

# Artificial Intelligence: A Survey for Policymakers

School of International & Public Affairs  
Columbia University

January 28, 2025 - Lecture 2

Neal Parikh · [nparikh.org](http://nparikh.org) · [neal@nparikh.org](mailto:neal@nparikh.org)

# Today

- crash course in machine learning
- **warning:** this is going to be a slog, but the jargon and concepts will keep getting revisited and reinforced as we go through applications and examples through the course
- you should plan to keep revisiting this material periodically
- some of this will continue into the lecture next week

# Preface

There is a great deal of mathematics in the early part because almost surely the future of science and engineering will be more mathematical than the past, and also I need to establish the nature of the foundations of our beliefs and their uncertainties. Only then can I show the weakness of our current beliefs and indicate future directions to be considered.

If you find the mathematics difficult, skip those early parts. Later sections will be understandable provided you are willing to forego the deep insights mathematics gives into the weaknesses of our current beliefs. General results are always stated in words, so the content will still be there but in a slightly diluted form.

**Richard Hamming**



In a sense you will never really grasp the whole problem of AI until you get inside and try your hand at finding what you mean and what machines can do. Before the checkers-playing program which learned was exposed in simple detail, you probably thought machines could not learn from experience — now you may feel what was done was not learning but clever cheating, though clearly the program modified its behavior depending on its experiences. *You must struggle with your own beliefs if you are to make any progress in understanding the possibilities and limitations of computers in the intellectual area.* To do this adequately you must formalize your beliefs and then criticize them severely, arguing one side against the other, until you have a fair idea of the strengths and weaknesses of both sides. Most students start out anti-AI; some are pro-AI; and if you are either one of these then you must try to undo your biases in this important matter. ...You must make up your own mind on this important topic. False beliefs will mean you will not participate significantly in the inevitable and extensive computerization of your organization and society generally. In many senses the computer revolution has only begun!

**Richard Hamming**

# **Machine Learning Background**

Machine learning is a way of writing software by giving examples rather than writing explicit rules.

# Algorithm for finding the maximum entry in a list

```
function find_maximum(numbers):  
    max_value <- first_number_of(numbers)  
    for each number in numbers:  
        if number > max_value:  
            max_value <- number  
    return max_value
```

# Finding spy planes

in August 2017, BuzzFeed News publishes articles finding

- military contractors flying over SF Bay Area
- secret US Marshals plane hunting drug cartel kingpins in Mexico
- Air Force special operations planes flying over US

## US Federal Agents Flew A Secret Spy Plane To Hunt Drug Cartel Leaders In Mexico

Neither the US Marshals Service nor the Mexican government wants to talk about their joint efforts to hunt drug kingpins. But BuzzFeed News spotted a Marshals spy plane circling around the time of a prominent capture in Sinaloa.

Posted on August 3, 2017, at 8:00 a.m.



**Peter Aldhous**

BuzzFeed News Reporter

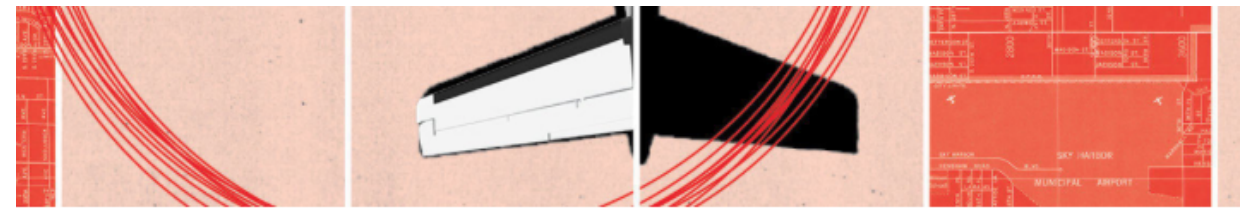


**Karla Zabudovsky**

BuzzFeed News Reporter

# Finding spy planes

1. pull public dataset  
(not intended for this)
2. train a simple machine learning model
3. validate  
(here, 'do journalism')

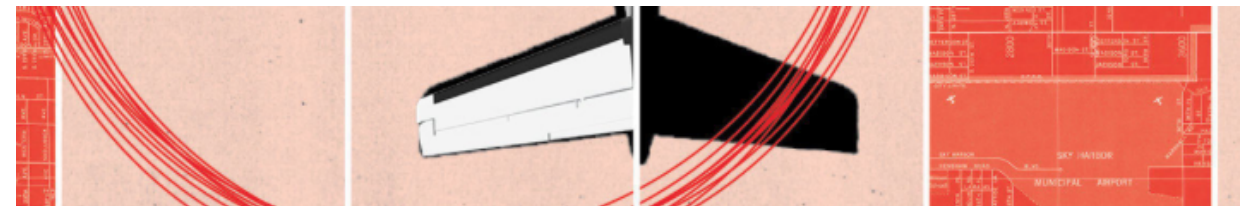


## **BuzzFeed News Trained A Computer To Search For Hidden Spy Planes. This Is What We Found.**

From planes tracking drug traffickers to those testing new spying technology, US airspace is buzzing with surveillance aircraft operated for law enforcement and the military.

# Finding spy planes

1. pull 4 months of flight-tracking data from Flightradar24
2. extract 'features': turning rates, speeds, altitudes, manufacturers
3. train a binary classifier to distinguish between previously identified planes and not
4. validate



## **BuzzFeed News Trained A Computer To Search For Hidden Spy Planes. This Is What We Found.**

From planes tracking drug traffickers to those testing new spying technology, US airspace is buzzing with surveillance aircraft operated for law enforcement and the military.

# Examples

## Companies

- Adobe (font recognition using phone camera)
- Amazon (speculative shipping, Kindle browser prefetching)
- American Express (fraud detection, individual credit limits)
- Cheesecake Factory (predict food ingredient demand)
- C-SPAN (automatically name politicians on screen)
- HireVue (video analysis of job interviews for hiring/screening)
- Nest Thermostat (embedded control of smart thermostat)
- Target (market research, individualized product catalogues)
- USPS (handwriting recognition)
- Walmart (inventory, product placement)



# Examples

## NYC government

- Administration for Children's Services
- Criminal Justice Agency
- Cyber Command
- Department of Education
- Department of Health & Mental Hygiene
- Department of Social Services
- Fire Department
- Mayor's Office of Criminal Justice
- Police Department

# What is machine learning?

- no precise, universal technical or operational definition
  - (but you need to pick one to write and implement a policy!)
- usage evolved over time
- ‘classical’ usage is as a sub-discipline of AI research
- Example definition: “Machine learning is a field that develops algorithms designed to be applied to data sets, with the main areas of focus being prediction (regression), classification, and clustering or grouping tasks. These tasks are divided into two main branches, supervised and unsupervised ML.” (Athey)
- this (perfectly reasonable) definition won’t work for some purposes

# What is machine learning?

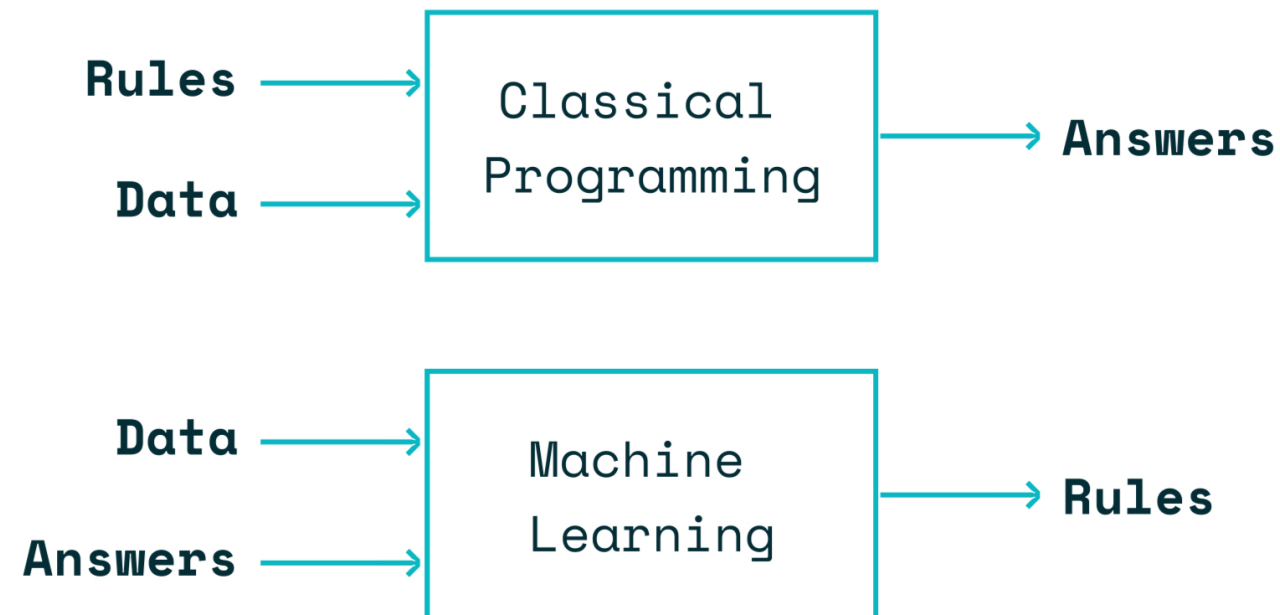
- intersection of computer science and statistics
- computationally tractable algorithms that learn from data
- the mathematical foundation of modern AI, but now also used in a huge variety of other domains

# What is machine learning?

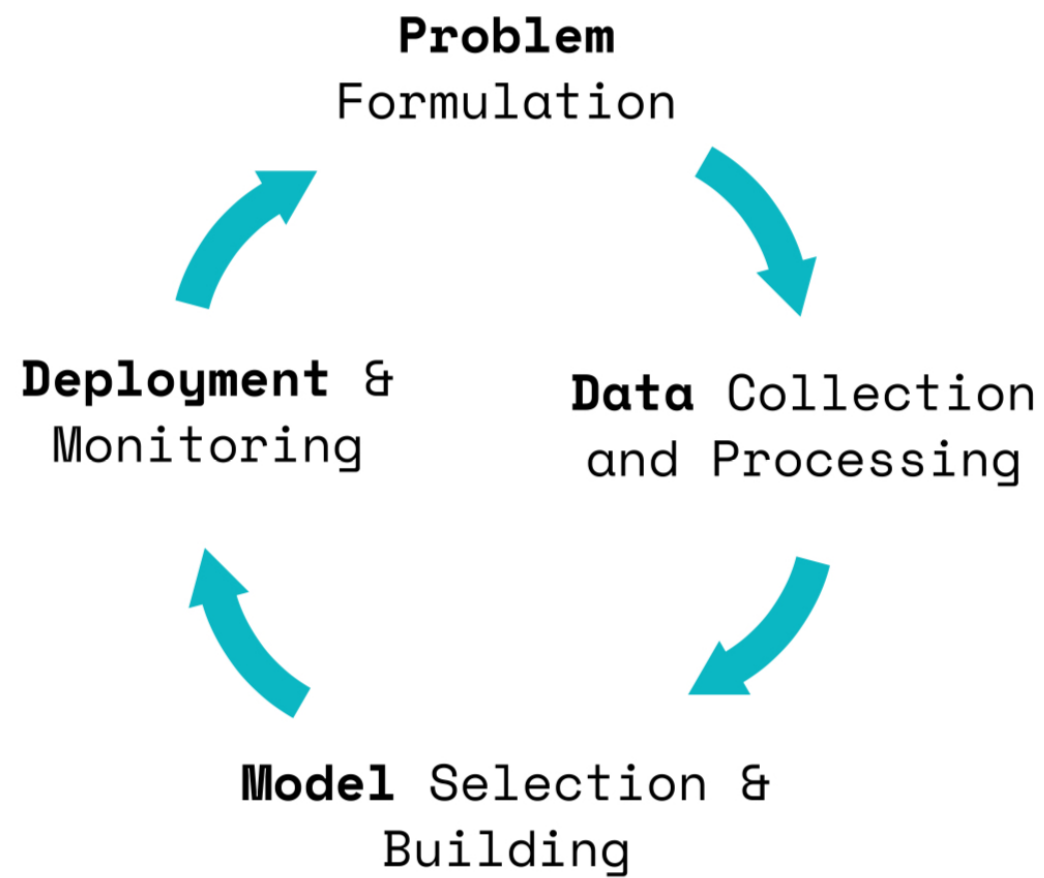
- modern usage: how to build learning procedures, i.e., how to use historical data to build a prediction rule
- **prediction rule**: algorithm mapping observable inputs to prediction of unknown quantity (the response)
- **focus is on making predictions**, and doing well on data you haven't yet seen
  - how to select the right prediction rule among several
- informally, is mostly interchangeable with the terms 'AI' and 'modern statistical prediction'
  - specialized, single-purpose systems can also be called 'AI'

# Machine learning vs classical programming

(F. Chollet)



# The AI Lifecycle



# (Crude) history of machine learning and AI

1950s                      Dartmouth conferences; chess & checkers; LISP; perceptron

1960s                      early foundational work; symbolic/logic methods; search (A\*); perceptrons

1970s                      logic programming; expert systems; backprop; AI winter

1980s                      backpropagation; probability; shift to machine learning; AI winter

---

1990s                      probabilistic/statistical revolution; graphical models; kernel methods; NLP

2000s                      Big Data; Internet applications (search, recsys); convex optimization

2010s—                    deep learning (NN) revolution; major software libraries; wide-spread uses

# ML and friends, or 'Big Tent' AI

- machine learning intersects with and builds on many other areas in applied math and computer science:
  - numerical linear algebra
  - mathematical optimization (optimization theory)
  - statistics
  - probability
  - information theory
  - control theory
  - many other areas in CS (algorithms, architecture, graphics, HCI, ...)
  - application domains (biology, finance, ...)
- it also increasingly uses tools and ideas from social sciences (economics, ...)



# Machine learning vs statistics

(Wasserman; Tibshirani)

## Statistics

## Computer Science

estimation/fitting

learning

regression/classification

supervised learning

clustering/density estimation

unsupervised learning

data

training sample

covariates

features, inputs

response

outputs

test set performance

generalization ability;  
out of sample performance

# Common Task Framework

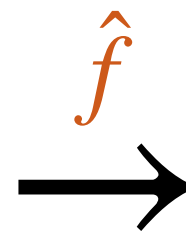
- a methodology for standardizing and evaluating model performance
- key components:
  - **shared dataset:** common ground for training and testing models
  - **predefined task:** clearly defined problem(s) for models to solve
  - **evaluation metrics:** standardized criteria to evaluate models
- benefits:
  - facilitates fair comparison between different algorithms
  - promotes reproducibility and transparency in research
  - drives innovation through focused competition on shared tasks
- part of “frictionless reproducibility” (see week 2 reading)

# Models

# Tabular data task (generic)

	F1	F2	F3	F4
1	#	#	#	#
2	#	#	#	#
3	#	#	#	#
4	#	#	#	#
5	#	#	#	#
6	#	#	#	#
7	#	#	#	#
8	#	#	#	#
	$w_1$	$w_2$	$w_4$	$w_4$

$X$



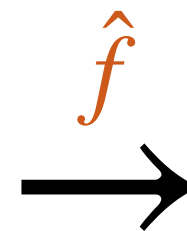
	output
1	#
2	#
3	#
4	#
5	#
6	#
7	#
8	#

$y$

# Tabular data task (generic)

	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	$w_1$	$w_2$	$w_4$	$w_4$

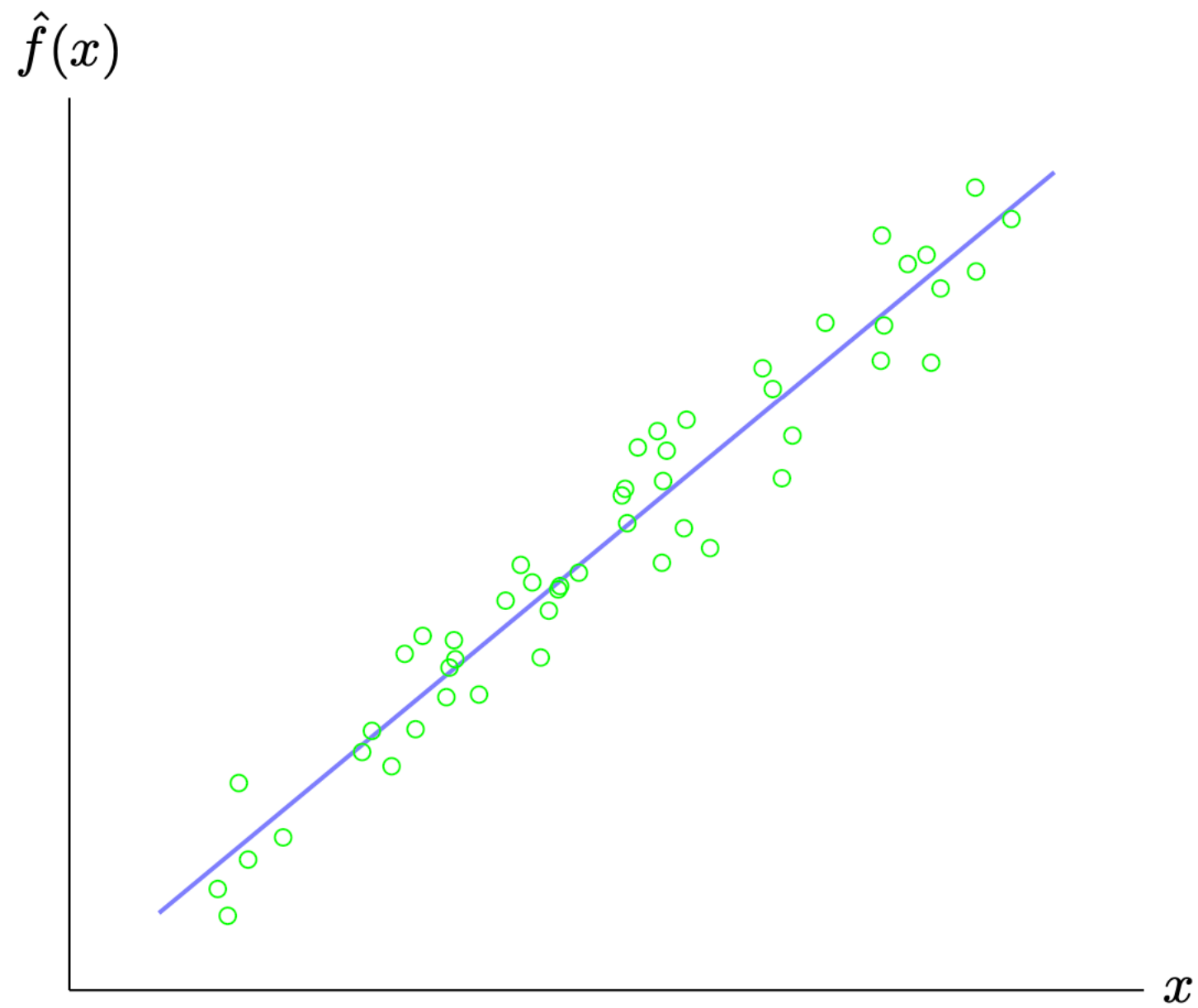
$X$



	income
1	\$35,000
2	\$62,500
3	\$88,000
4	\$42,750
5	\$110,250
6	\$75,500
7	\$143,000
8	\$53,600

$y$

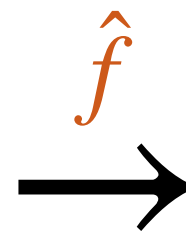
# Regression



# Tabular data task (binary classification)

	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	$w_1$	$w_2$	$w_4$	$w_4$

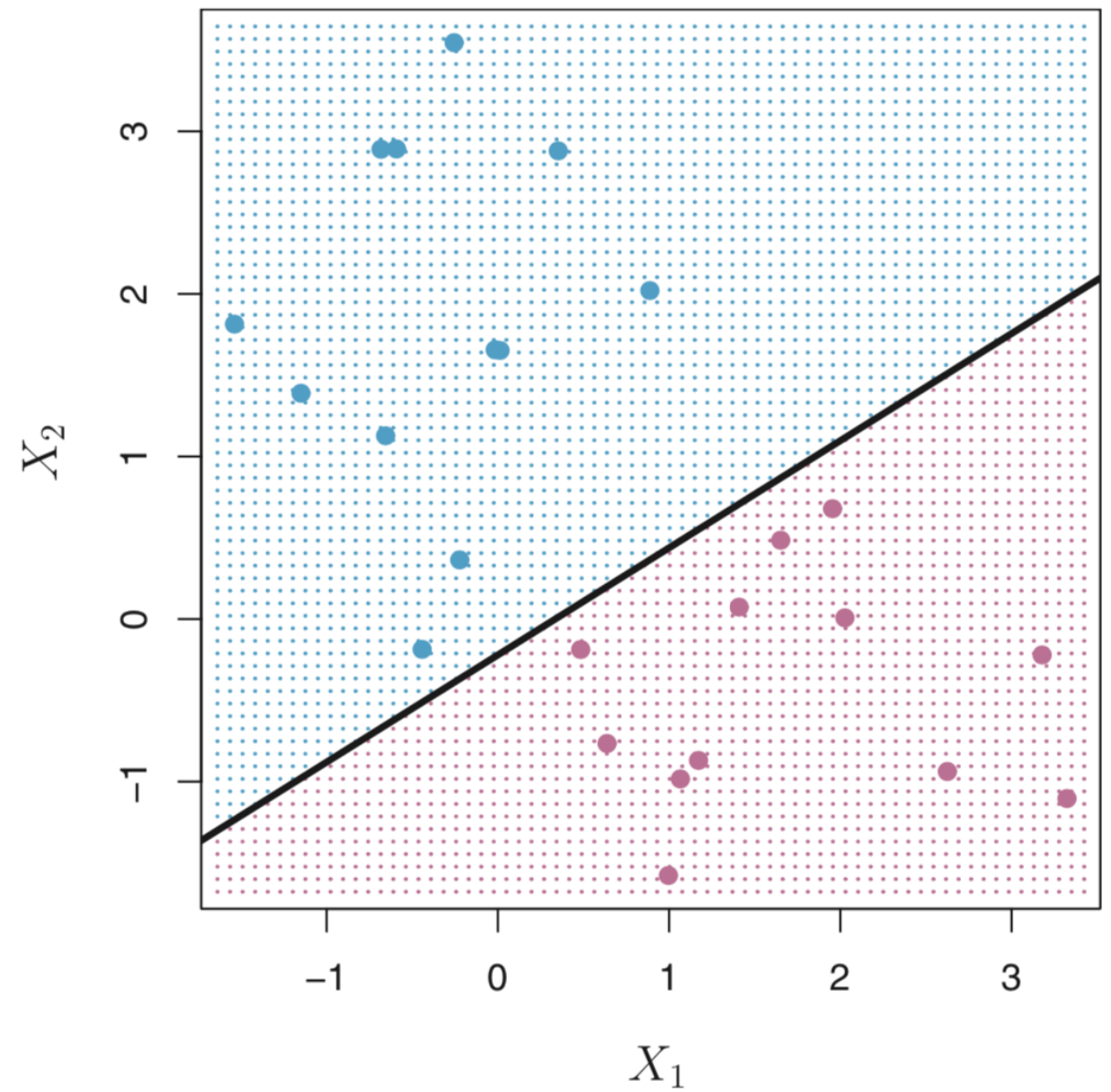
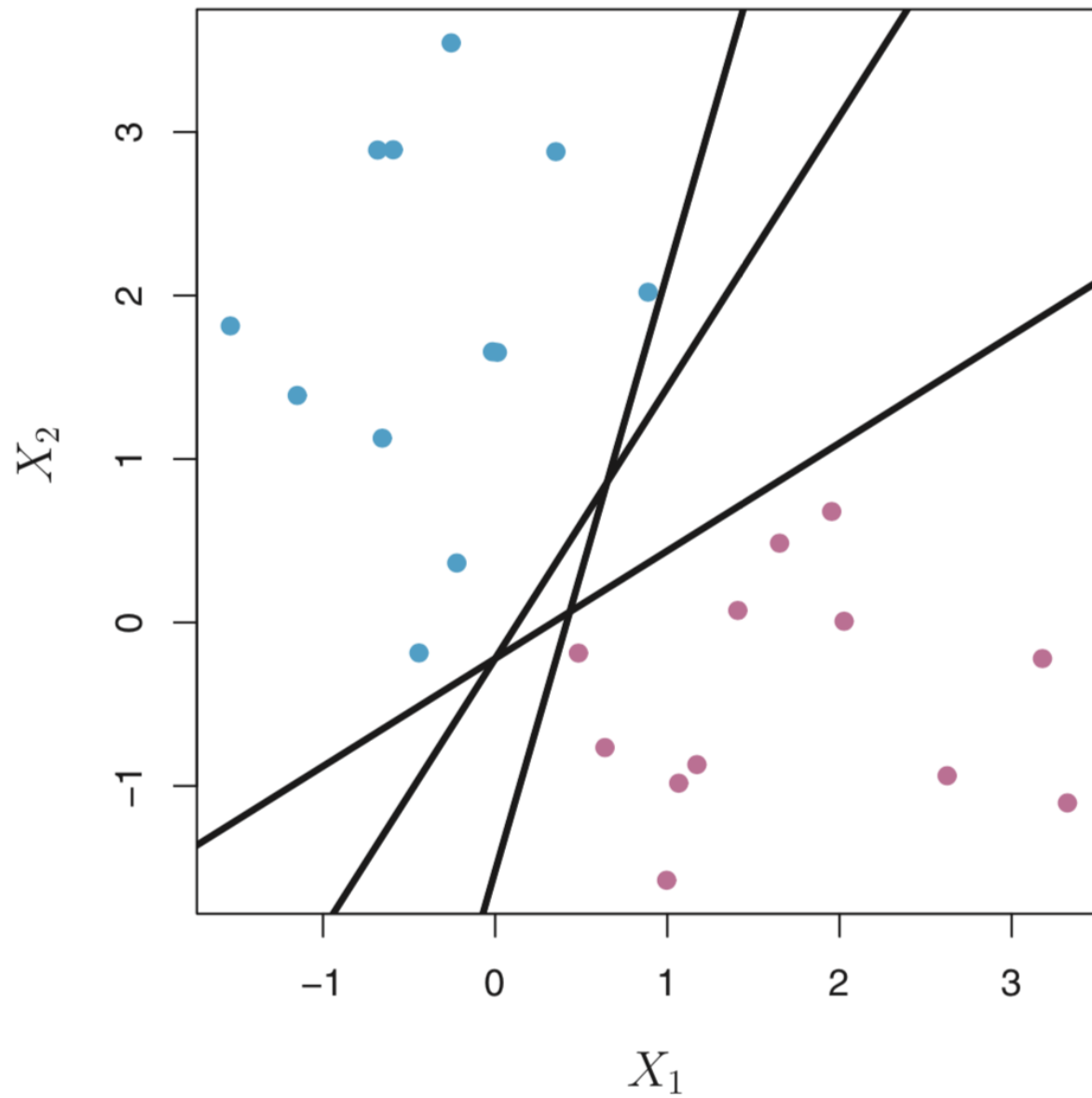
$X$



	graduate?
1	N
2	Y
3	N
4	N
5	Y
6	Y
7	Y
8	N

$y$

# (Binary) classification

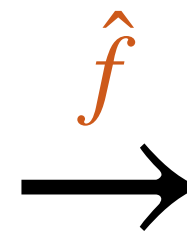




# Tabular data task (binary classification)

	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	$w_1$	$w_2$	$w_4$	$w_4$

$X$



	graduate?
1	N
2	Y
3	N
4	N
5	Y
6	Y
7	Y
8	N

$y$

# Some subtleties in supervised learning

	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	$w_1$	$w_2$	$w_4$	$w_4$



	graduate?
1	N
2	Y
3	N
4	N
5	Y
6	Y
7	Y
8	N



“Ground truth” labels

# Some subtleties in supervised learning

“Open category” problem

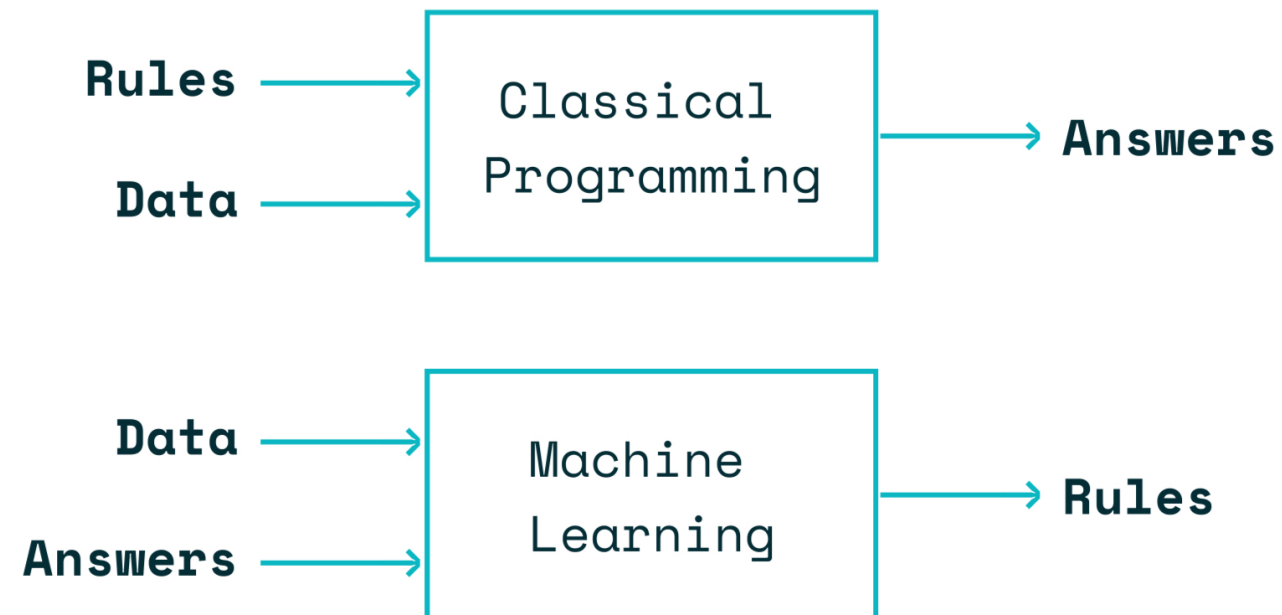


# Some standard (supervised) methods

Output	Model
(Binary) probability	Logistic regression
Continuous number	Linear regression
Structured output and/or Language/audio/image/video input	Deep neural network (with appropriate architecture)

# **Learning, Evaluation, and Model Selection**

# Models vs programs



The fundamental goal in machine learning is to *generalize* from the data you have to the data you have not yet seen.

This is distinct from what is sometimes the goal in social sciences, in which we may want to understand, *e.g.*, the statistical properties of a fixed historical dataset.



# The holdout method

The train/test split and estimation of generalization performance

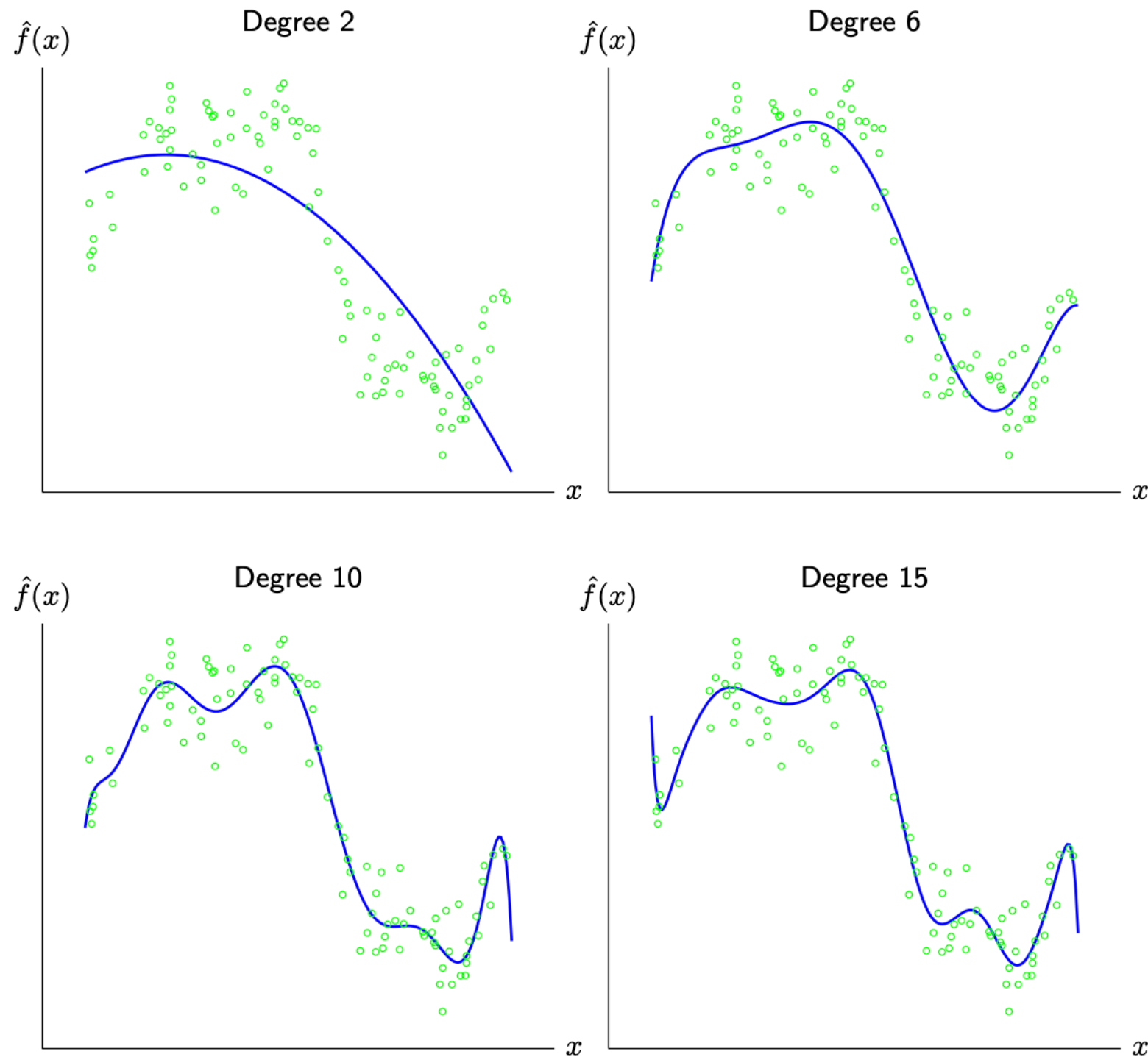
	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	⚙	⚙	⚙	⚙



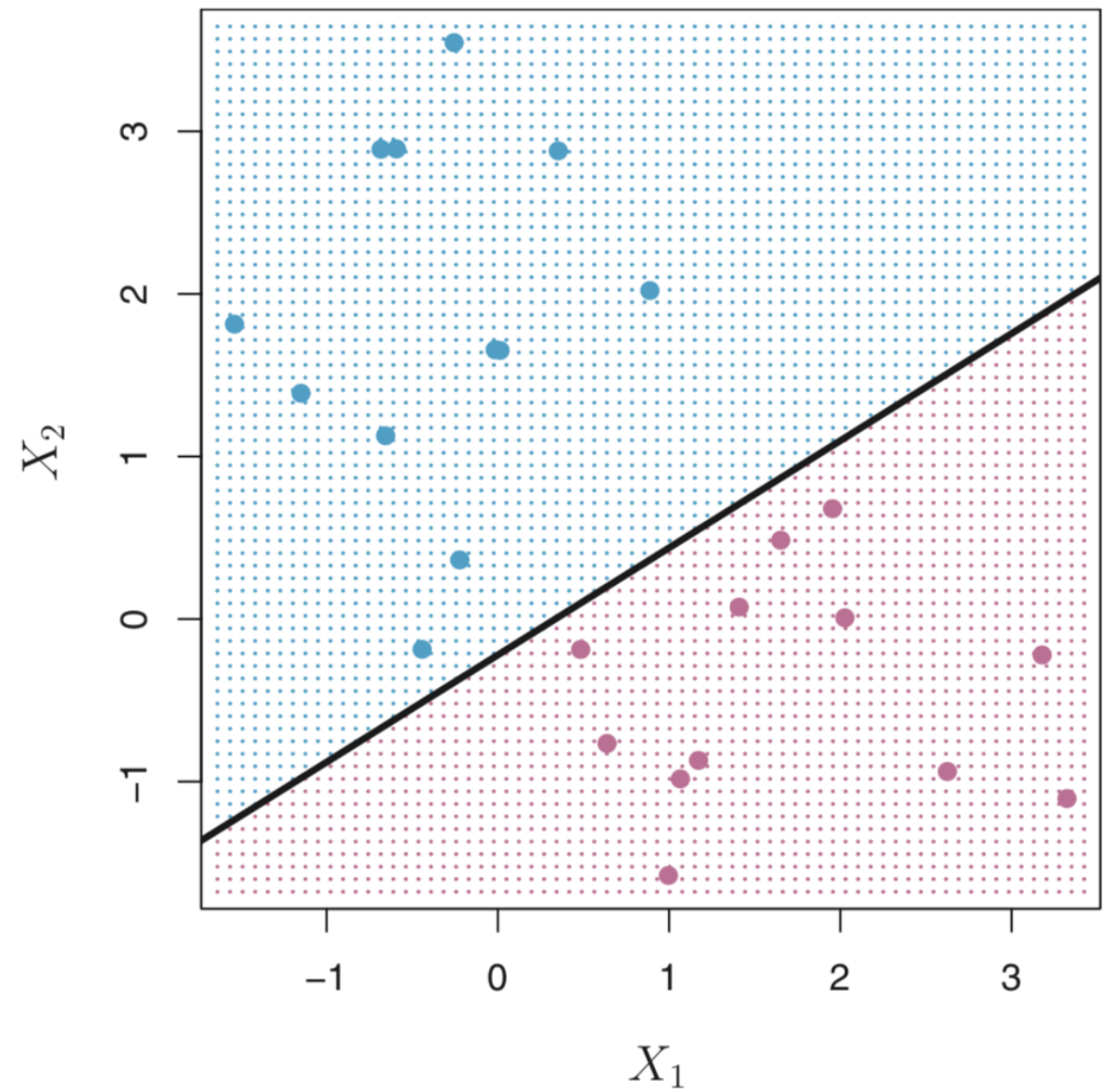
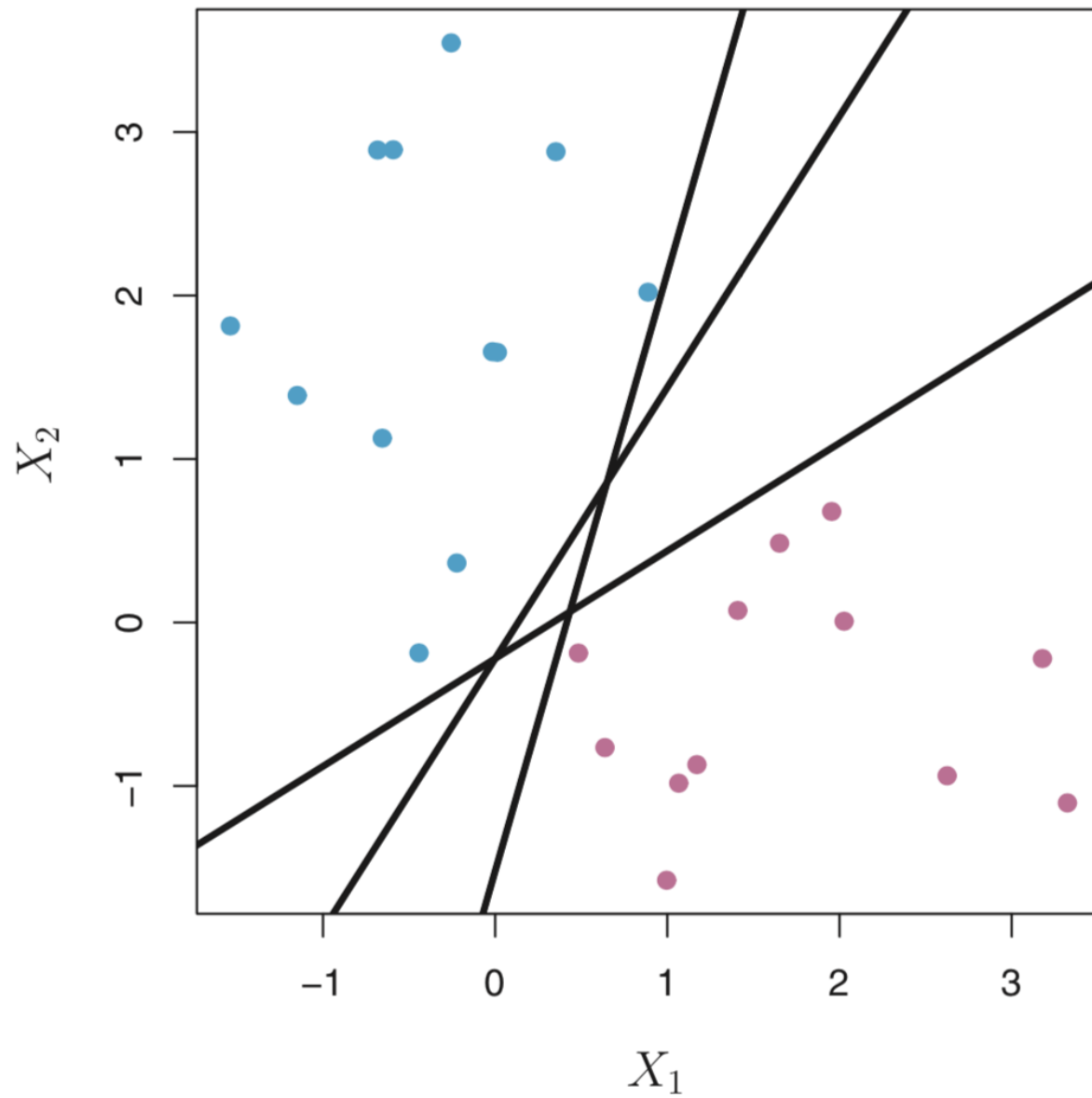
	graduate?
1	N
2	Y
3	N
4	N
5	Y
6	Y
7	Y
8	N

(an instance of a somewhat more sophisticated method called cross validation)

# Model complexity, overfitting, & underfitting

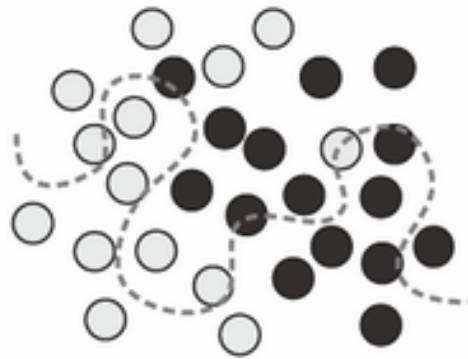


# Binary classification

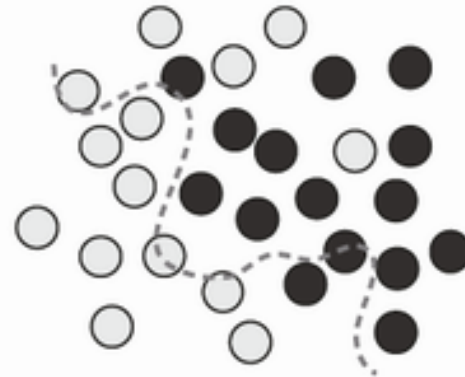


# Learning and overfitting

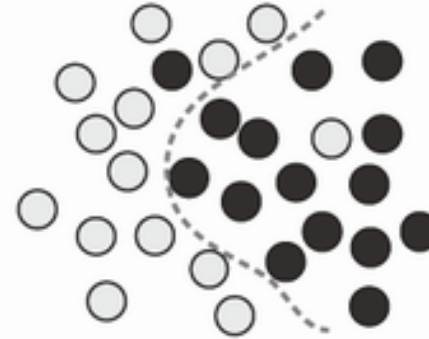
Before training:  
the model starts  
with a random initial state.



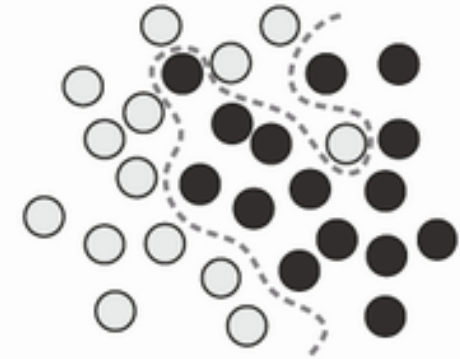
Beginning of training:  
the model gradually  
moves toward a better fit.



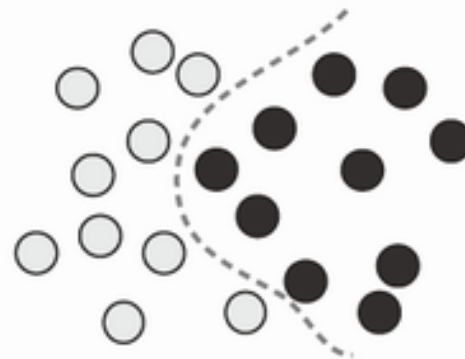
Further training: a robust  
fit is achieved, transitively,  
in the process of morphing  
the model from its initial  
state to its final state.



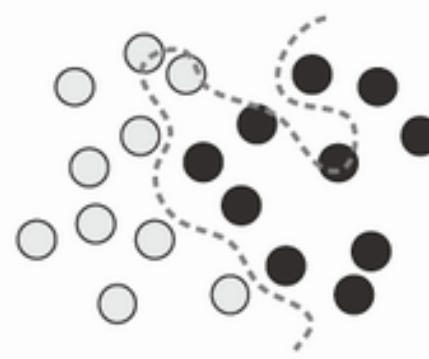
Final state: the model  
overfits the training data,  
reaching perfect training loss.



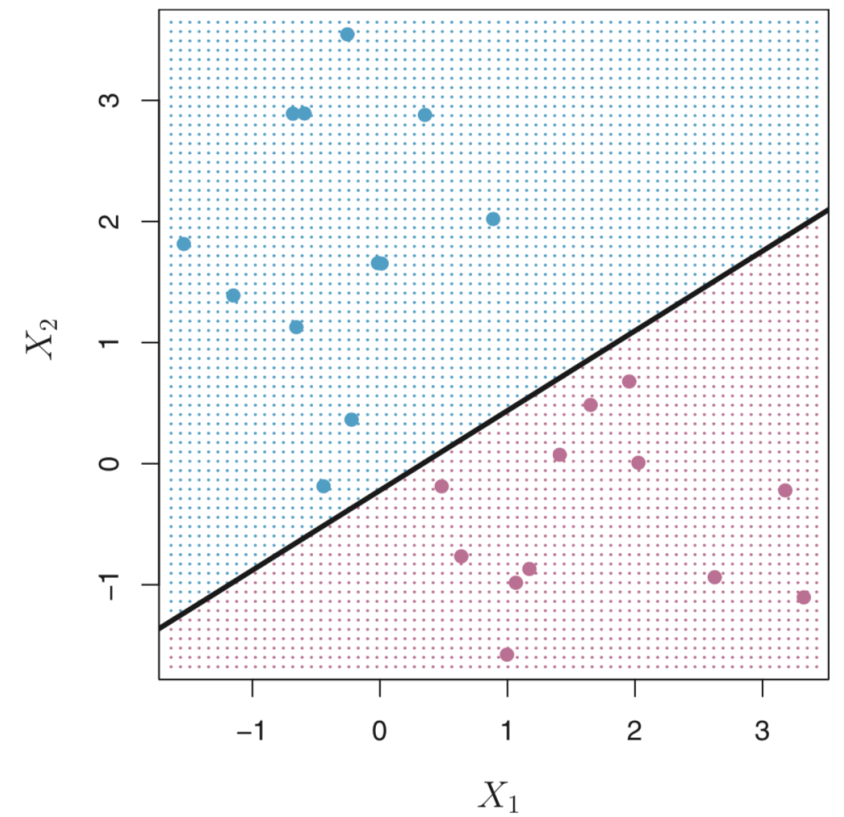
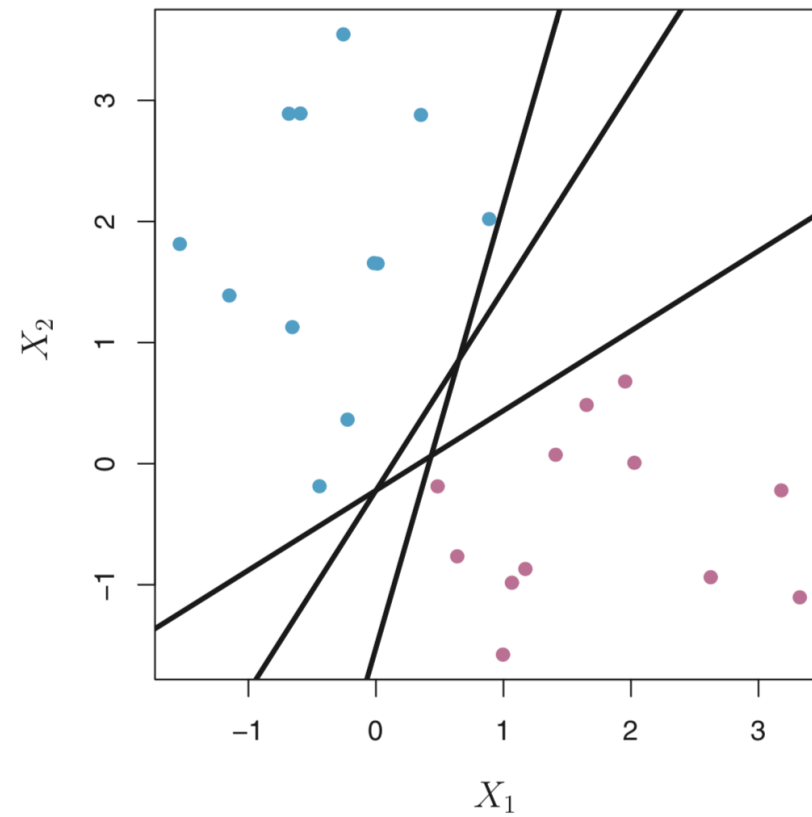
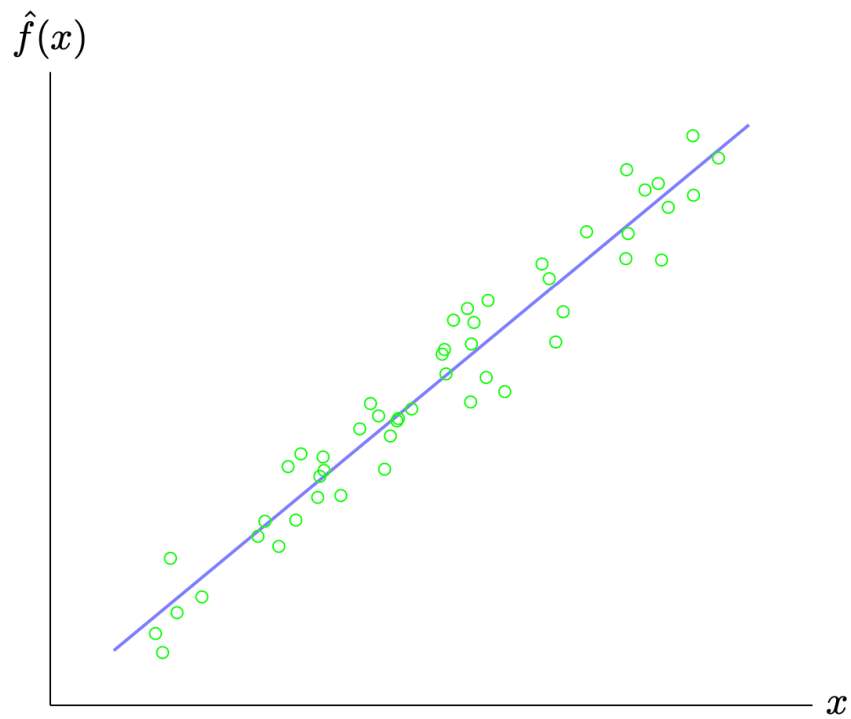
Test time: performance  
of robustly fit model  
on new data points



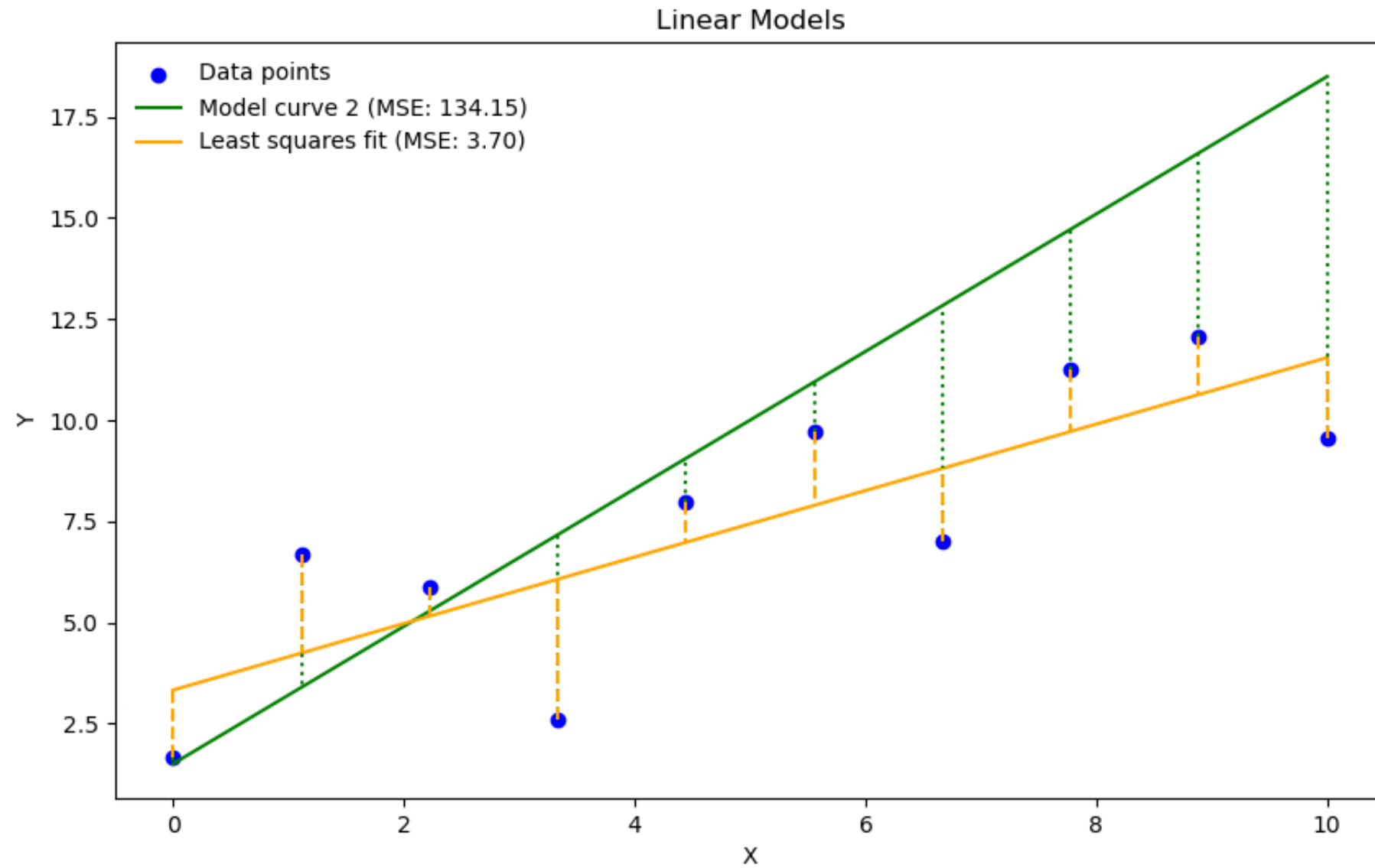
Test time: performance  
of overfit model  
on new data points



# Loss functions for learning



# Least squares and mean squared error



# Evaluating (binary) classifiers

- with probabilistic classifiers, need to pick decision threshold
- compute evaluation metric for classifier
- many possible evaluation metrics
- other loss functions used for other models/tasks, and models can be evaluated using metrics other than just the loss (if the other metrics, e.g., correspond more closely to some business/policy consideration, or are additional concerns like fairness)

# Confusion matrix

TP	FN
FP	TN



# Error rate

TP	FN
FP	TN

$$\text{error rate} = \frac{\text{FP} + \text{FN}}{N}$$

what fraction of all predictions were wrong?

# Accuracy

TP	FN
FP	TN

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{N}$$

what fraction of all predictions were right?

# Precision

TP	FN
FP	TN

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

of positive predictions, how many were right?

# Recall / sensitivity / true positive rate

TP	FN
FP	TN

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

of all positives, what proportion were identified?

# Specificity / true negative rate

TP	FN
FP	TN

$$\text{TNR} = \frac{\text{TN}}{\text{FP} + \text{TN}}$$

of all negatives, what proportion were identified?

# False positive rate

TP	FN
FP	TN

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

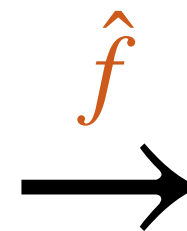
proportion of wrongly identified positives (false alarms)

# Problem Formulation

# Problem formulation process

	age	gpa	gender	zip
1	14	1.8	M	10011
2	15	3.2	M	10023
3	14	2.6	F	10029
4	16	2.9	M	10033
5	17	3.4	F	10040
6	16	3.9	F	10016
7	15	3.6	M	10027
8	18	2.3	M	10032
	$w_{\text{age}}$	$w_{\text{gpa}}$	$w_{\text{gender}}$	$w_{\text{zip}}$

$X$



	output
1	N
2	Y
3	N
4	N
5	Y
6	Y
7	Y
8	N

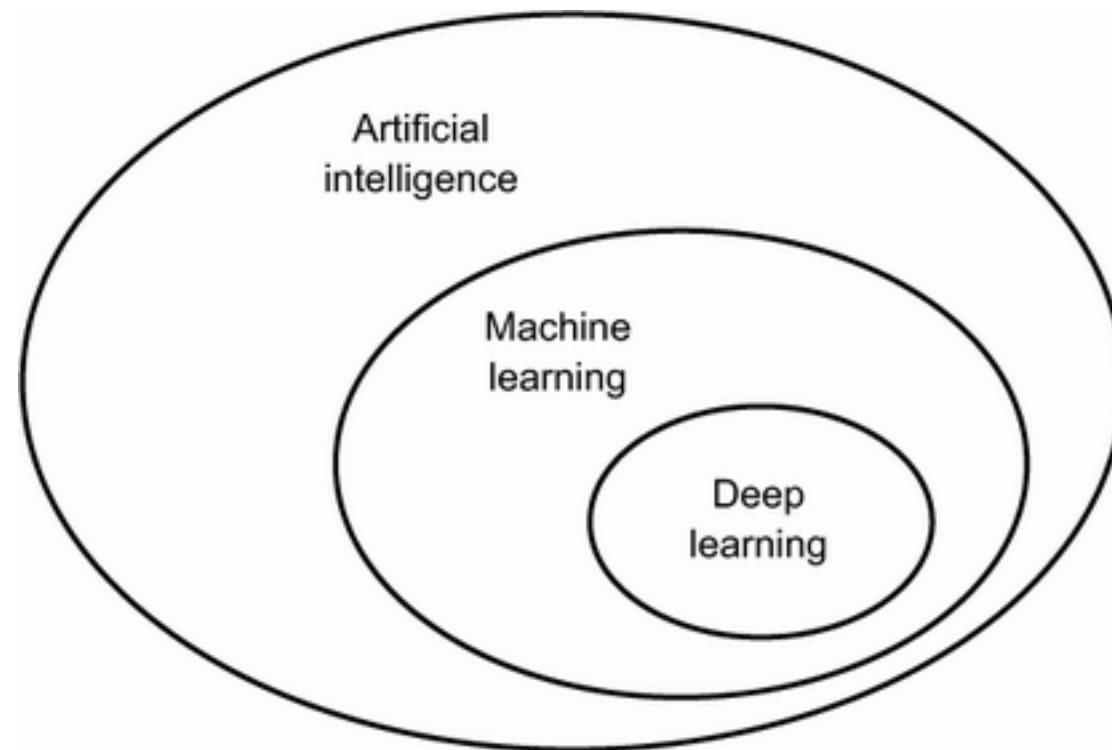
$y$

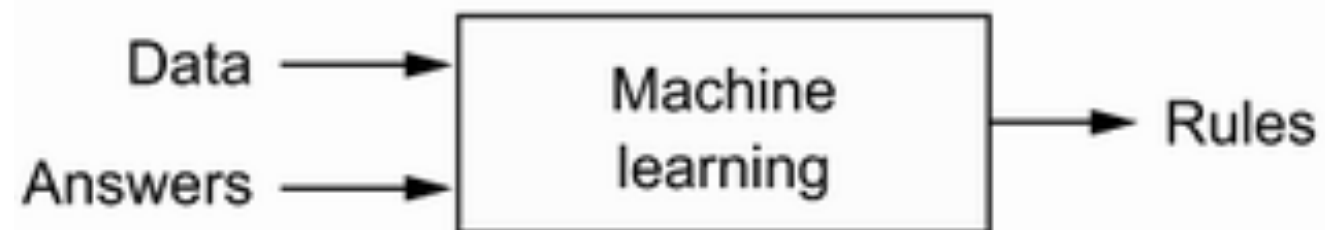
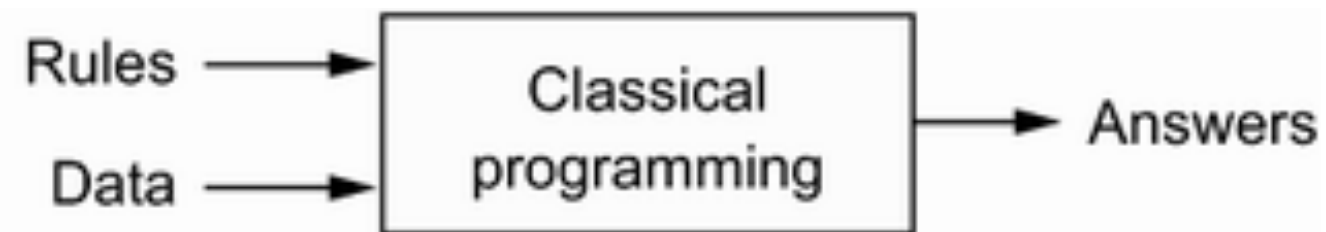


# Sample problem formulation process

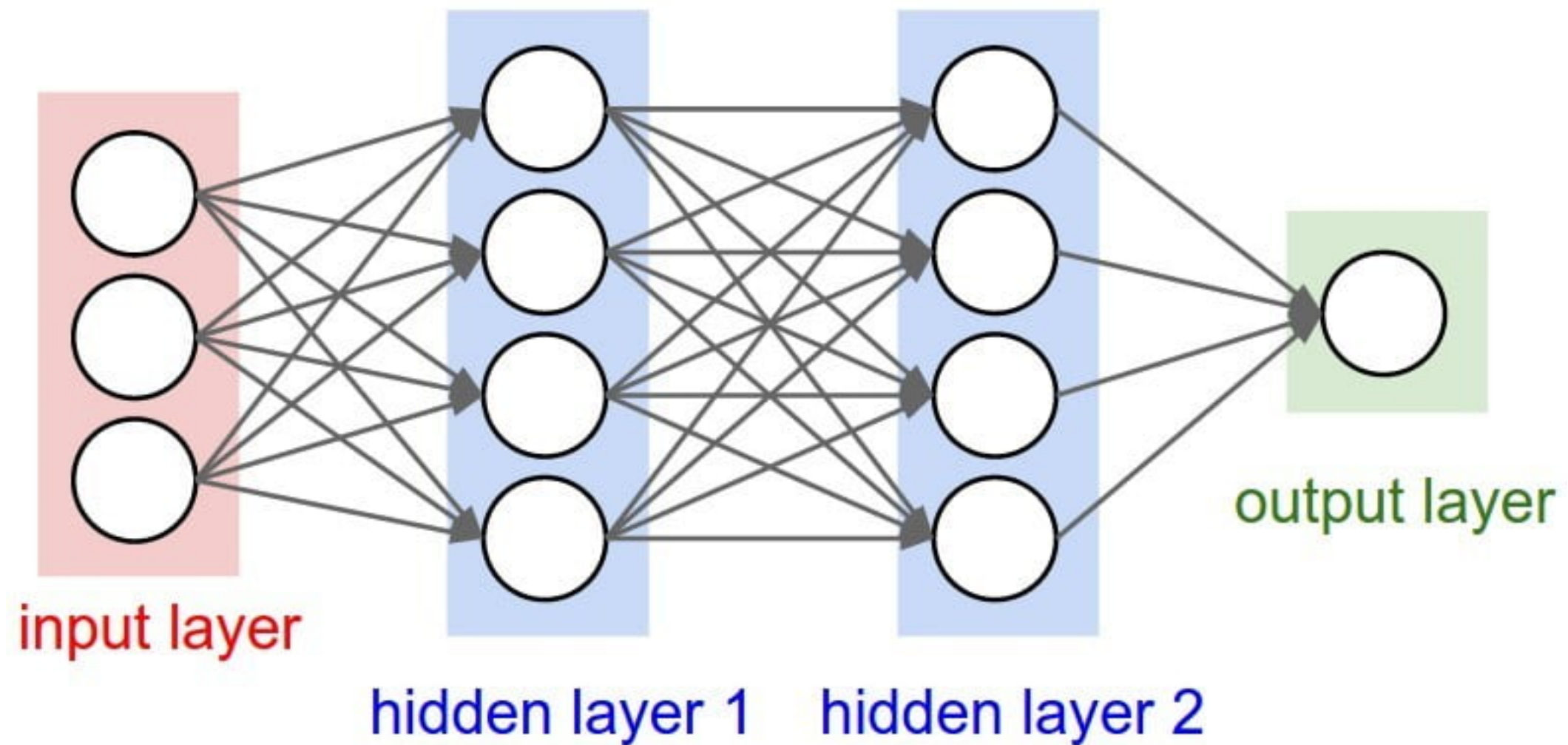
1. Determine task: output type  $y$  and type of input example  $x_i$
2. Identify task type (eg, binary classification) based on  $y$ , and consider availability and encoding of ground truth for  $y$
3. Brainstorm potentially relevant features + their encoding
4. Brainstorm dataset availability (where to get full  $X$ , if at all)
5. Pick a model (and loss function) based on task type
6. Think about model evaluation (eg, FP vs FN)
7. Then can think about deployment, monitoring, etc.

# Deep Learning

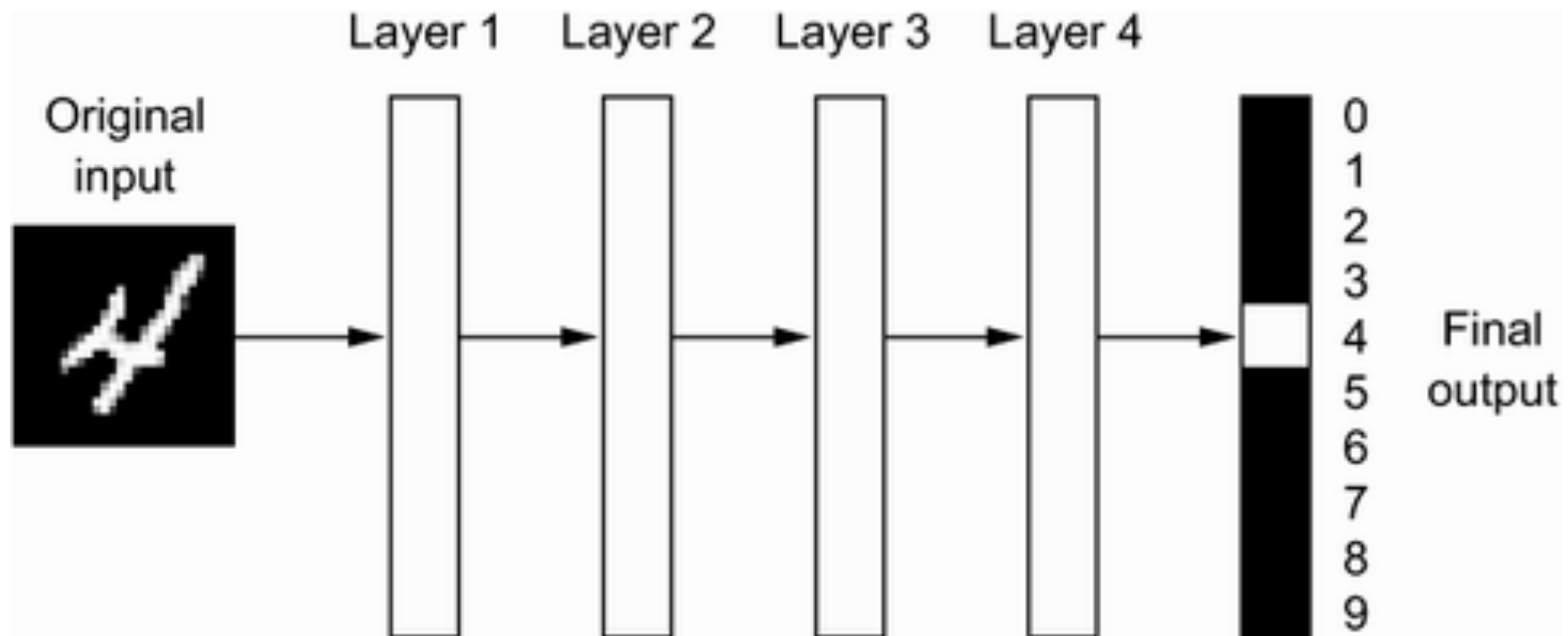


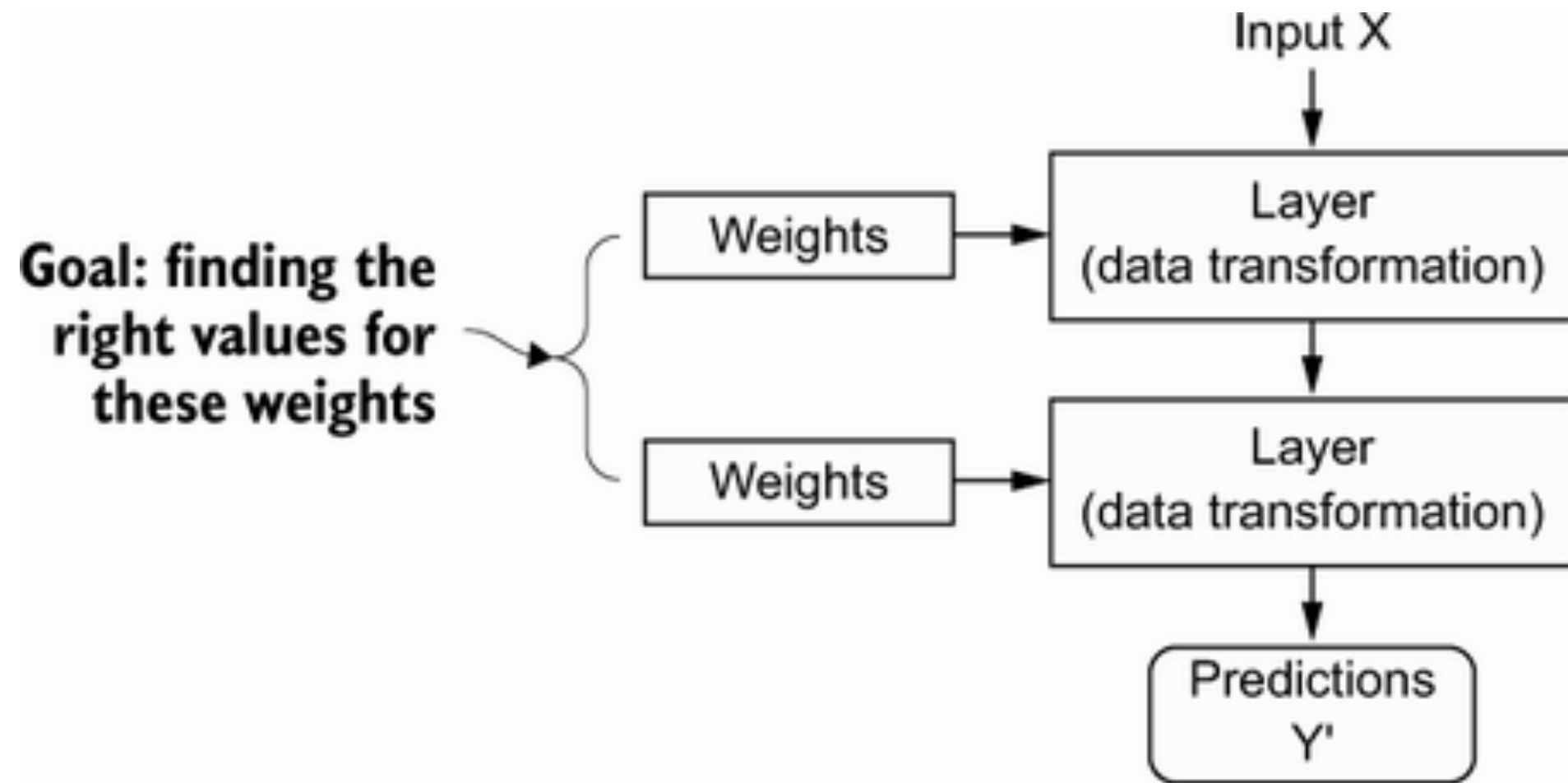


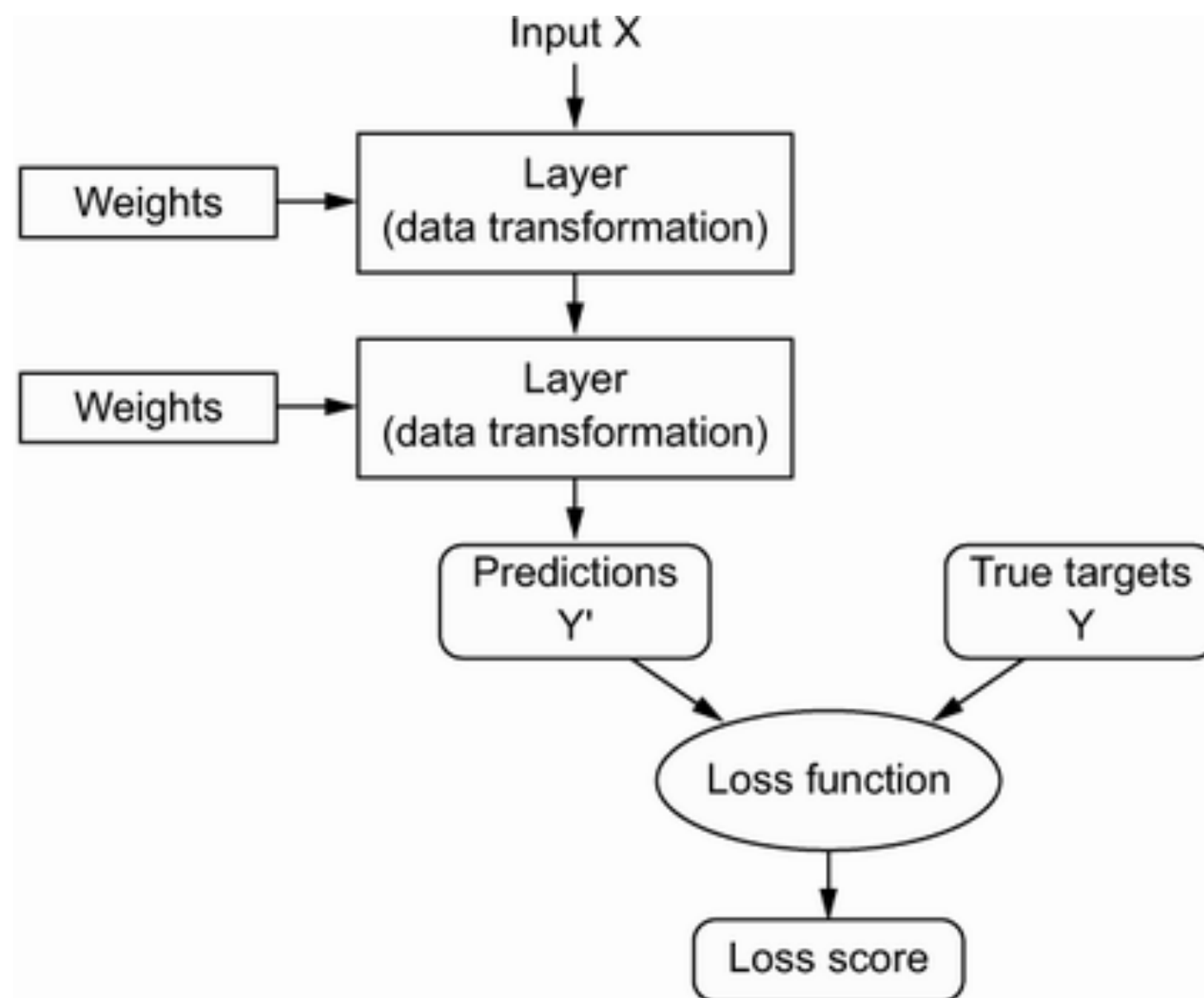
# Neural network



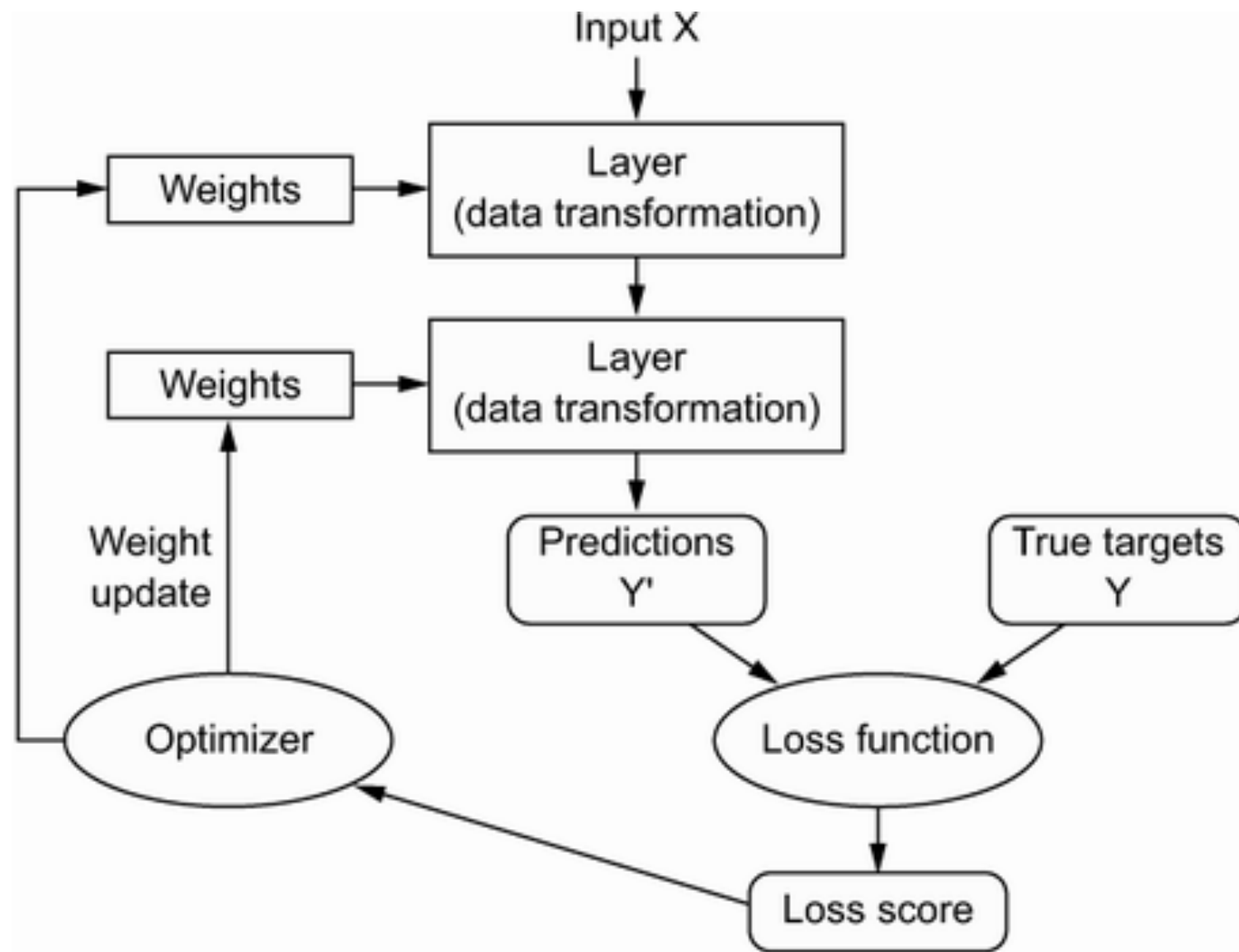
# Image classification



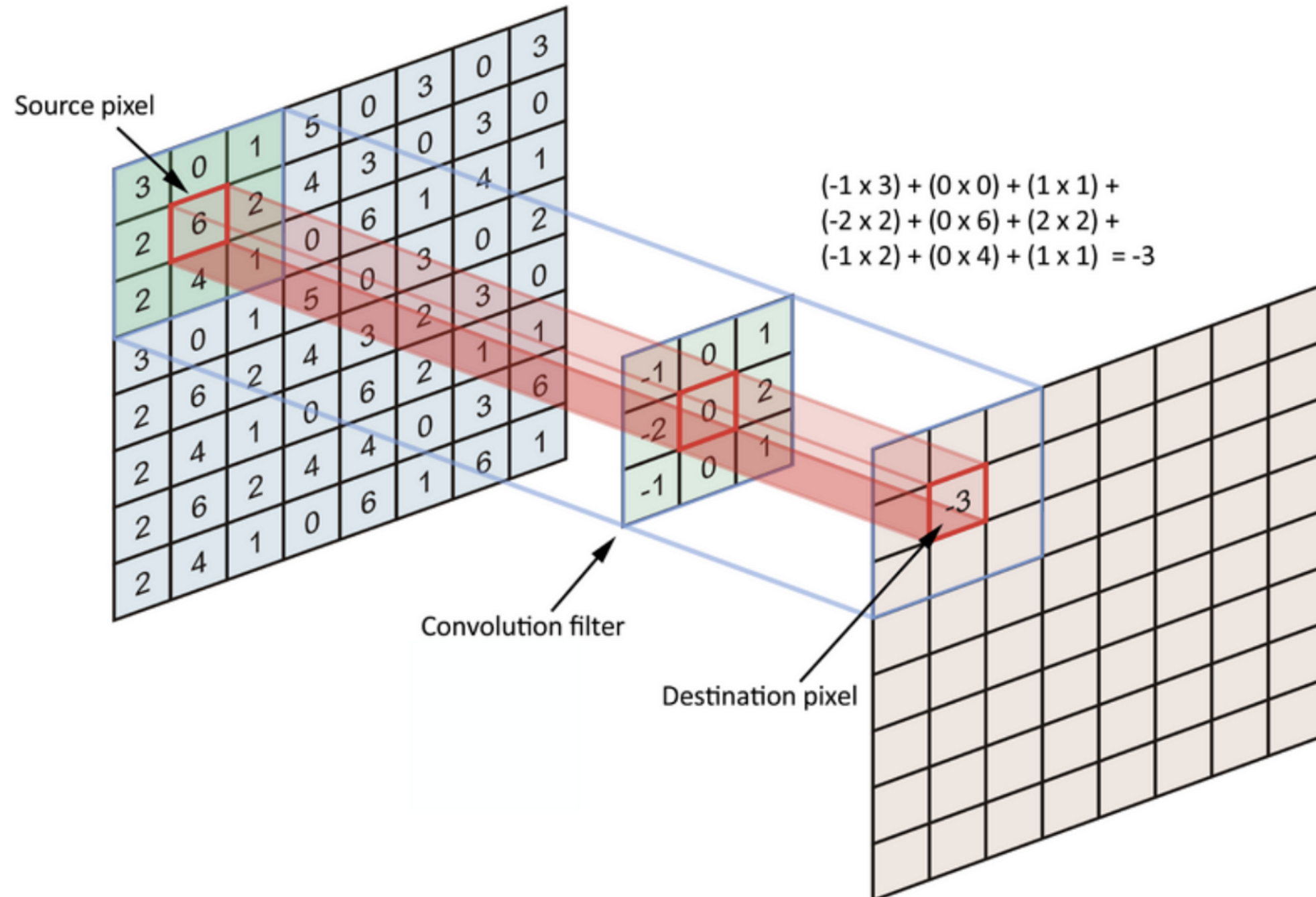




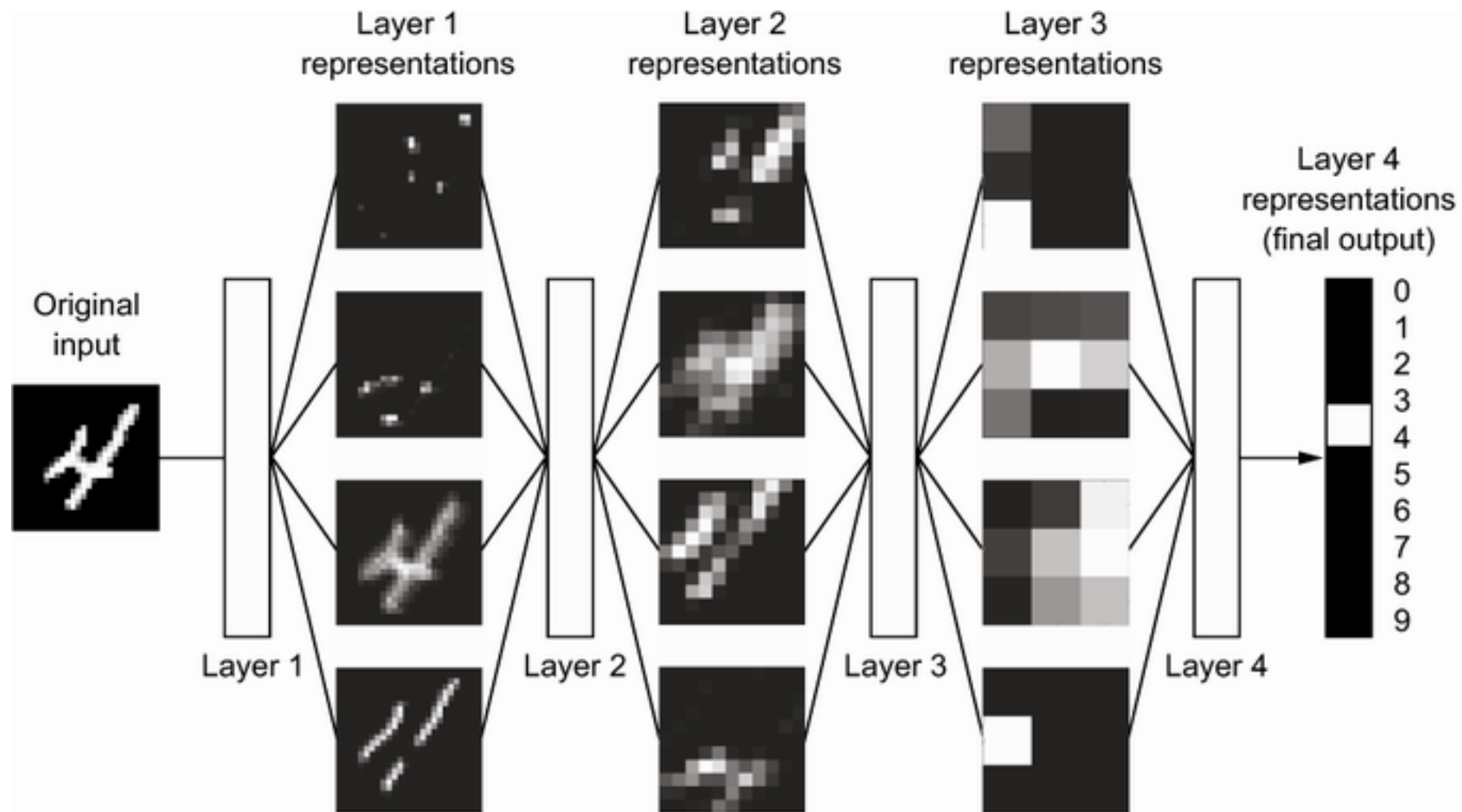




# Convolutional neural network (CNN)



# Learned data representations



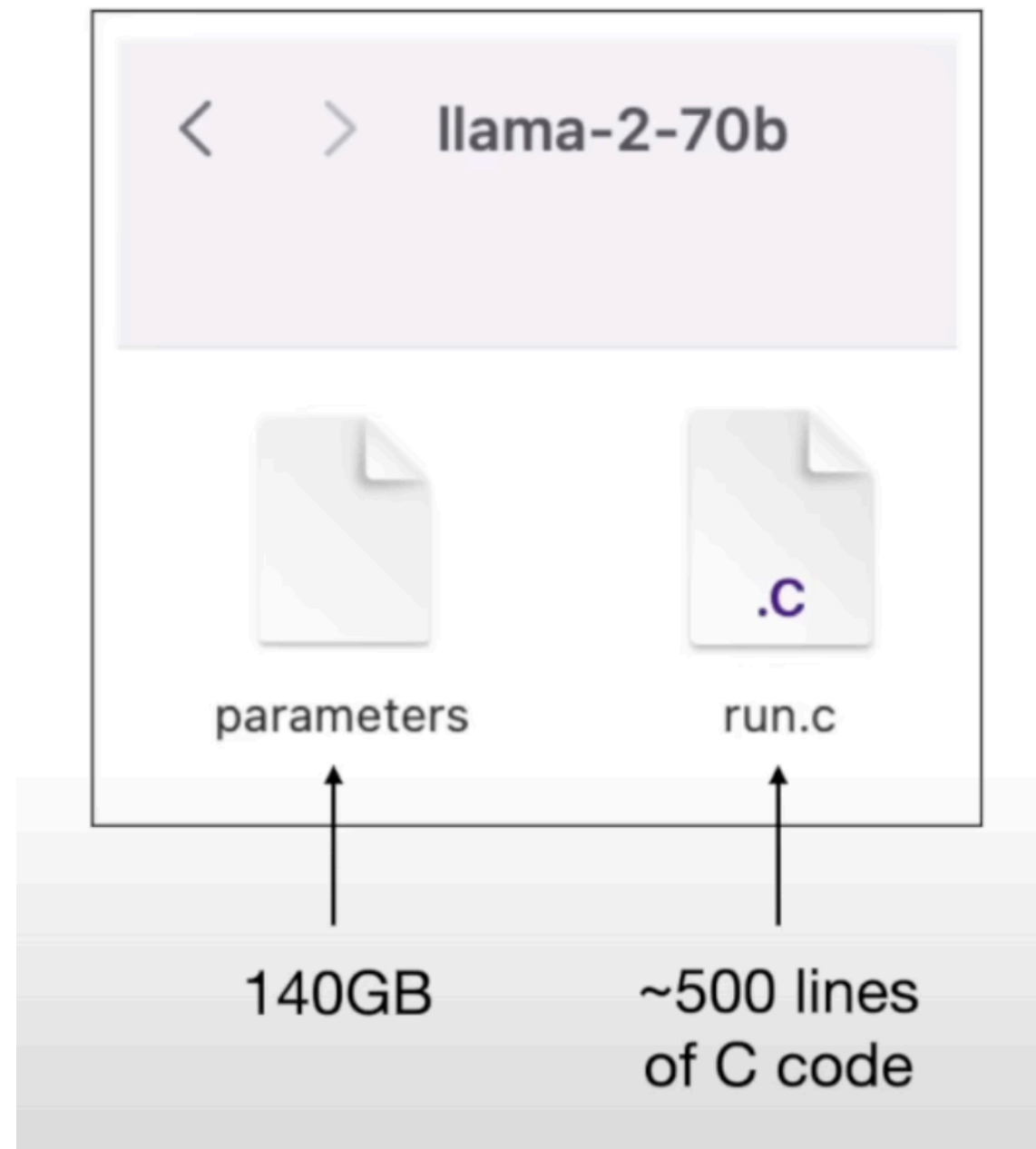
# Language Models

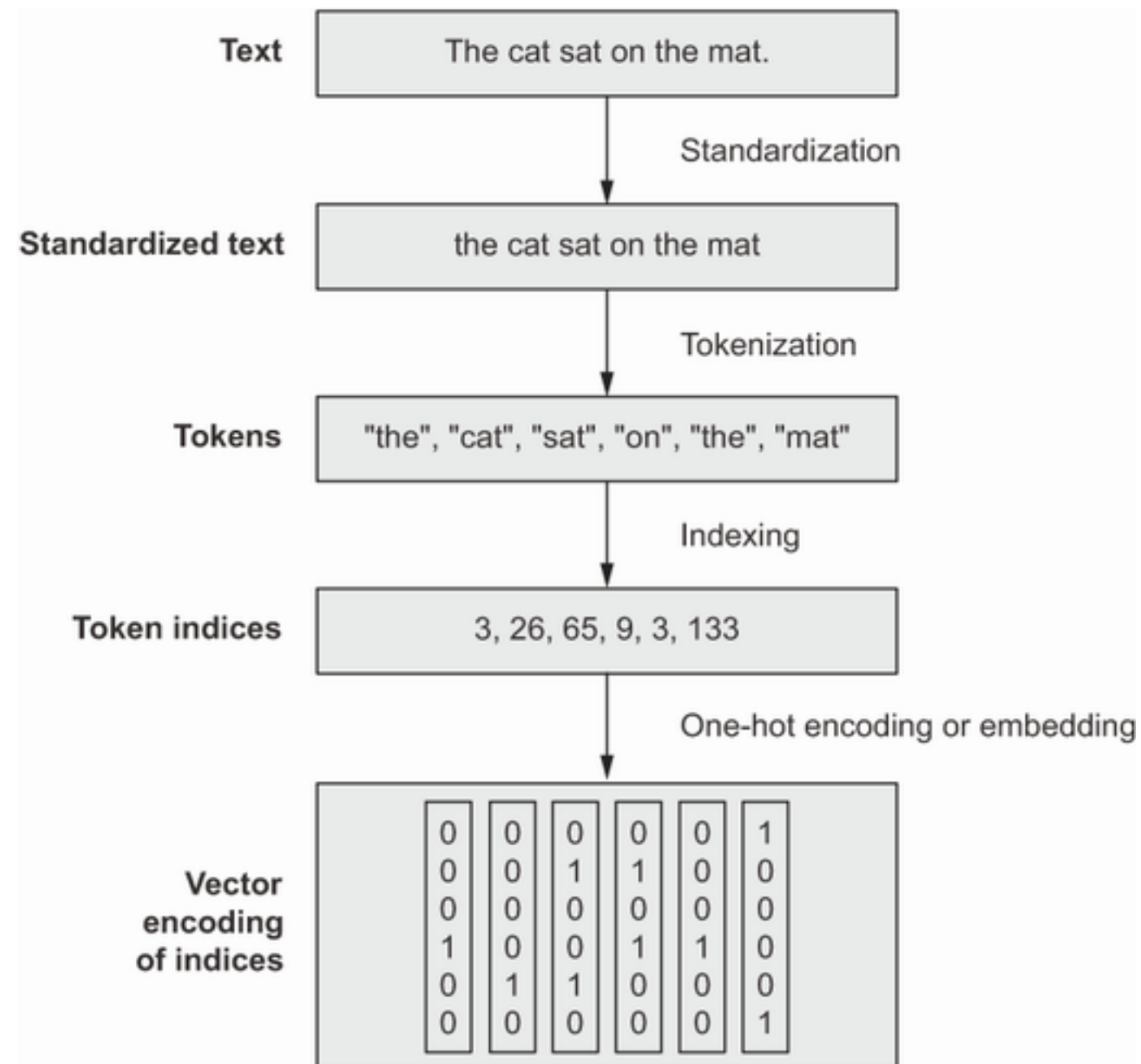
# Topics

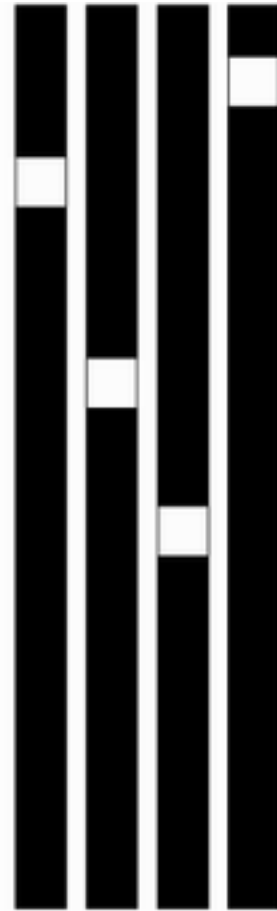
- word embeddings
- self-supervised learning
- transformer architecture and attention mechanism
- sampling to predict the next word
- autoregressive models
- LLM training process
- encoding various tasks as next token prediction
- prompt engineering
- to revisit later: attacks against LLMs, safety/security issues, etc.

# Large Language Models

(Karpathy)

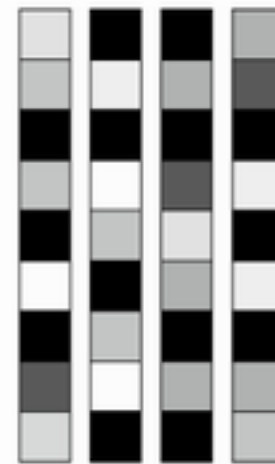






One-hot word vectors:

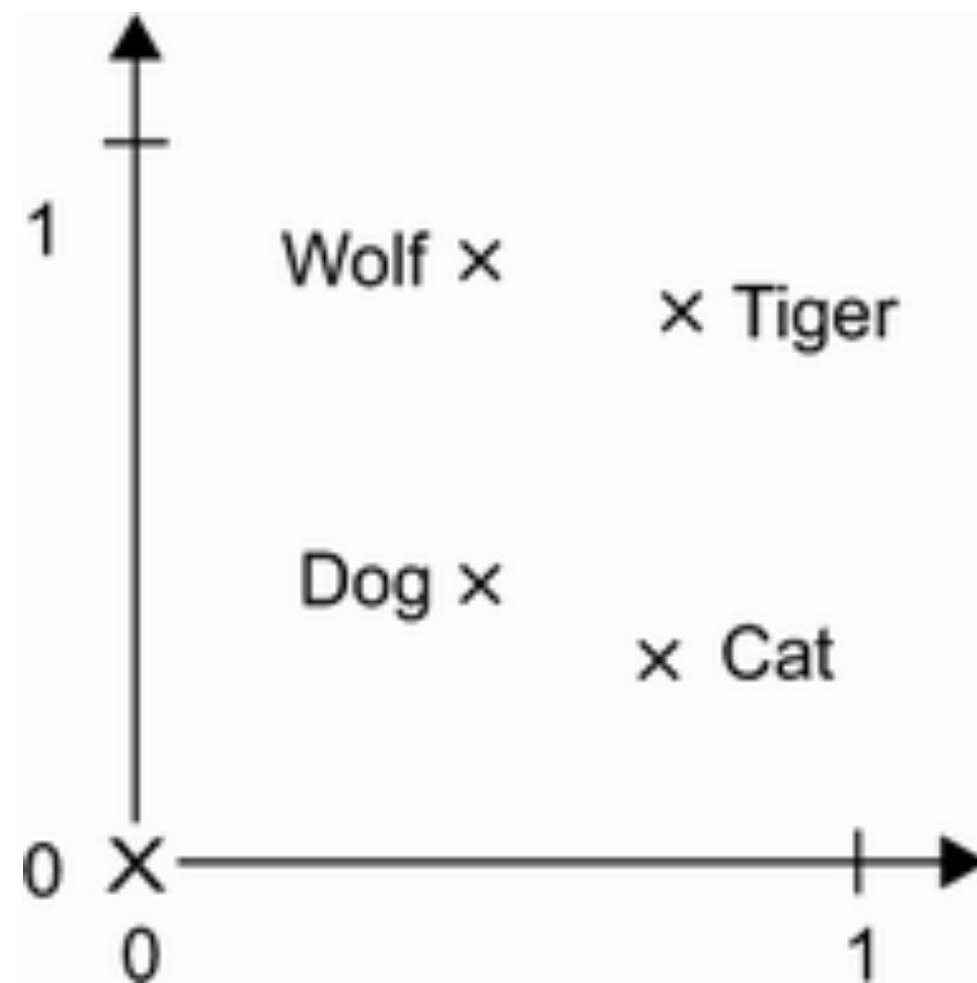
- Sparse
- High-dimensional
- Hardcoded



Word embeddings:

- Dense
- Lower-dimensional
- Learned from data





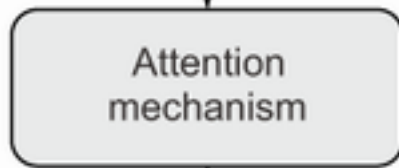
# Transformers, attention, and GPTs

- deep learning models currently used in modern LMs are called “transformers”, designed for sequential data like text
- use “attention” to help the model understand *context*, and help the model use long-range dependencies when predicting the next word
- GPT = “generative pre-trained transformer”
  - multiple transformer layers (to capture increasing complexity)
  - **pre-training**: train on tons of general text by predicting the next word given previous words (autoregressive model)
  - **fine-tuning**: train on data specific to task (e.g., Q&A)

Original  
representation



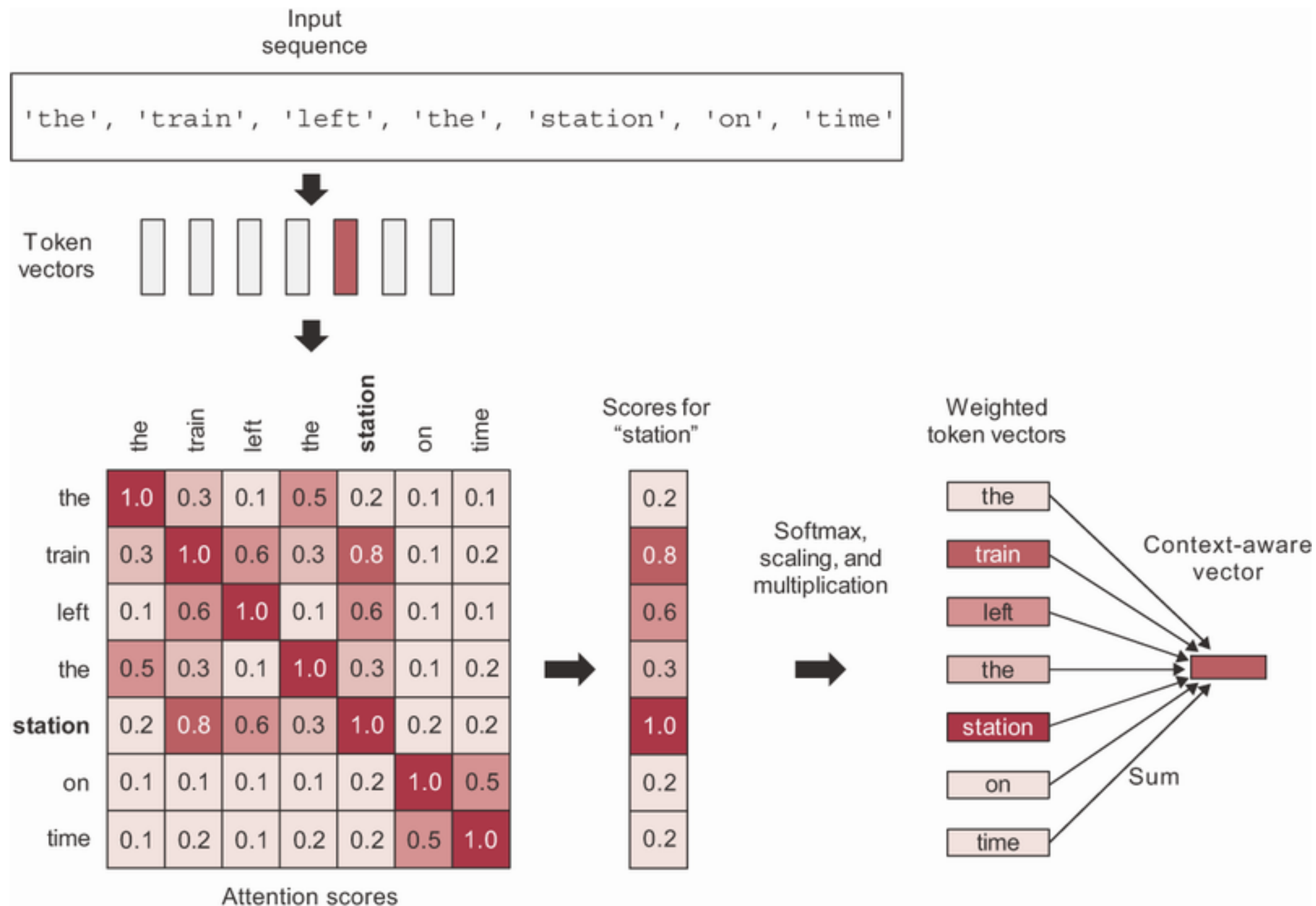
Attention  
mechanism



Attention scores

New  
representation

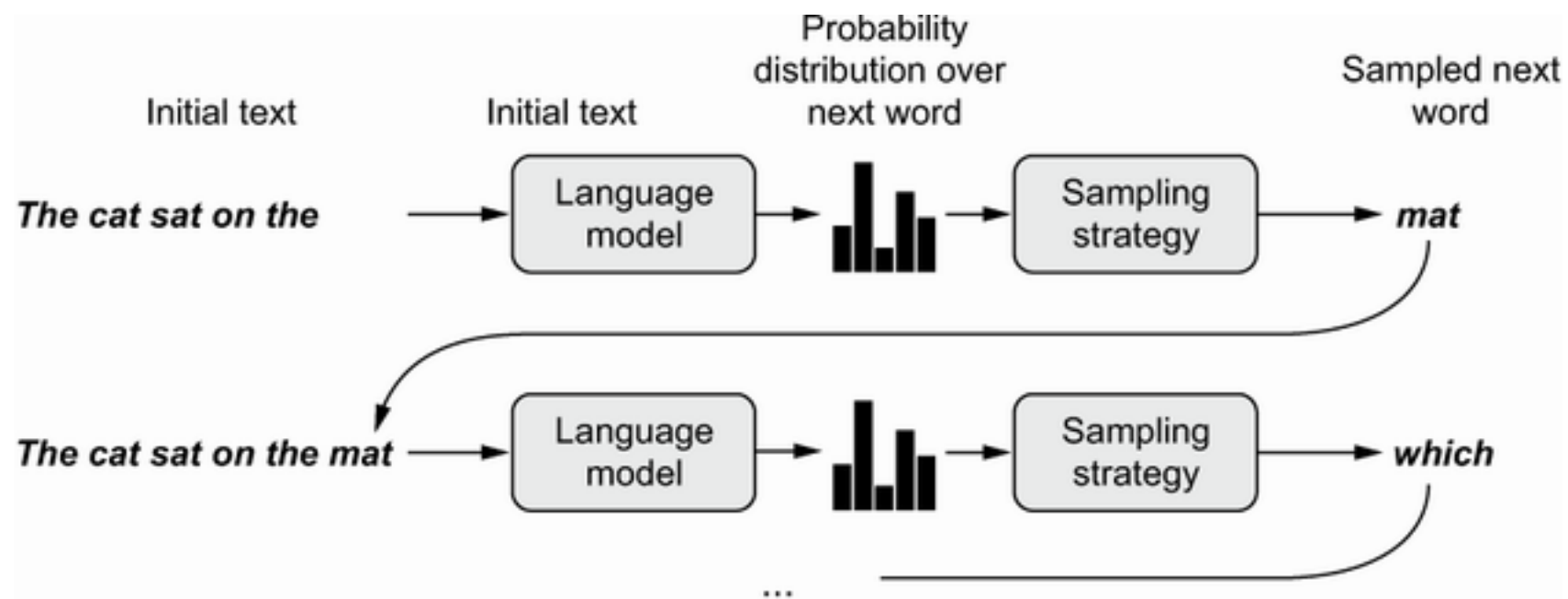




# Sampling

- sampling
- stochastic sampling
- temperature
- (there are also other fancier strategies)

# Sampling



# Predicting the next word

*The best thing about AI is its ability to*

learn	4.5%
predict	3.5%
make	3.2%
understand	3.1%
do	2.9%

# Successive completions

```
Out[•]= { The best thing about AI is its ability to,  
The best thing about AI is its ability to learn,  
The best thing about AI is its ability to learn from,  
The best thing about AI is its ability to learn from experience,  
The best thing about AI is its ability to learn from experience.,  
The best thing about AI is its ability to learn from experience. It,  
The best thing about AI is its ability to learn from experience. It's,  
The best thing about AI is its ability to learn from experience. It's not }
```



# Selecting maximum probability words

The best thing about AI is its ability to learn from experience. It's not just a matter of learning from experience, it's learning from the world around you. The AI is a very good example of this. It's a very good example of how to use AI to improve your life. It's a very good example of how to use AI to improve your life. The AI is a very good example of how to use AI to improve your life. It's a very good example of how to use AI to

# Different samplings

The best thing about AI is its ability to learn. I've always liked the

---

The best thing about AI is its ability to really come into your world and just

---

The best thing about AI is its ability to examine human behavior and the way it

---

The best thing about AI is its ability to do a great job of teaching us

---

The best thing about AI is its ability to create real tasks, but you can

# Temperature-based sampling

The best thing about AI is its ability to automate processes and make decisions quickly and accurately. AI can be used to automate mundane tasks, such as data entry, and can also be used to make complex decisions, such as predicting customer behavior or analyzing large datasets. AI can also be used to improve customer service, as it can quickly and accurately respond to customer inquiries. AI can also be used to improve the accuracy of medical diagnoses and to automate the process of drug discovery.

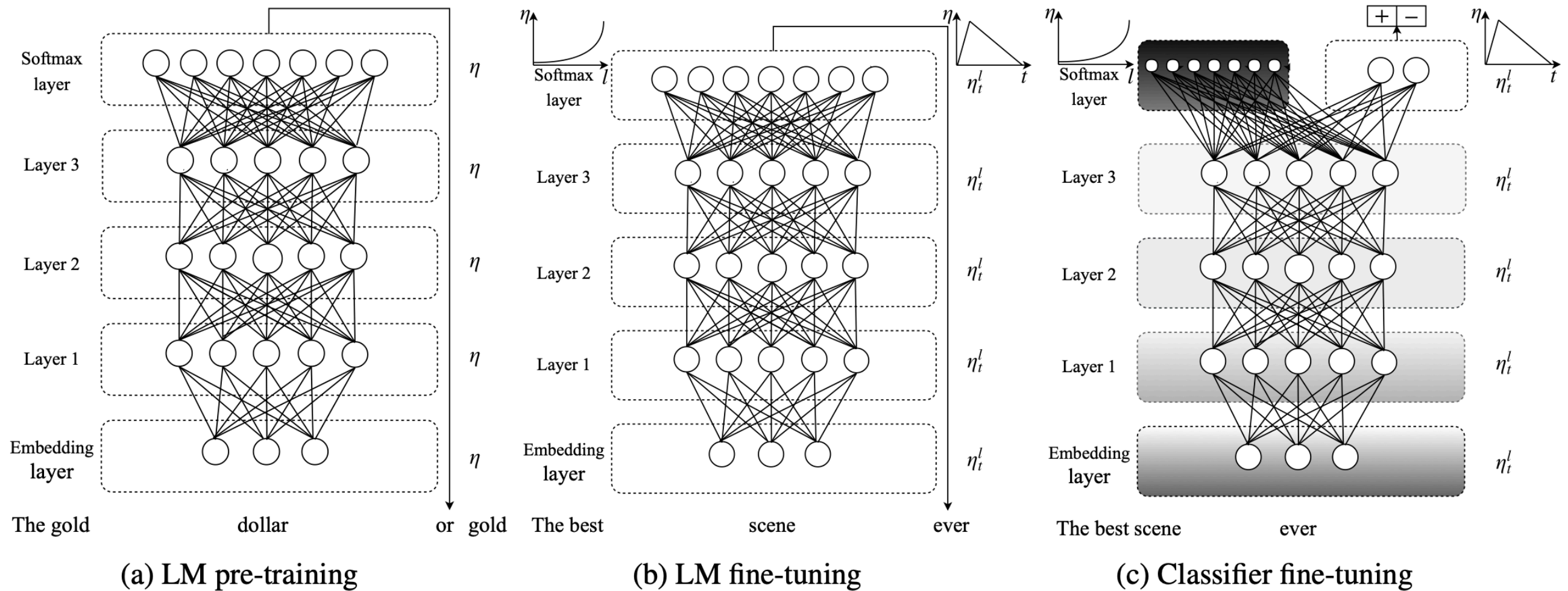
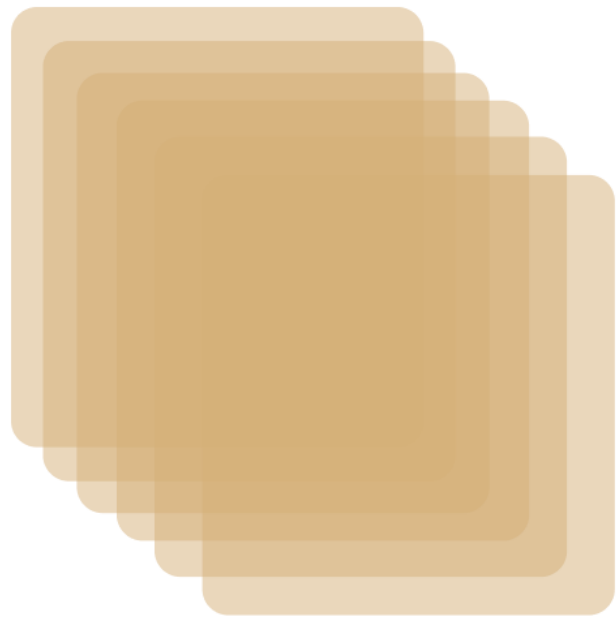
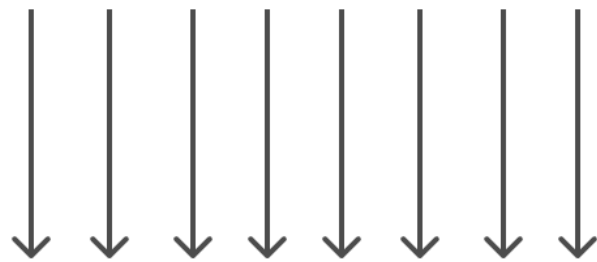


Figure 1: ULMFiT consists of three stages: a) The LM is trained on a general-domain corpus to capture general features of the language in different layers. b) The full LM is fine-tuned on target task data using discriminative fine-tuning (*‘Discr’*) and slanted triangular learning rates (STLR) to learn task-specific features. c) The classifier is fine-tuned on the target task using gradual unfreezing, *‘Discr’*, and STLR to preserve low-level representations and adapt high-level ones (shaded: unfreezing stages; black: frozen).

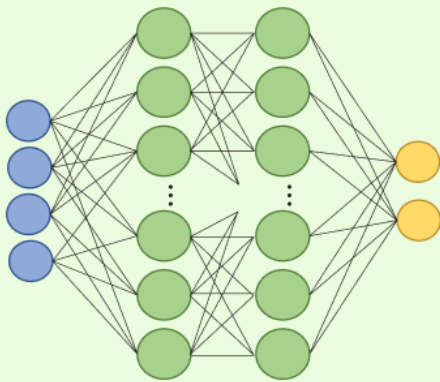
## Prompts Dataset



Sample many prompts



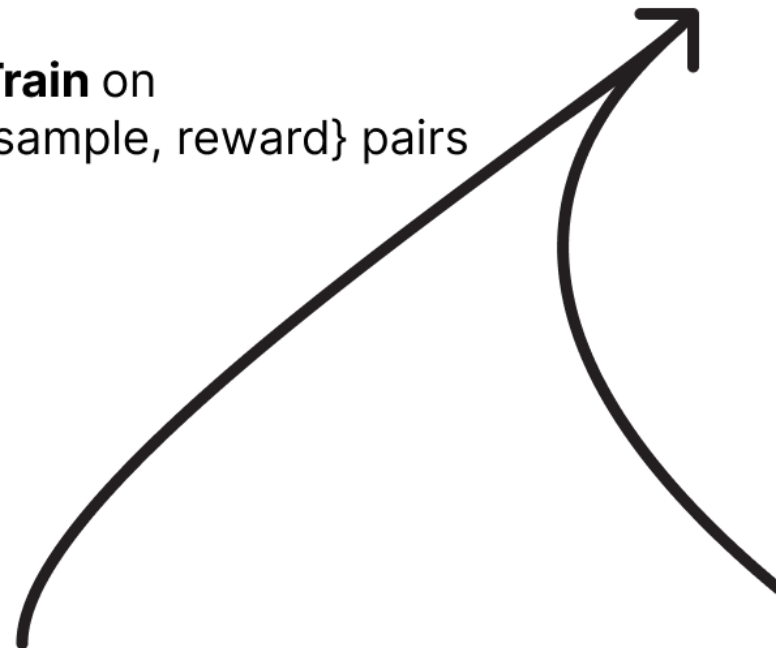
## Initial Language Model



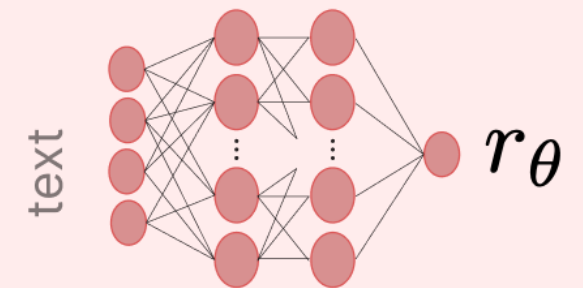
Lorem ipsum dolor  
sit amet, consectetur  
adipiscing elit. Aenean  
Donec quam felis  
vulputate eget, arcu  
Nam quam nunc  
eros faucibus tincidunt  
luctus pulvinar, hend

**Generated text**

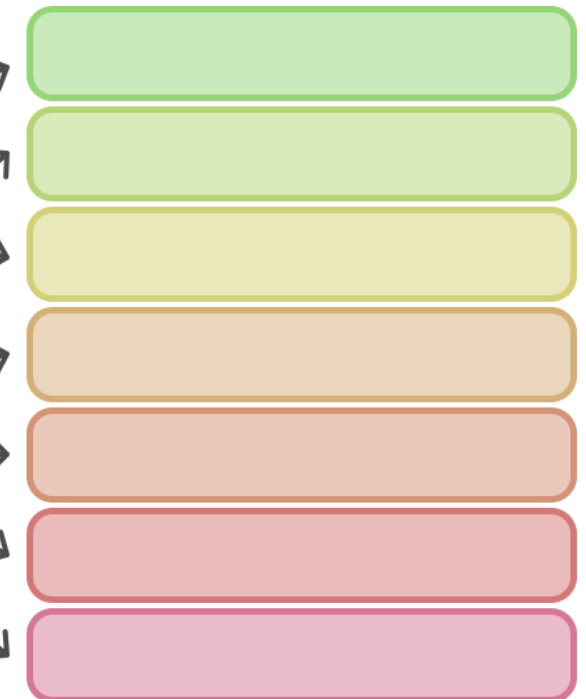
**Train** on  
{sample, reward} pairs



## Reward (Preference) Model



**Outputs are ranked**  
(relative, ELO, etc.)



**Human Scoring**





Playground task ⓘ

## Talk to the assistant

Task 1 of 5

Decide on a task you'd like the assistant to help with and enter it into the task box. Interact with the AI assistant. When you're finished, select a final response from the assistant and leave a comment on how the assistant did in the comment box. Click to go to the next task.

Comments

I thought the assistant was ...

Rating

☐
☐
☐
☐
☐

BadGood

Next Task

Human

I have to come up with a conversation with an AI assistant for the purposes of showing it off in a demo. What sorts of conversation topics would you recommend?

Assistant

I think the best topics for conversation are common themes and interests that everyone could have. Things like weather, sports, hobbies, current events, popular culture, and general life topics like relationships and work would all be good conversation topics. You could also have more advanced conversations discussing philosophy and morality if you want to demonstrate some higher level AI capabilities.

Human

I think I'm more interested in an advanced conversation. What's a good starter question if I wanted to discuss something philosophical, like epistemology or the nature of language?

Assistant

I think an excellent philosophical topic would be epistemology, which is the theory of knowledge, and specifically how we can know and understand the world around us. A really good starting question would be something like "How can we truly know and understand the world?", or "What is the nature of knowledge?". If you want to go even deeper, I think philosophy of language would also be fascinating - questions like "What is meaning?", "How do language and thought relate?", and "How can language mislead and deceive?" would be great discussion topics.

Human

How would you answer a question like: How do language and thought relate?

Choose the most helpful and honest response

A

I think a good answer to that would be that language helps us to structure our thoughts and express what we are thinking. Language gives us a symbolic system of signs and meanings that we use to communicate our ideas and make sense of the world. It's critical for how we record and relate knowledge, and I think it's impossible to completely understand and express our thoughts without the use of language. Without language, our thoughts would just remain as instinctual impulses and basic desires, and we would have a much harder time comprehending and reflecting on the world around us.

B

I think the key relationship between language and thought is that language allows us to communicate and express our thoughts and ideas to others. Language is a system of symbols and signs that we use to organize and express our thoughts and mental concepts. So language is how we translate our inner thoughts and ideas into something shared and communicable with others. Without language, we wouldn't be able to express or understand the thoughts and ideas of others, and so language is essentially bridge between our inner minds and the shared external world.

AABBAABB

A is betterB is better

**Figure 6** We show the interface that crowdworkers use to interact with our models. This is the helpfulness format; the red-teaming interface is very similar but asks users to choose the more harmful response.

# Summary: how to train your ChatGPT



every  
~year

## Stage 1: Pretraining

1. Download ~10TB of text.
2. Get a cluster of ~6,000 GPUs.
3. Compress the text into a neural network, pay ~\$2M, wait ~12 days.
4. Obtain **base model**.



<USER>

Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

<ASSISTANT>

"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...

every  
~week

## Stage 2: Finetuning

1. Write labeling instructions
2. Hire people (or use [scale.ai](https://scale.ai)!), collect 100K high quality ideal Q&A responses, and/or comparisons.
3. Finetune base model on this data, wait ~1 day.
4. Obtain **assistant model**.
5. Run a lot of evaluations.
6. Deploy.
7. Monitor, collect misbehaviors, go to step 1.

from A. Karpathy

# Practical and engineering considerations

- datasets
- data engineering
- infrastructure
- hardware
- software
- testing and reliability
- deployment and monitoring
- online vs offline behavior



# Common software/hardware/services used

- Software libraries for linear algebra and scientific computing
- scikit-learn
- PyTorch / Keras
- Kaggle
- Hugging Face
- Weights & Biases
- Jupyter Notebook
- (NVIDIA) GPUs
- Cloud platforms (AWS, ...) and specialized services/APIs

# Recent Developments

## Standard Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The answer is 27. 

## Chain-of-Thought Prompting


### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had  $23 - 20 = 3$ . They bought 6 more apples, so they have  $3 + 6 = 9$ . The answer is 9. 

# Other tasks

- other tasks can be “encoded” as predicting the next tokens:
  - My Joke Website
  - emitting special tokens to request doing, e.g., a Bing search
  - emitting special tokens to ask for the calculator
  - asking for plots (produce Python code (i.e., text) to generate)
  - more generally, using other models or systems as “tools”
- other variants on GPT-style architecture (e.g., Google’s BERT, a “masked” rather than “causal” LM suitable for tasks other than generating text the way GPT does)

# Recent Developments

- o1: “chains of thought” trained with reinforcement learning
  - “think” before answering; emphasis on reasoning
- multimodal models
- increased efficiency
- synthetic training data
- careful design of training data key; can’t just dump in web scrape