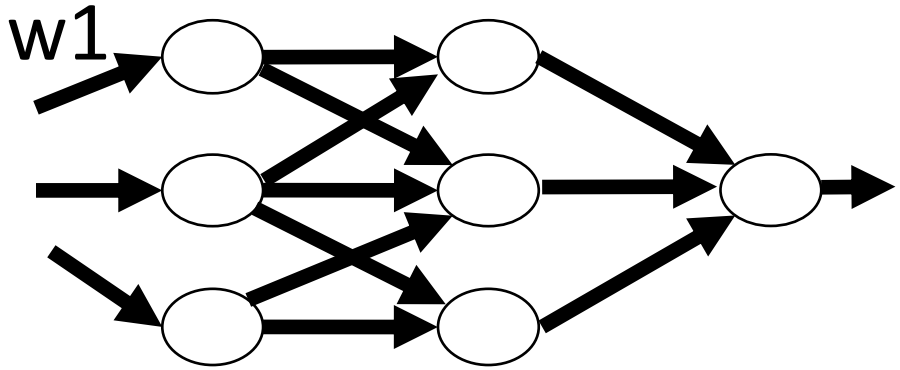
Putting Together the Pieces!

Three Views on Artificial Neural Nets

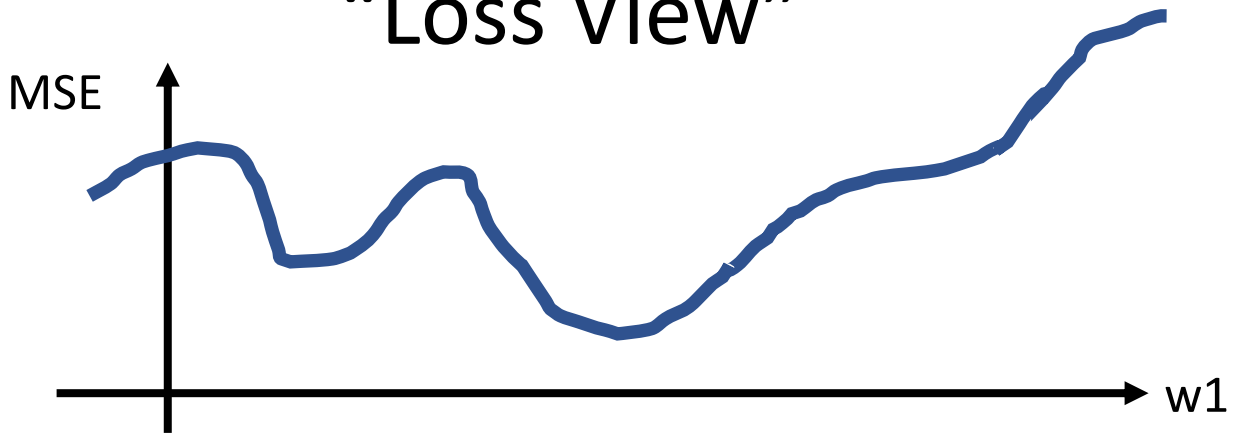
"Data View"



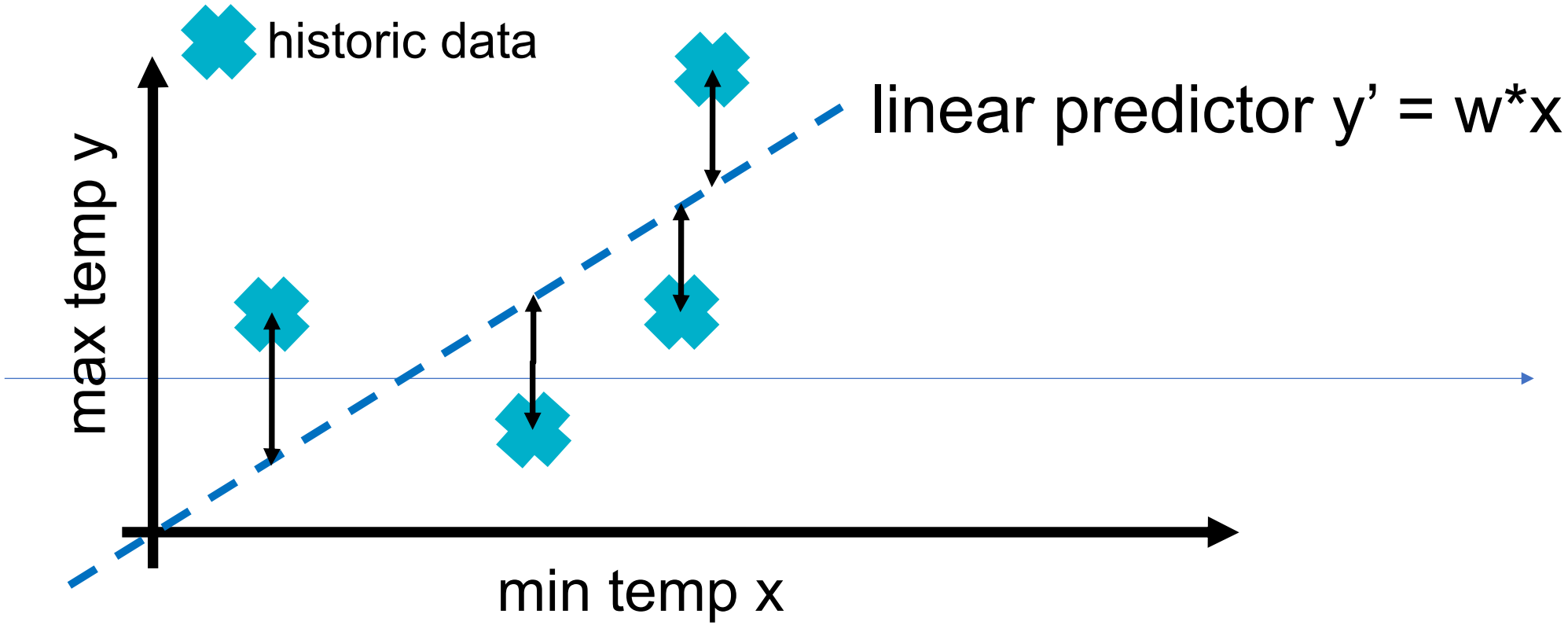
"Model View"



"Loss View"

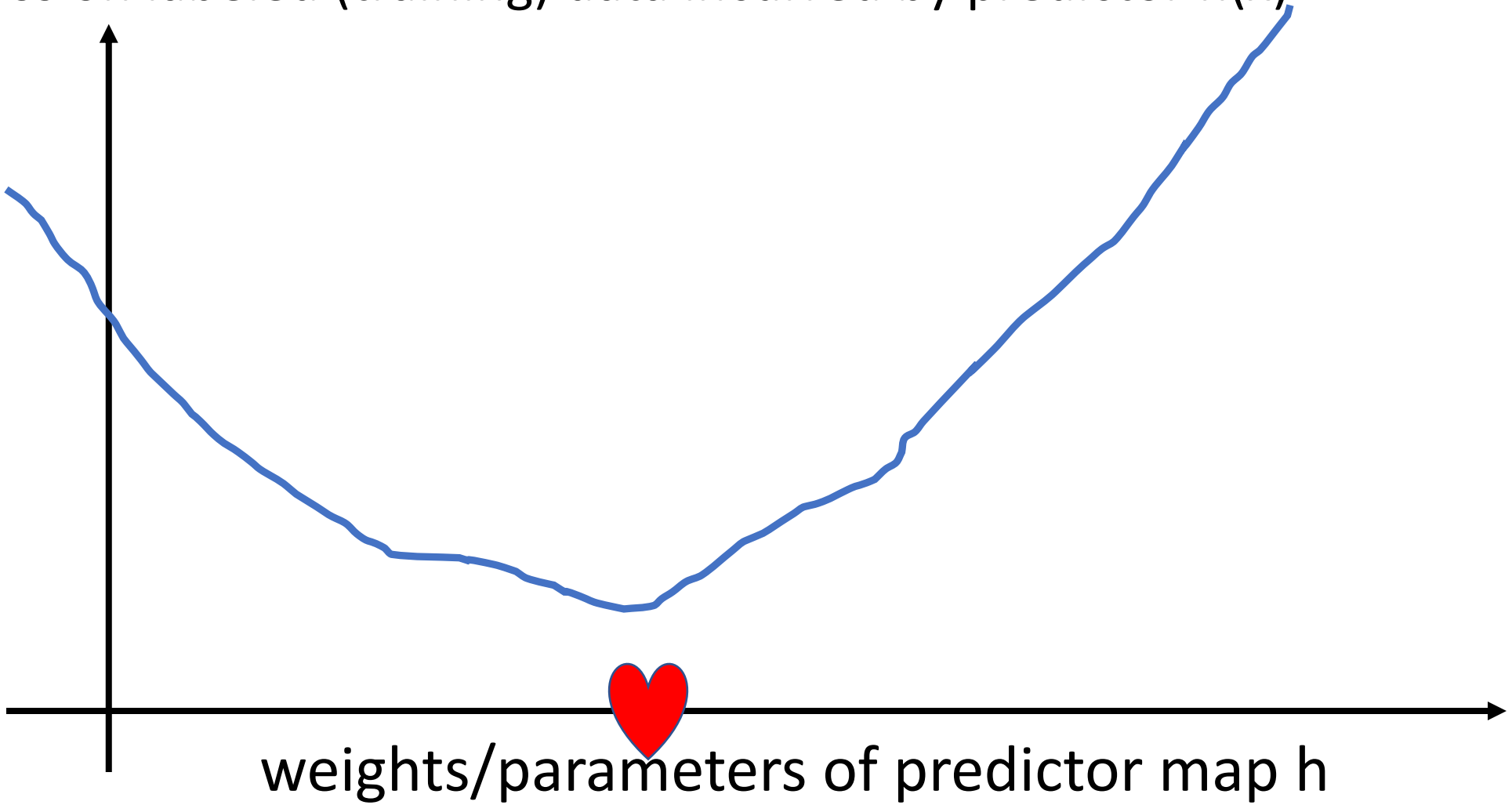


learn predictor y'=h(x) by tuning weights w to minimize loss

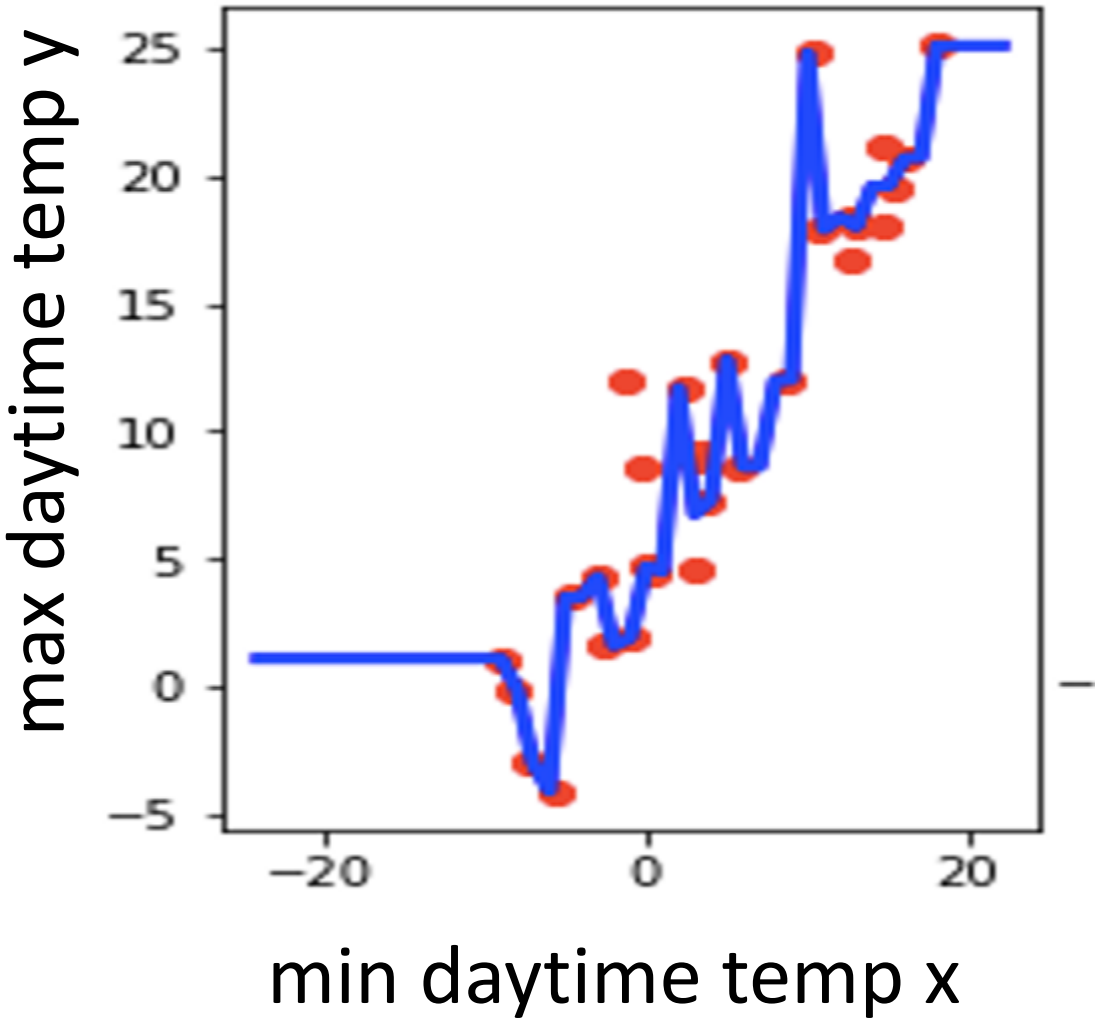


Machine Learning = Optimization

average loss on labeled (training) data incurred by predictor h(x)

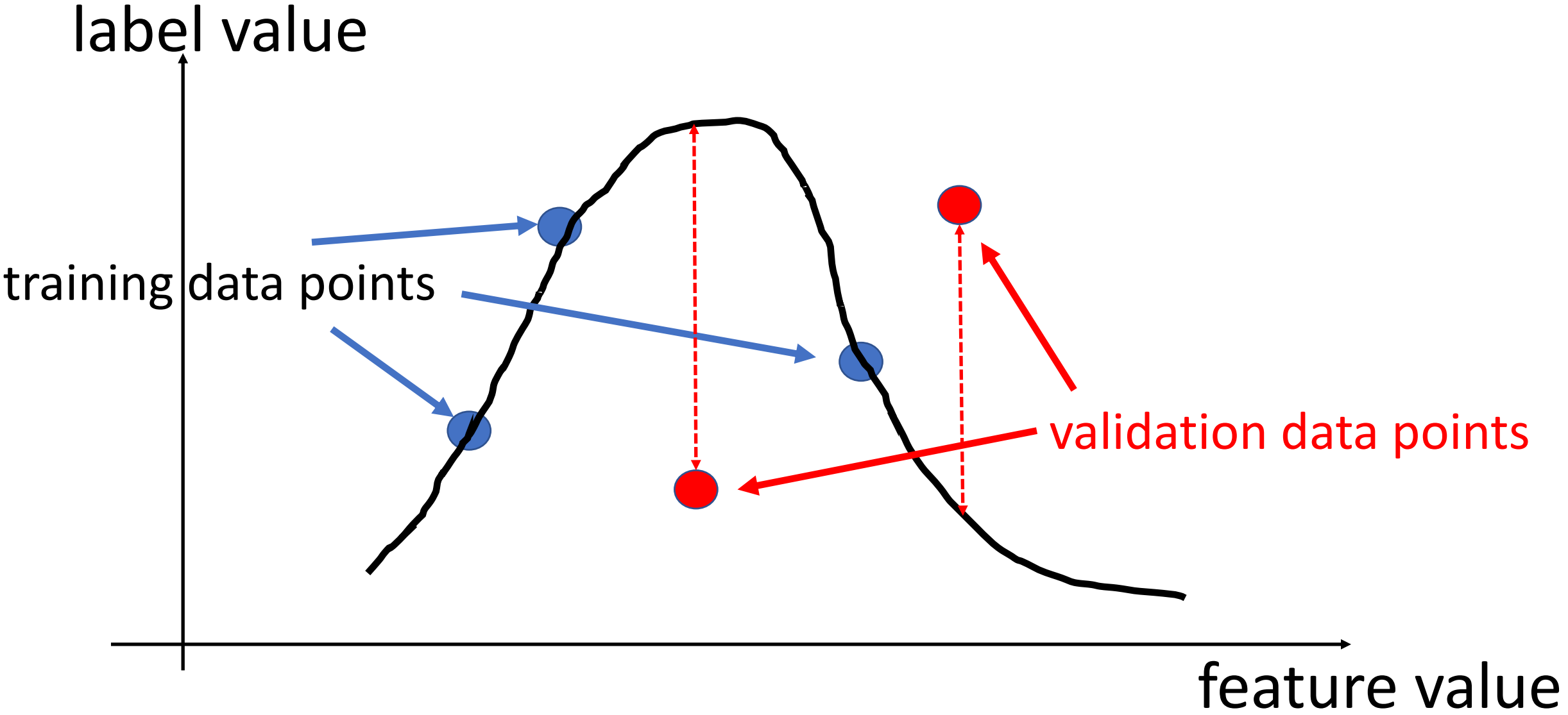


Key Challenge in Machine Learning - Overfitting

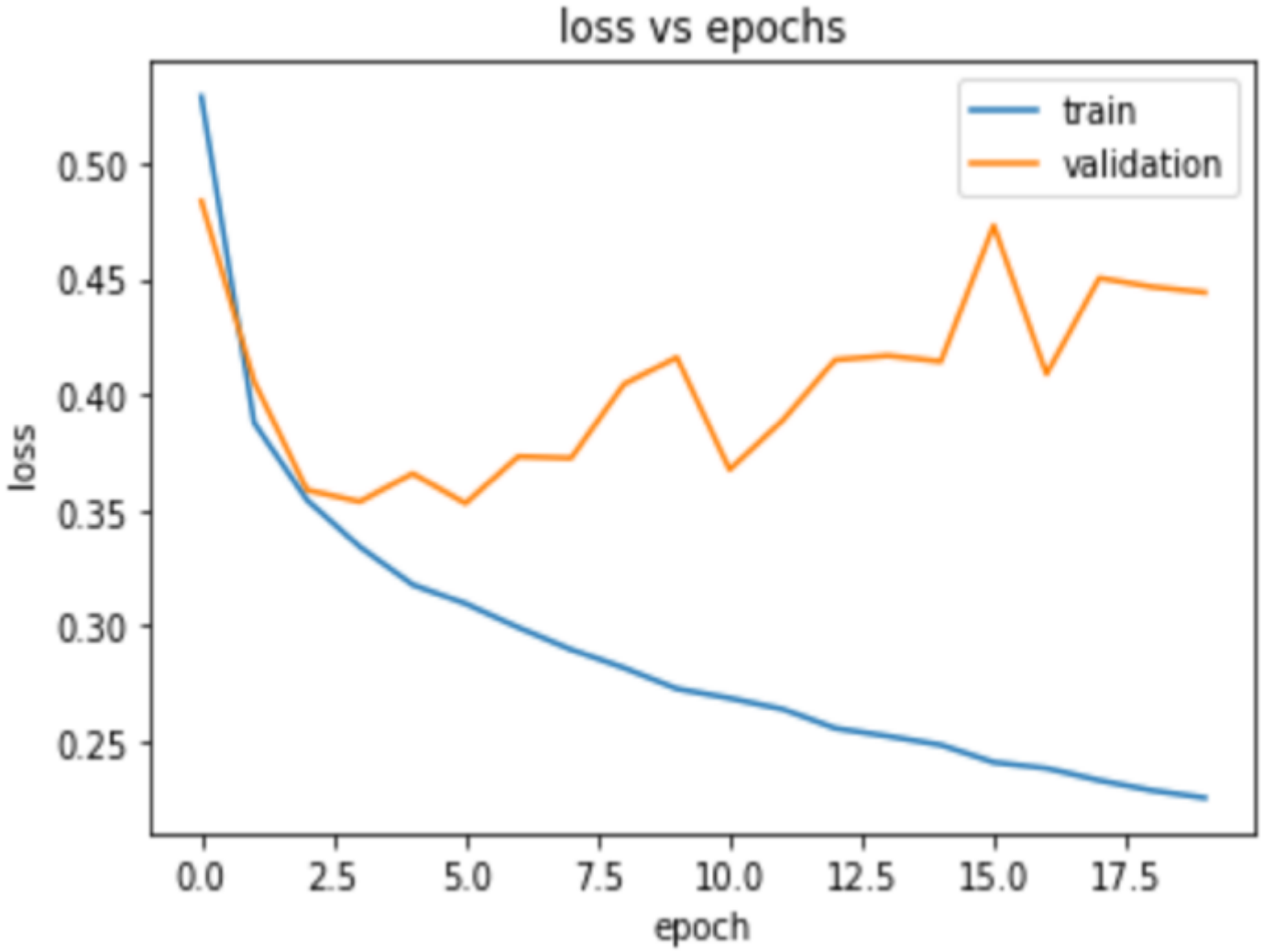


small training error but poor predictor map!

Detecting Overfitting by Validation



Look at the Validation Set !!!



Training, Validation and Test Set

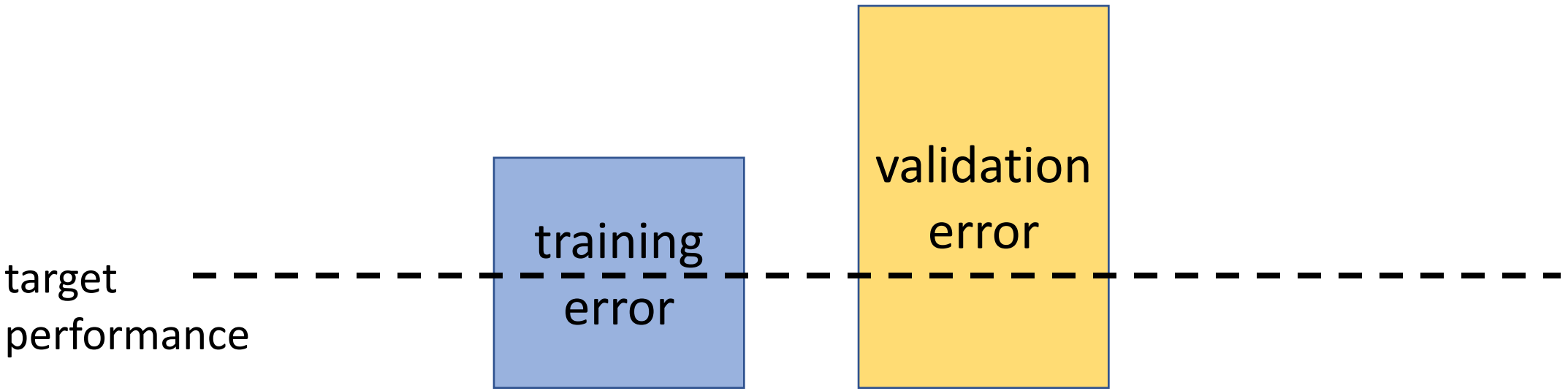
- training set: used to adjust weights
- validation set: used to adjust hyperparameters (number of layers..)
- test set: final performance evaluation

Test Set

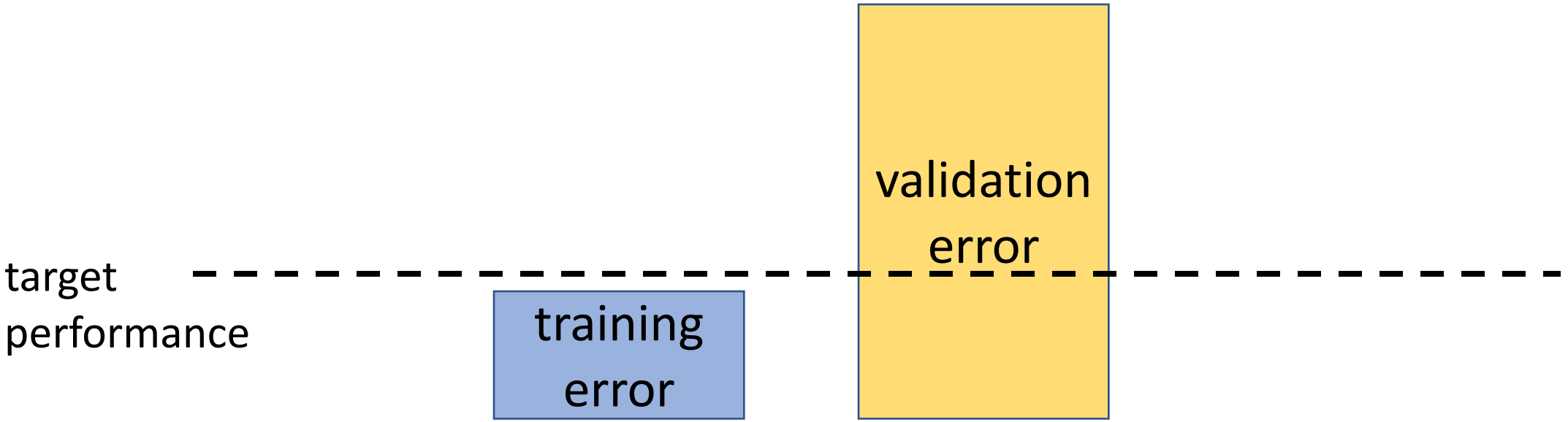
- used for final performance evaluation

- results on test set MUST NOT BE used for model adjustment!

Diagnosing ML Methods



Case 1: Overfitting



possible remedies:

reduce hypothesis space or use more training data

Reducing Hypothesis Space

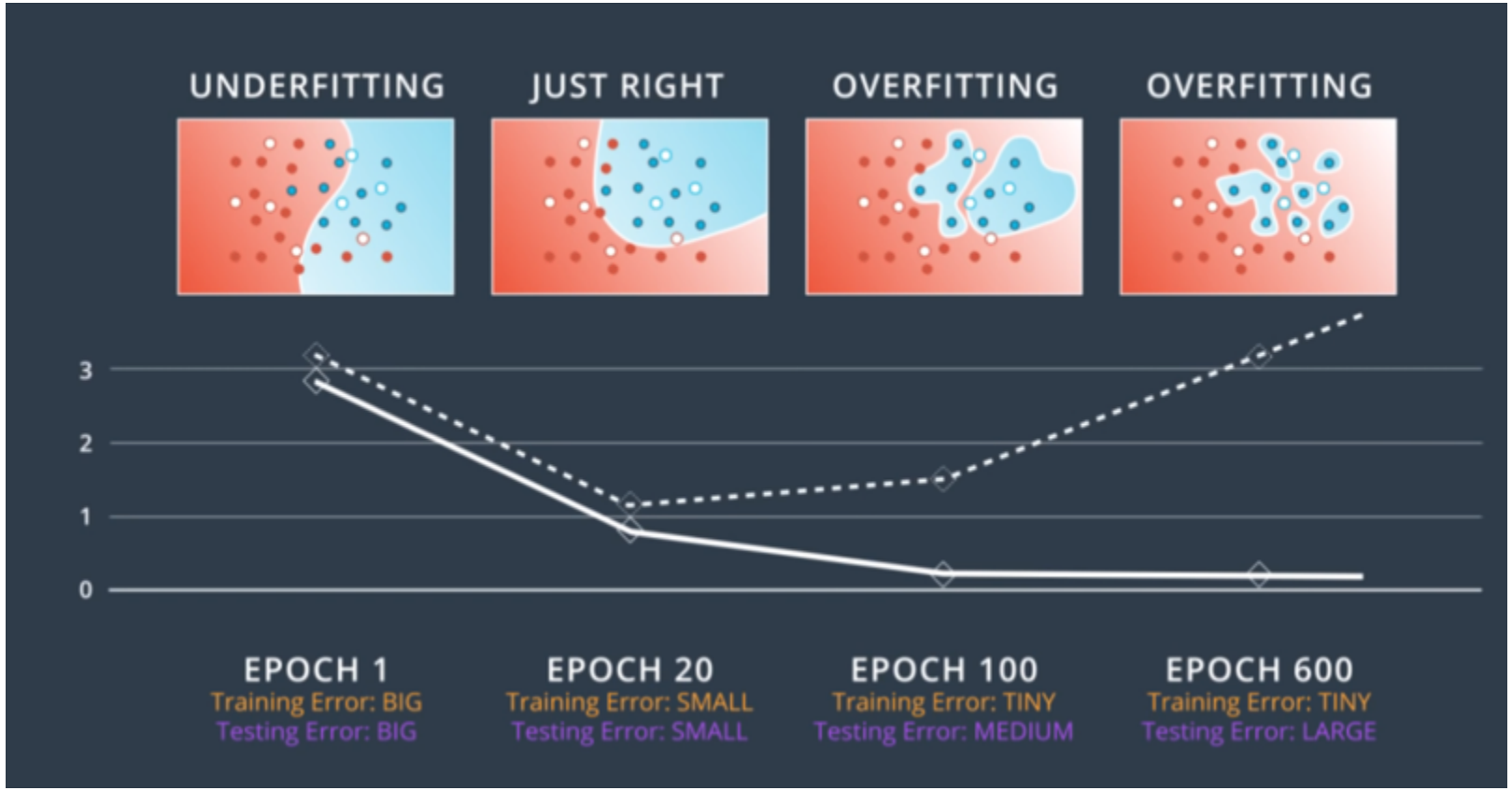
- use fewer neurons in hidden layers
- use fewer features (manually choose relevant features)
- use fewer layers
- use fewer iterations of gradient method (such that we search only a smaller subset of the nominal space)

Reducing Hypothesis Space

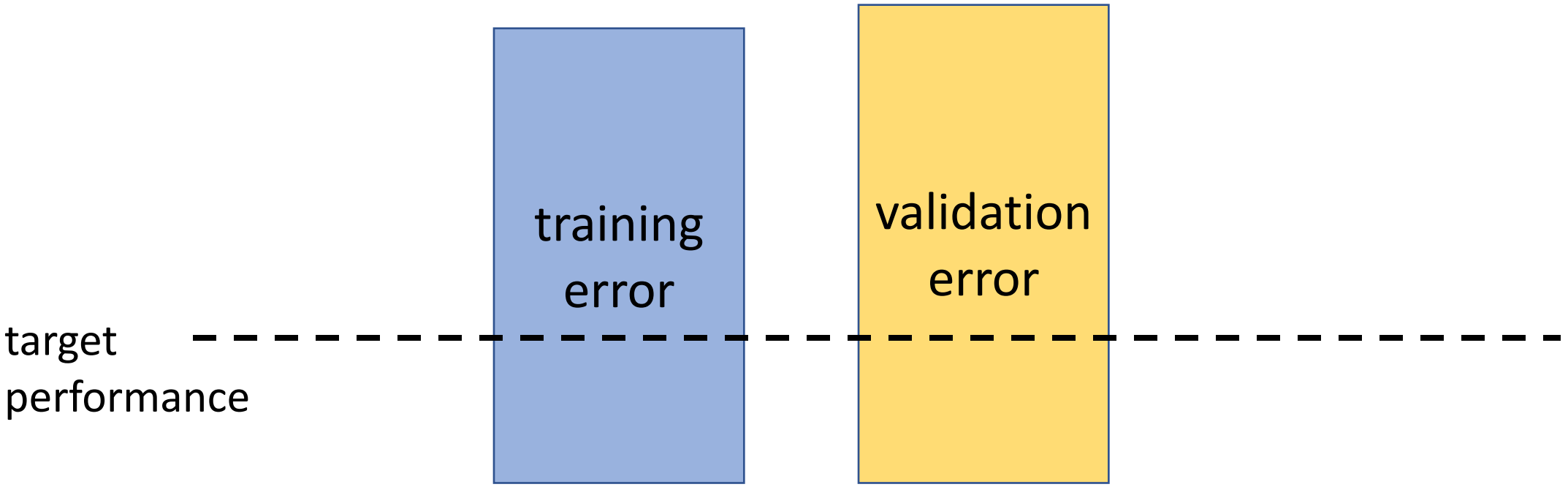
- use fewer neurons in hidden layers
- use fewer features (manually choose relevant features)
- use fewer layers
- use fewer iterations of gradient descent (search only a smaller subset of the nominal space)

"early stopping"

Reducing Effective Hypothesis Space



Case 2: Underfitting



possible remedy:
enlarge hypothesis space

Enlarging Hypothesis Space

- use more neurons in hidden layers (wider layers)
- use more features (manually choose relevant features)
- use more layers (make network deeper)
- use more iterations of gradient descent (search

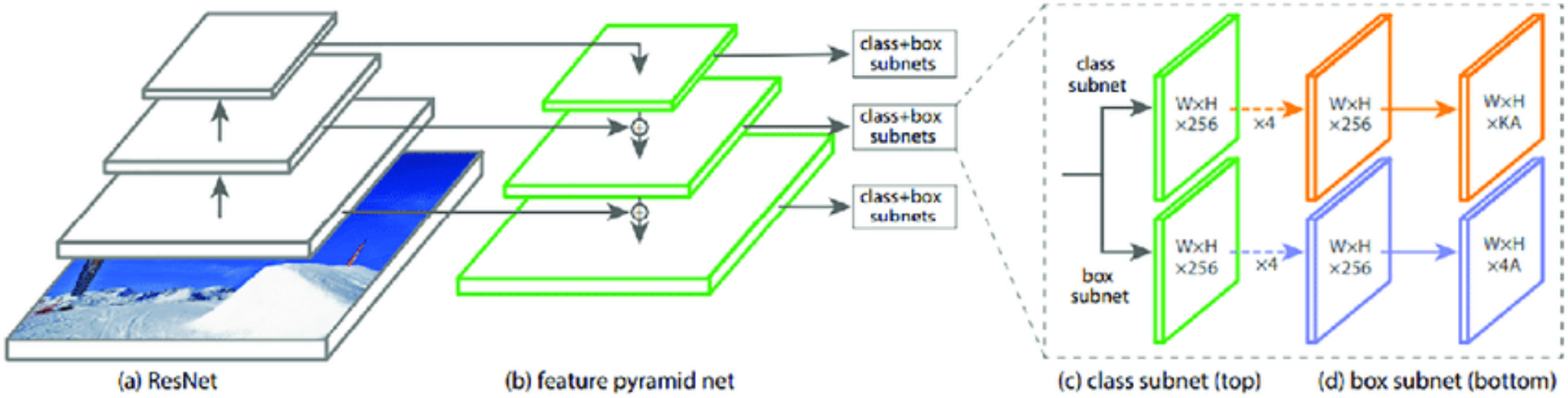
a larger portion of the entire hypothesis space)

That’s All !

Do not hesitate to ask !

Use slack discussion forum!

The Model (Network Architecture)



“RetinaNet” – storing one single configuration of all weights for this model results in 500 MB file !

Encoding of Label Values

- one hot
- softmax vs. sigmoid
- different encoding of label values in multiclass problems
- https://gombru.github.io/2018/05/23/cross_entropy_loss/

Table 4.1 Choosing the right last-layer activation and loss function for your model

Problem type	Last-layer activation	Loss function
Binary classification	`sigmoid`	`binary_crossentropy`
Multiclass, single-label classification	`softmax`	`categorical_crossentropy`
Multiclass, multilabel classification	`sigmoid`	`binary_crossentropy`
Regression to arbitrary values	None	`mse`
Regression to values between 0 and 1	`sigmoid`	`mse` or `binary_crossentropy`