https://introml.mit.edu/

6.390 Intro to Machine Learning

Lecture 12: Unsupervised Learning

Shen Shen

May 3, 2024

(many slides adapted from Phillip Isola and Tamara Broderick)

Logistics

- This is the last regular lecture; next week is the last regular week.
- Friday, May 10
  - Lecture time, discuss future topics (generativeAI).
  - By the end of the day, all assignments will be due.
- Tuesday, May 14
  - 20-day extensions applicable through this day.
  - Last regular OHs (for checkoffs/hw etc); afterwards only Instructor OHs.
  - 6-8pm, 10-250, final exam review.
- The end-of-term subject evaluations are open. We'd love to hear your thoughts on 6.390: this provides valuable feedback for us and other students, for future semesters!
- Check out final exam logistics on introML homepage.

Outline

- $\boxed{\text{Recap: Supervised learning and reinforcement learning}}$
- Unsupervised learning
- Clustering: k-means algorithm
  - Clustering vs. classification
  - Initialization matters
  - k matters
- Auto-encoder
  - Compact representation
- Unsupervised learning again -- representation learning and beyond

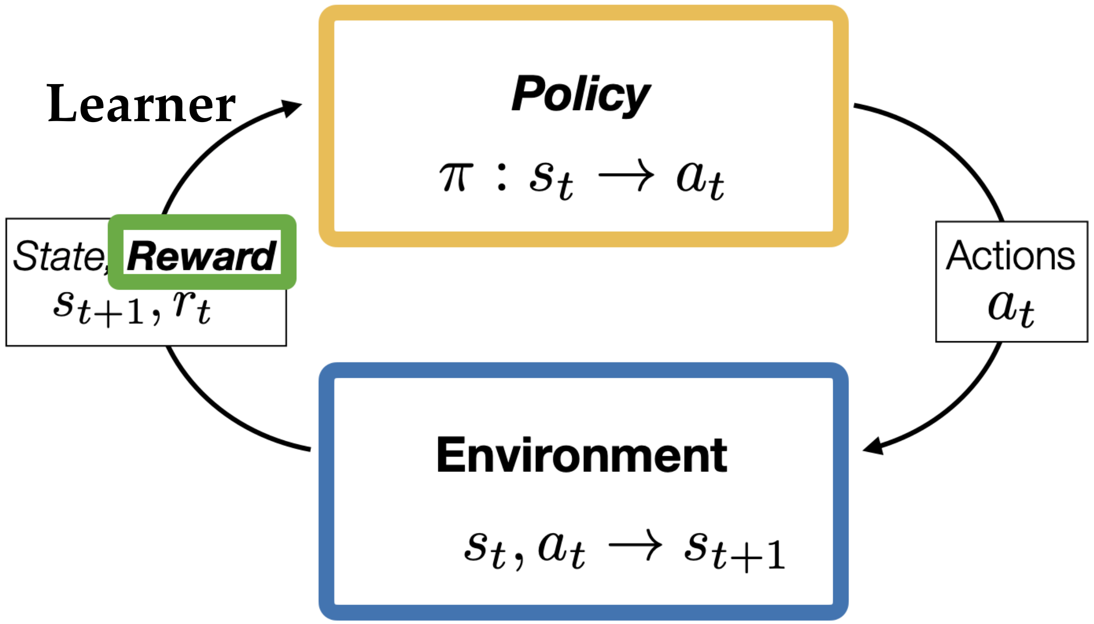
Supervised learning

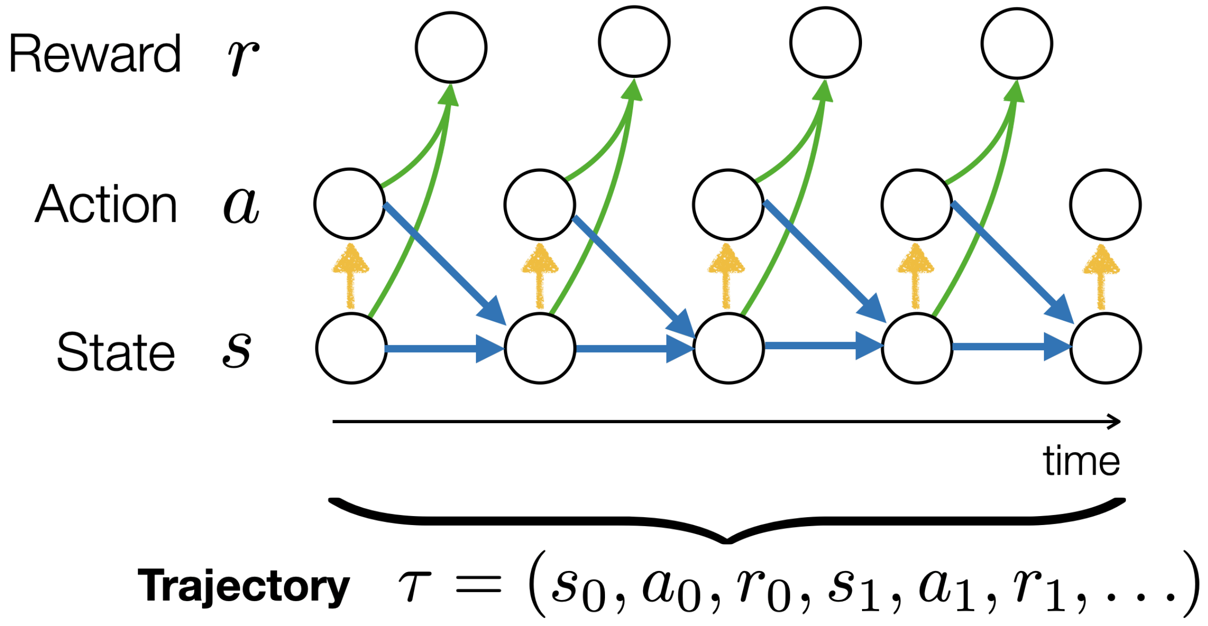
Training data

$$\{x^{(1)}, y^{(1)}\}$$
$$\{x^{(2)}, y^{(2)}\} \quad \longrightarrow \quad \boxed{\text{Learner}} \quad \longrightarrow \quad f : x \to y$$
$$\{x^{(3)}, y^{(3)}\}$$
$$\dots$$

- **explicit** supervision via labels y.
- both regression or classification are trying to predict accurate, exact labels.

Reinforcement learning





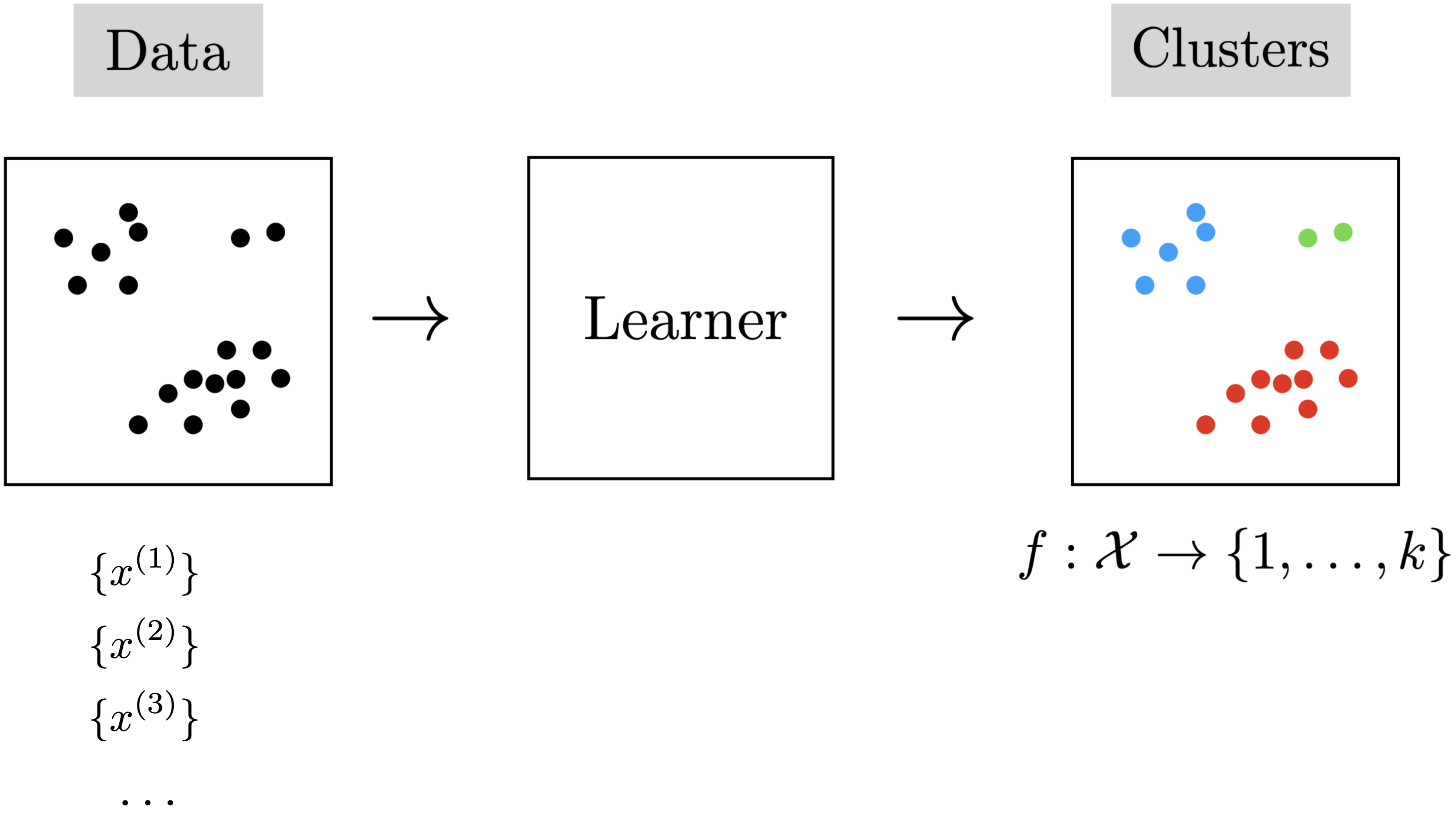
- **implicit(evaluative), sequential, iterative** supervision via rewards.

Outline

- Recap: Supervised learning and reinforcement learning
- $\boxed{\text{Unsupervised learning}}$
- Clustering: k-means algorithm
  - Clustering vs. classification
  - Initialization matters
  - k matters
- Auto-encoder
  - Compact representation
- Unsupervised learning again -- representation learning and beyond

Unsupervised learning

- **No** supervision, i.e., no labels nor rewards.
- try to learn something "interesting" using **only** the features
  - Clustering: learn "similarity" of the
  - Autoencoder: learn compression/reconstruction/representation
- useful paradigm on its own; often empowers downstream tasks.


Data
Clusters
Learner
$\{x^{(1)}\}$
$\{x^{(2)}\}$
$\{x^{(3)}\}$
$\dots$
$f : \mathcal{X} \to \{1, \dots, k\}$

Data

Autoencoder

$\{x^{(1)}\}$

$\{x^{(2)}\}$ $\rightarrow$ $\boxed{\text{Learner}}$ $\rightarrow$ Representations

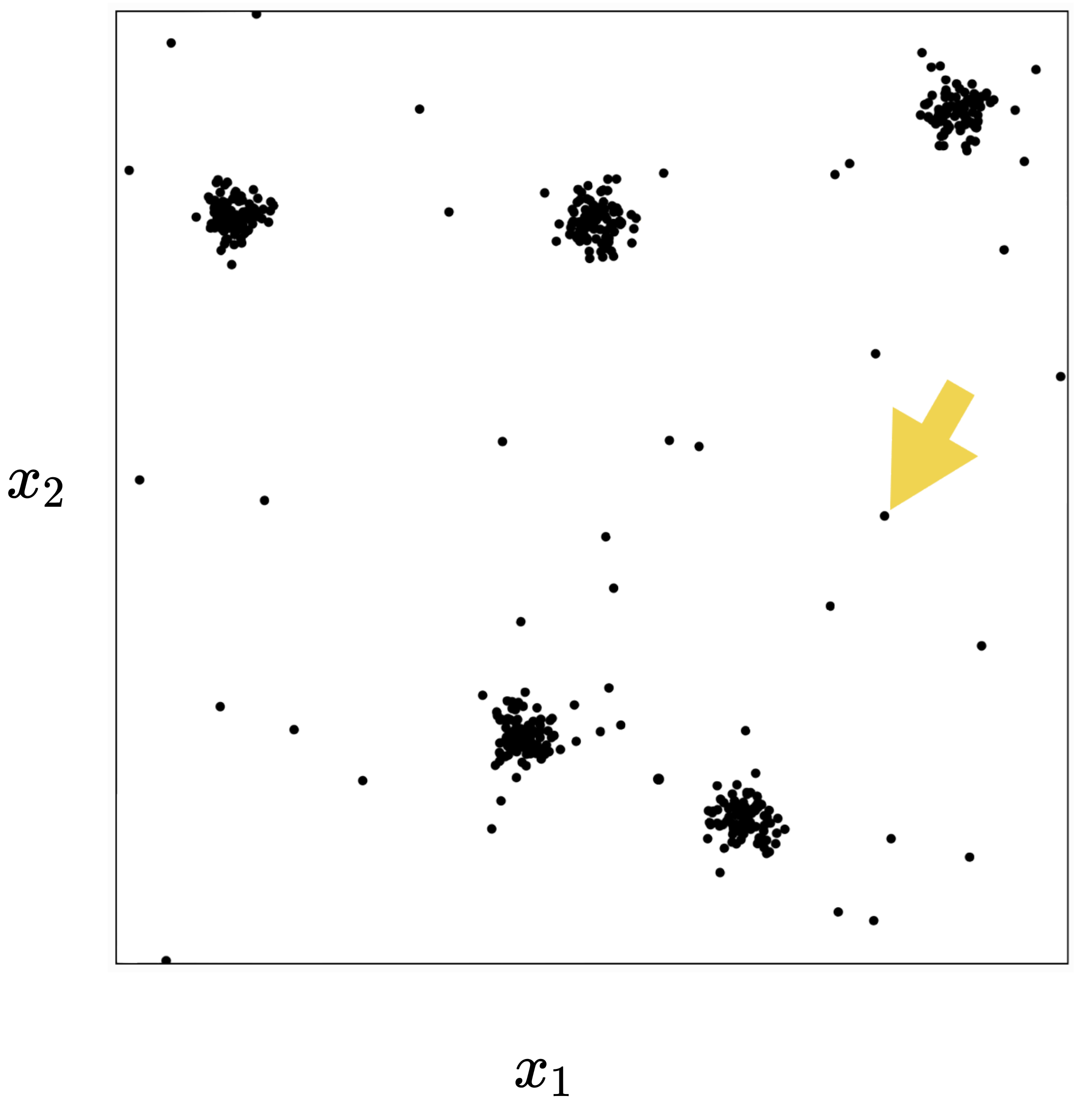
$\{x^{(3)}\}$

$\ldots$

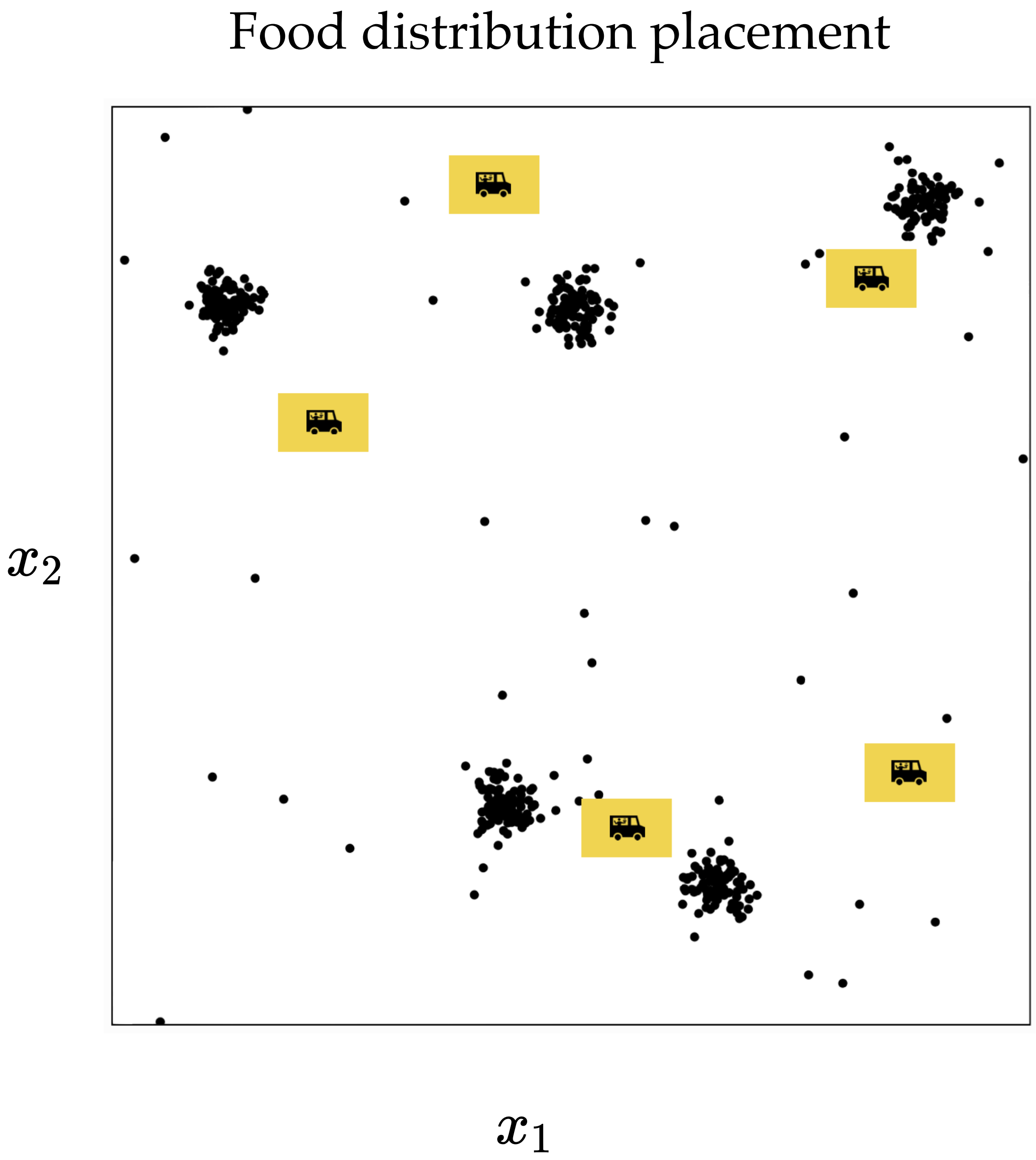
Outline

- Recap: Supervised learning and reinforcement learning
- Unsupervised learning
- Clustering: k-means algorithm
  - Clustering vs. classification
  - Initialization matters
  - k matters
- Auto-encoder
  - Compact representation
- Unsupervised learning again -- representation learning and beyond

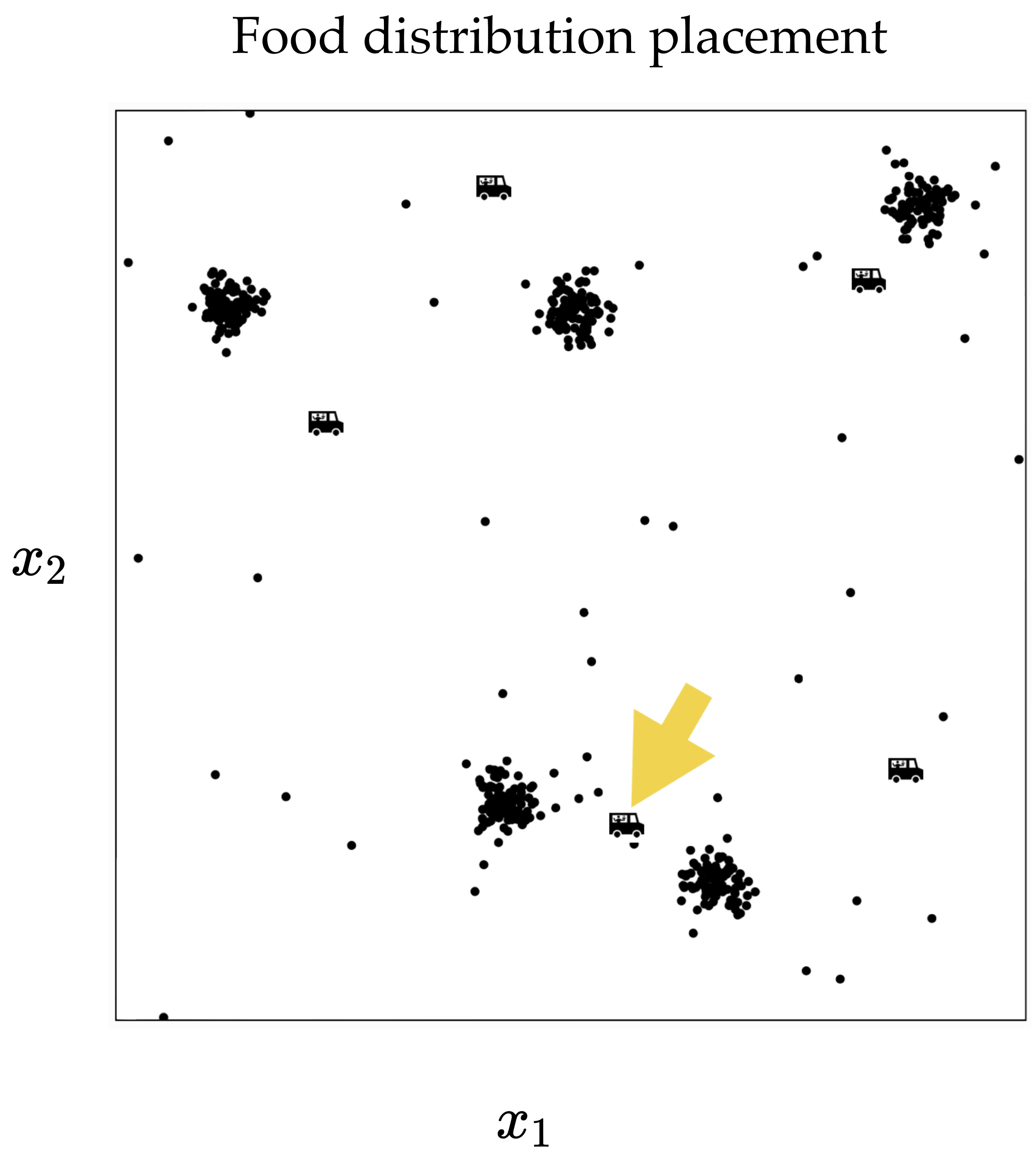
Food distribution placement



- x_1: longitude, x_2: latitude
- Person i location $x^{(i)}$

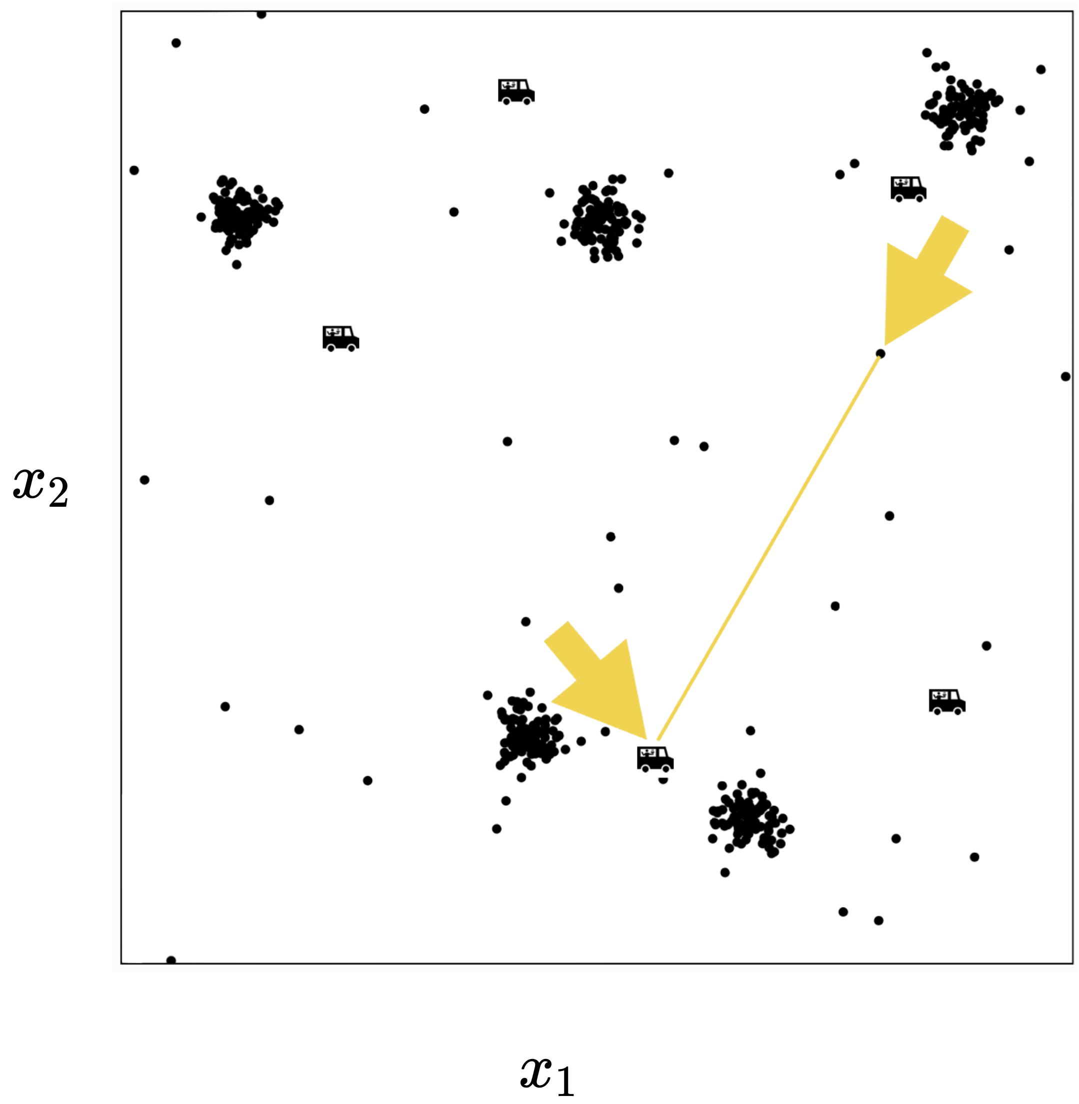


- x_1: longitude, x_2: latitude
- Person i location $x^{(i)}$
- Q: where should I have my k food trucks park?



- x_1: longitude, x_2: latitude
- Person i location $x^{(i)}$
- Q: where should I have my k food trucks park?
- Food truck j location $\mu^{(j)}$
- Want to minimize the "loss" of people we serve

Food distribution placement



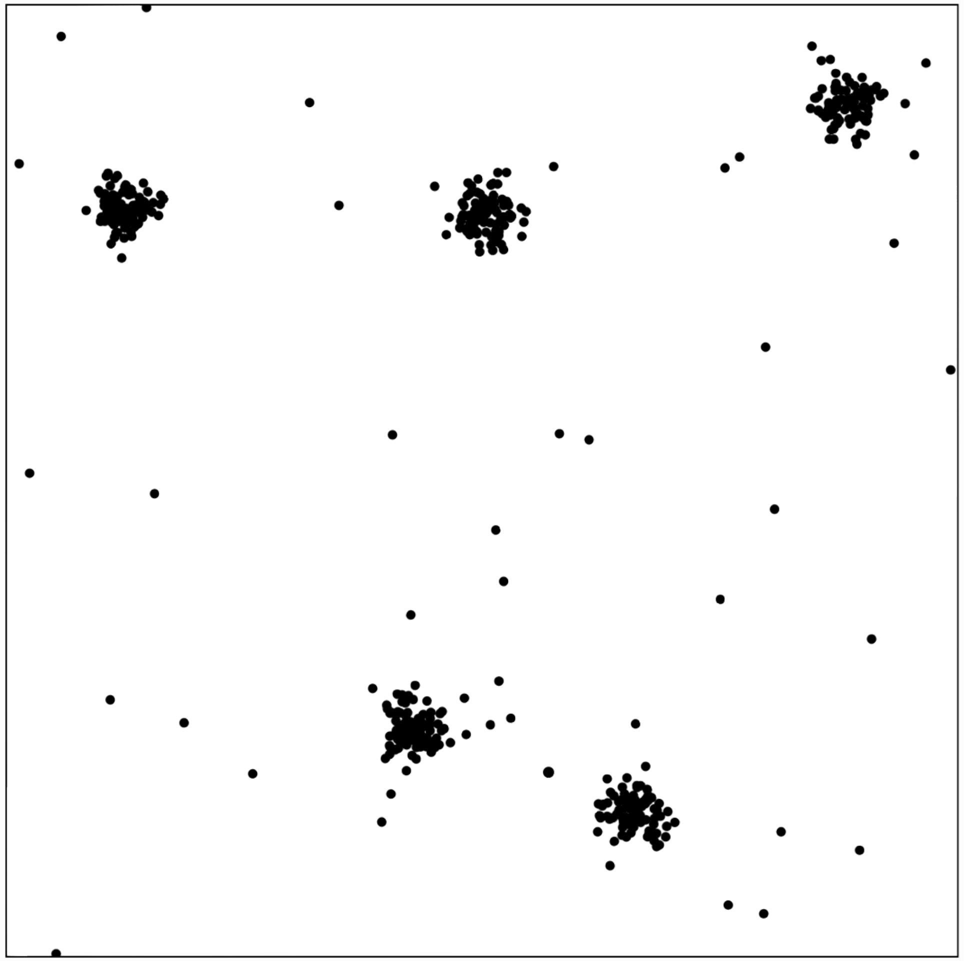
- x_1: longitude, x_2: latitude
- Person i location $x^{(i)}$
- Q: where should I have my k food trucks park?
- Food truck j location $\mu^{(j)}$

  Want to minimize the "loss" of people we serve
- Loss if i walks to truck j : $\left\|x^{(i)} - \mu^{(j)}\right\|_2^2$
- Index of the truck where person i is **chosen** to walk to: $y^{(i)}$

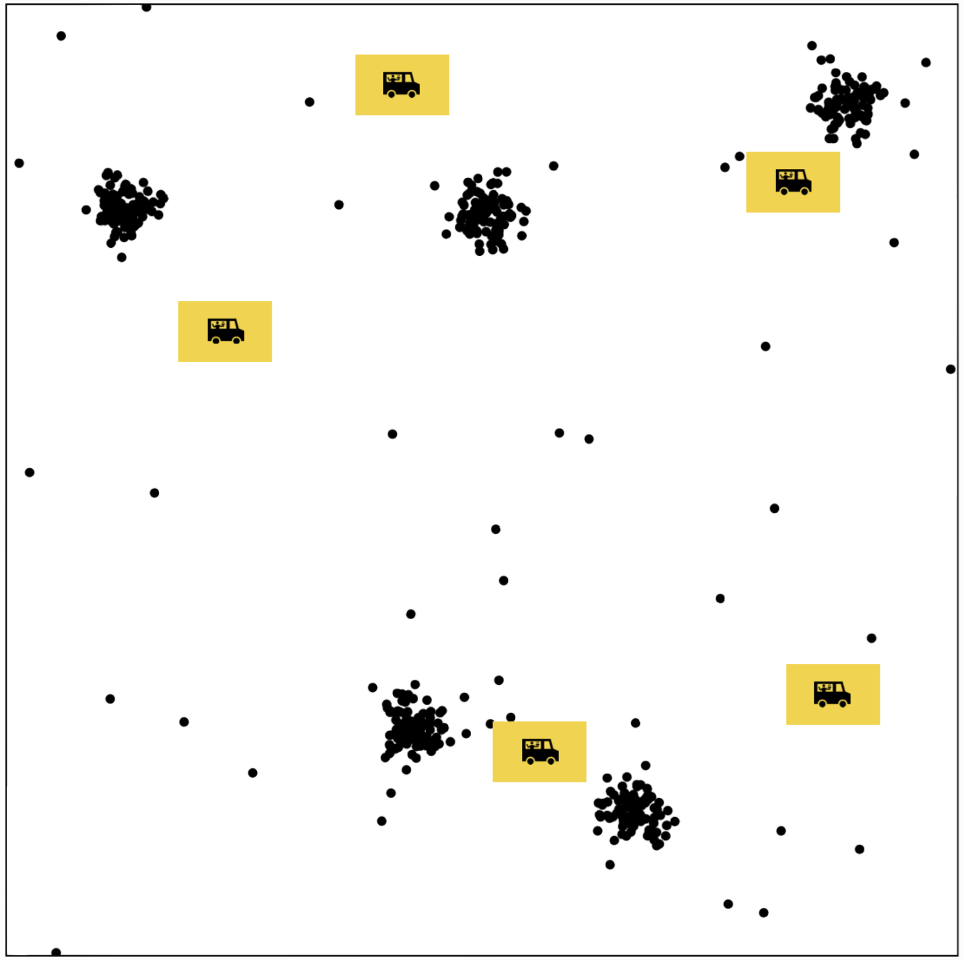
Loss over all people

$$\sum_{i=1}^{n} \sum_{j=1}^{k} \mathbf{1}\left\{y^{(i)} = j\right\} \left\|x^{(i)} - \mu^{(j)}\right\|_2^2$$

k-means objective

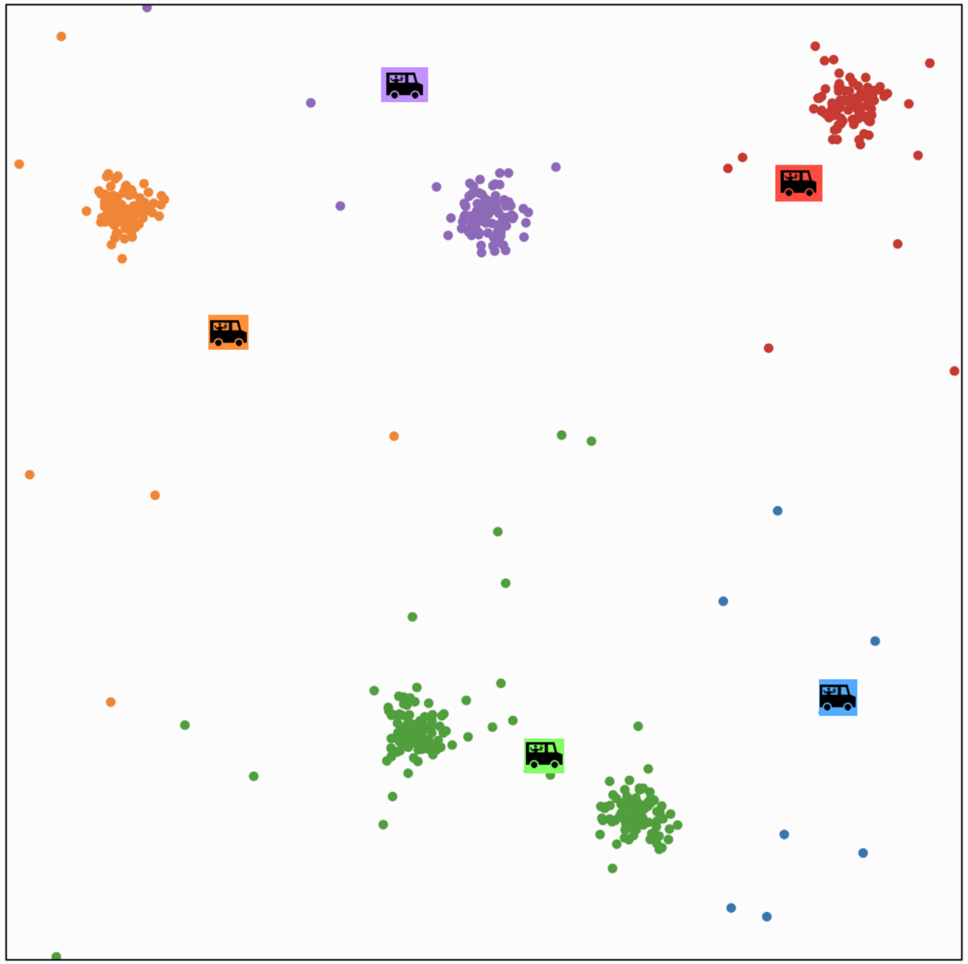


$$\text{K-MEANS}(k, \tau, \{x^{(i)}\}_{i=1}^{n})$$



$\text{K-MEANS}(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

1 μ random initialization



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

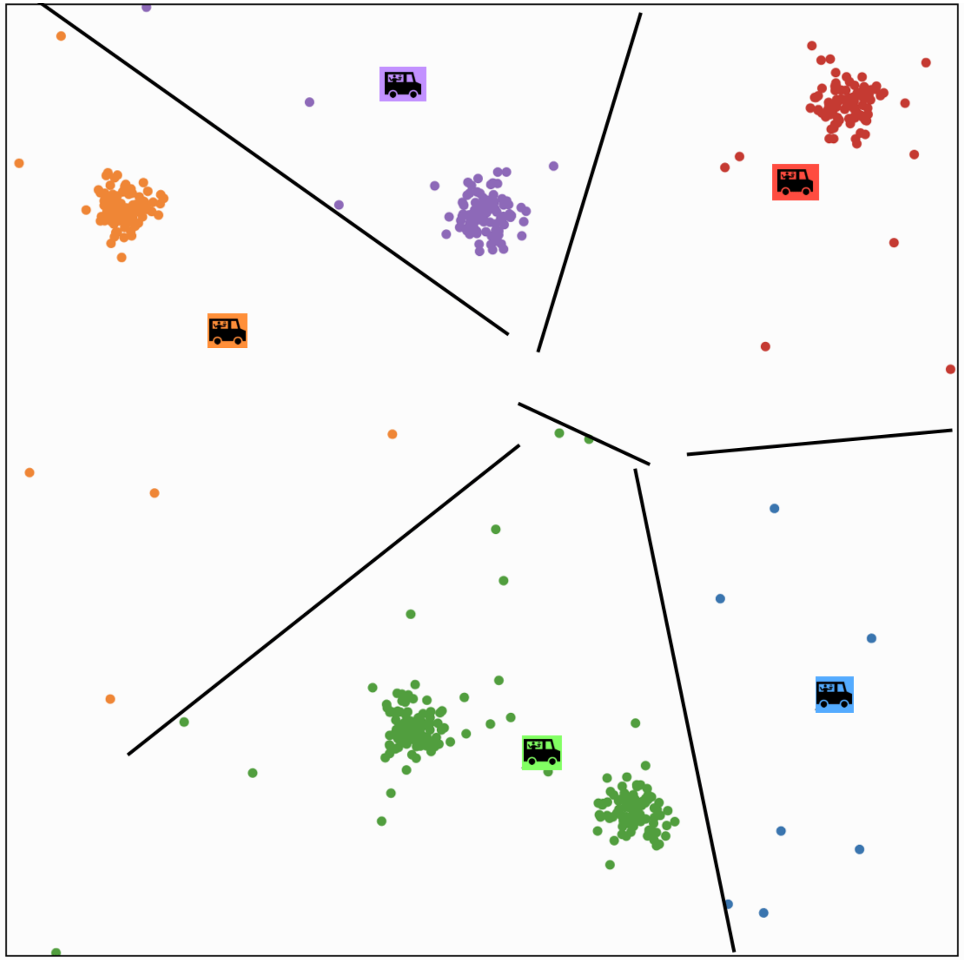
μ random initialization

for $t = 1$ to τ

for $i = 1$ to n

$$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$$

each person i gets assigned to a food truck j, color-coded.



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

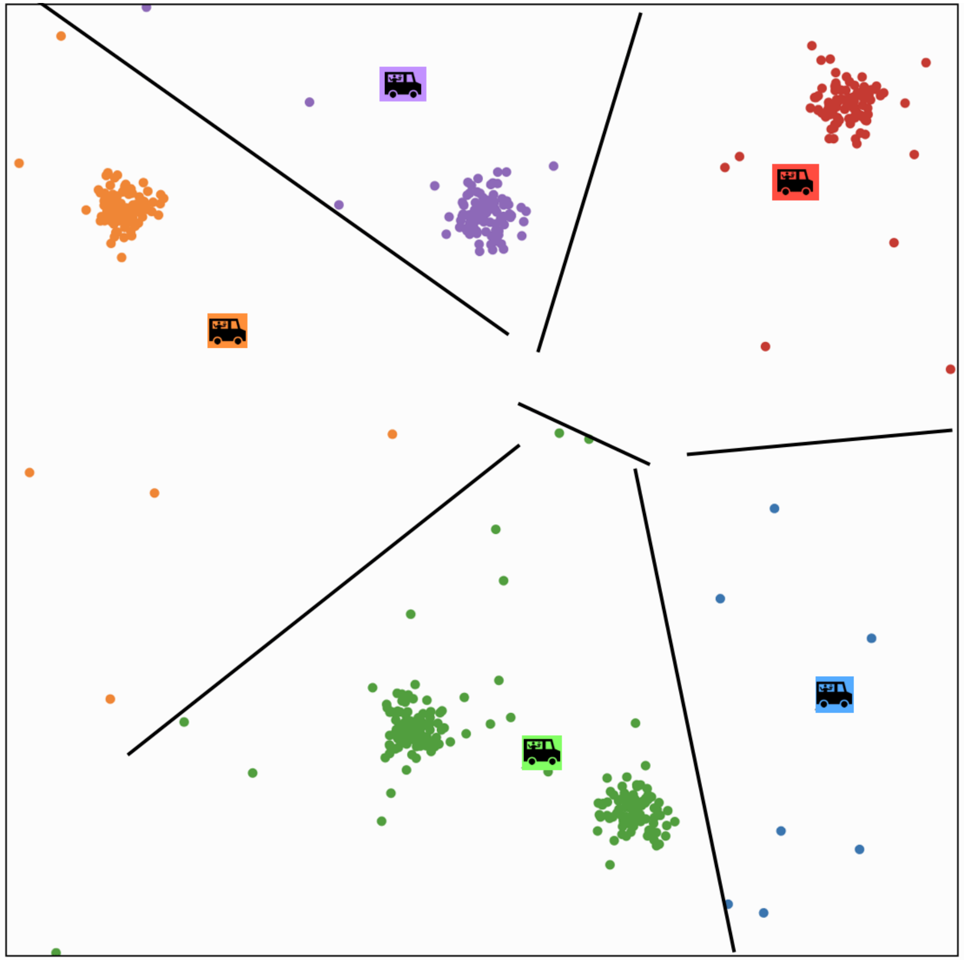
μ random initialization

for $t = 1$ to τ

for $i = 1$ to n

$$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$$

each person i gets assigned to a food truck j, color-coded.



$\text{K-MEANS}(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

$\mu =$ random initialization

for $t = 1$ to τ

for $i = 1$ to n

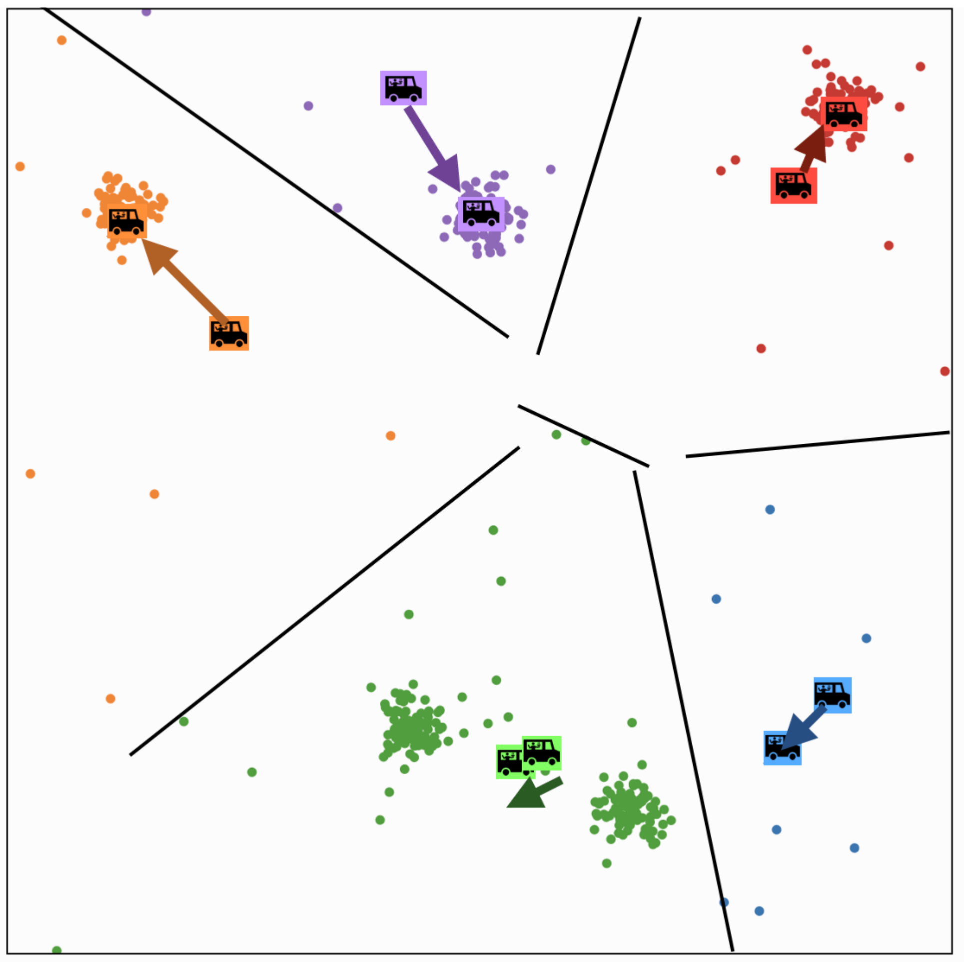
$$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$$

for $j = 1$ to k

$$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \text{j}\right) x^{(i)}$$

food truck j gets moved to the "central" location of all ppl assigned to it

$$N_j = \sum_{i=1}^{n} \mathbf{1}\left\{y^{(i)} = j\right\}$$



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

μ = random initialization

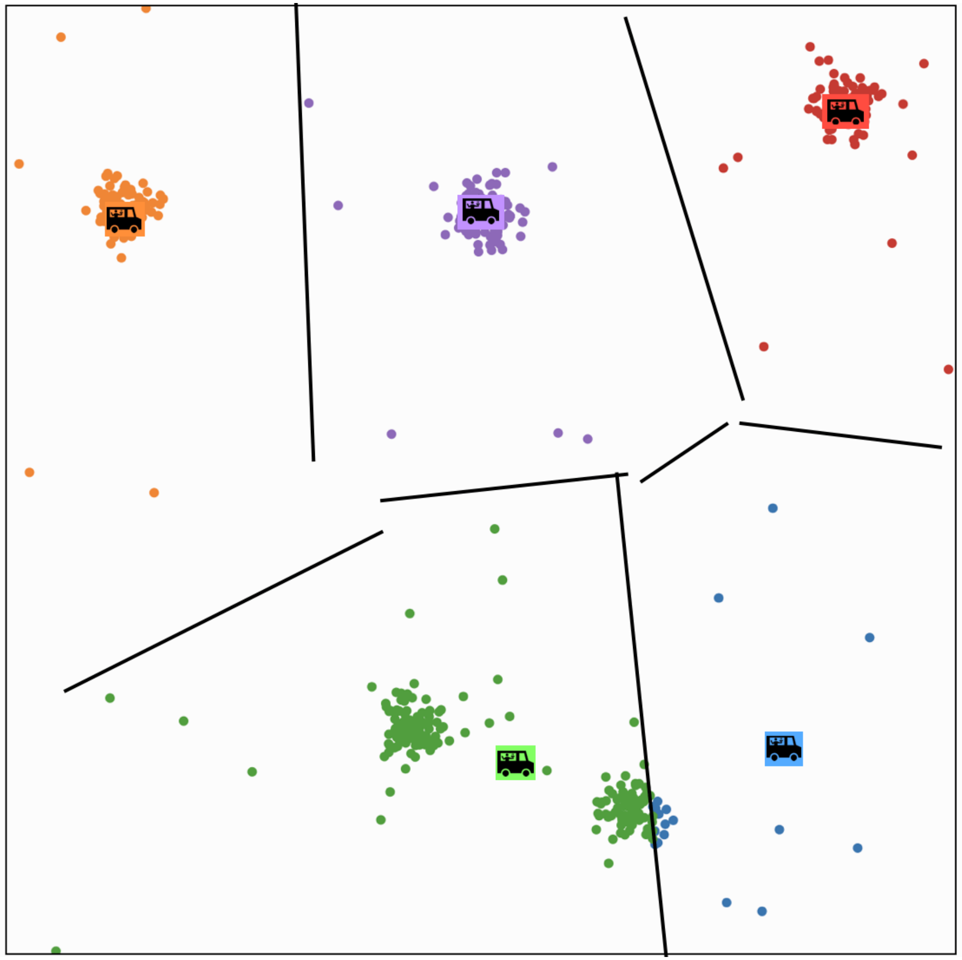
for $t = 1$ to τ

for $i = 1$ to n

$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$

for $j = 1$ to k

$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = j\right) x^{(i)}$



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

$\mu =$ random initialization

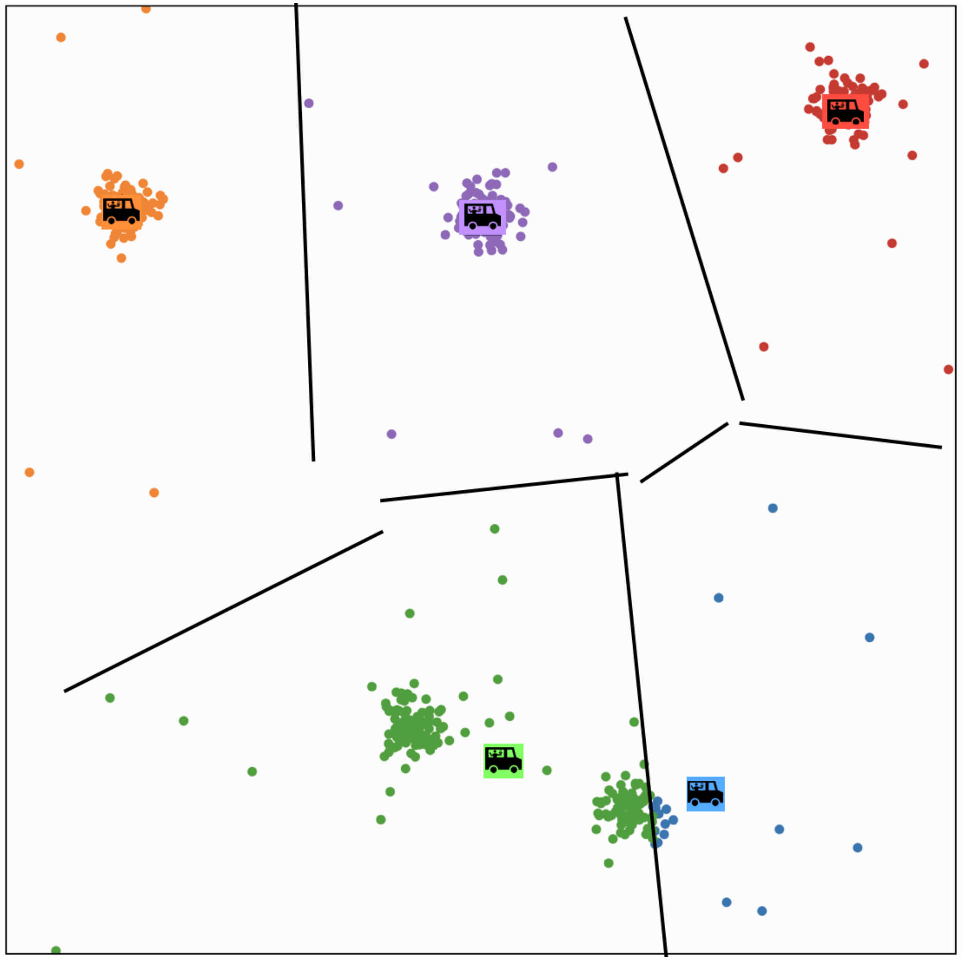
for $t = 1$ to τ

for $i = 1$ to n

$$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$$

for $j = 1$ to k

$$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \mathrm{j}\right) x^{(i)}$$



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

$\mu =$ random initialization

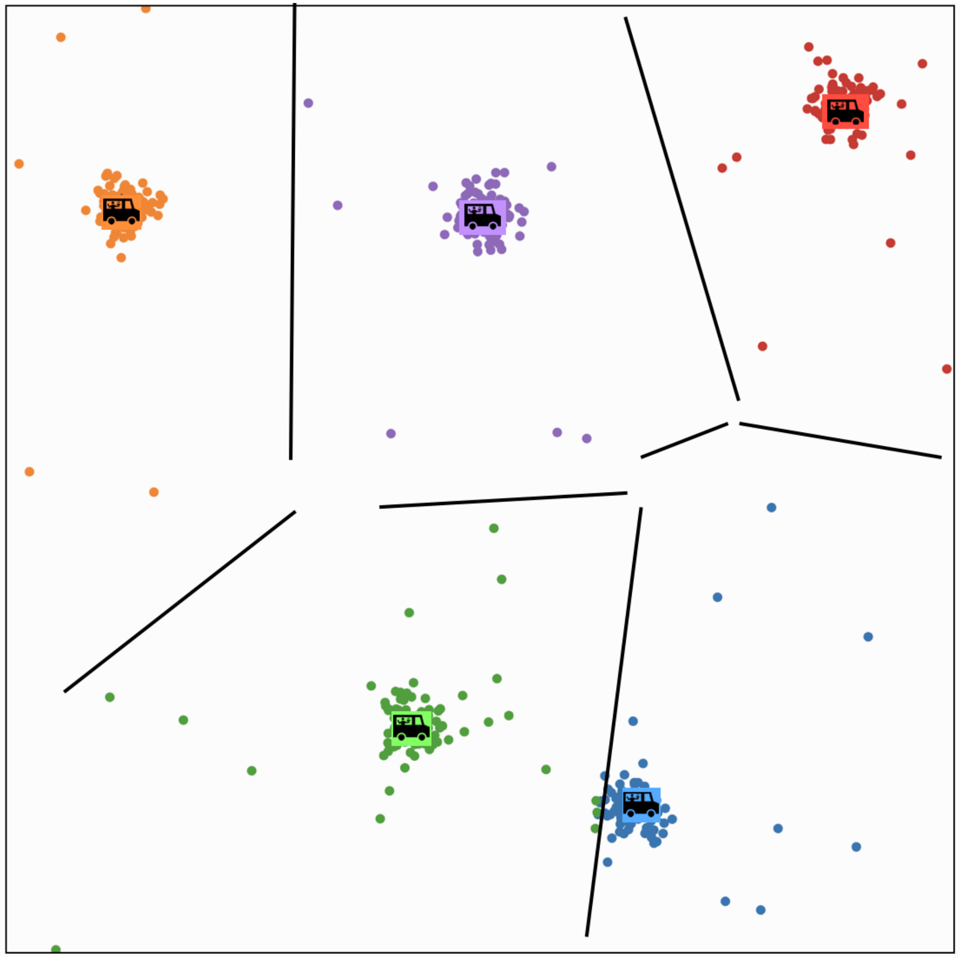
for $t = 1$ to τ

for $i = 1$ to n

$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$

for $j = 1$ to k

$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \mathrm{j}\right) x^{(i)}$



$$\text{K-MEANS}(k, \tau, \{x^{(i)}\}_{i=1}^{n})$$

$$\mu = \text{random initialization}$$

$$\textbf{for } t = 1 \text{ to } \tau$$

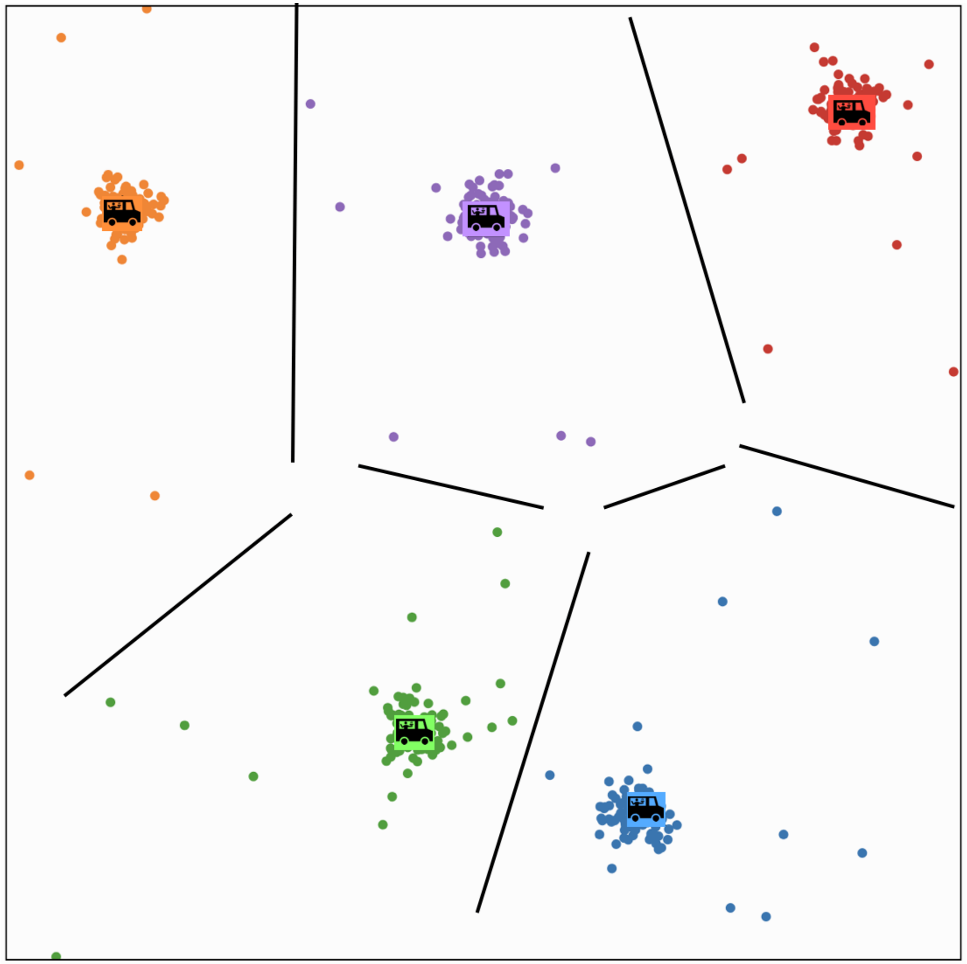
$$\textbf{for } i = 1 \text{ to } n$$

$$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$$

$$\textbf{for } j = 1 \text{ to } k$$

$$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \text{j}\right) x^{(i)}$$

continue (ppl assignment then truck movement) update at some point.



$\text{K-MEANS}(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

μ = random initialization

for $t = 1$ to τ

for $i = 1$ to n

$y^{(i)} = \arg\min_j \left\| x^{(i)} - \mu^{(j)} \right\|^2$

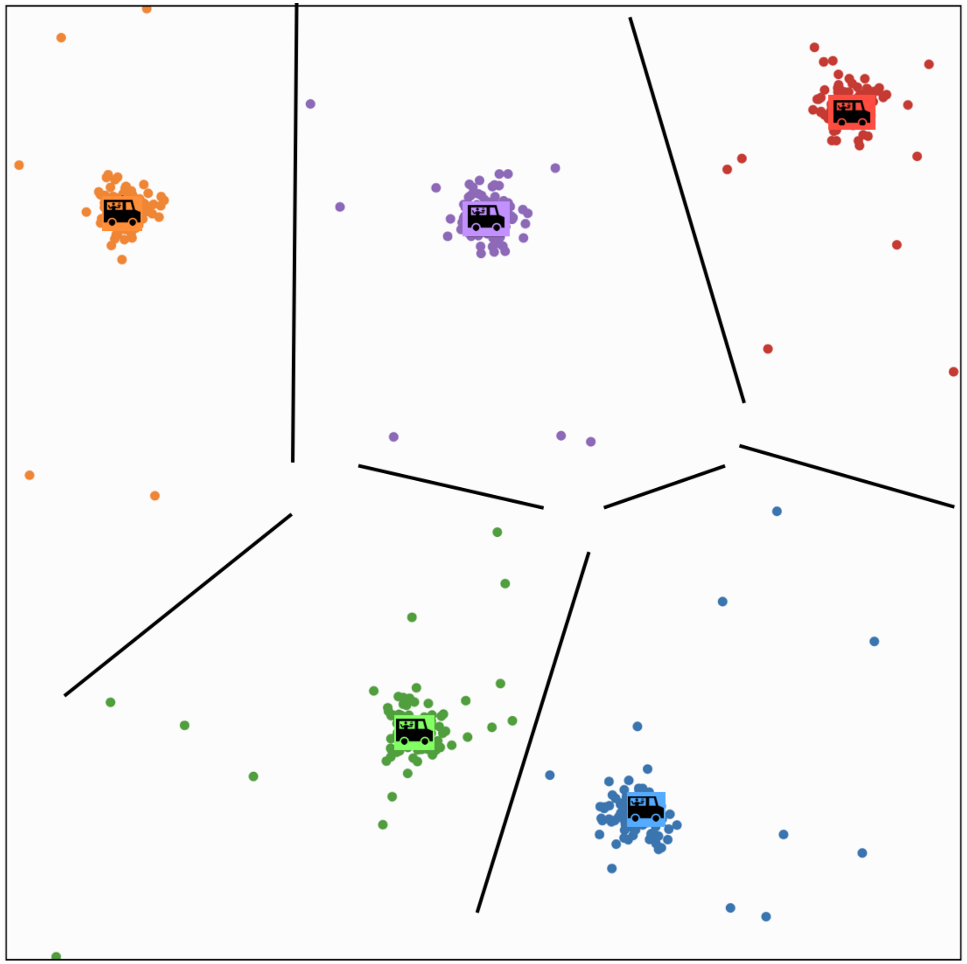
for $j = 1$ to k

$\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \mathrm{j}\right) x^{(i)}$

if $y == y_{\text{old}}$

break

(ppl assignment and truck location) will stop changing



K-MEANS$(k, \tau, \{x^{(i)}\}_{i=1}^{n})$

μ, y = random initialization

for $t = 1$ to τ

  $y_{old} = y$

  for $i = 1$ to n

    $y^{(i)} = \arg\min_j \left\|x^{(i)} - \mu^{(j)}\right\|^2$

  for $j = 1$ to k

    $\mu^{(j)} = \frac{1}{N_j} \sum_{i=1}^{n} 1\left(y^{(i)} = \mathrm{j}\right) x^{(i)}$

  if $y == y_{\text{old}}$

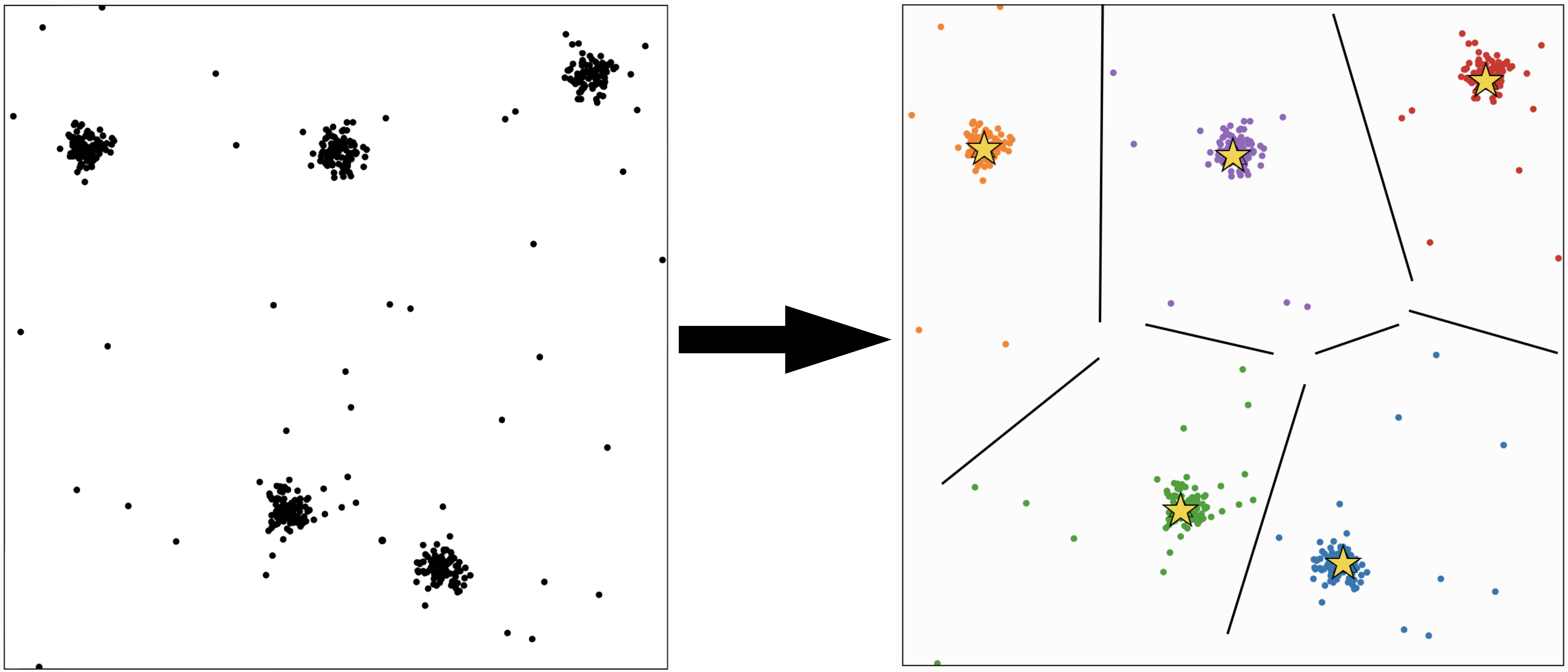
    break

return μ, y

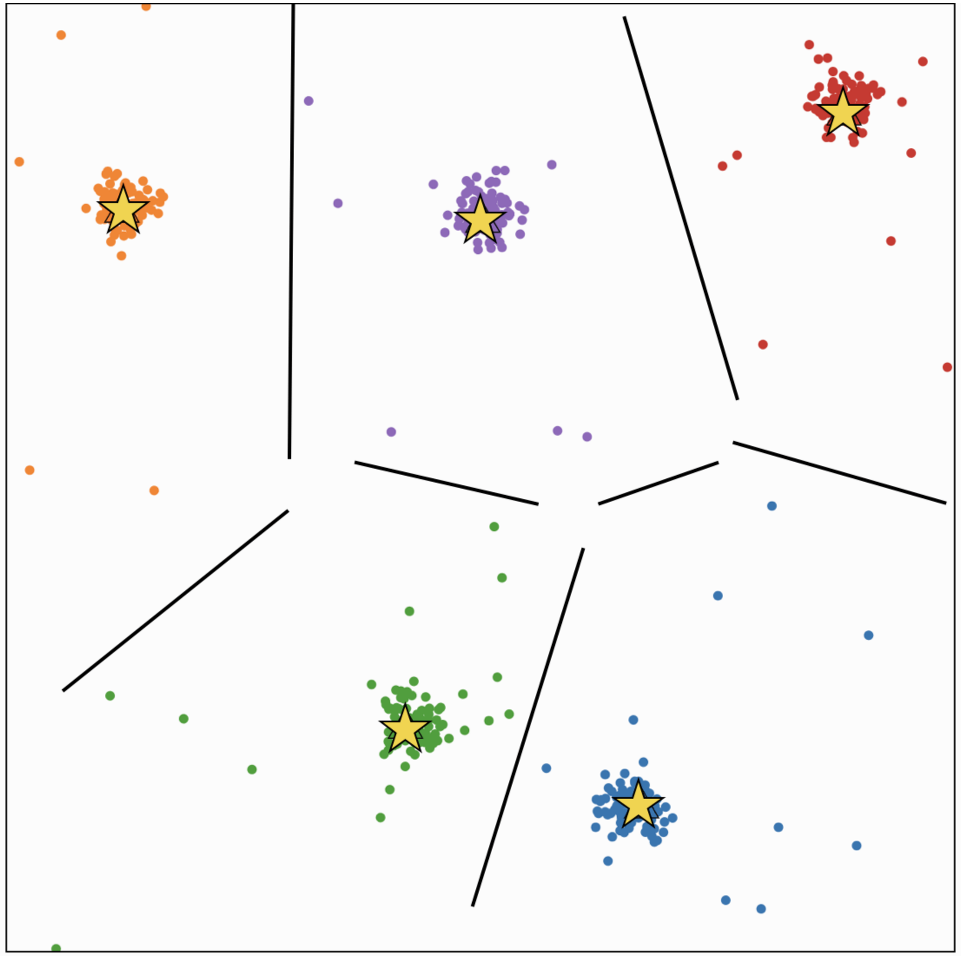
Outline

- Recap: Supervised learning and reinforcement learning
- Unsupervised learning
- Clustering: k-means algorithm
  - Clustering vs. classification
  - Initialization matters
  - k matters
- Auto-encoder
  - Compact representation
- Unsupervised learning again -- representation learning and beyond

k-means

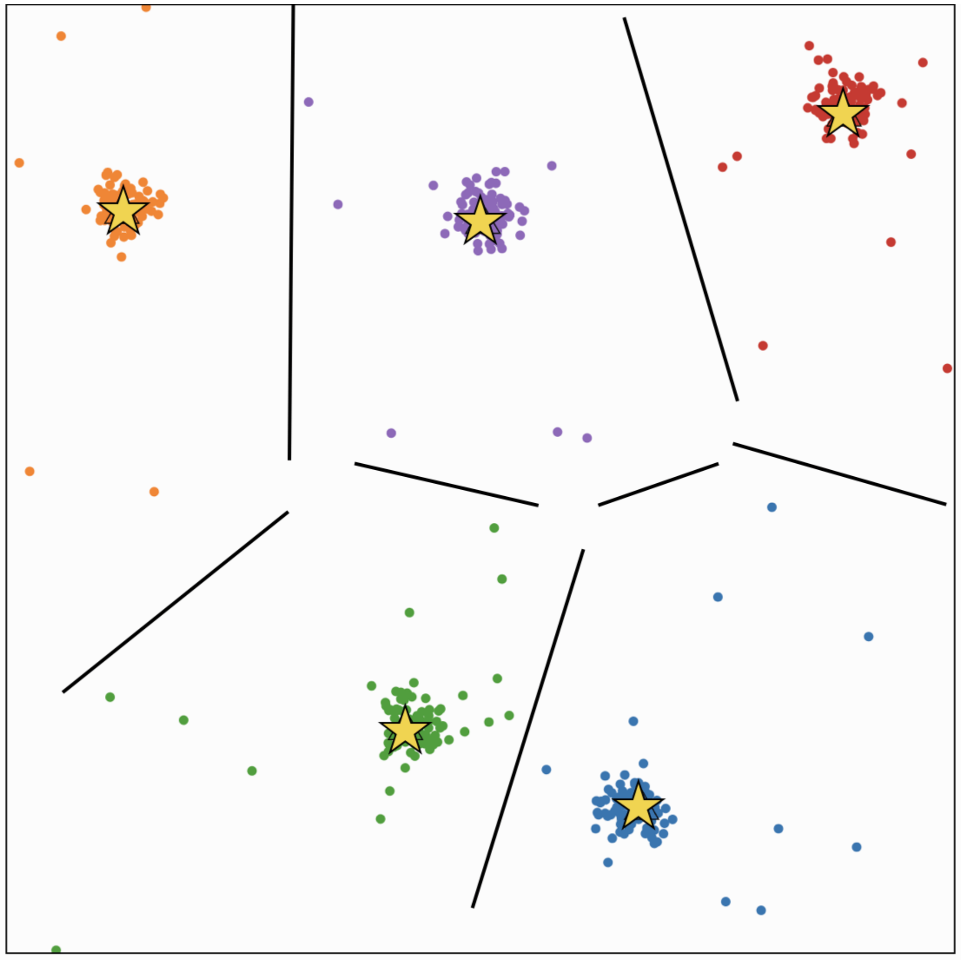


Compare to classification



- Did we just do k-class classification?
- Looks like we assigned label $y^{(i)}$, which takes k different values, to each feature vector $x^{(i)}$
- But we didn't use any **labeled** data
- The "labels" here don't have meaning; I could permute them and have the same result
- Output is really a partition of the data

Compare to classification

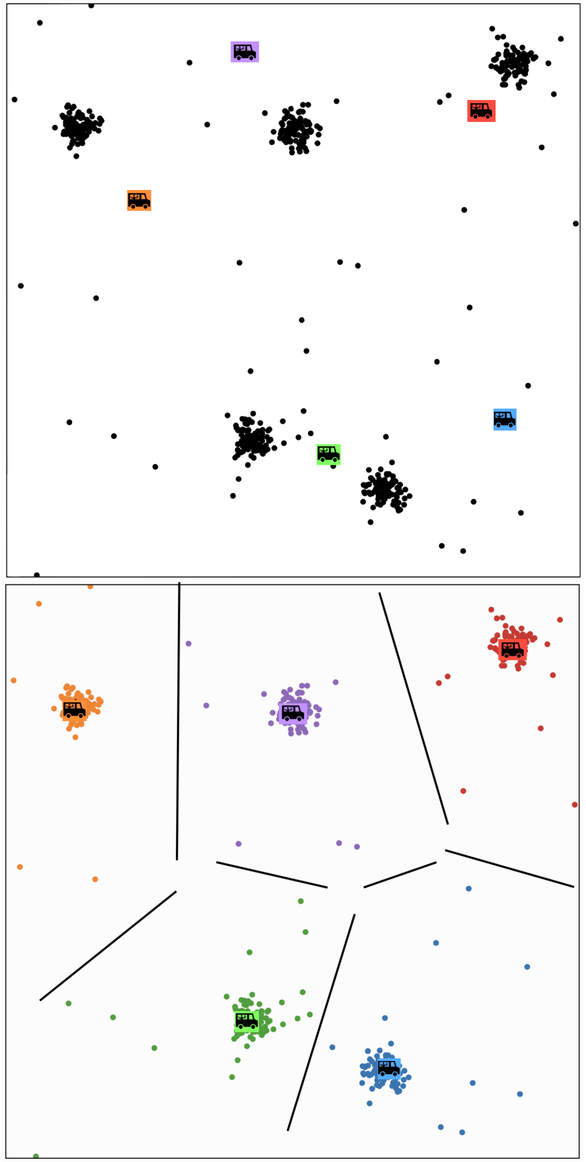


- So what did we do?
- We clustered the data: we grouped the data by similarity
- Why not just plot the data? We should! Whenever we can!
- But also: Precision, big data, high dimensions, high volume
- An example of unsupervised learning: no labeled data, and we're finding patterns

Outline

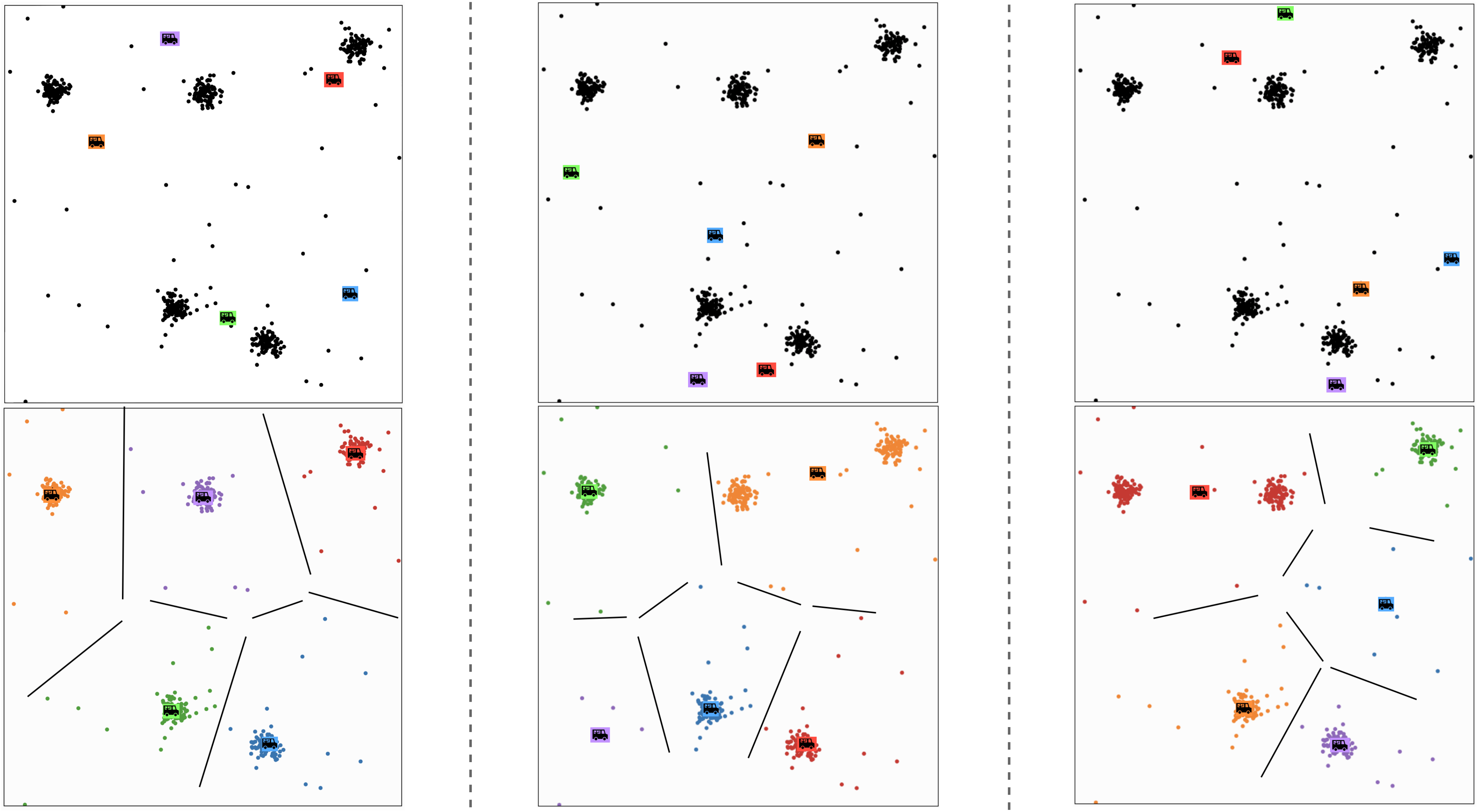
- Recap: Supervised learning and reinforcement learning
- Unsupervised learning
- Clustering: k-means algorithm
  - Clustering vs. classification
  - Initialization matters
  - k matters
- Auto-encoder
  - Compact representation
- Unsupervised learning again -- representation learning and beyond

Effect of initialization

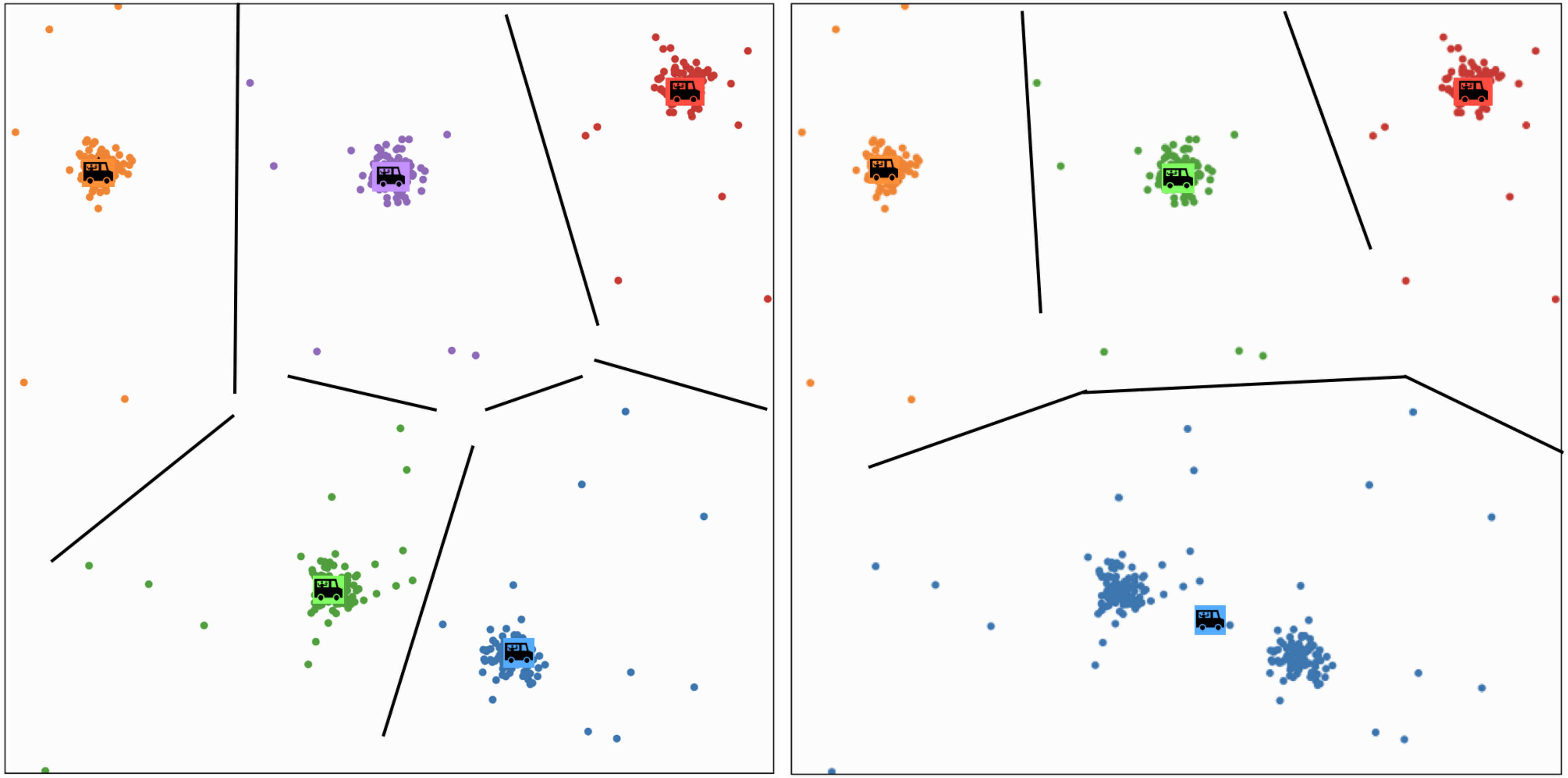


- A theorem say if run for enough outer iterations (line 2), the k-means algorithm will converge to a local minimum of the k-means objective
- That local minimum could be bad!
- The initialization can make a big difference.
- Some options: random restarts.

Effect of initialization



Effect of k



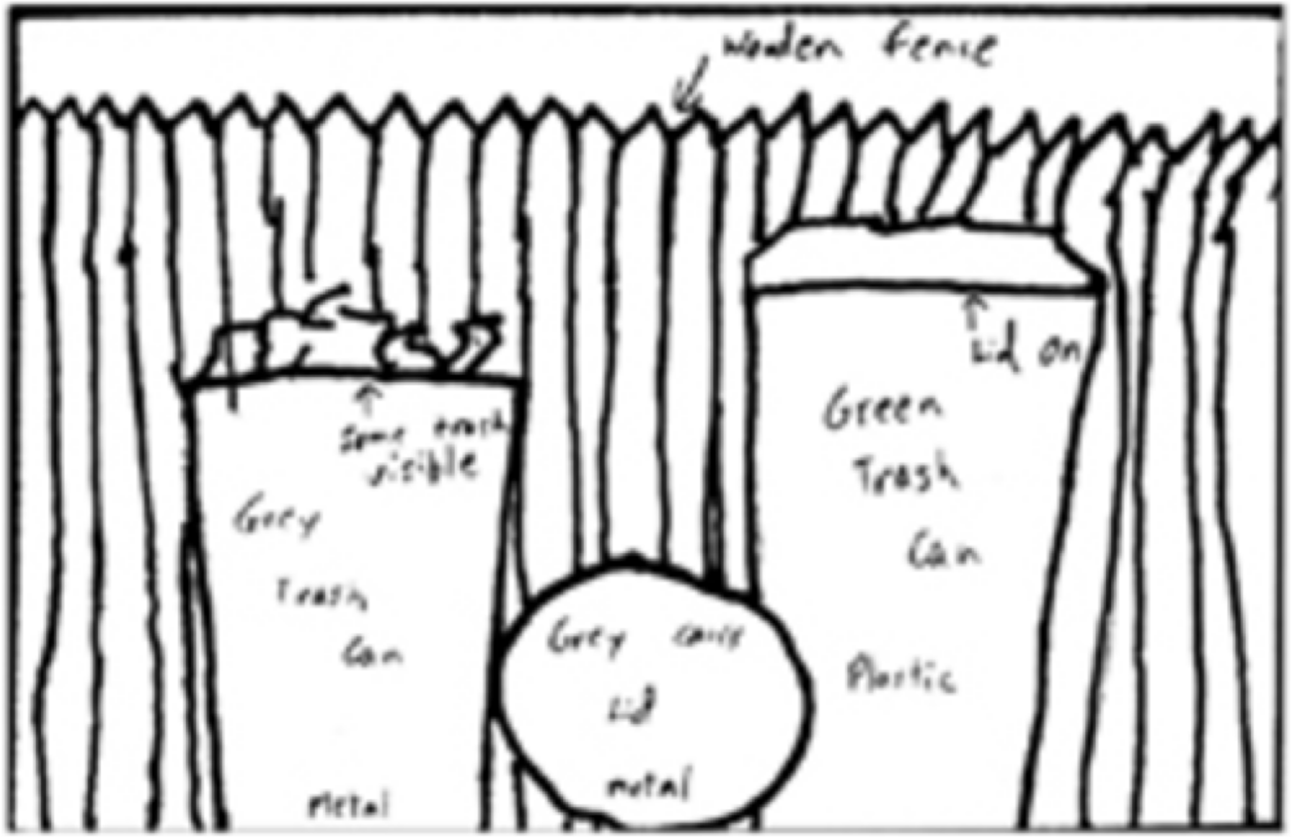
Outline

- Recap: Supervised learning and reinforcement learning
- Unsupervised learning
- Clustering: k-means algorithm
    - Clustering vs. classification
    - Initialization matters
    - k matters
- Auto-encoder
    - Compact representation
- Unsupervised learning again -- representation learning and beyond

Observed image

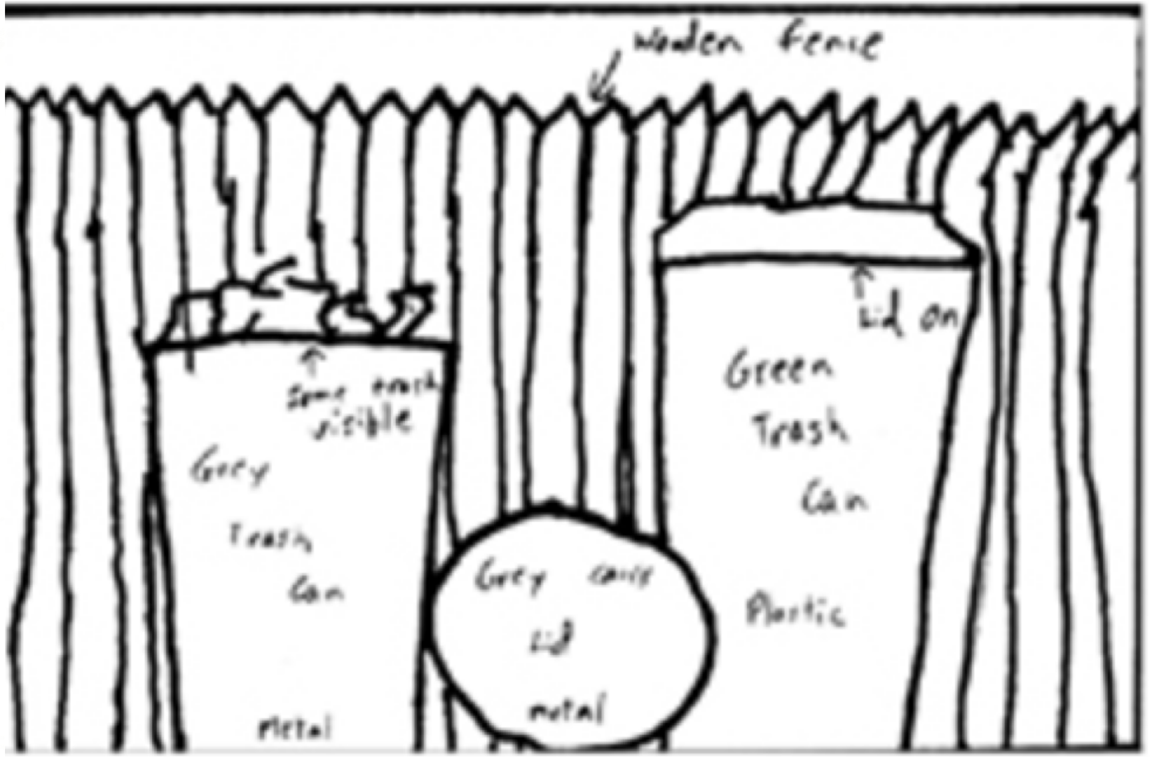
[Bartlett, 1932]
[Intraub & Richardson, 1989]

Drawn from memory



Observed image

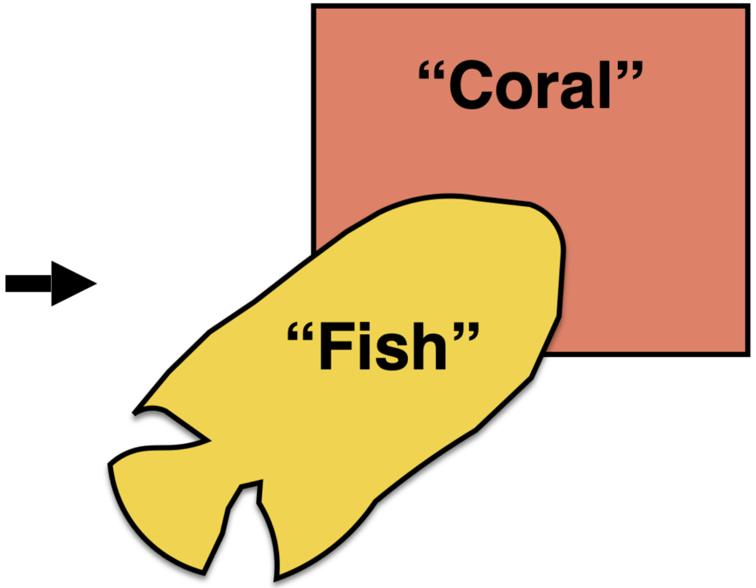
Drawn from memory



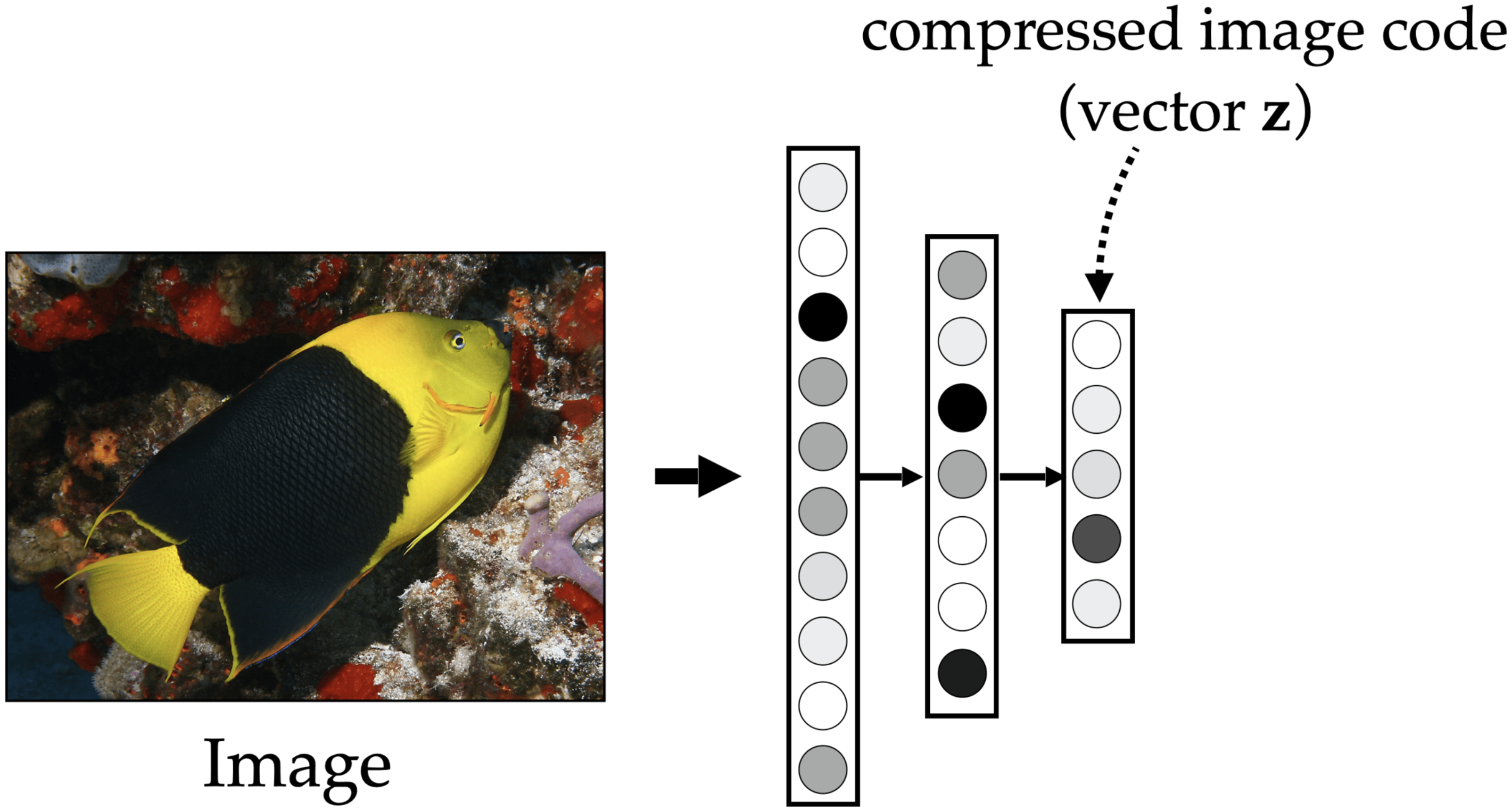
"I stand at the window and see a house, trees, sky. Theoretically I might say there were 327 brightnesses and nuances of colour. Do I have "327"? No. I have sky, house, and trees.”

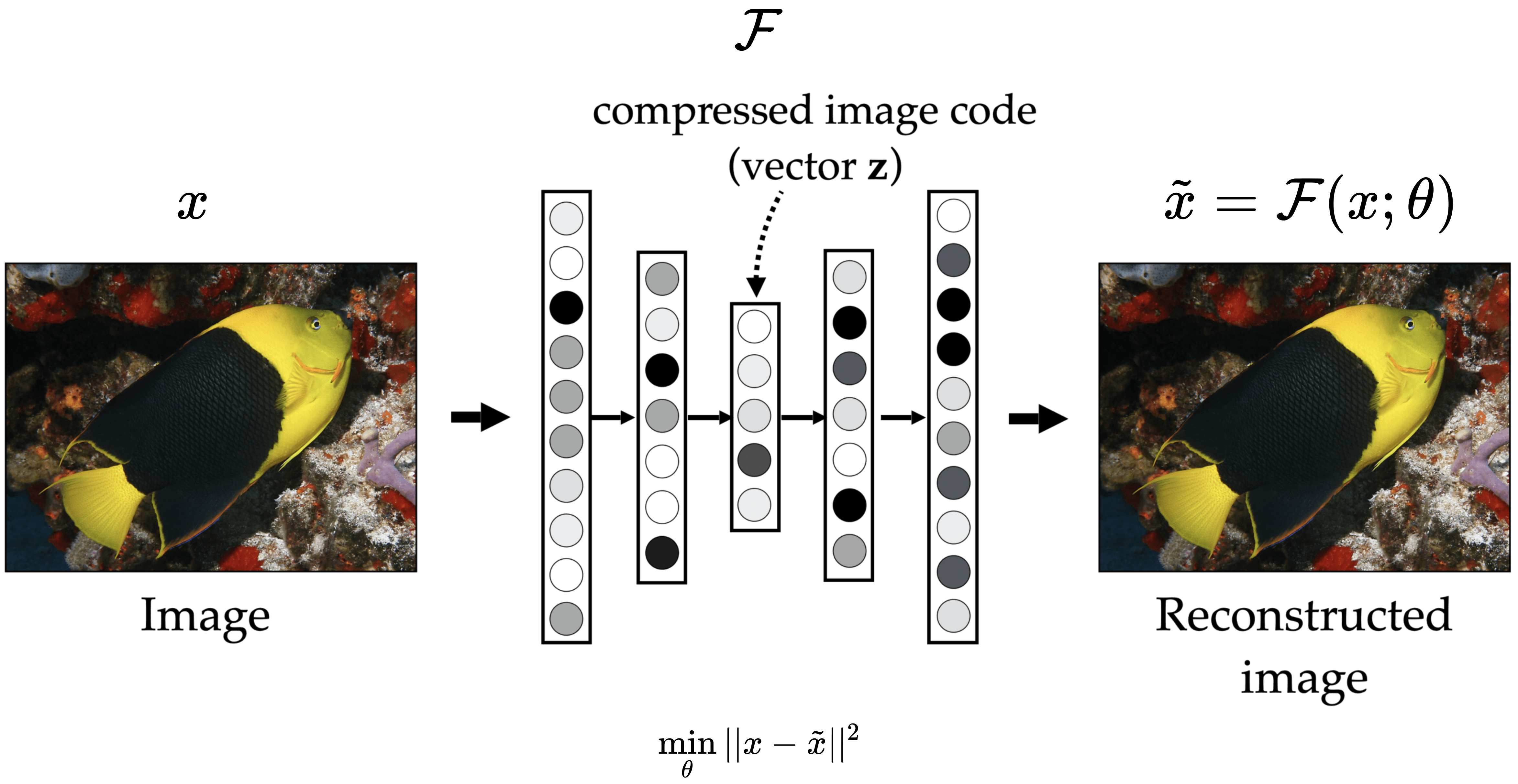
— Max Wertheimer, 1923

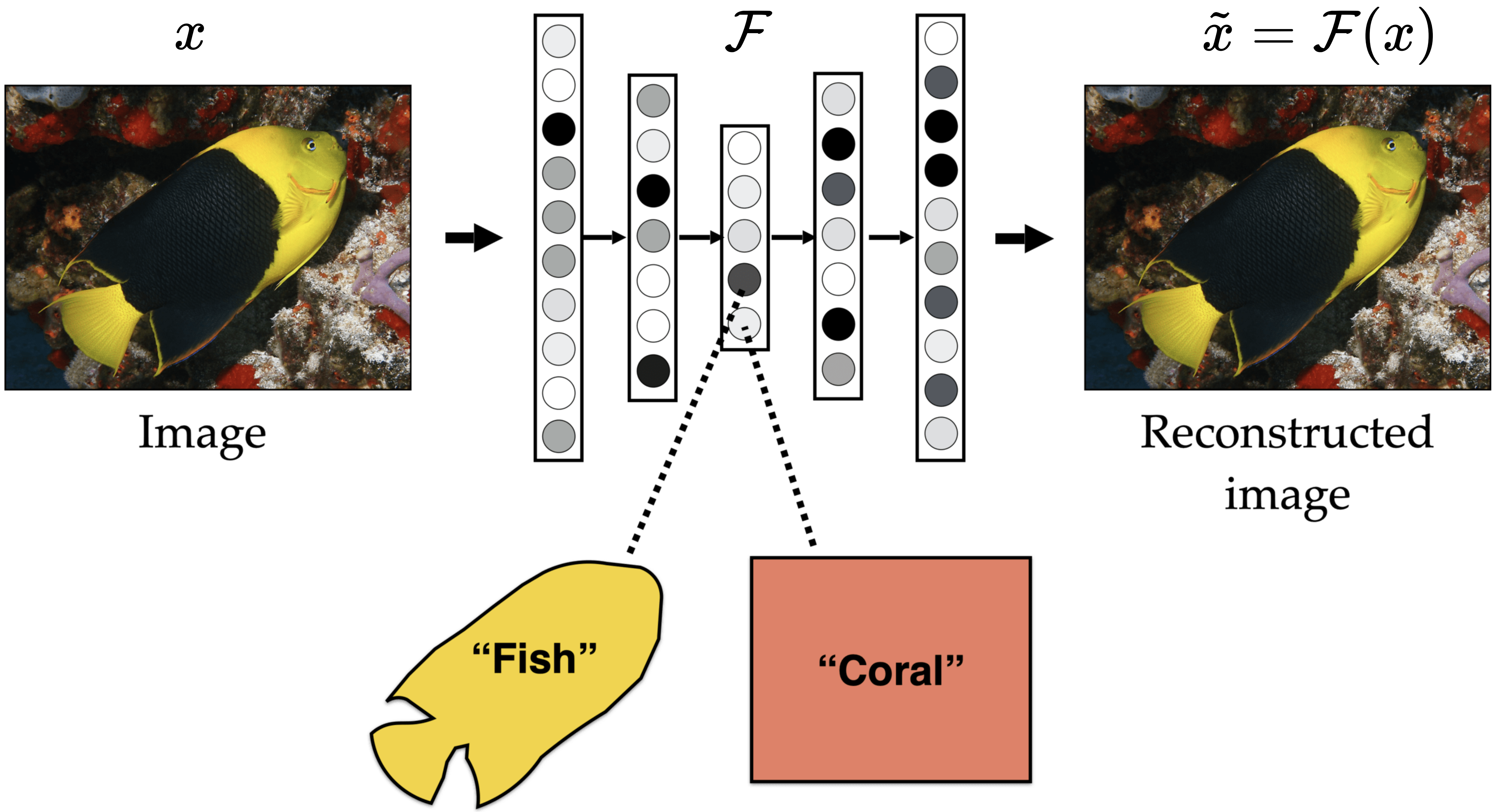
Image

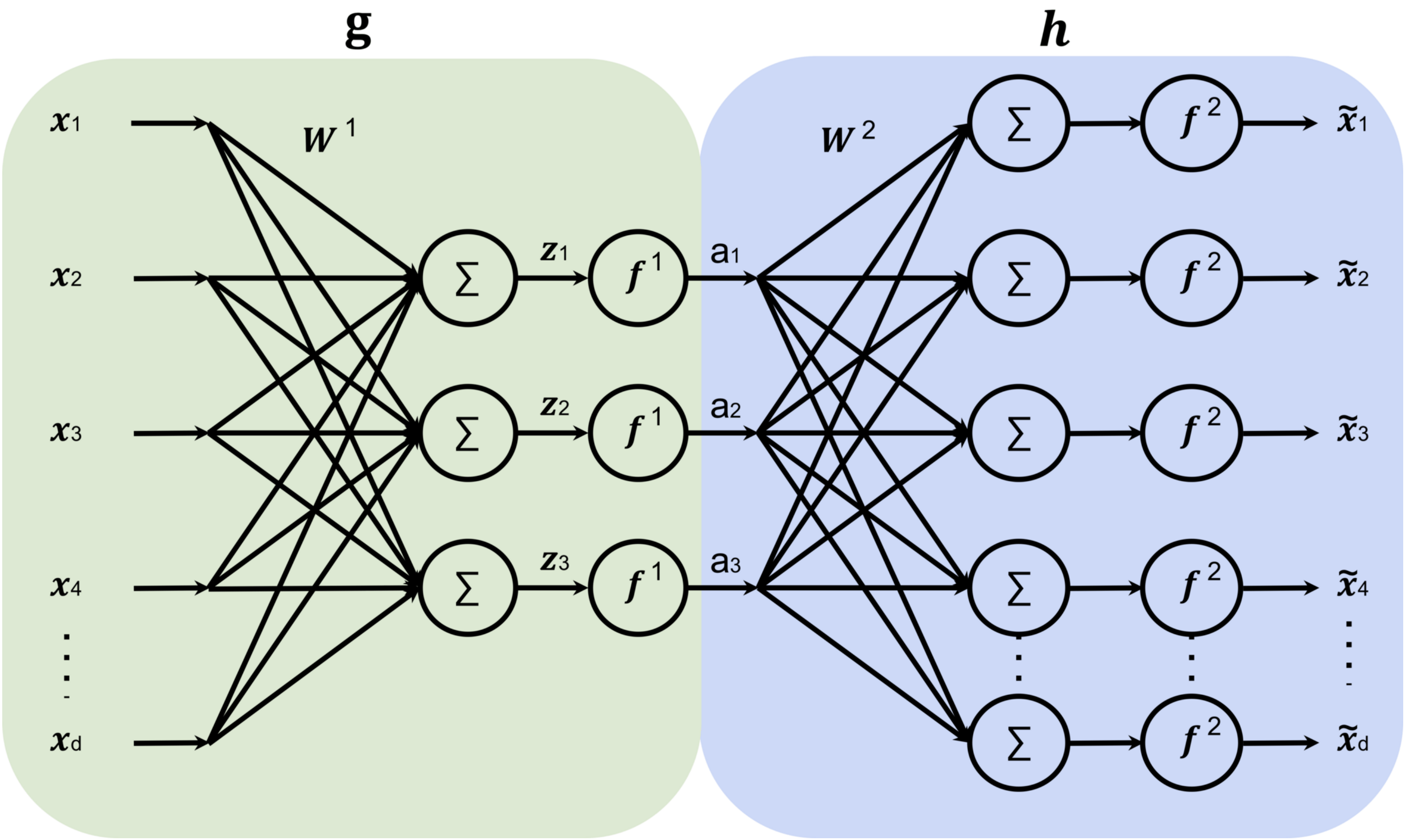


Compact mental representation


compressed image code
(vector **z**)
Image


$\mathcal{F}$
compressed image code
(vector $\mathbf{z}$)
x
$\tilde{x} = \mathcal{F}(x;\theta)$
Image
Reconstructed image
$\min_{\theta} ||x - \tilde{x}||^2$


x
$\mathcal{F}$
$\tilde{x} = \mathcal{F}(x)$
Image
Reconstructed image
"Fish"
"Coral"


g
h
x1
x2
x3
x4
xd
W 1
W 2
z1
z2
z3
f 1
a1
a2
a3
Σ
f 2
x̃1
x̃2
x̃3
x̃4
x̃d

Autoencoder

Data

$$\{\mathbf{x}^{(i)}\}_{i=1}^{N} \quad \rightarrow$$

Learner

Objective

$$\mathcal{L}(f(\mathbf{x}), \mathbf{x}) = \|f(\mathbf{x}) - \mathbf{x}\|_2^2$$

Hypothesis space

Neural net with a bottleneck

Optimizer

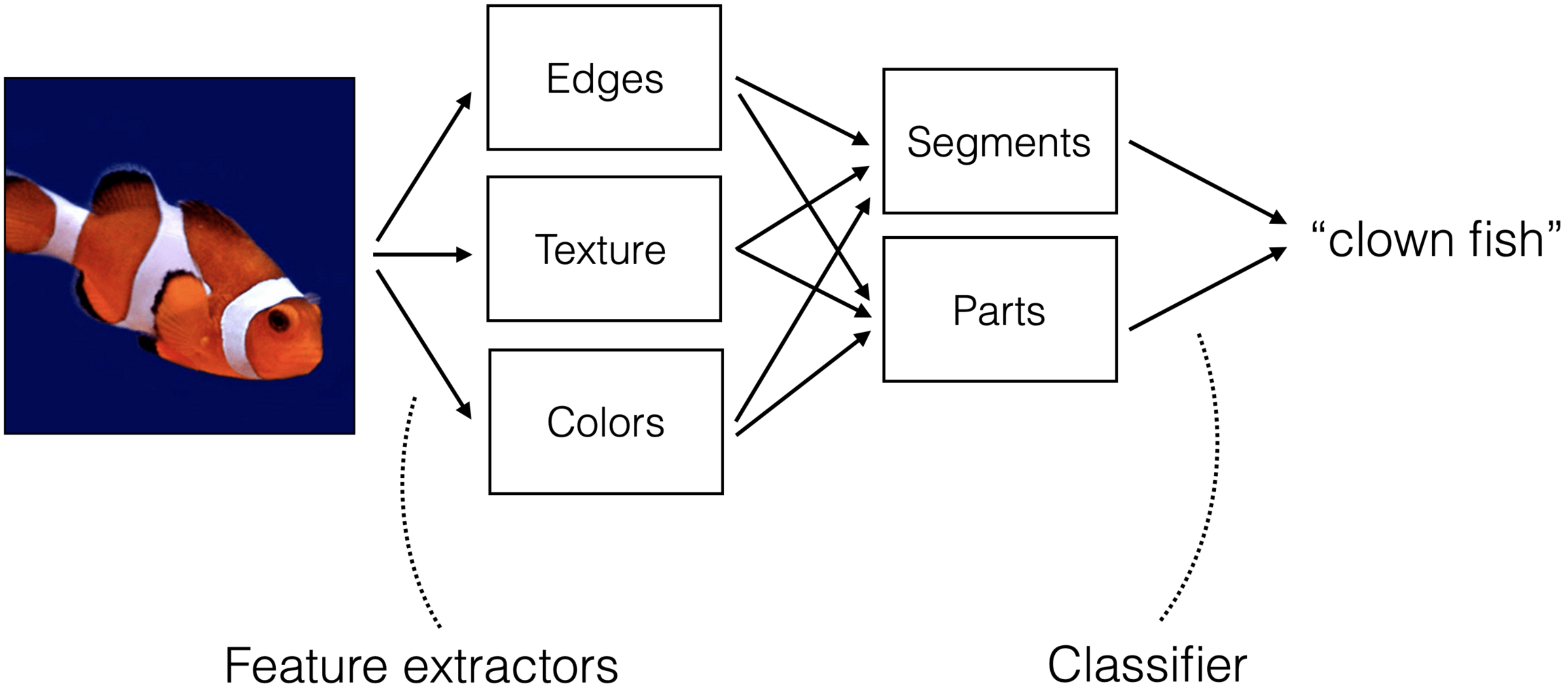
SGD

$$\rightarrow f$$

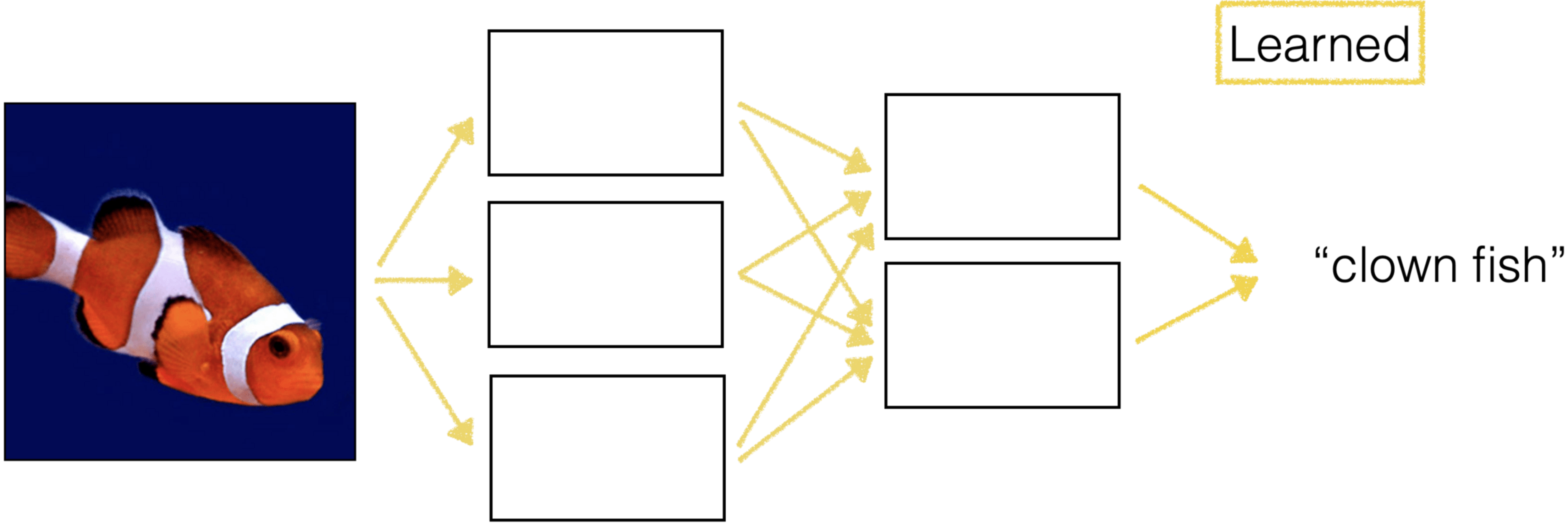
Outline

- Recap: Supervised learning and reinforcement learning
- Unsupervised learning
- Clustering: k-means algorithm
    - Clustering vs. classification
    - Initialization matters
    - k matters
- Auto-encoder
    - Compact representation
- Unsupervised learning again -- representation learning and beyond

Classical object recognition



Deep learning



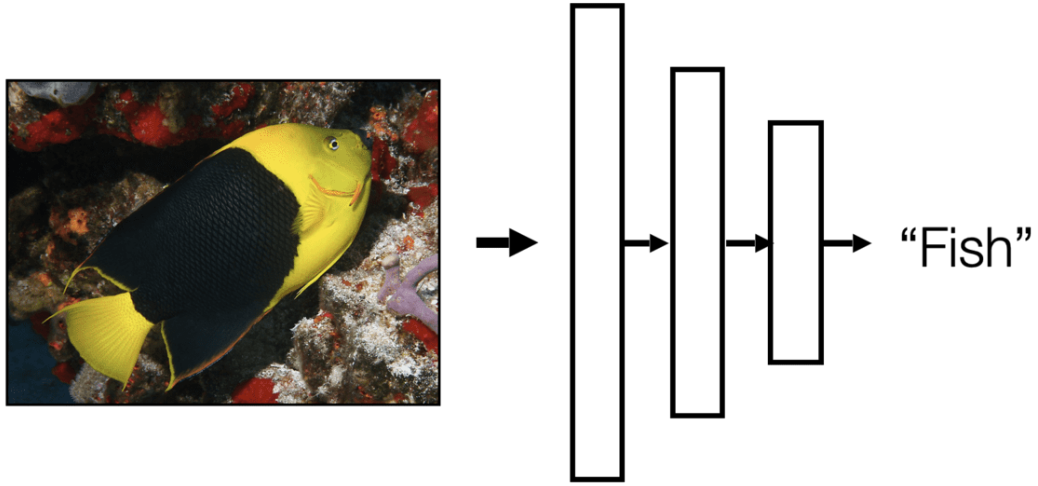
Transfer learning

"Generally speaking, a good representation is one that makes a subsequent learning task easier." — *Deep Learning*, Goodfellow et al. 2016

?

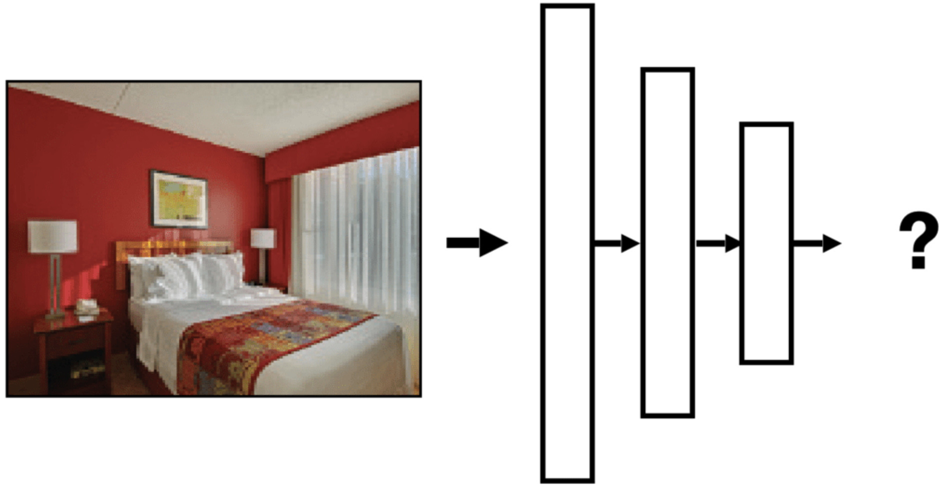
Training

Object recognition

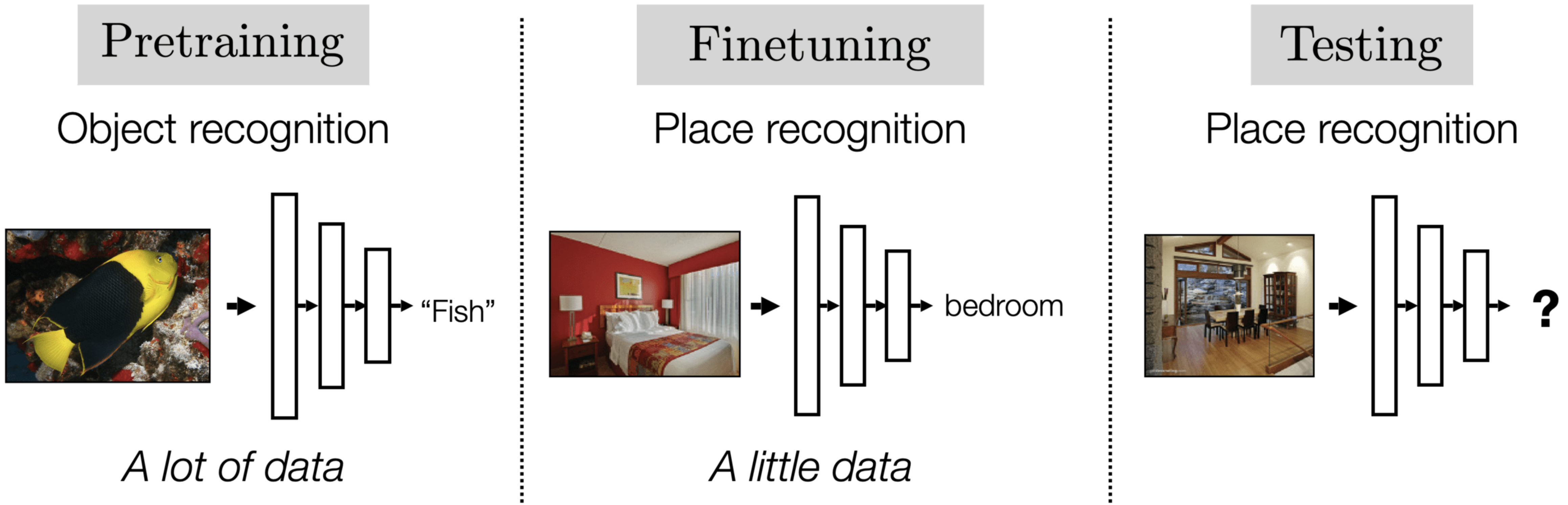


Testing

Place recognition



Often, what we will be "tested" on is to learn to do a new thing.



Finetuning starts with the representation learned on a previous task, and adapts it to perform well on a new task.

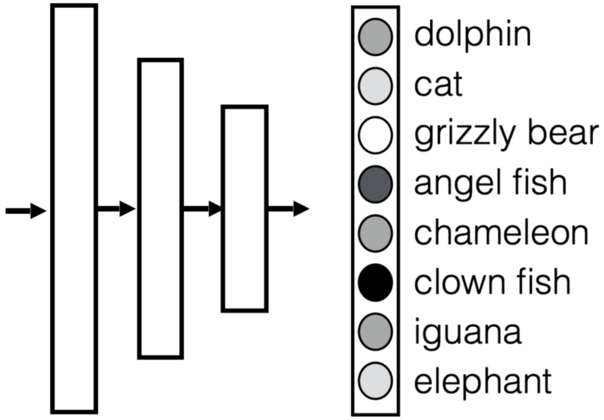
Finetuning in practice

- Pretrain a network on task A (often object recognition), resulting in parameters $\mathbf{W}$ and $\mathbf{b}$
- Initialize a second network with some or all of $\mathbf{W}$ and $\mathbf{b}$
- Train the second network on task B, resulting in parameters $\mathbf{W}'$ and $\mathbf{b}'$

Finetuning in practice

Pretraining

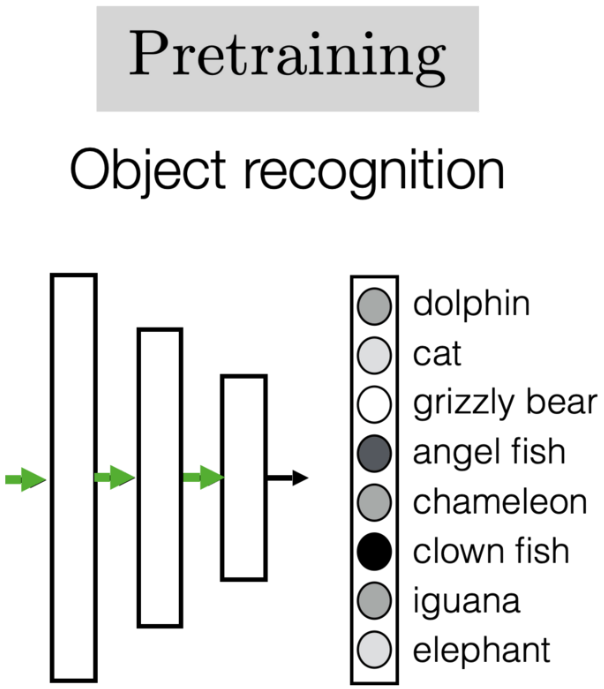
Object recognition

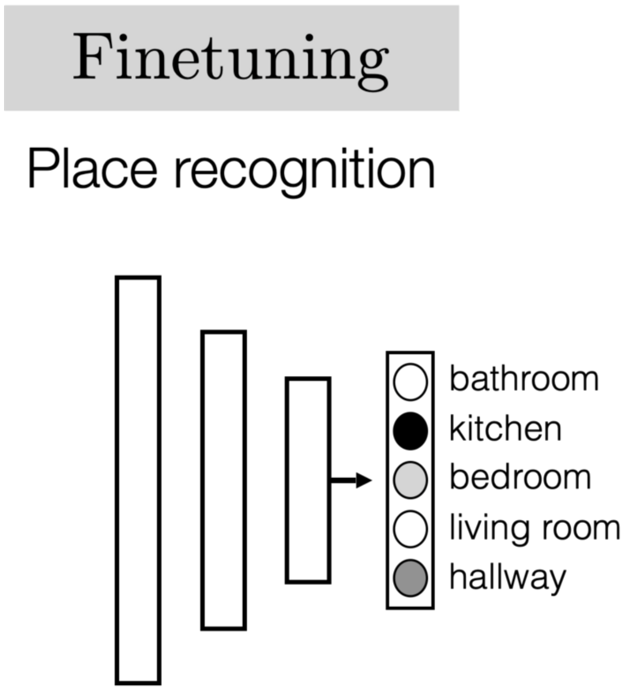


Finetuning

Place recognition

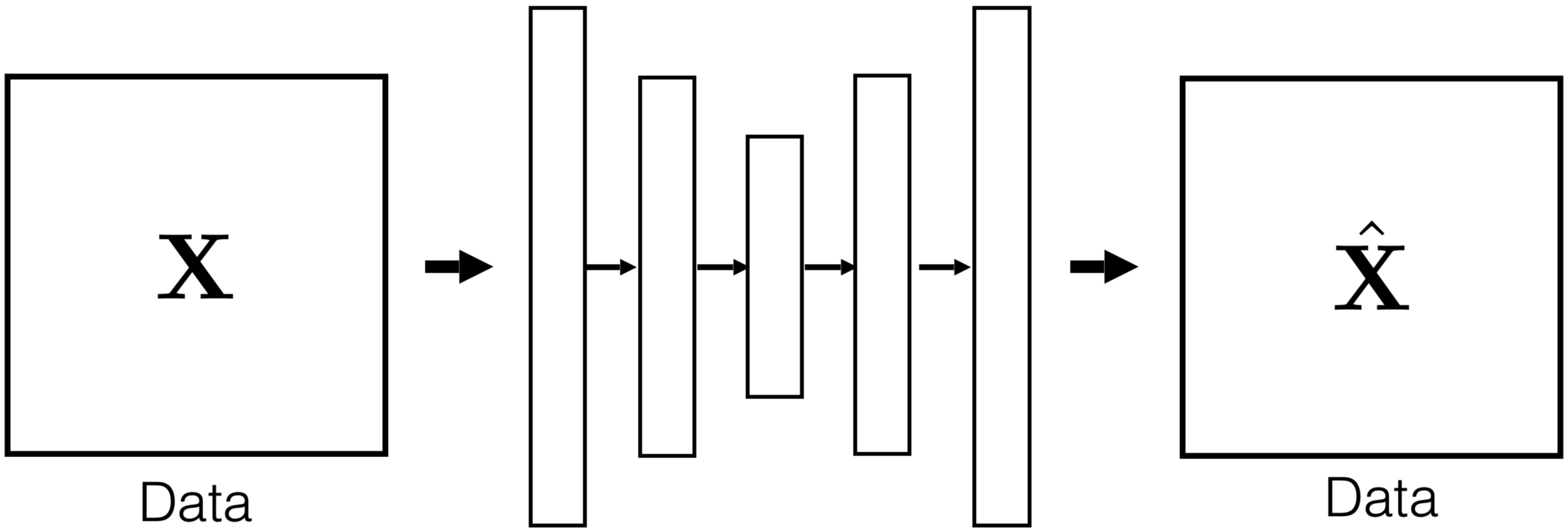
Finetuning in practice



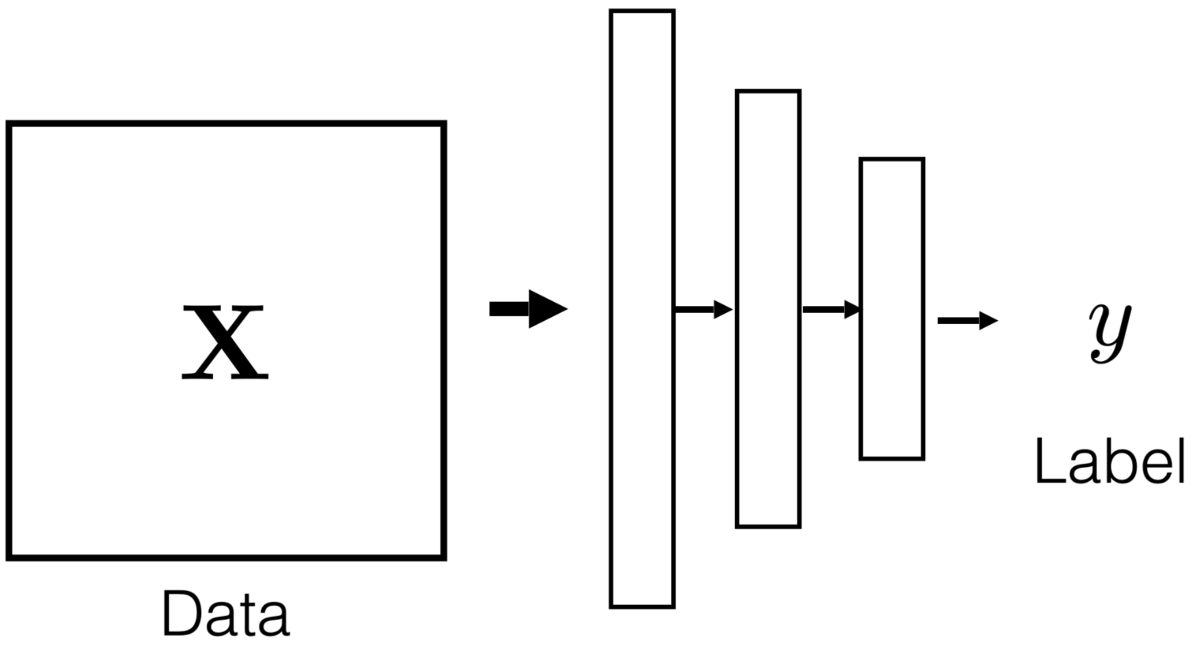


The “learned representation” is just the weights and biases, so that’s what we transfer

Data compression

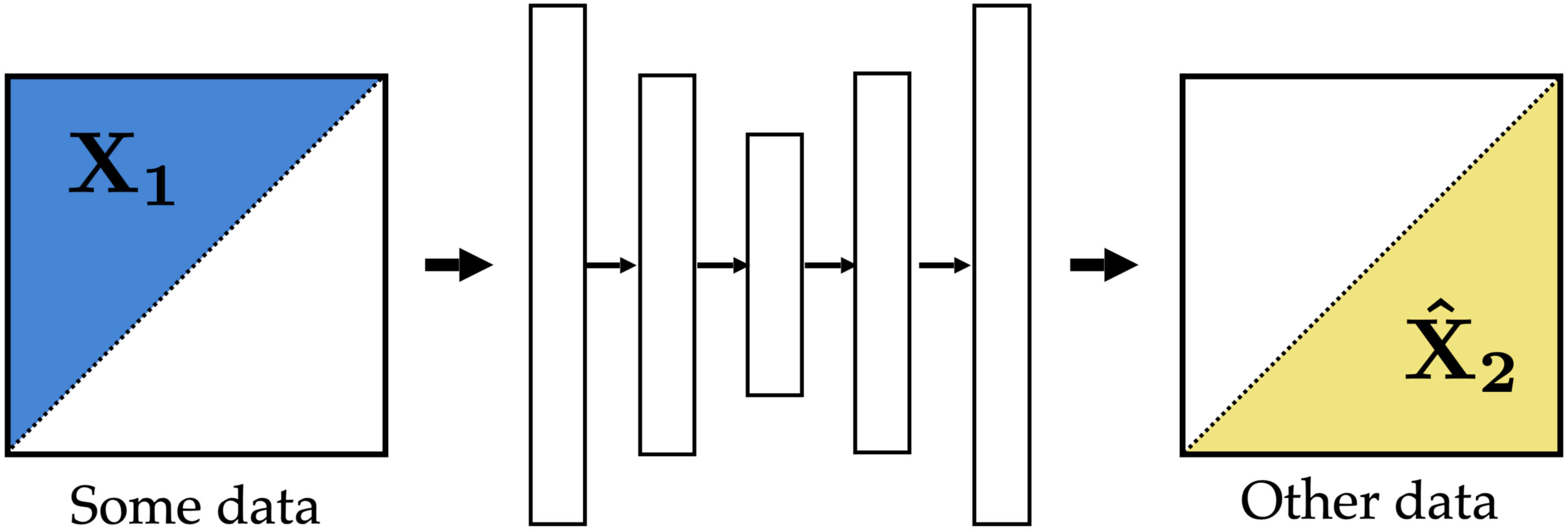


Label prediction

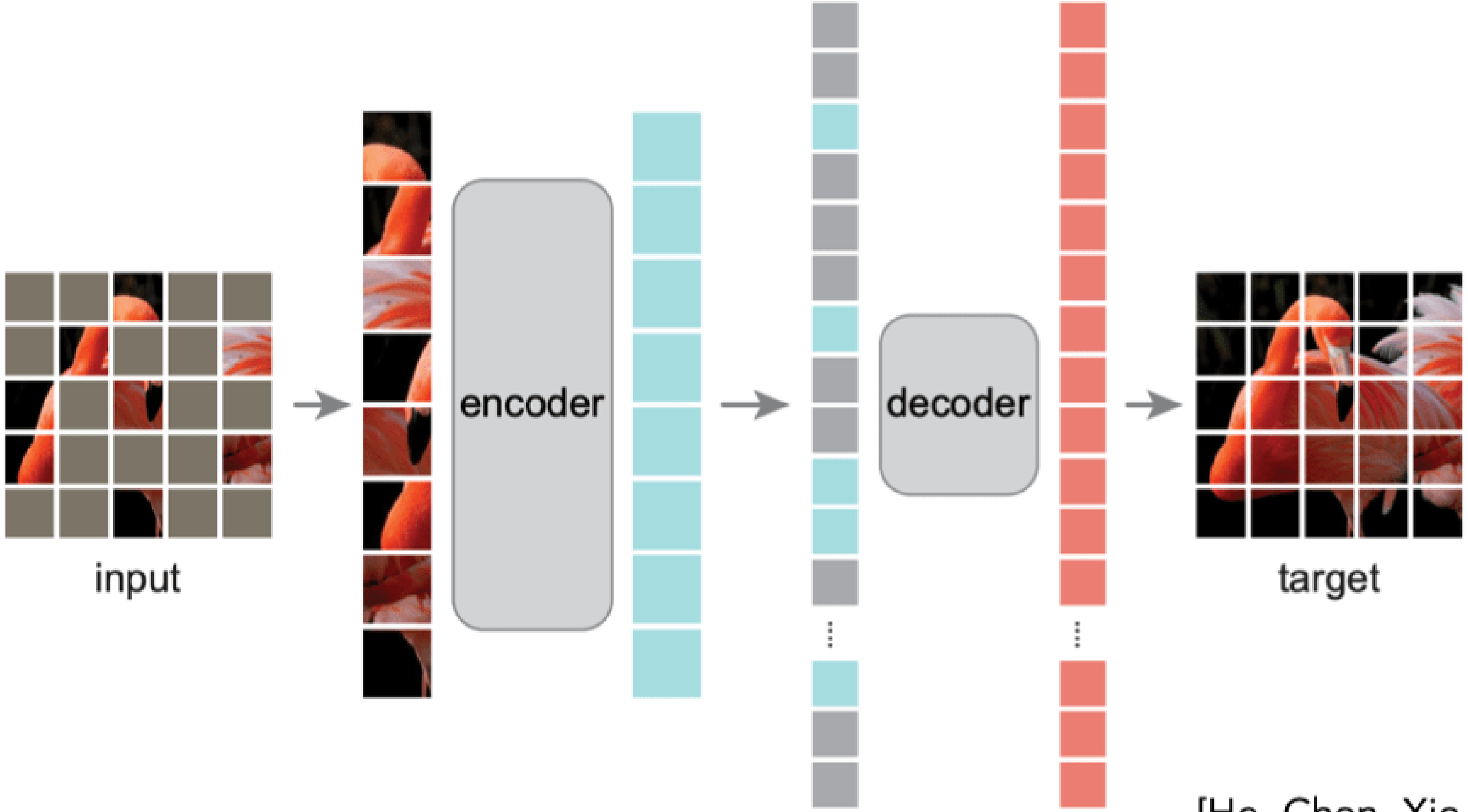


e.g., image classification

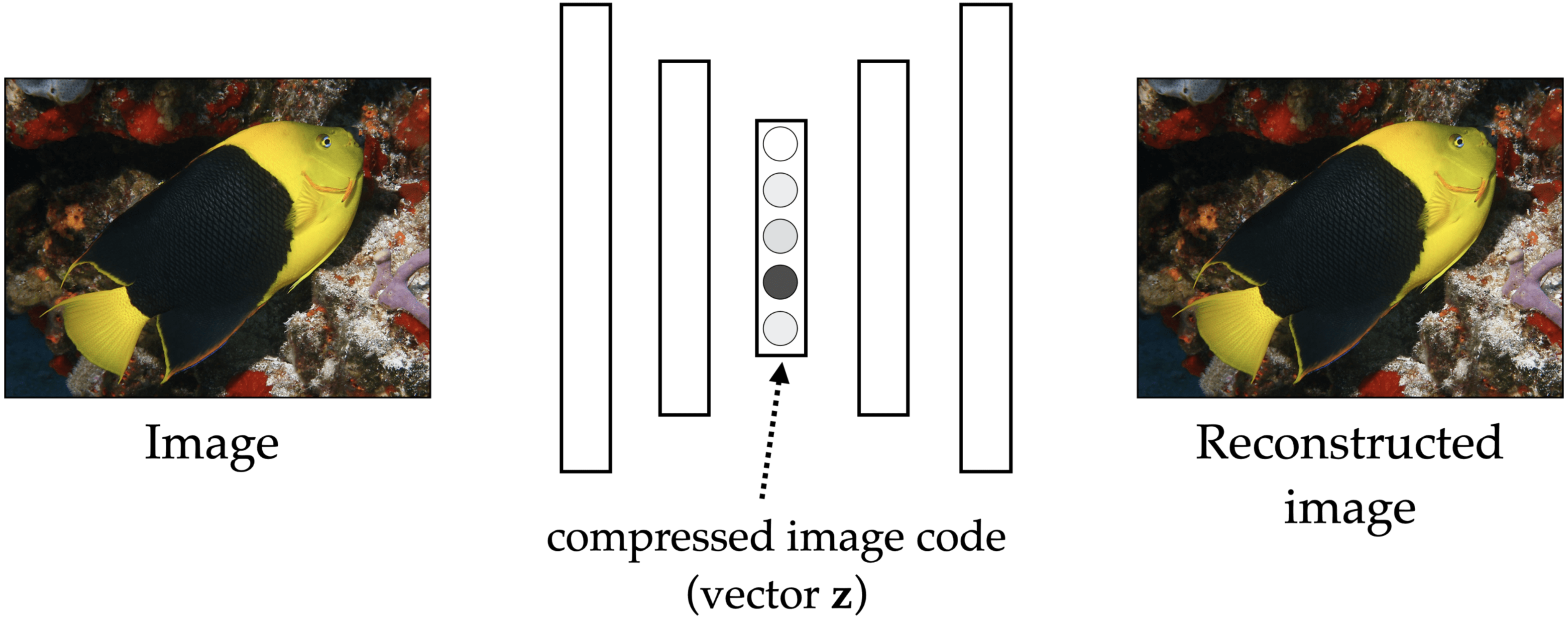
Data prediction
aka "self-supervised learning"



Self-supervision (masking)



[He, Chen, Xie, et al. 2021]

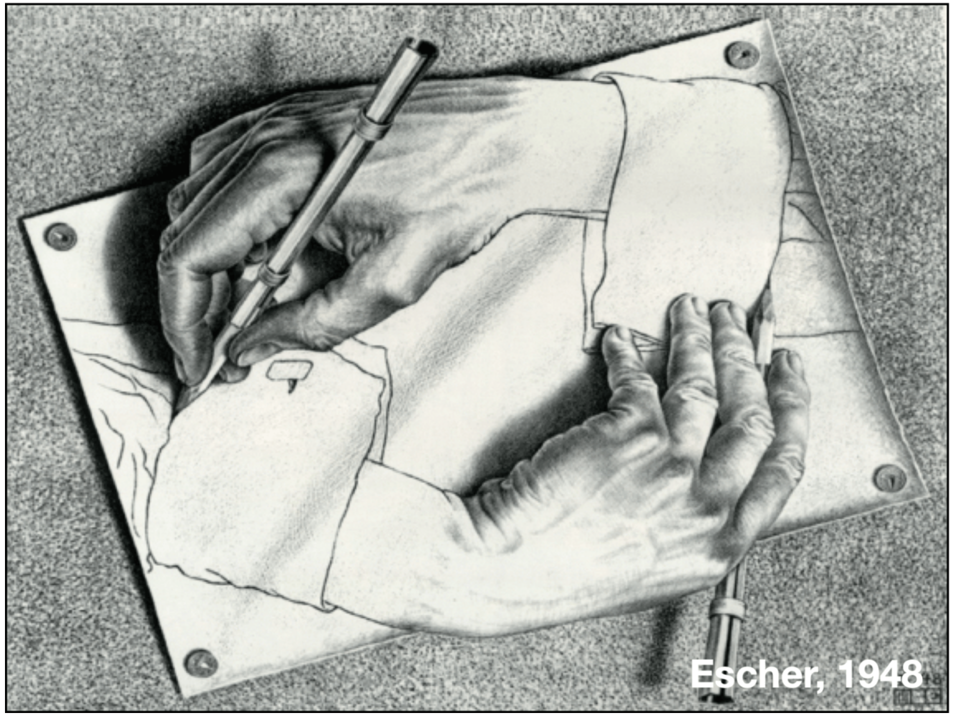


Is the code informative about object class y?

Logistic regression:

$$y = \sigma(\mathbf{W}\mathbf{z} + \mathbf{b})$$

Self-supervised learning



Common trick:

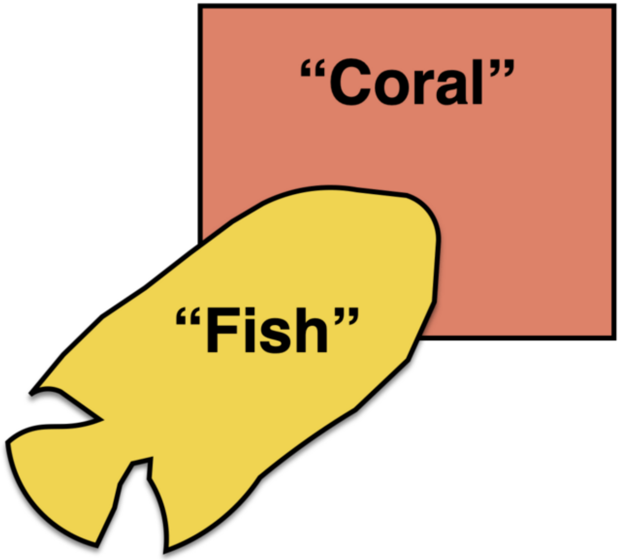
- Convert "unsupervised" problem into "supervised" empirical risk minimization

- Do so by cooking up "labels" (prediction targets) from the raw data itself

Representation learning

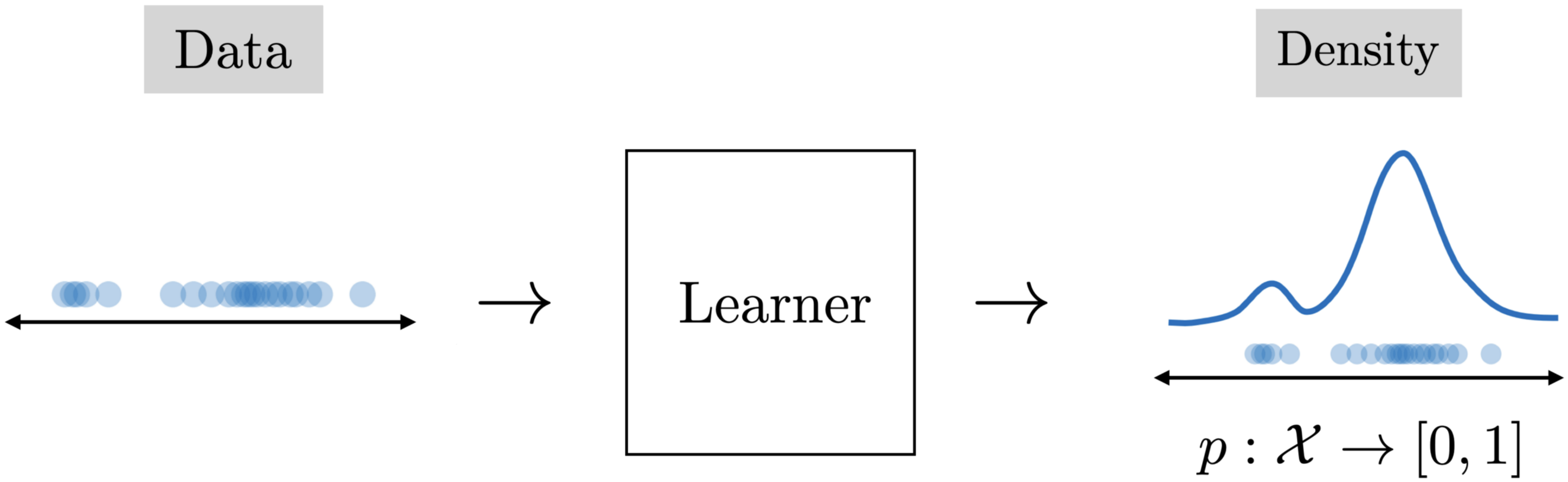
Good representations are:

1. Compact (*minimal*)
2. Explanatory (*sufficient*)
3. Disentangled (*independent factors*)
4. Interpretable
5. *Make subsequent problem solving easy*

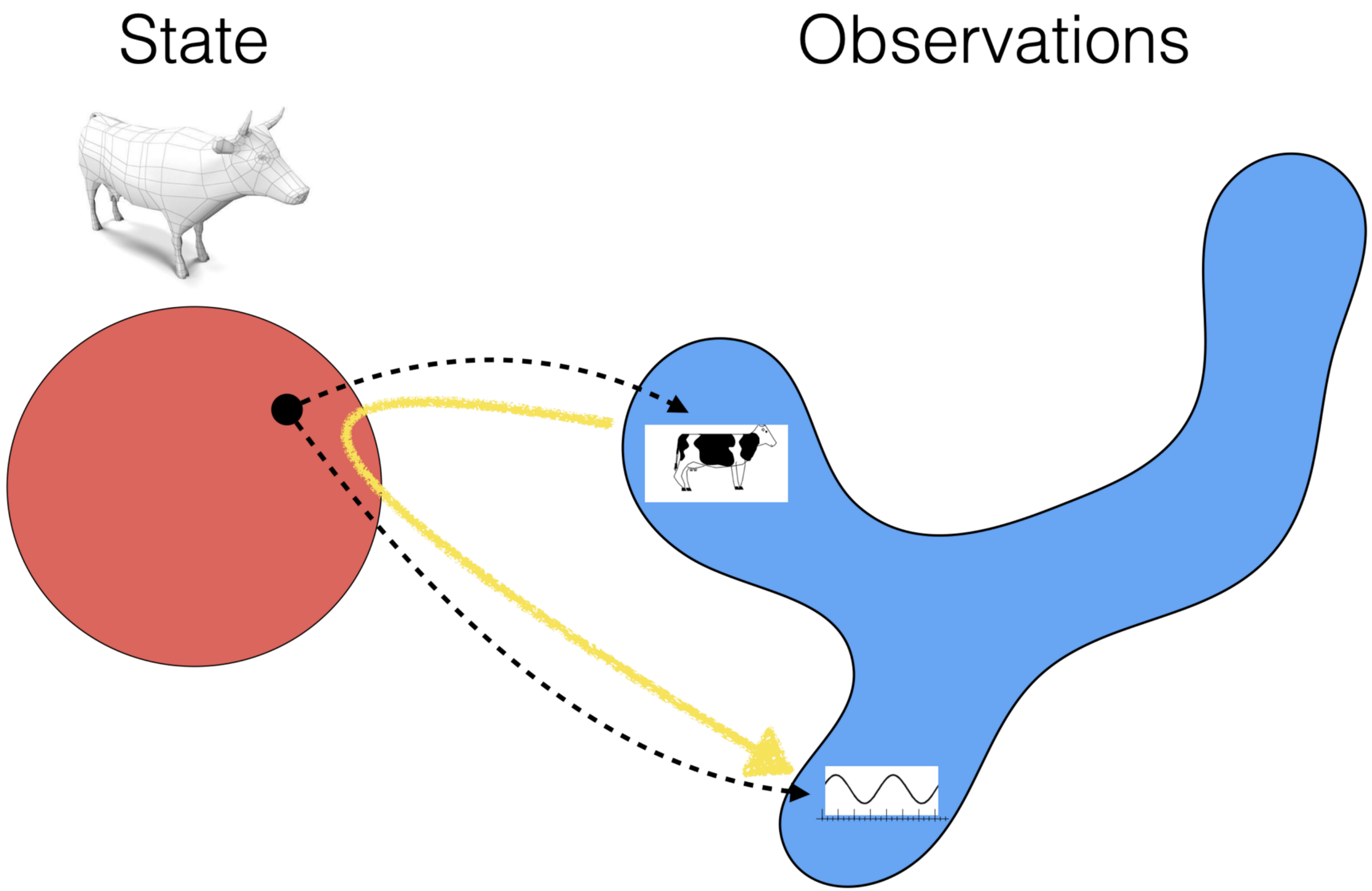


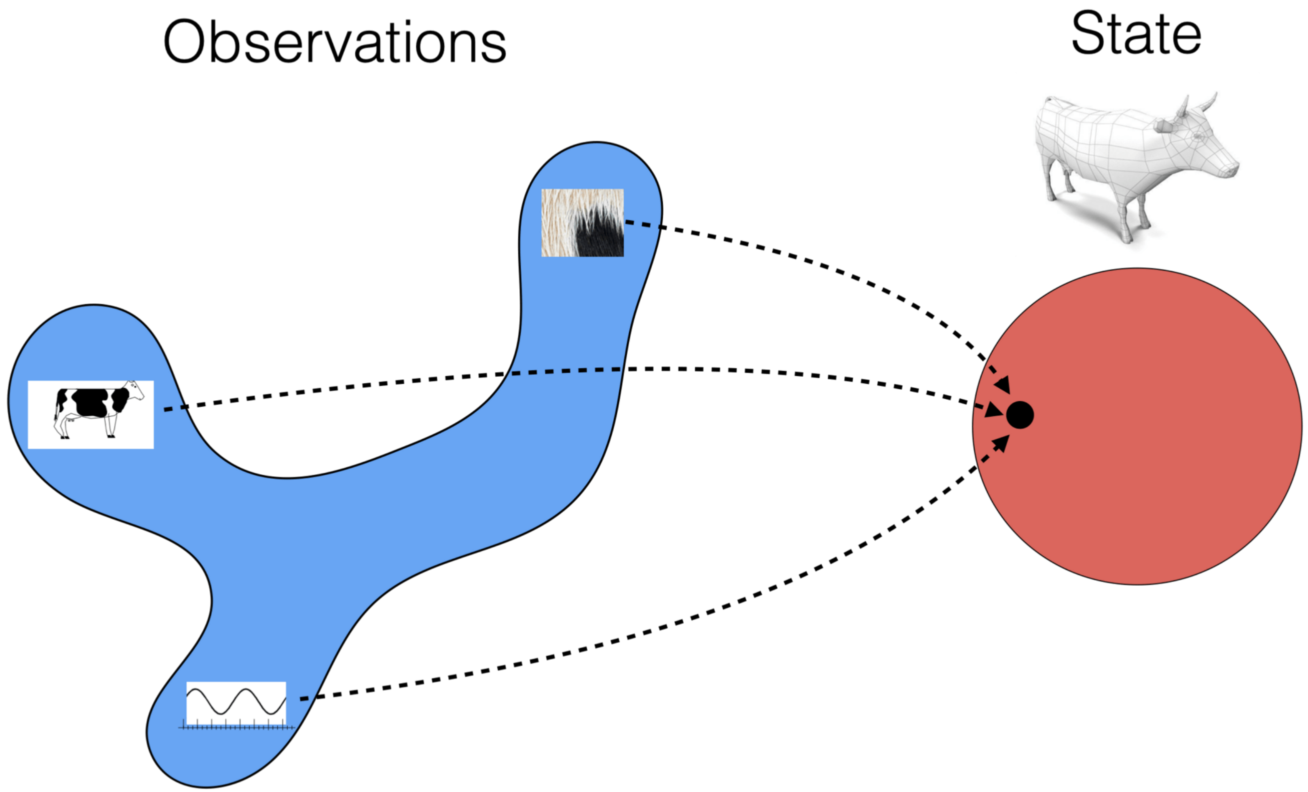
[See "Representation Learning", Bengio 2013, for more commentary]

(Density estimation)



[figs modified from: http://introtodeeplearning.com/materials/2019_6S191_L4.pdf]


State
Observations



The way you measure the world does not change the underlying state

Dall-E 2 (UnCLIP):

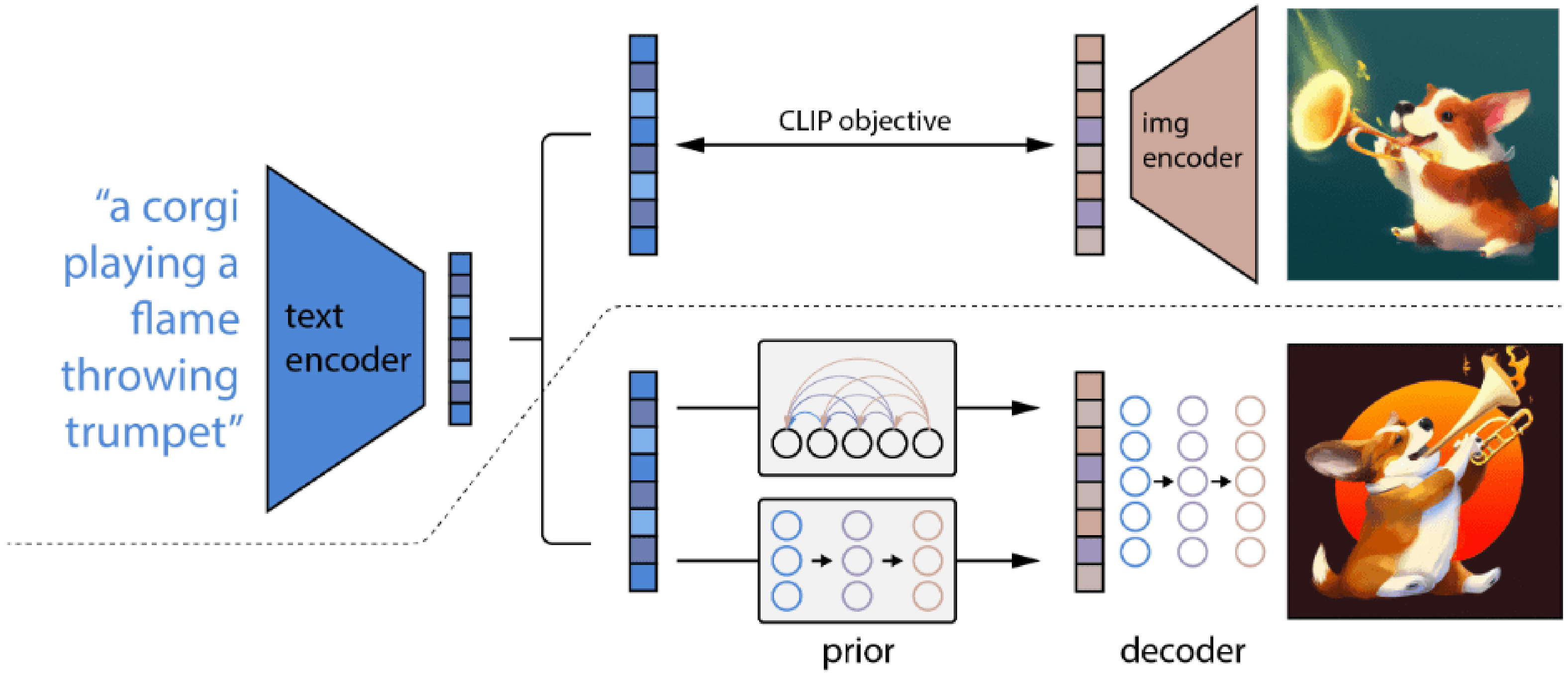


Figure 2: A high-level overview of unCLIP. Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process: a CLIP text embedding is first fed to an autoregressive or diffusion prior to produce an image embedding, and then this embedding is used to condition a diffusion decoder which produces a final image. Note that the CLIP model is frozen during training of the prior and decoder.

[https://arxiv.org/pdf/2204.06125.pdf]

We'd appreciate your feedback on the lecture.

Thanks!