innovate achieve lead

SE ZG568 / SS ZG568, Applied Machine Learning
Lecture No. 15 [11- May-2025]

Topic of Discussion

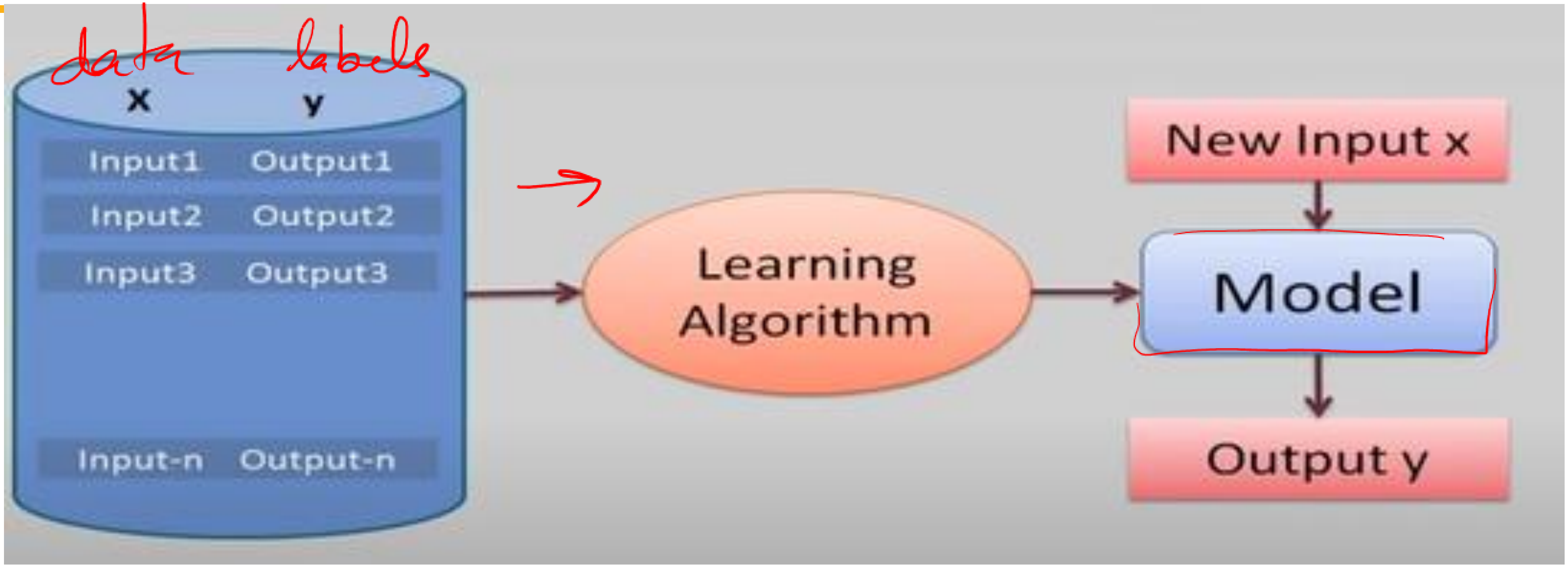
Unsupervised Learning

K Means Clustering

Acknowledgement

Prof. Tamara Broderick, MIT Institute for Foundations of Data Science (MIFODS)

Supervised Learning



$\boxed{f} : x \rightarrow y$

Image Source: Prof. Sudeshna Sarkar's lecture

Unsupervised Learning

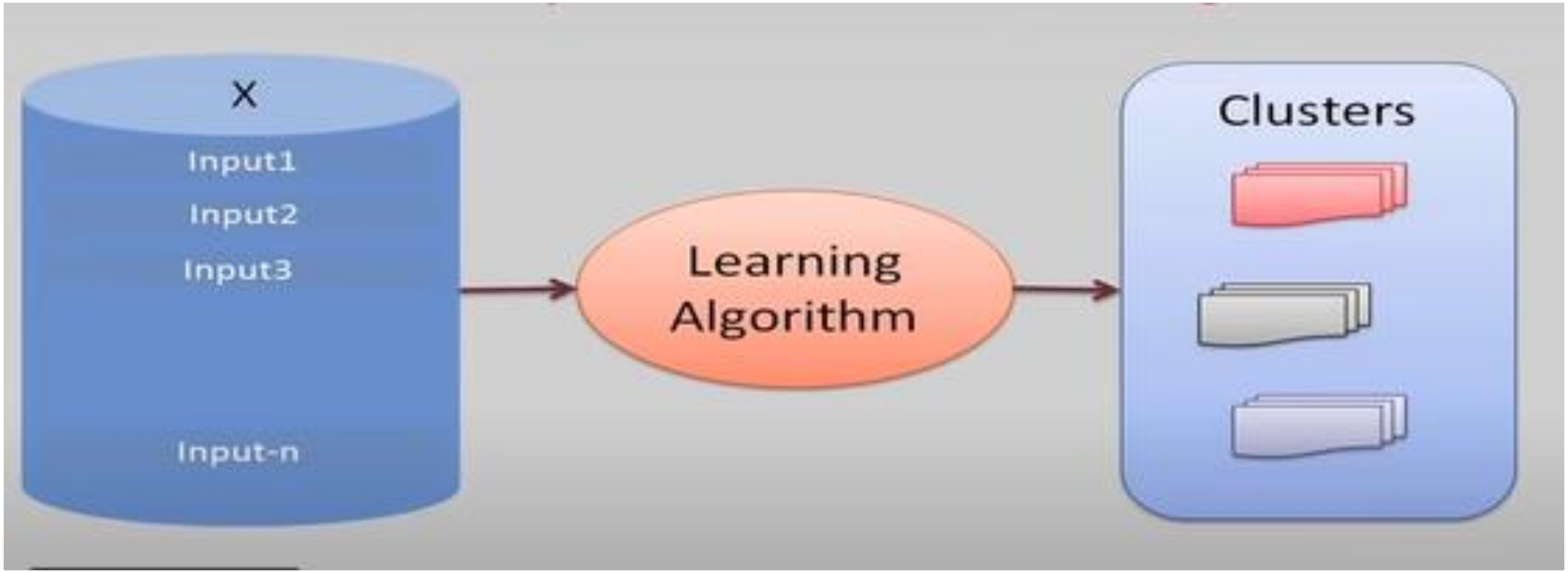
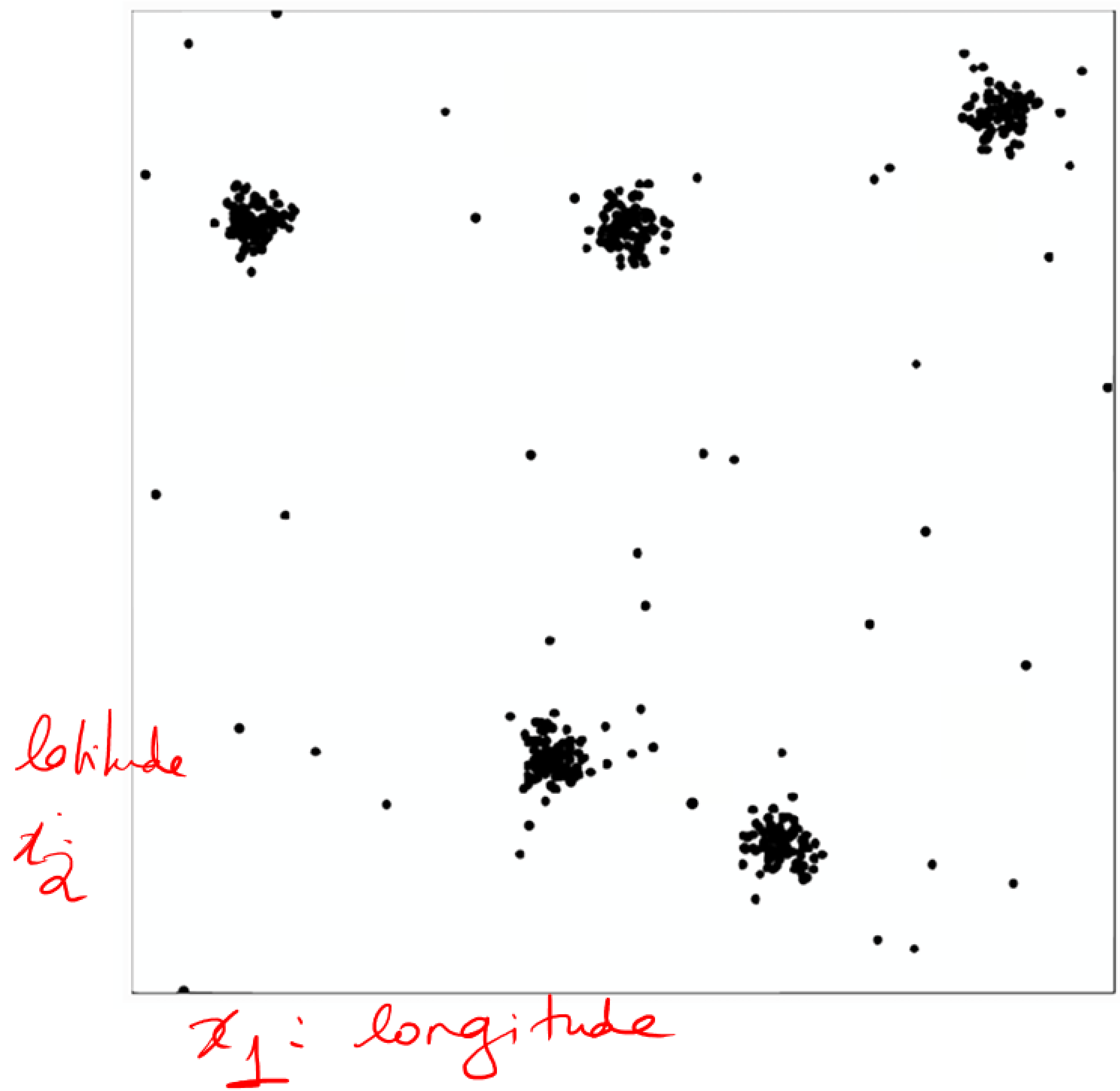


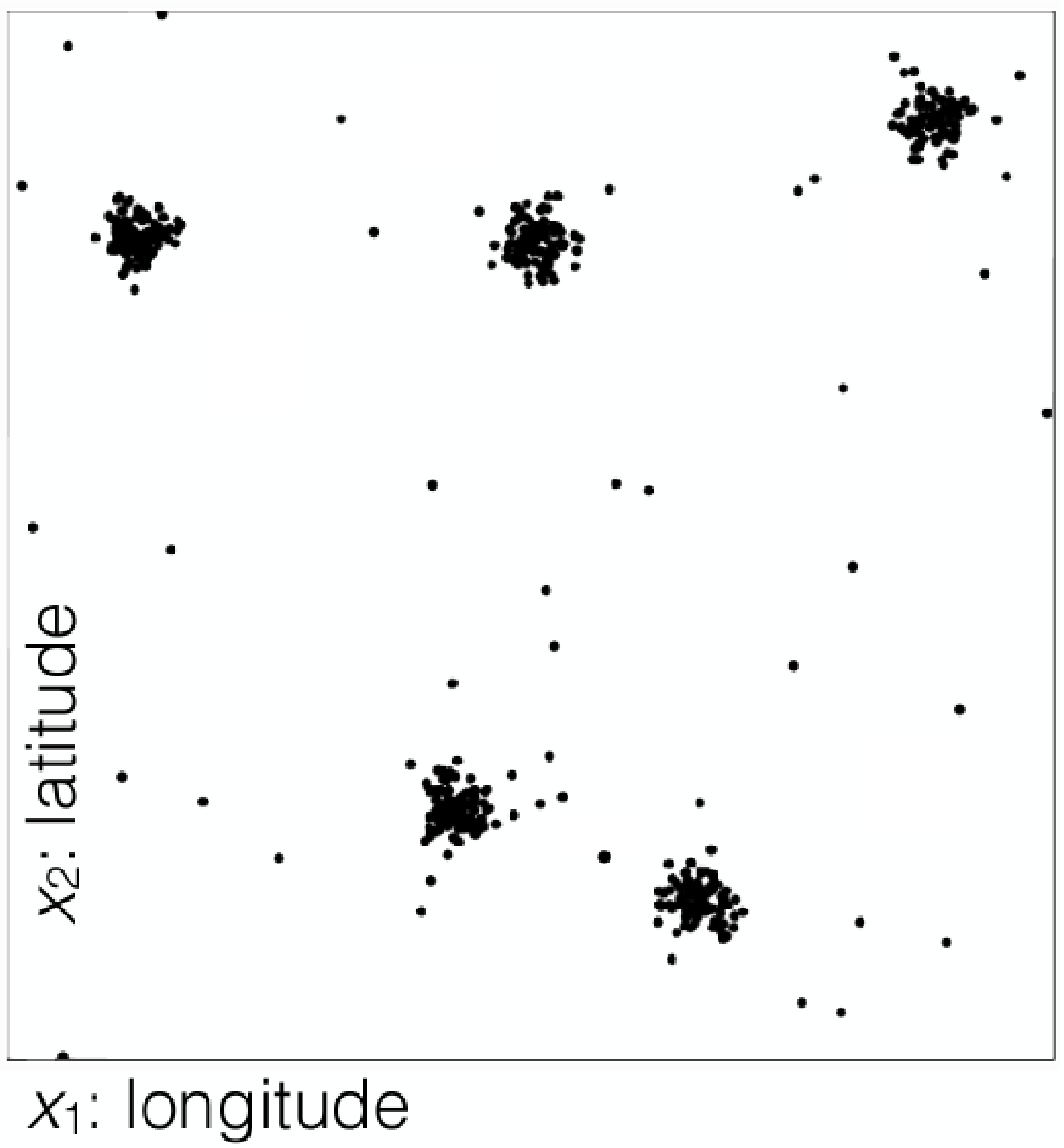
Image Source: Prof. Sudeshna Sarkar's lecture

Food distribution placement



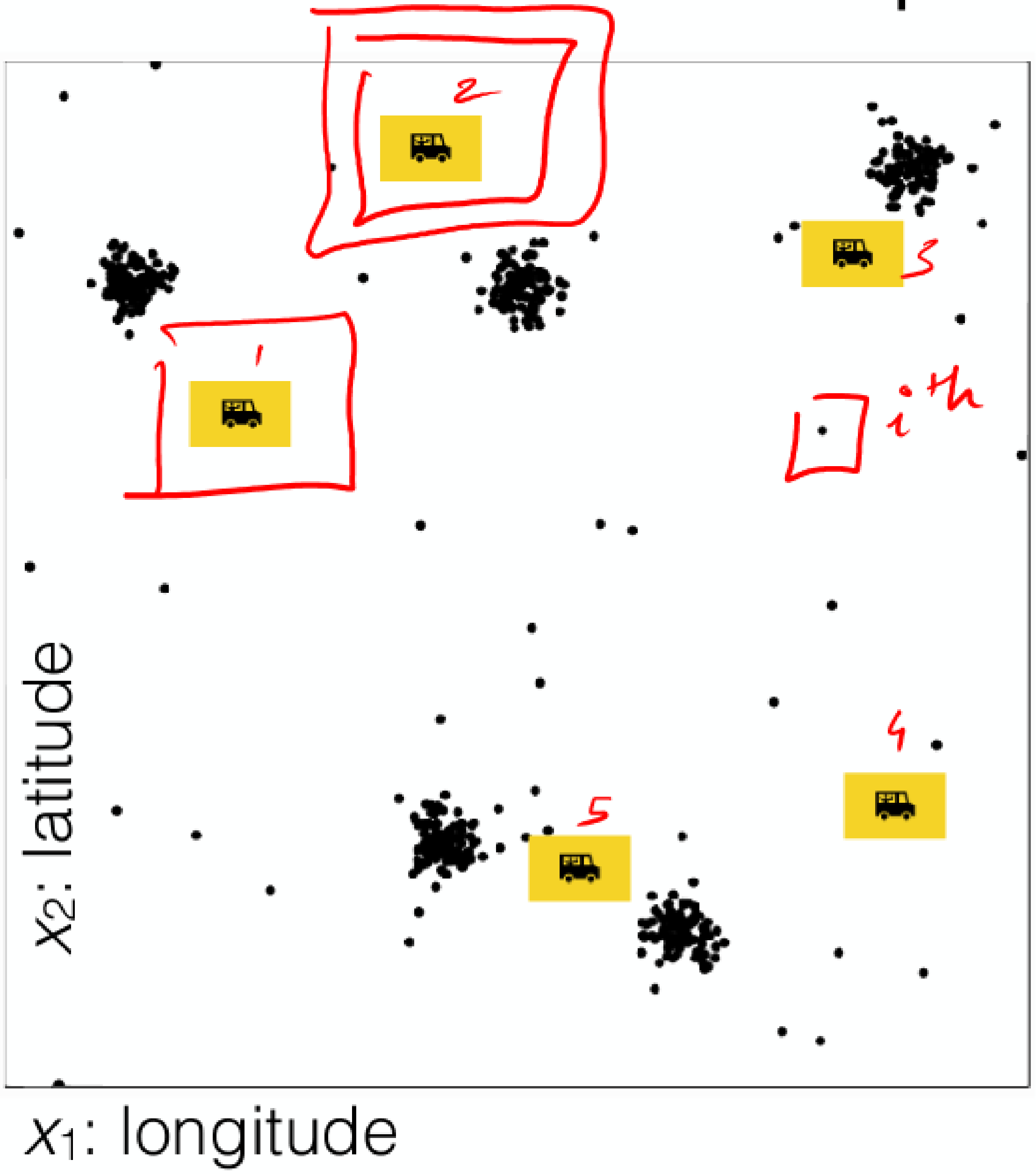
- Where should I have my *k* food trucks park?

Food distribution placement



- Where should I have my *k* food trucks park?
- Want to minimize the loss of people we serve

Food distribution placement



$k = 5$ → people

- Where should I have my *k* food trucks park?
- Want to minimize the loss of people we serve
- Person *i* location $x^{(i)}$
- Food truck *j* location $\mu^{(j)}$

$$x^{(i)} = \begin{bmatrix} x_1^{(i)} & x_2^{(i)} \end{bmatrix}$$

$x_1^{(i)}$ ↓ longitude, $x_2^{(i)}$ ↓ latitude

$j = 1$ to k

$$\left.\begin{aligned} \mu^{(1)} &= \begin{bmatrix} \mu_1^{(1)} & \mu_2^{(1)} \end{bmatrix} \\ \mu^{(2)} &= \begin{bmatrix} \mu_1^{(2)} & \mu_2^{(2)} \end{bmatrix} \end{aligned}\right\} \text{Truck position}$$

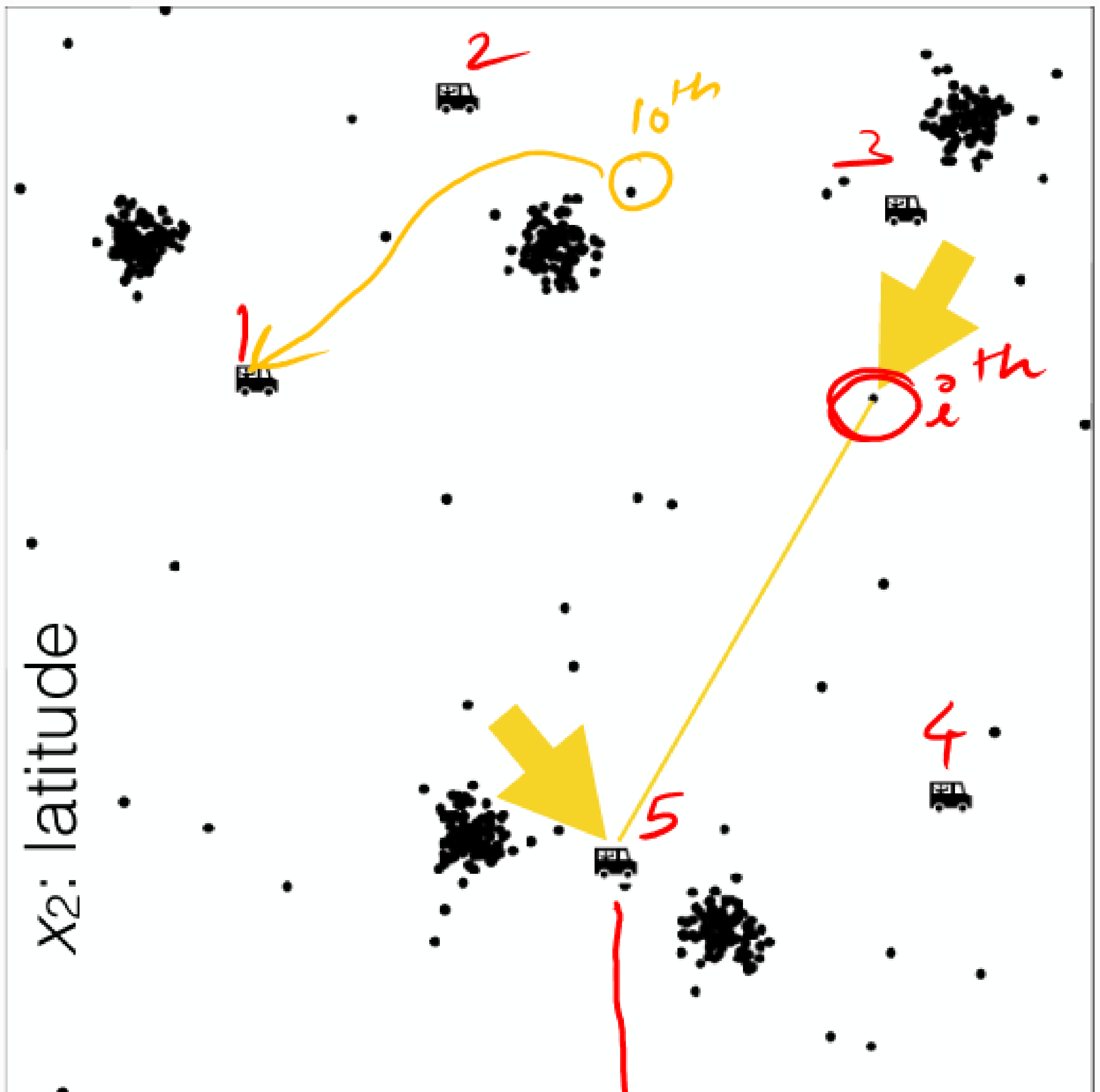
$i = 1$ to n

"n" people or "n" data instance

Food distribution placement

$x^{(10)}$

$\mu^{(1)}$



x_1: longitude

- Where should I have my k food trucks park?
- Want to minimize the loss of people we serve
- Person i location $x^{(i)}$
- Food truck j location $\mu^{(j)}$
- Index of truck where person i walks: $y^{(i)}$
- Loss if i walks to truck j:

$$\boxed{\|x^{(i)} - \mu^{(j)}\|_2^2}$$

$$x^{(i)} = \begin{bmatrix} x_1^{(i)} & x_2^{(i)} \end{bmatrix}$$

$$\mu^{(5)} = \begin{bmatrix} \mu_1^{(5)} & \mu_2^{(5)} \end{bmatrix}$$

$$\| x^{(i)} - \mu^{(5)} \|_2^2 = \left(\sqrt{(x_1^{(i)} - \mu_1^{(5)})^2 + (x_2^{(i)} - \mu_2^{(5)})^2} \right)^2$$

k-means algorithm

```
k-means(k, )
```

k-means algorithm

no of food truck

`k-means(k, τ)`

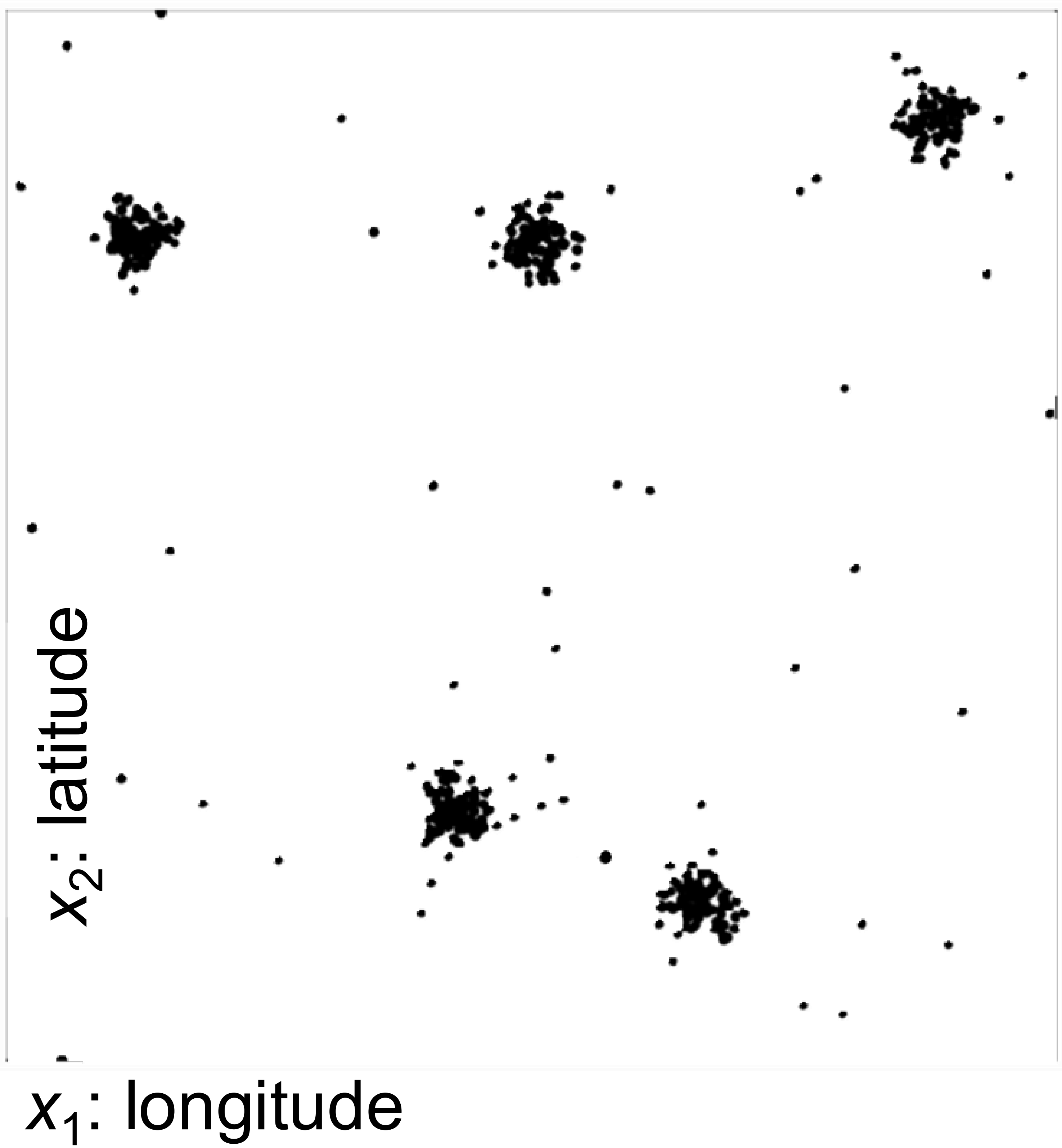
max. iter

k-means algorithm

`k-means(k, τ)`

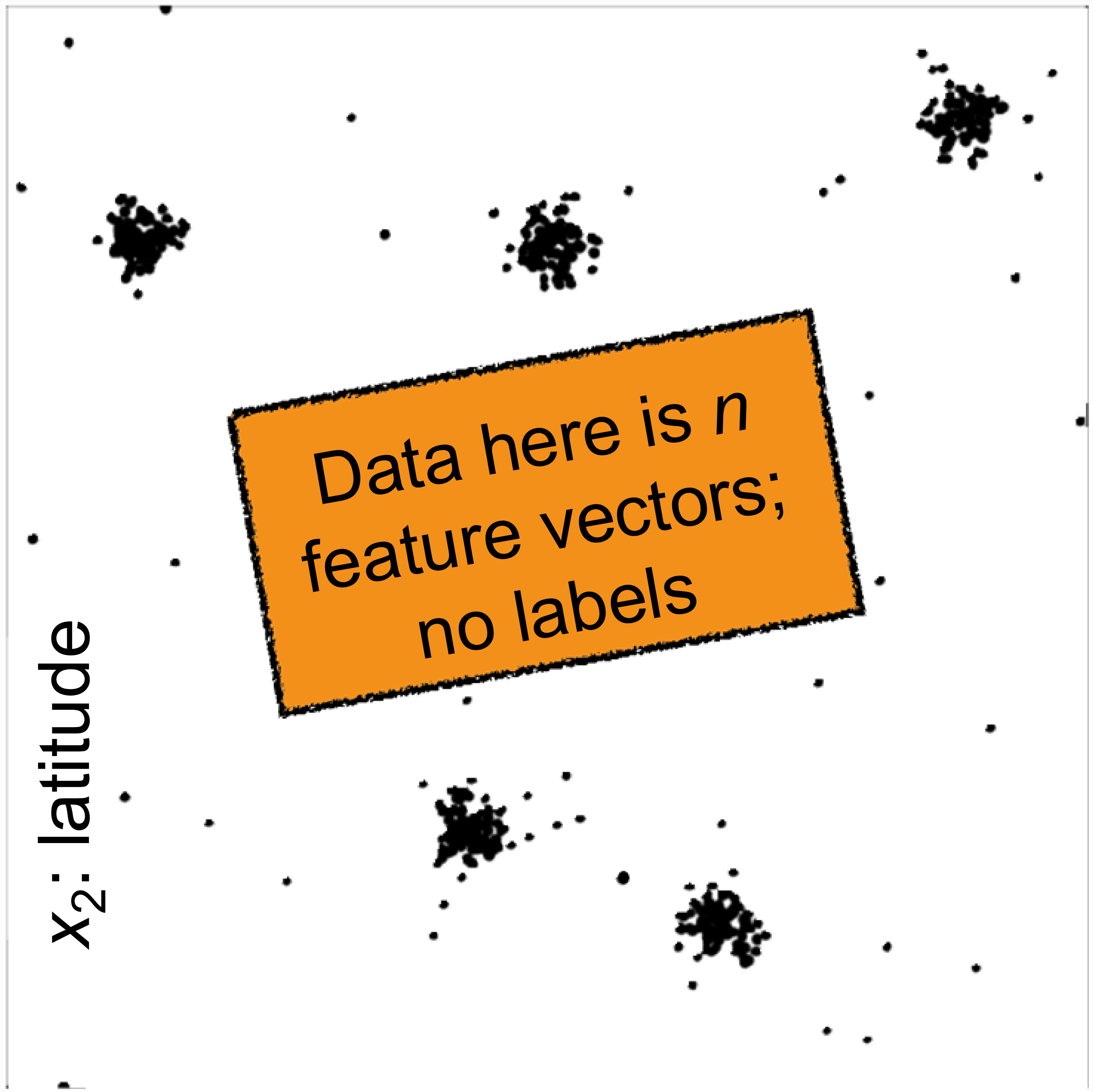
k-means algorithm

`k-means(k, τ)`



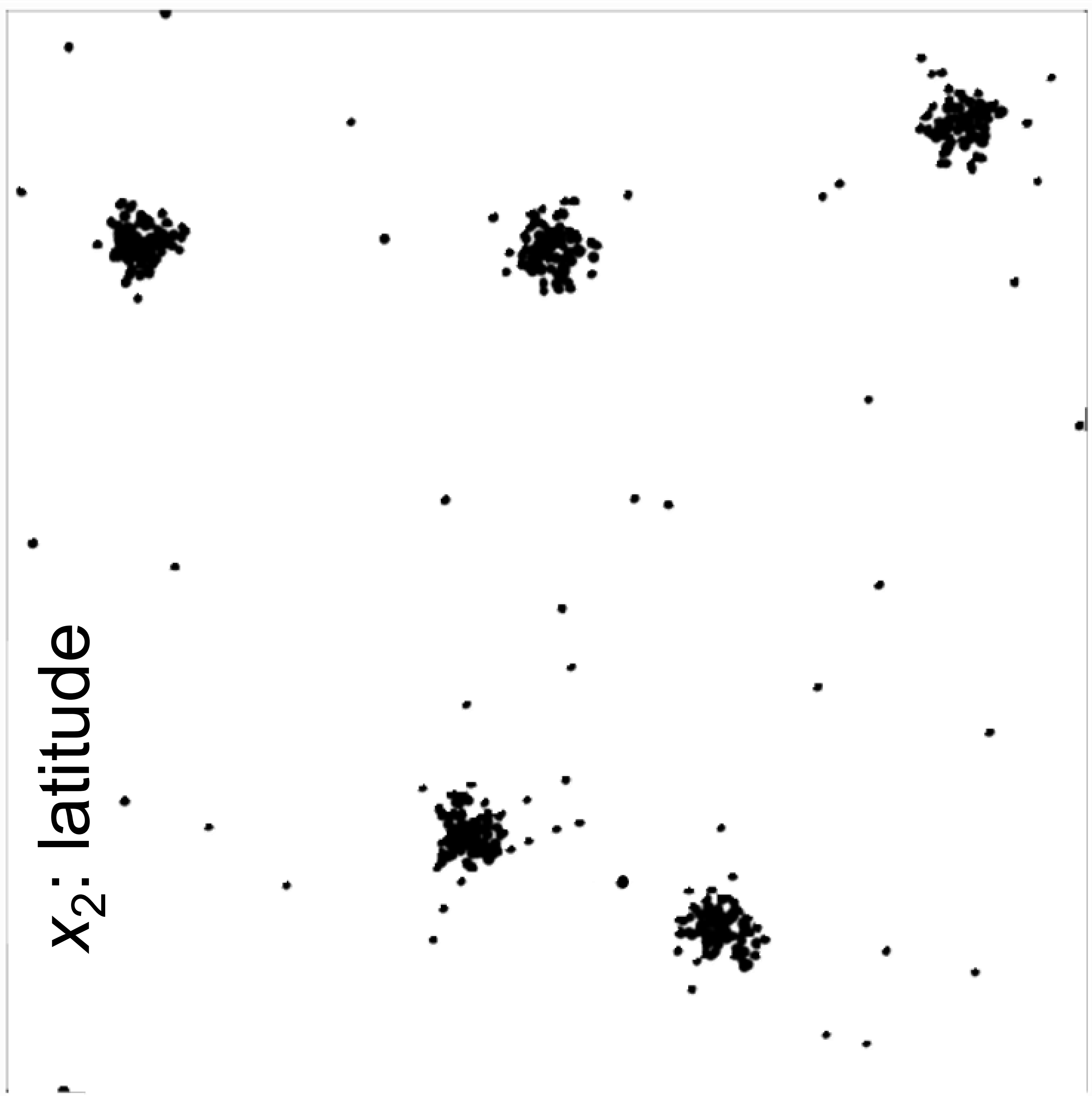
k-means algorithm

`k-means(k, **τ**)`



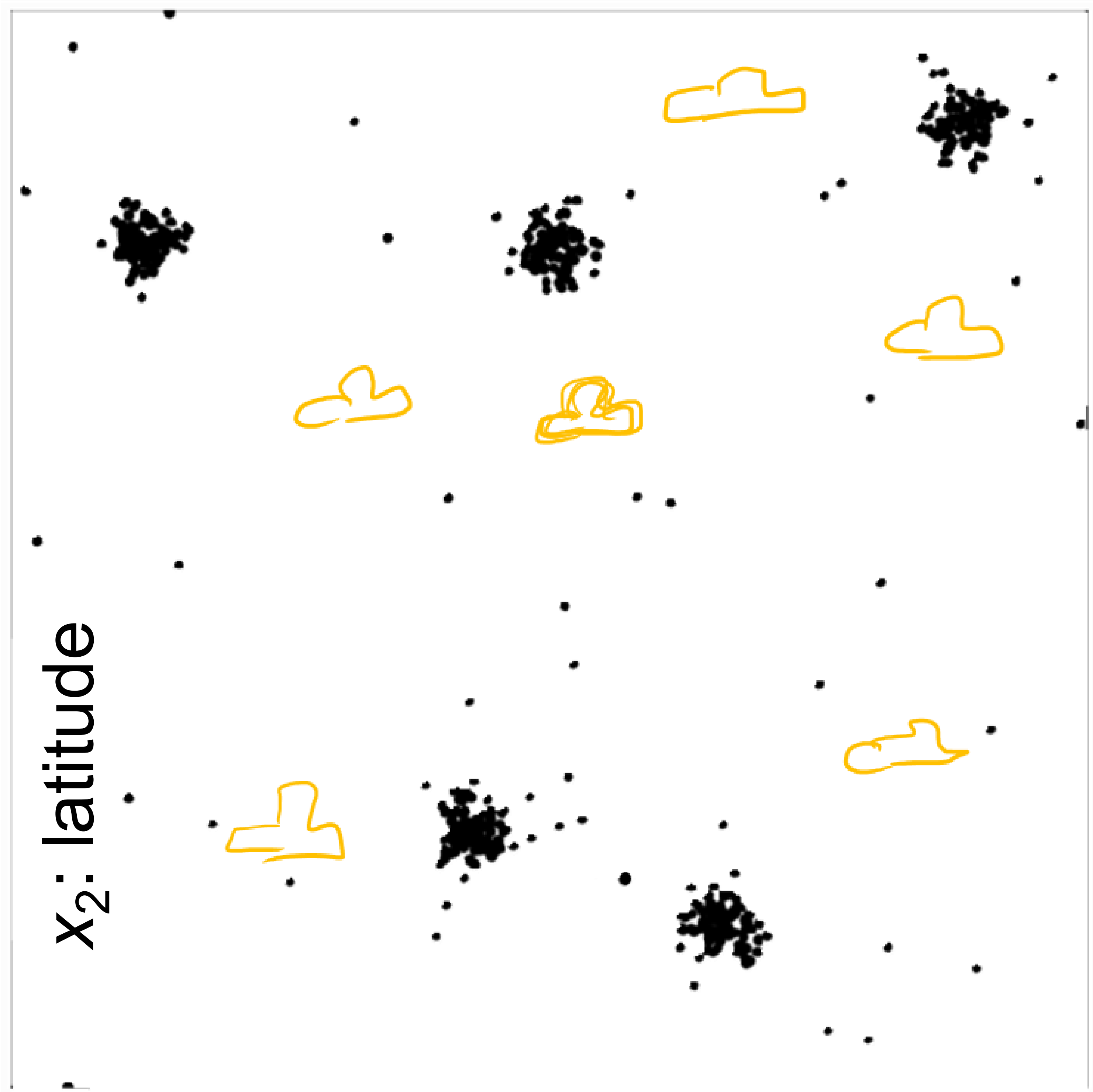
x_1: longitude

k-means algorithm




```
k-means(k, τ)
```

k-means algorithm

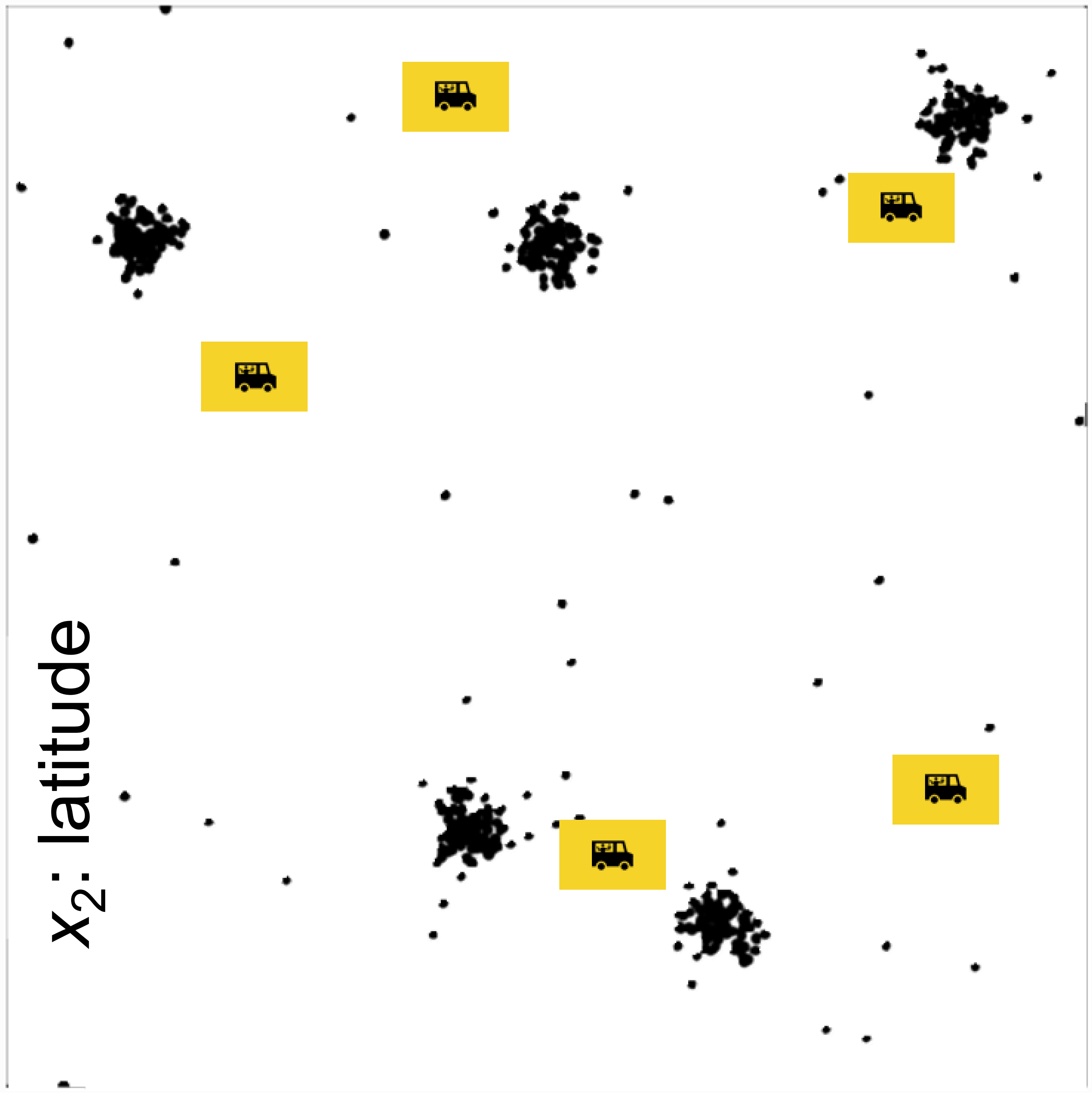


x_1: longitude

`k-means(k,` $\boldsymbol{\tau}$`)`

Init $\{\mu^{(j)}\}_{j=1}^{k}$

k-means algorithm

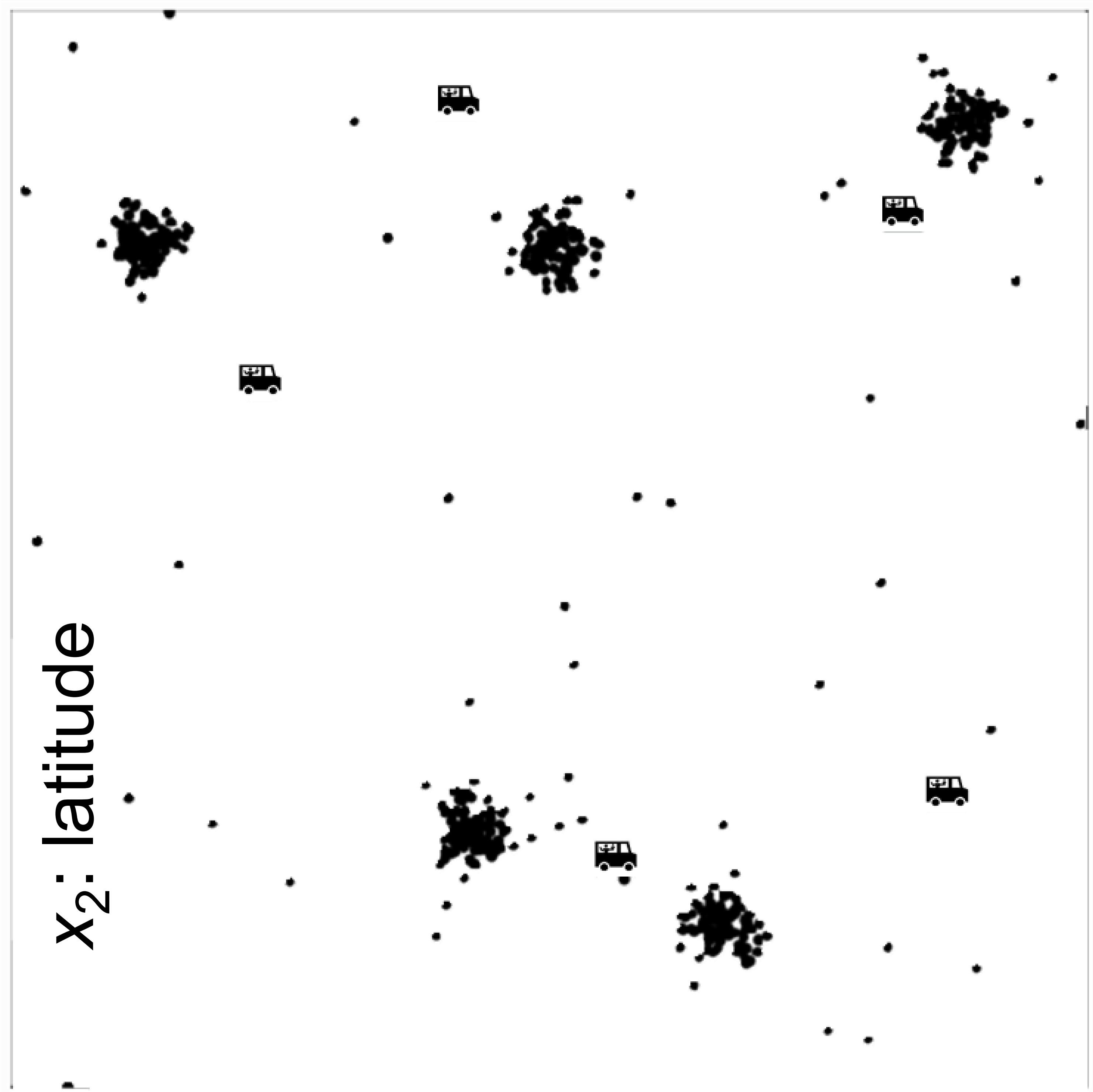


x_1: longitude

`k-means(k, τ)`

`Init` $\{\mu^{(j)}\}_{j=1}^{k}$

k-means algorithm



x_1: longitude

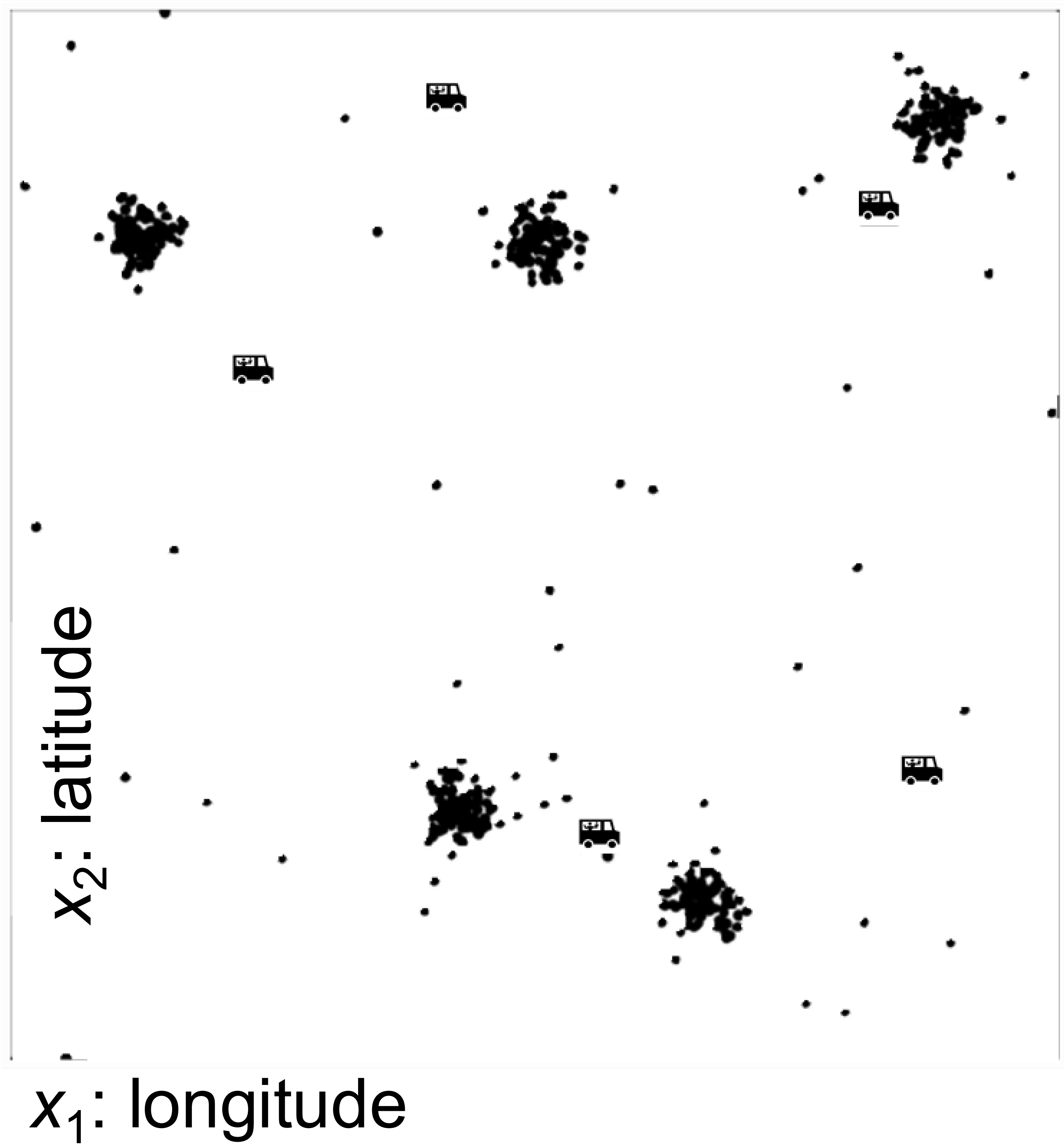
k-means(k, $\boldsymbol{\tau}$)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to $\boldsymbol{\tau}$

τ → max_iter

k-means algorithm

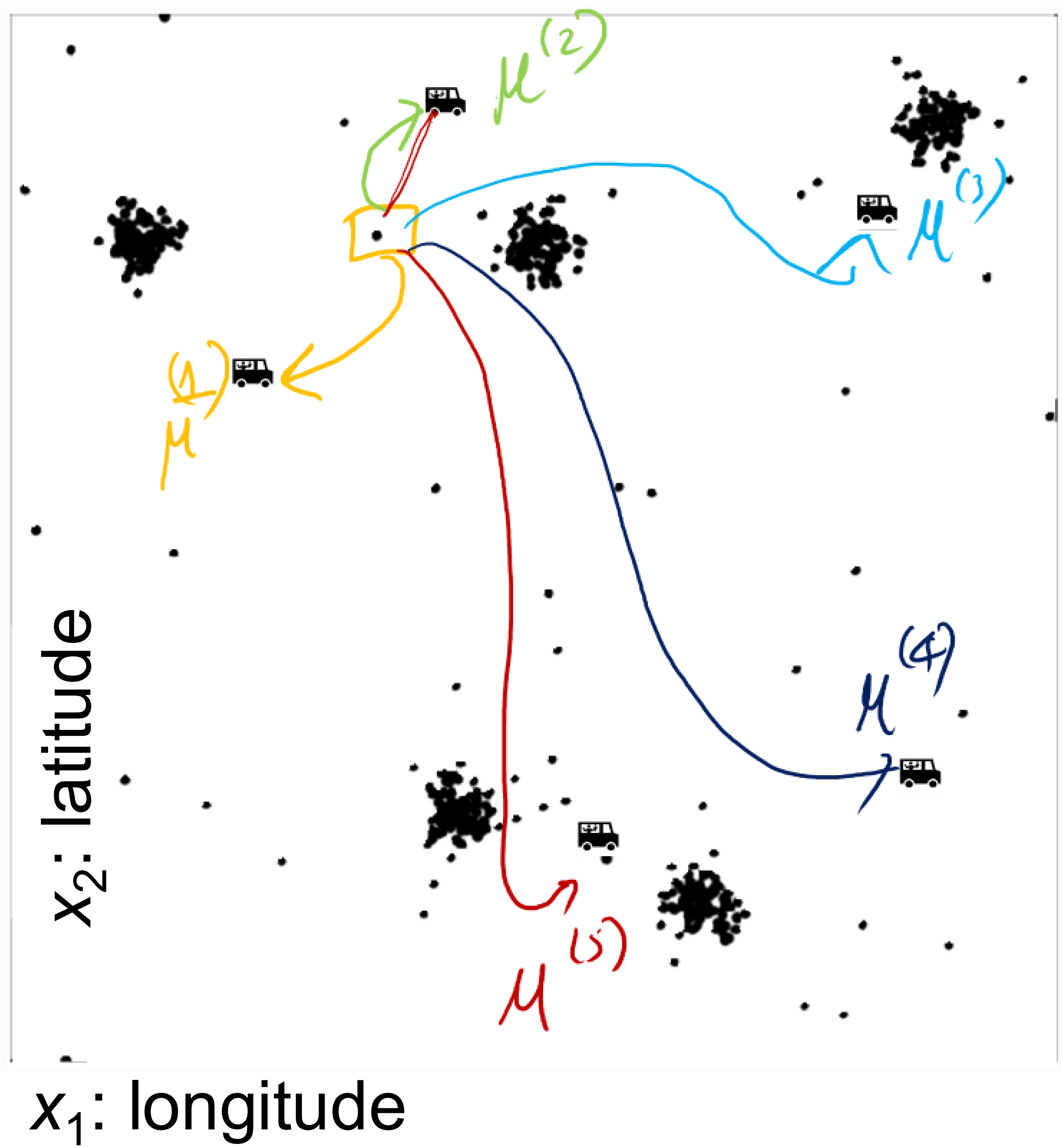



```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k
  for   t = 1 to τ

      for i = 1 to n
```

max iters

total no of people

k-means algorithm



$\min_x (x-5)^2$

$\arg\min_x (x-5)^2$

what is the val of x for which I get a min

k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for $t = 1$ to τ

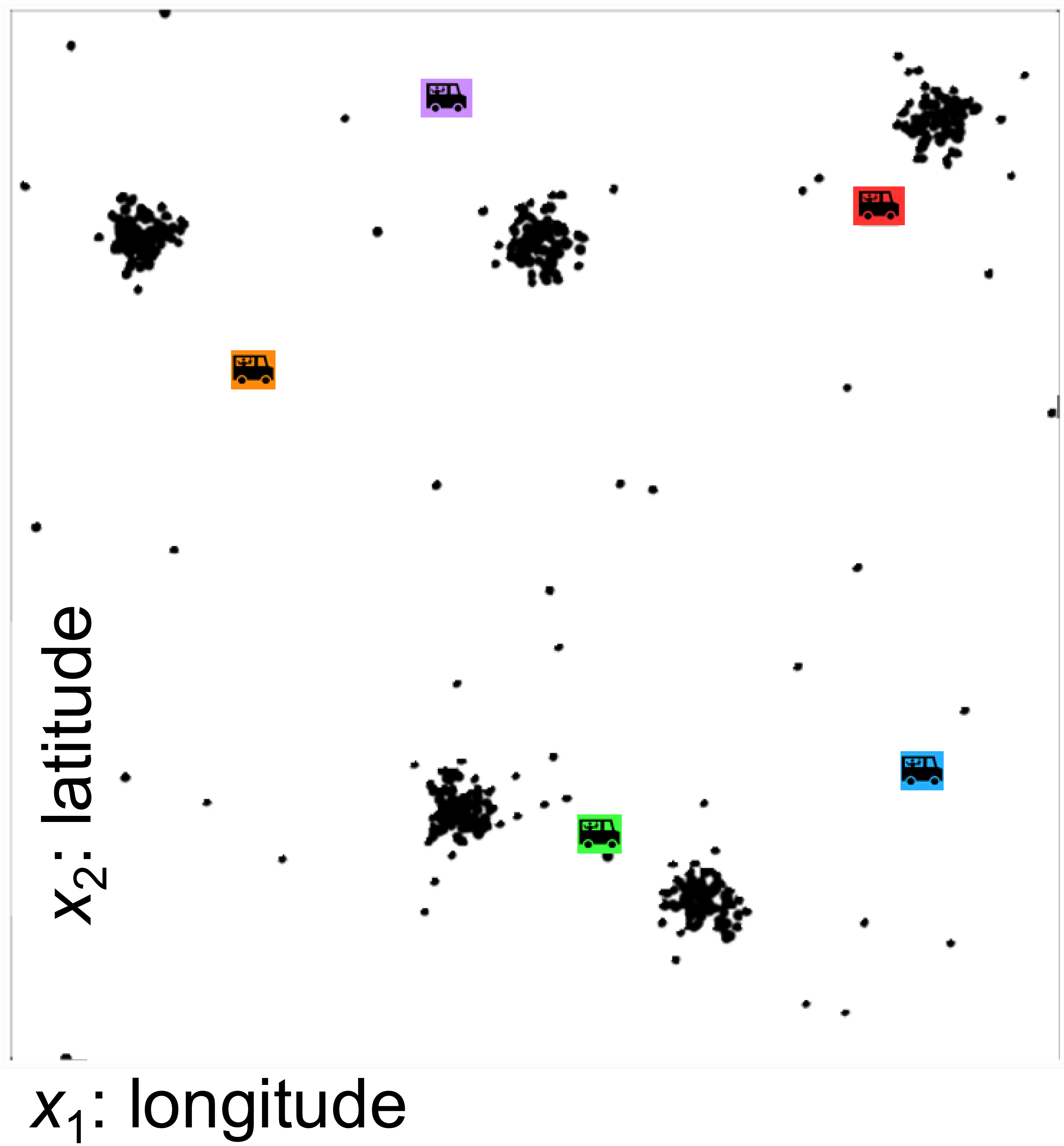
for $i = 1$ to n

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

Assignment of truck to the i th data instance

$y^{(i)} =$ 1 or 2 or 3 or 4 or 5

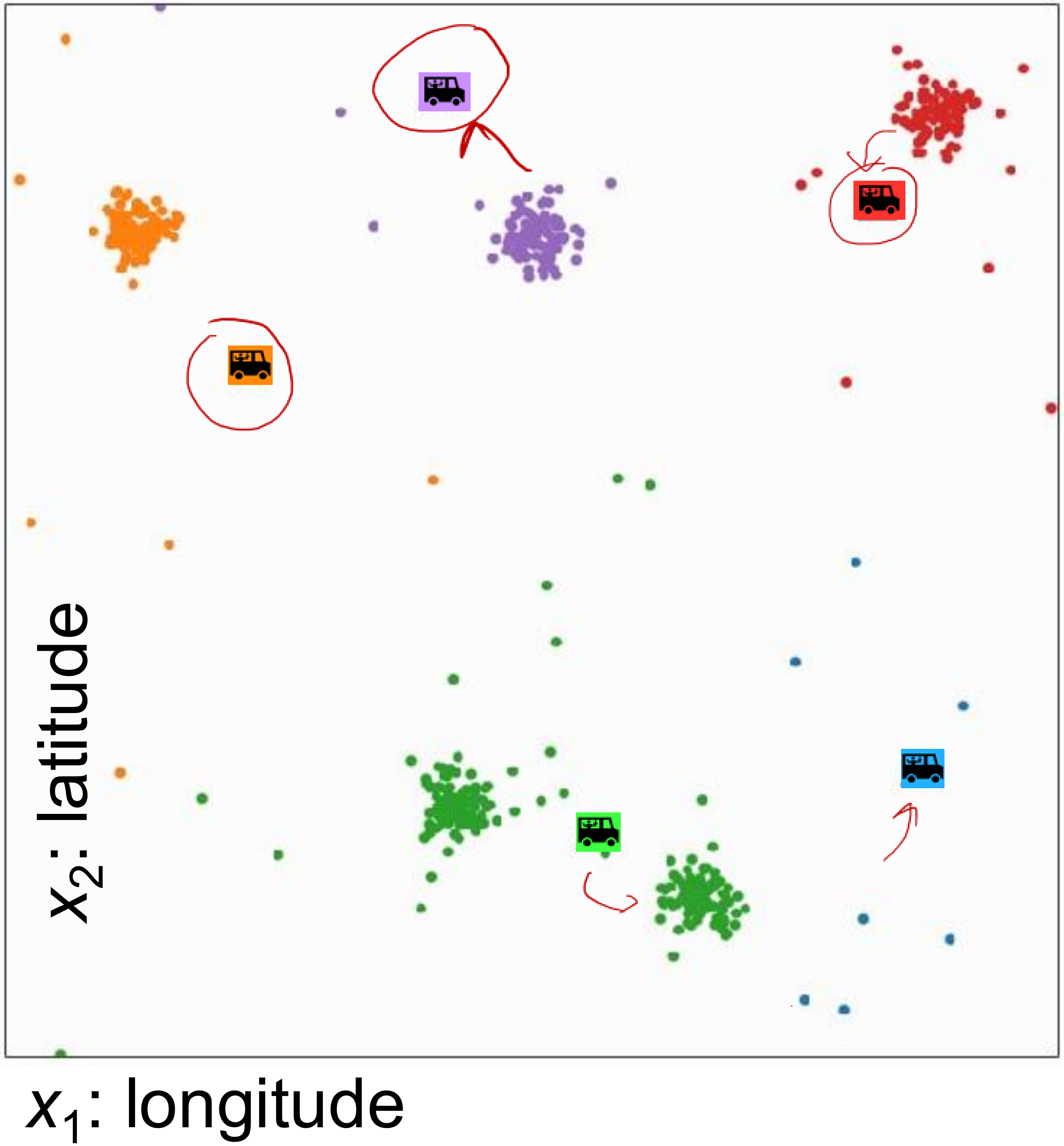
k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for  t = 1 to τ

   for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

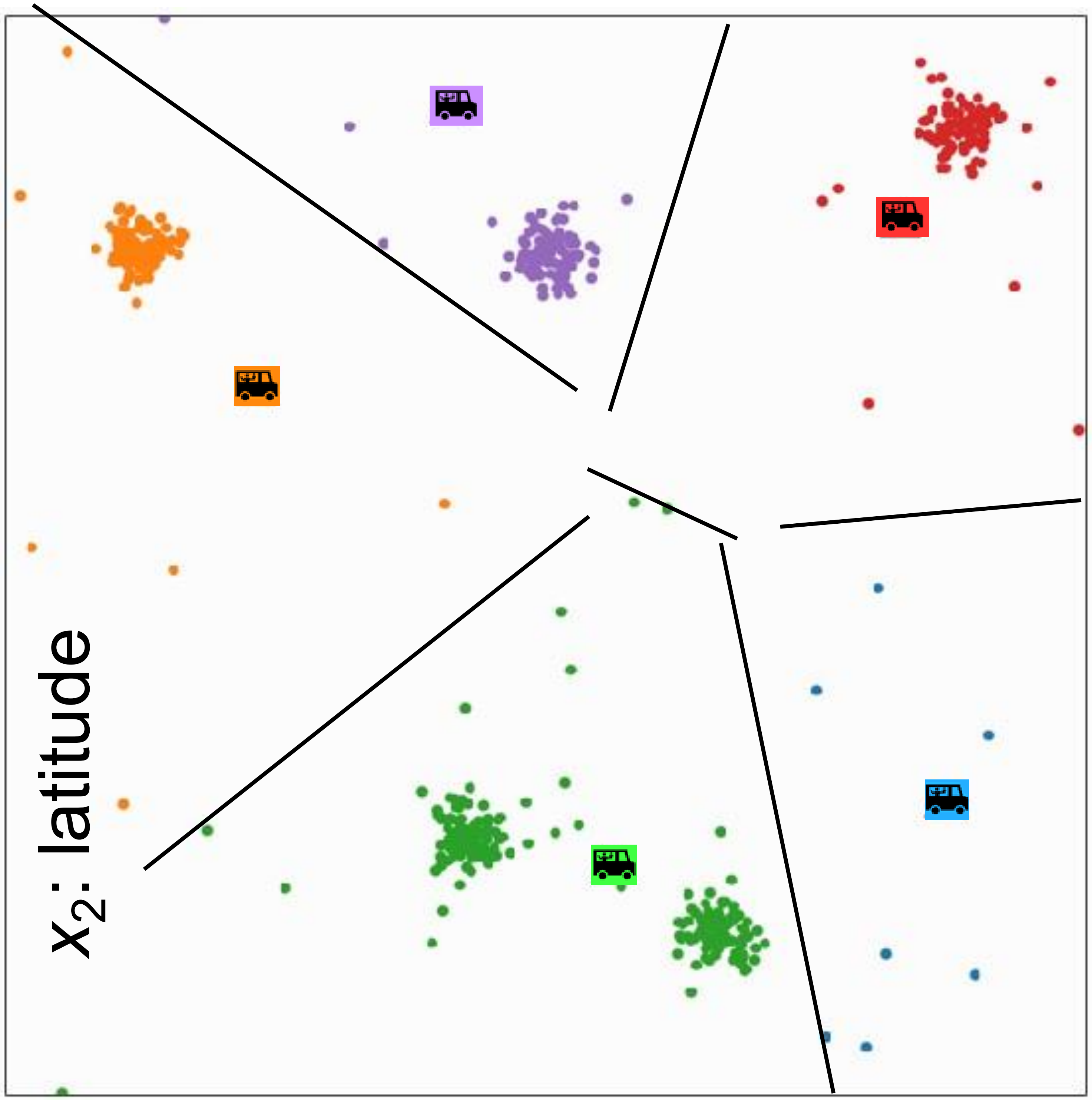
k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for    t = 1 to τ

   for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

k-means algorithm



k-means(k, τ)

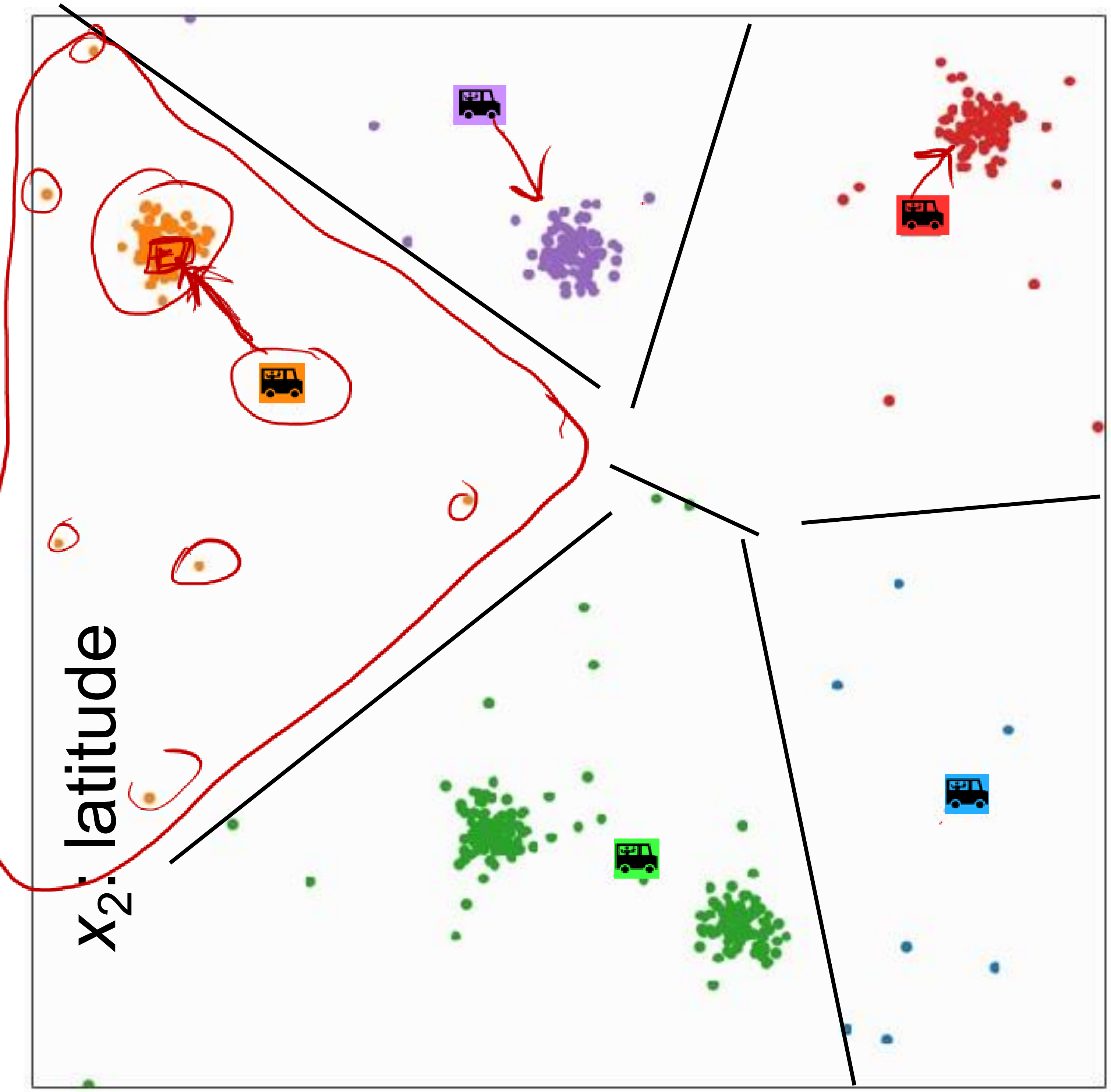
Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for   t = 1 to τ

   for i = 1 to n
```

assignment is done

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

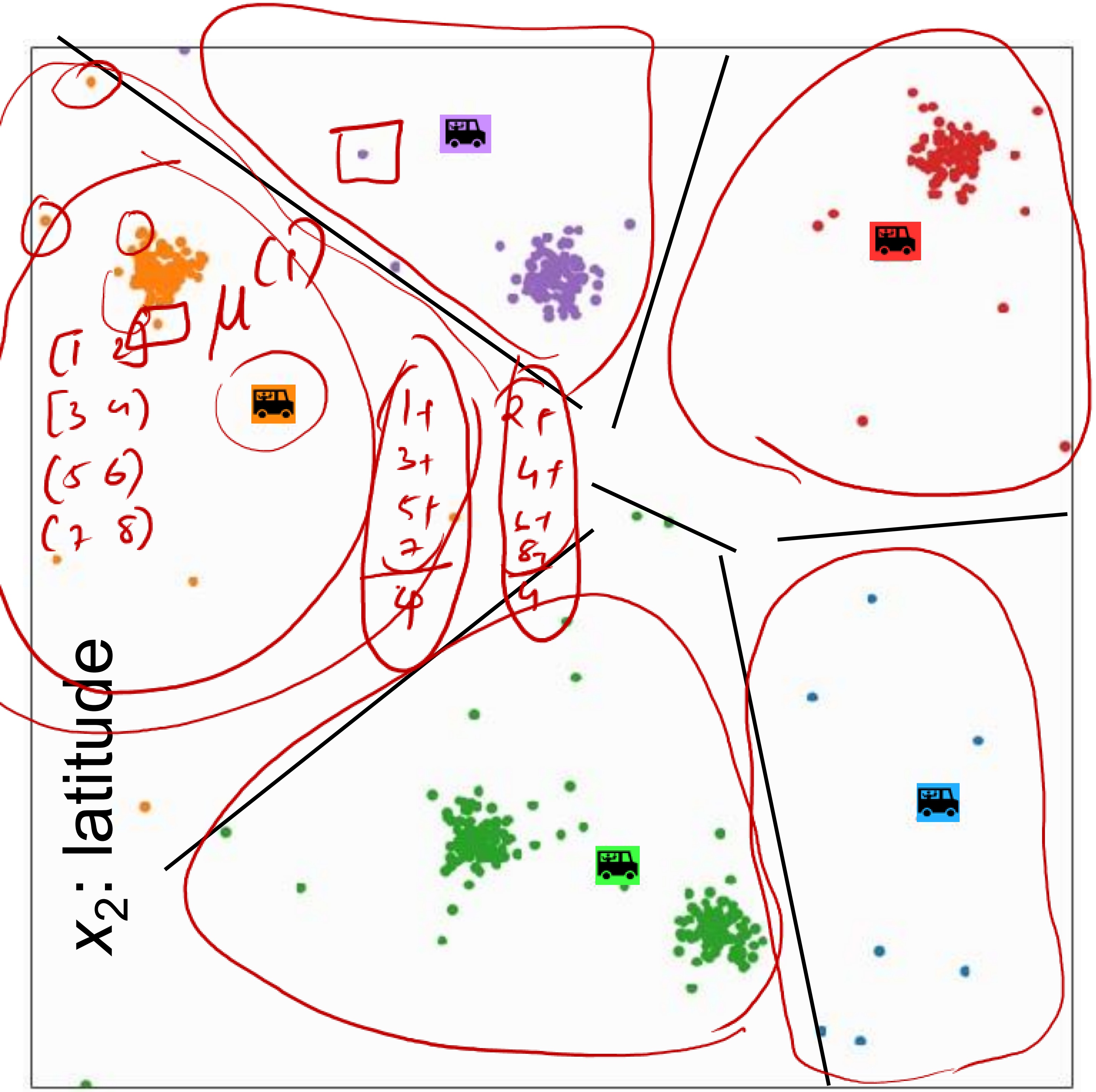
```
   for j = 1 to k
```

update the position of truck

k-means algorithm

3, 4, 5

$\frac{3+4+5}{3}$



x_1: longitude

```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k
  for   t = 1 to τ

     for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

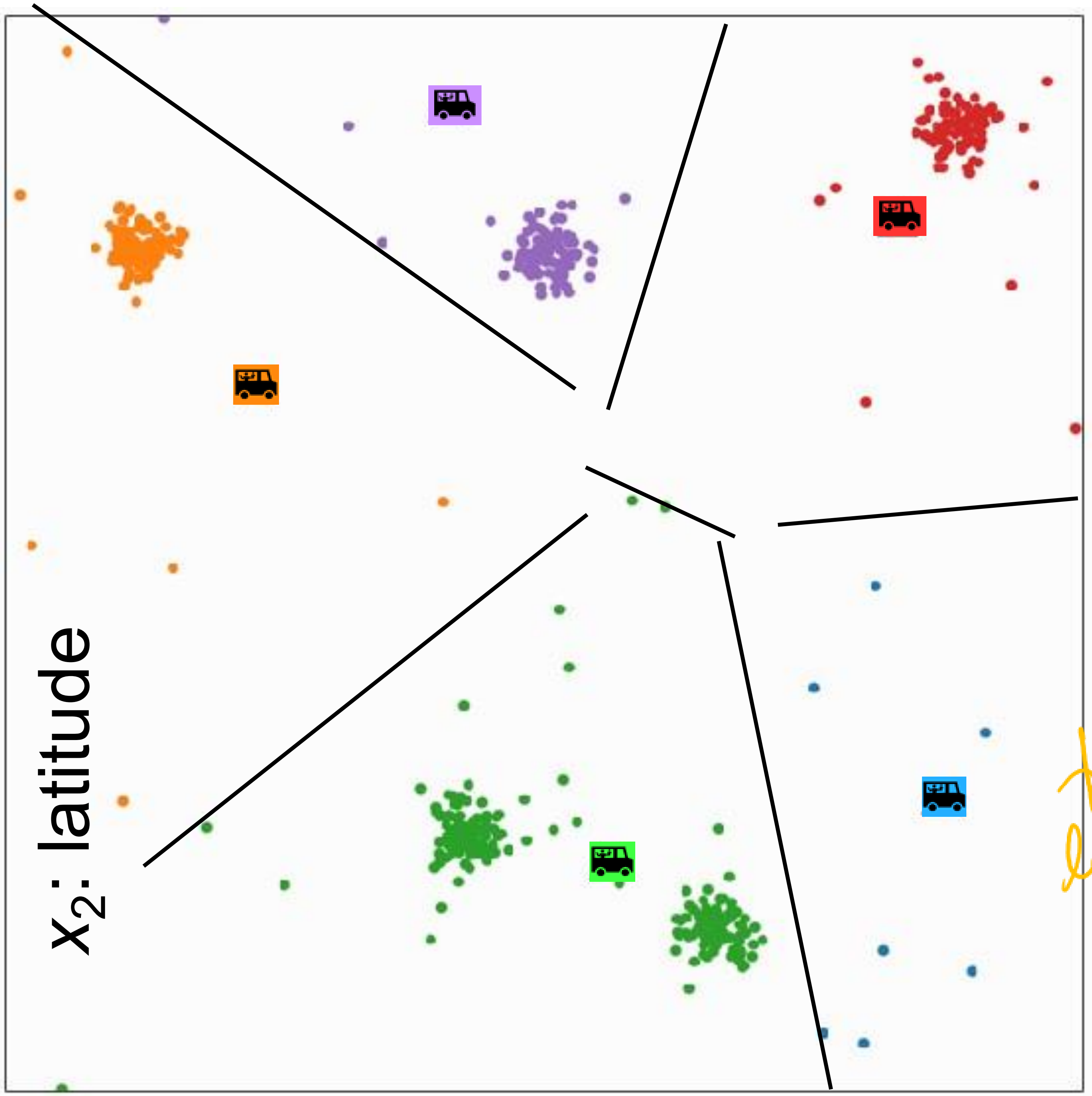
```
     for j = 1 to k
```

$$\mu^{(j)} = \frac{1}{\;}\frac{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\}}$$

$\mathbf{1}\{y^{(i)} = j\}$

if $y^{(i)} = j$ then 1
else 0

k-means algorithm



x_1: longitude

k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

for i = 1 to n

assignment updation

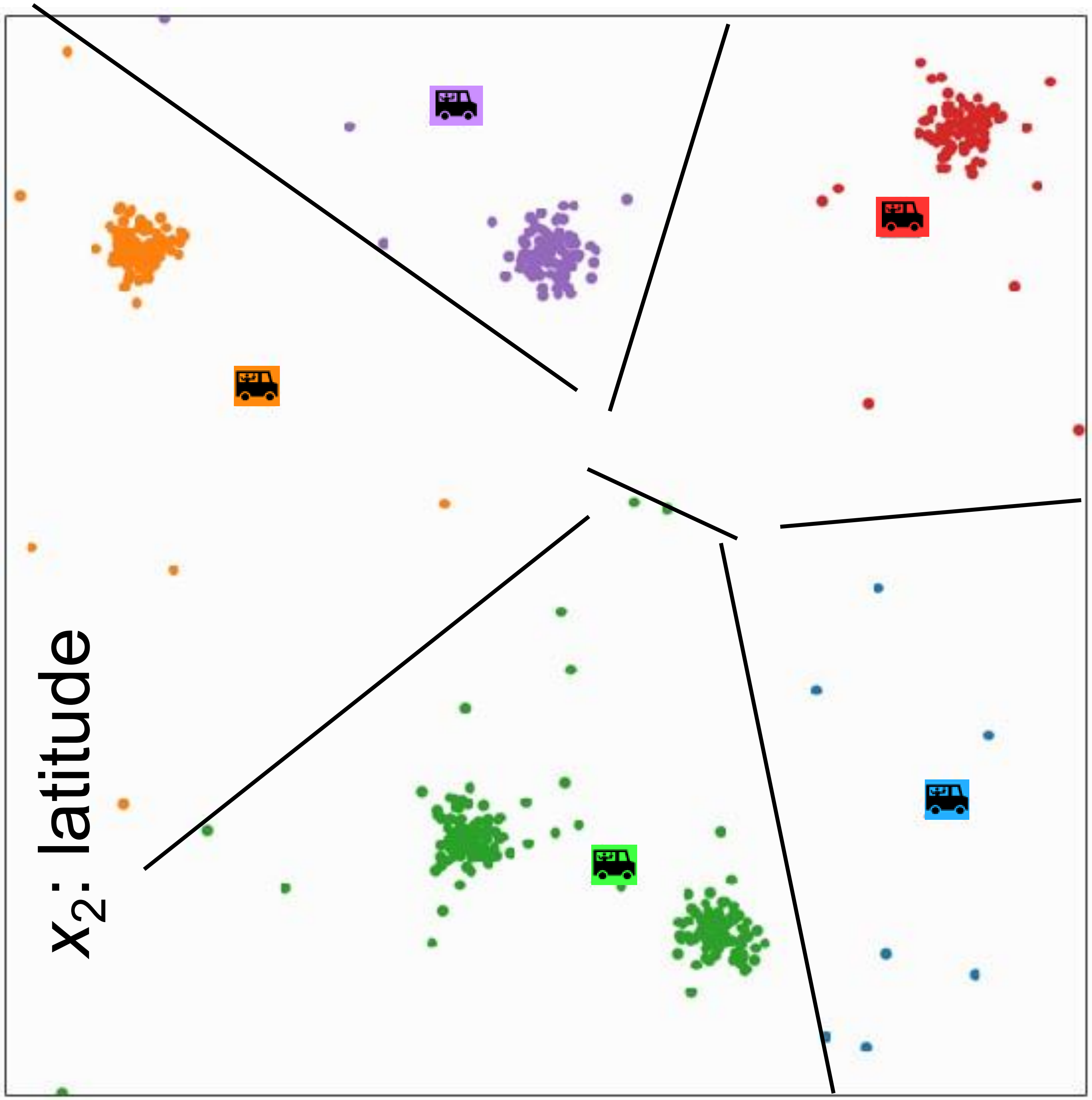
$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

truck location updation

$\boxed{\mu^{(j)}} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\}}$

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

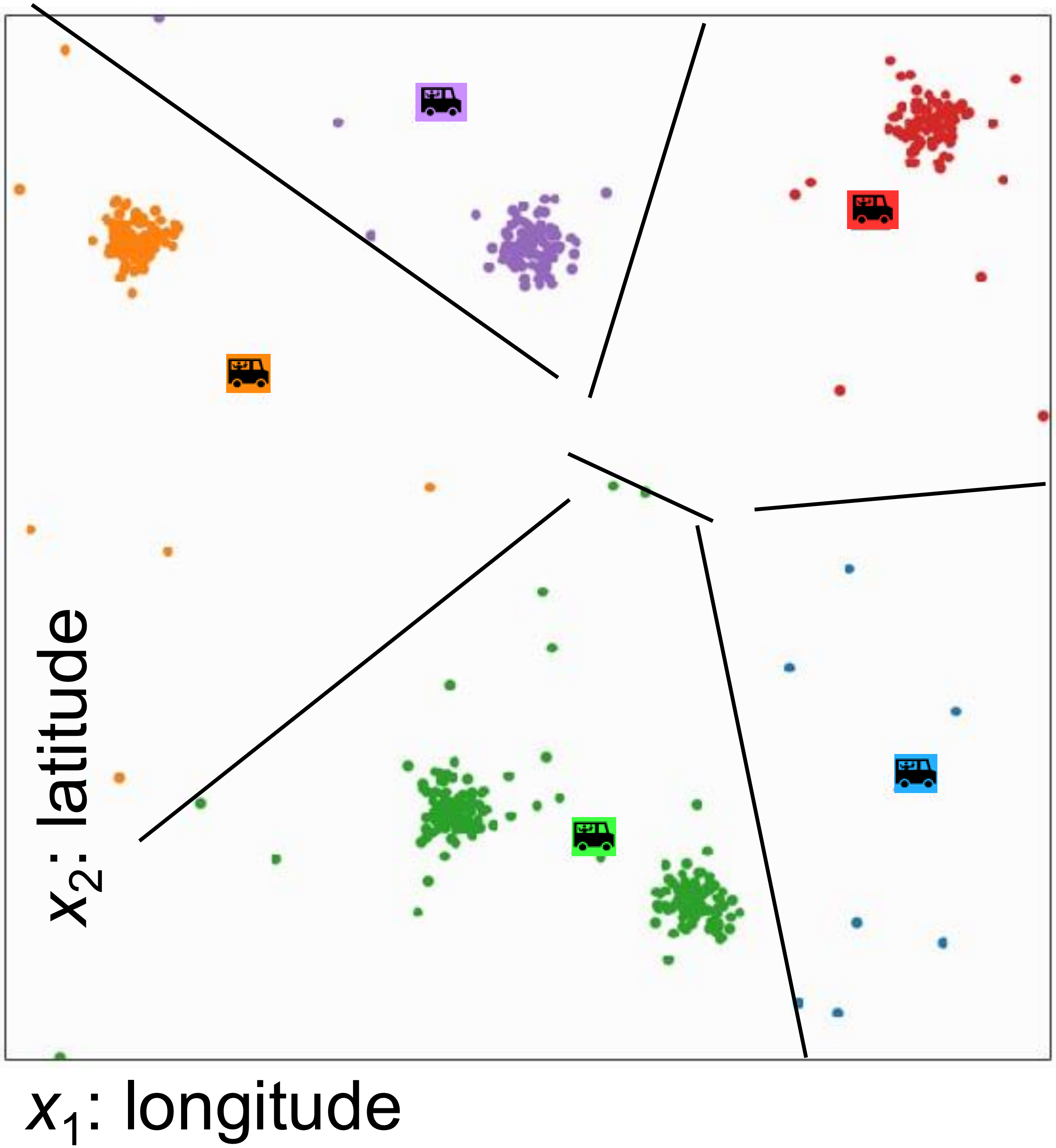
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

$$\mu^{(j)} = \frac{\boxed{\sum_{i=1}^{n}} \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\}}$$

k-means algorithm



k-means(k, $\boldsymbol{\tau}$)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to $\boldsymbol{\tau}$

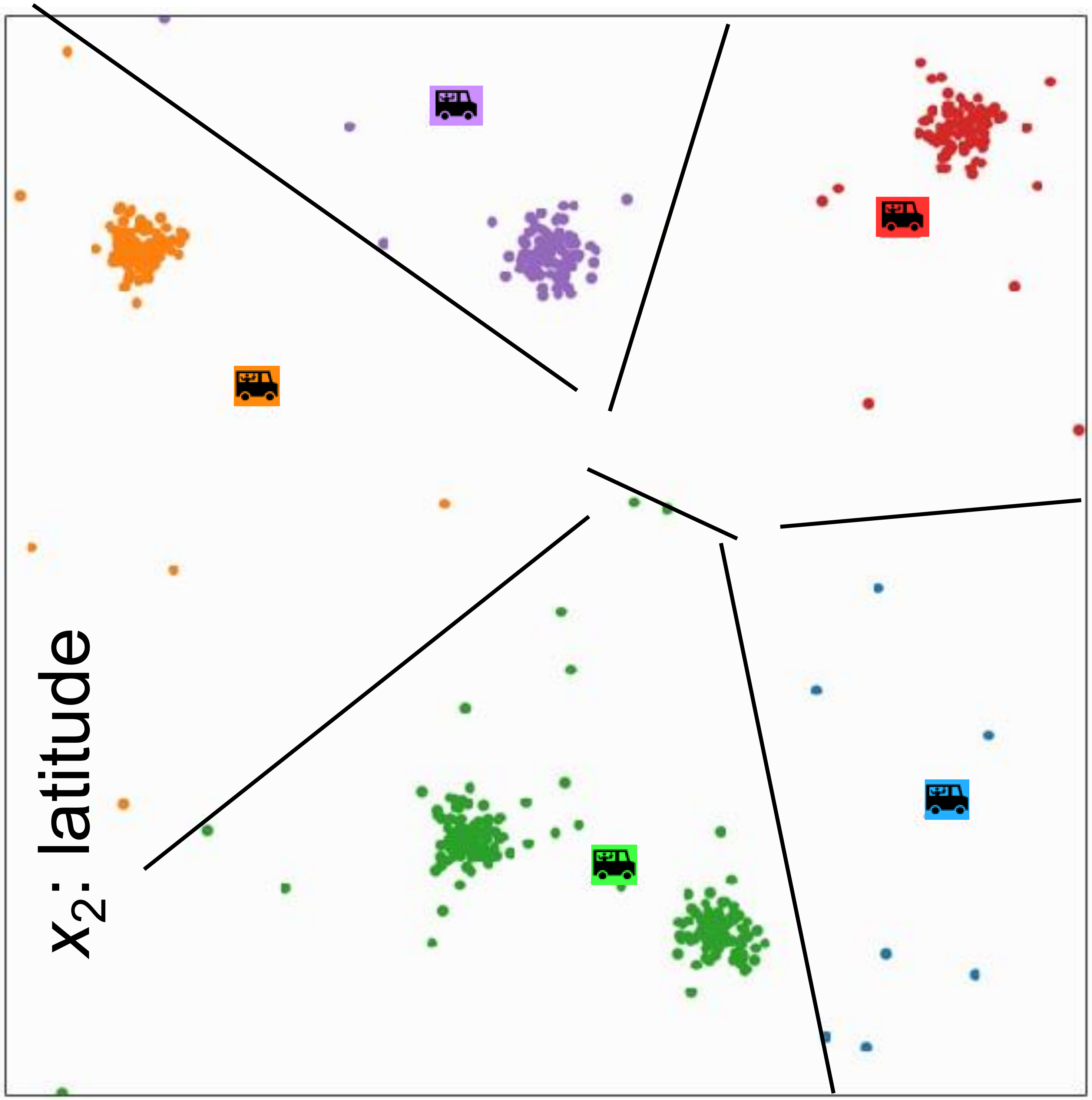
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm



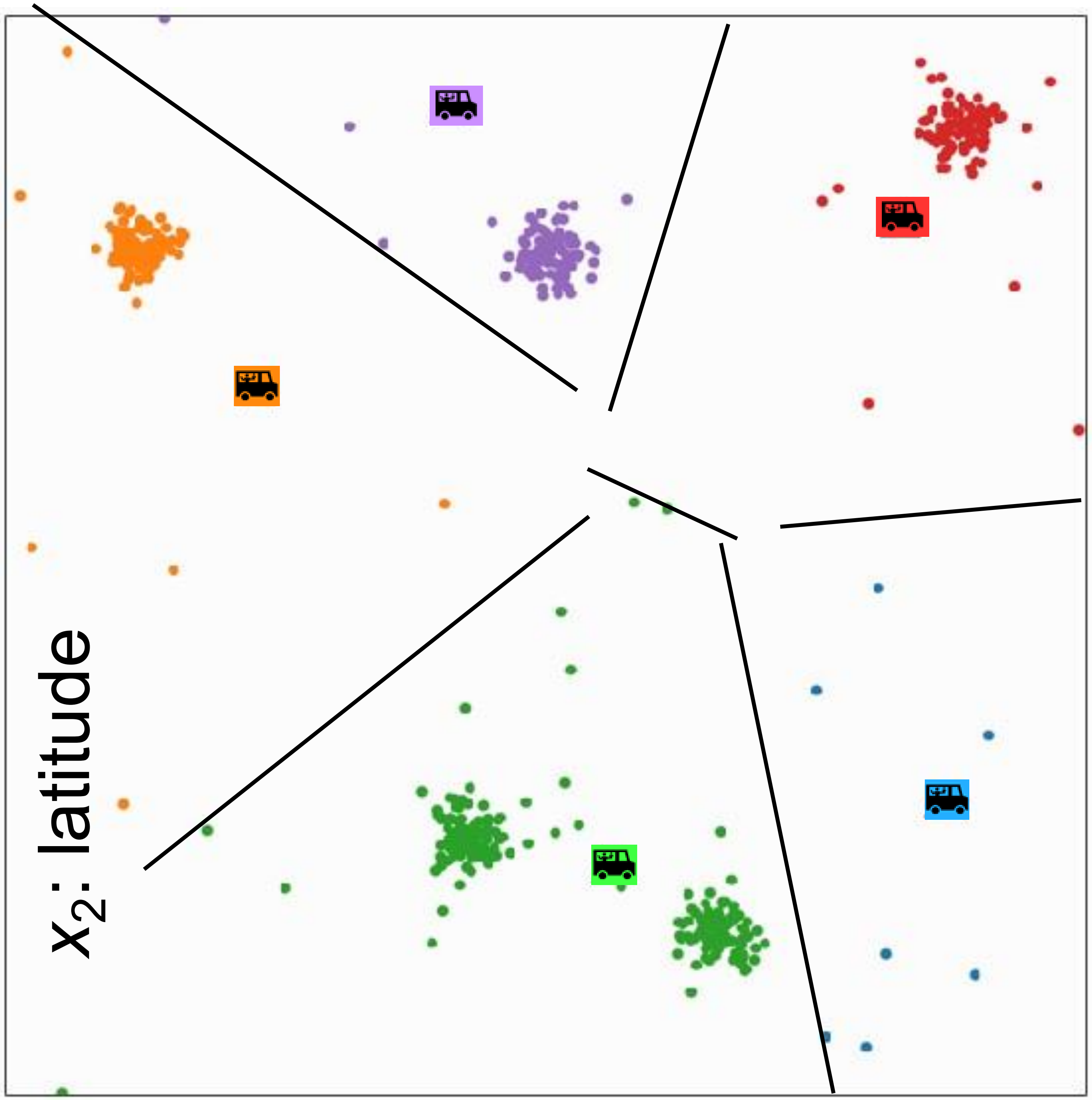

```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for   t = 1 to τ

   for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

```
   for j = 1 to k
```

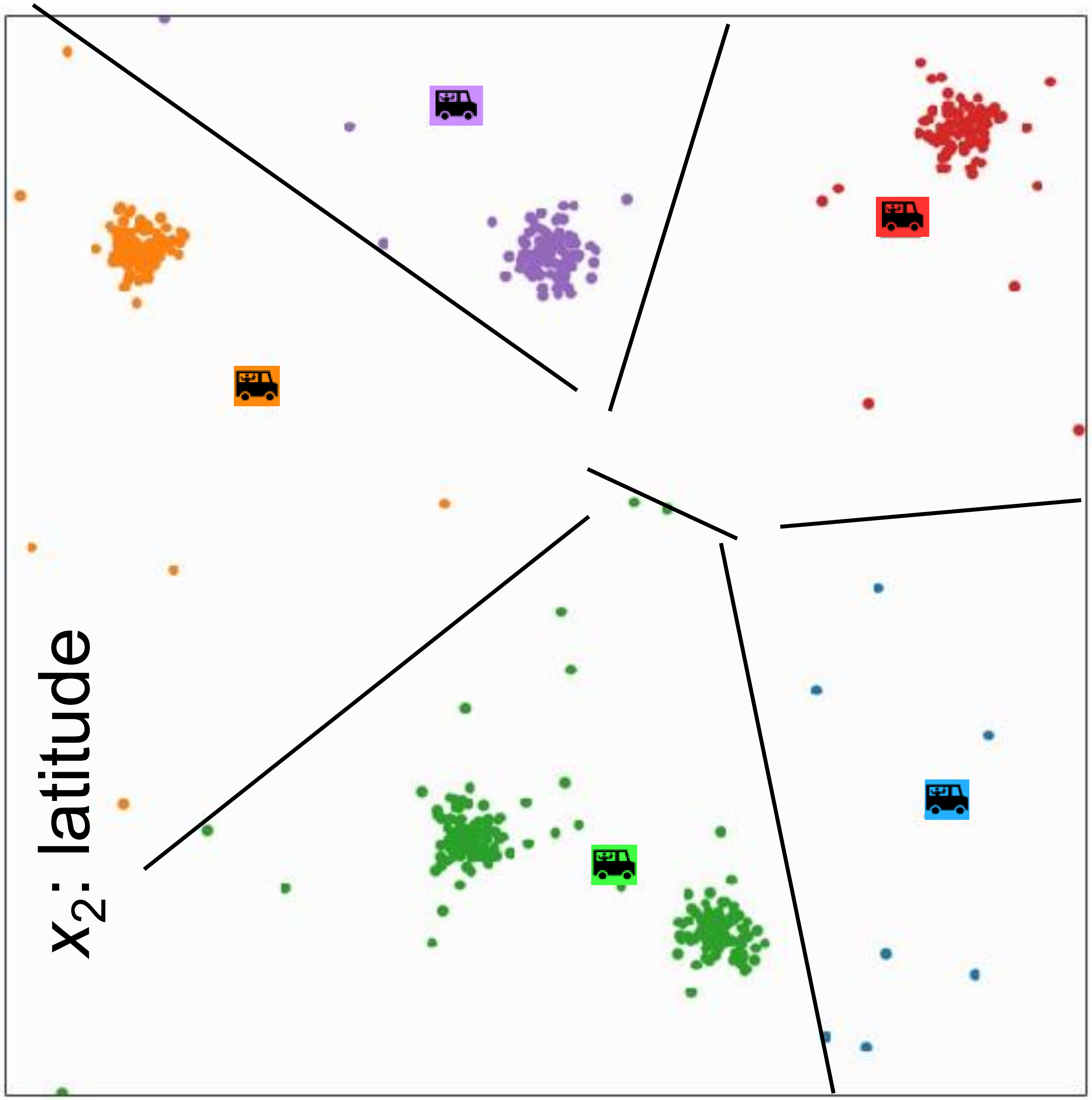
$$\mu^{(j)} = \frac{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\}}$$

k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for   t = 1 to τ

   for i = 1 to n
```
$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$
```
   for j = 1 to k
```
$$\mu^{(j)} = \frac{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\}}$$

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

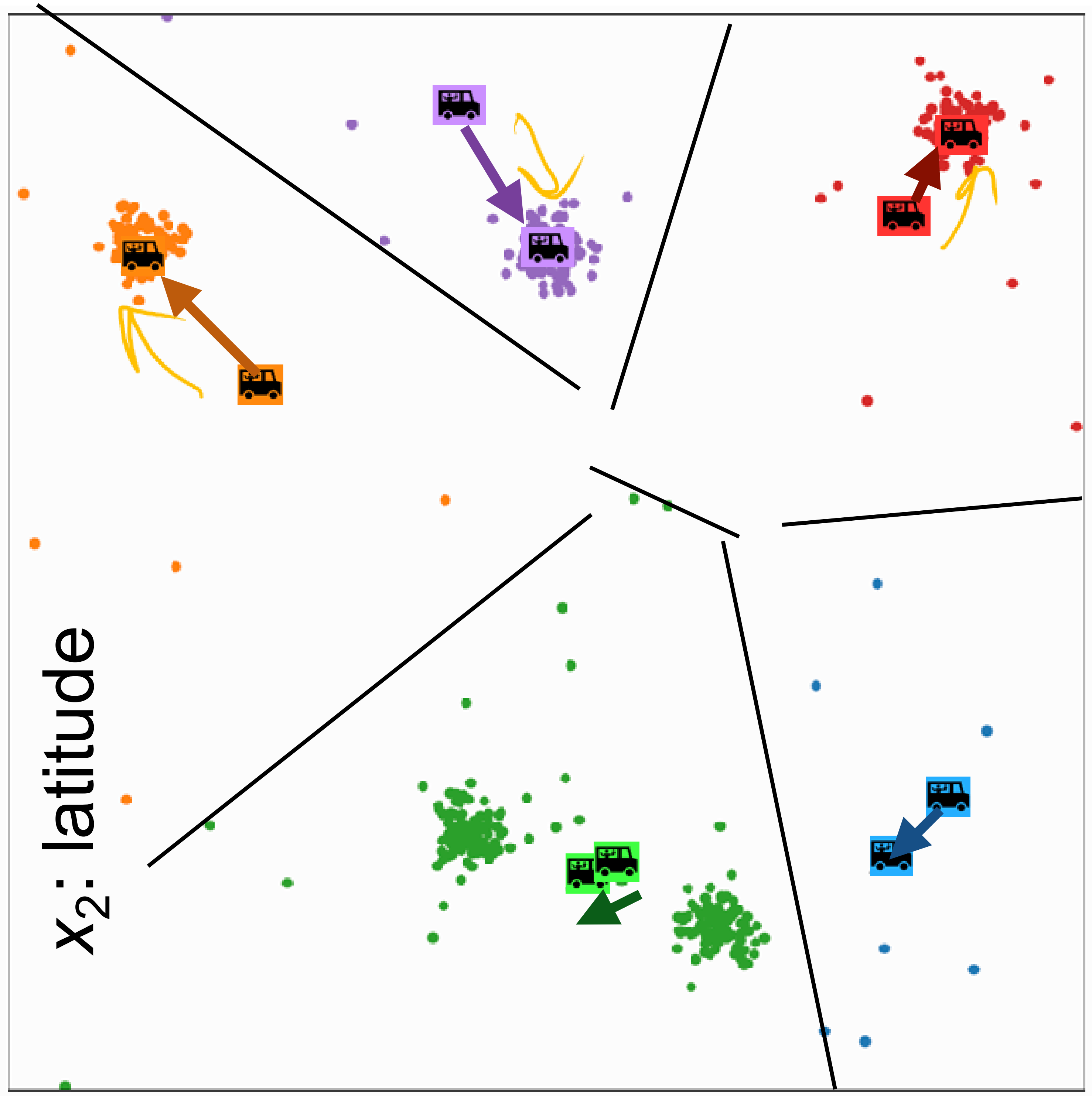
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm



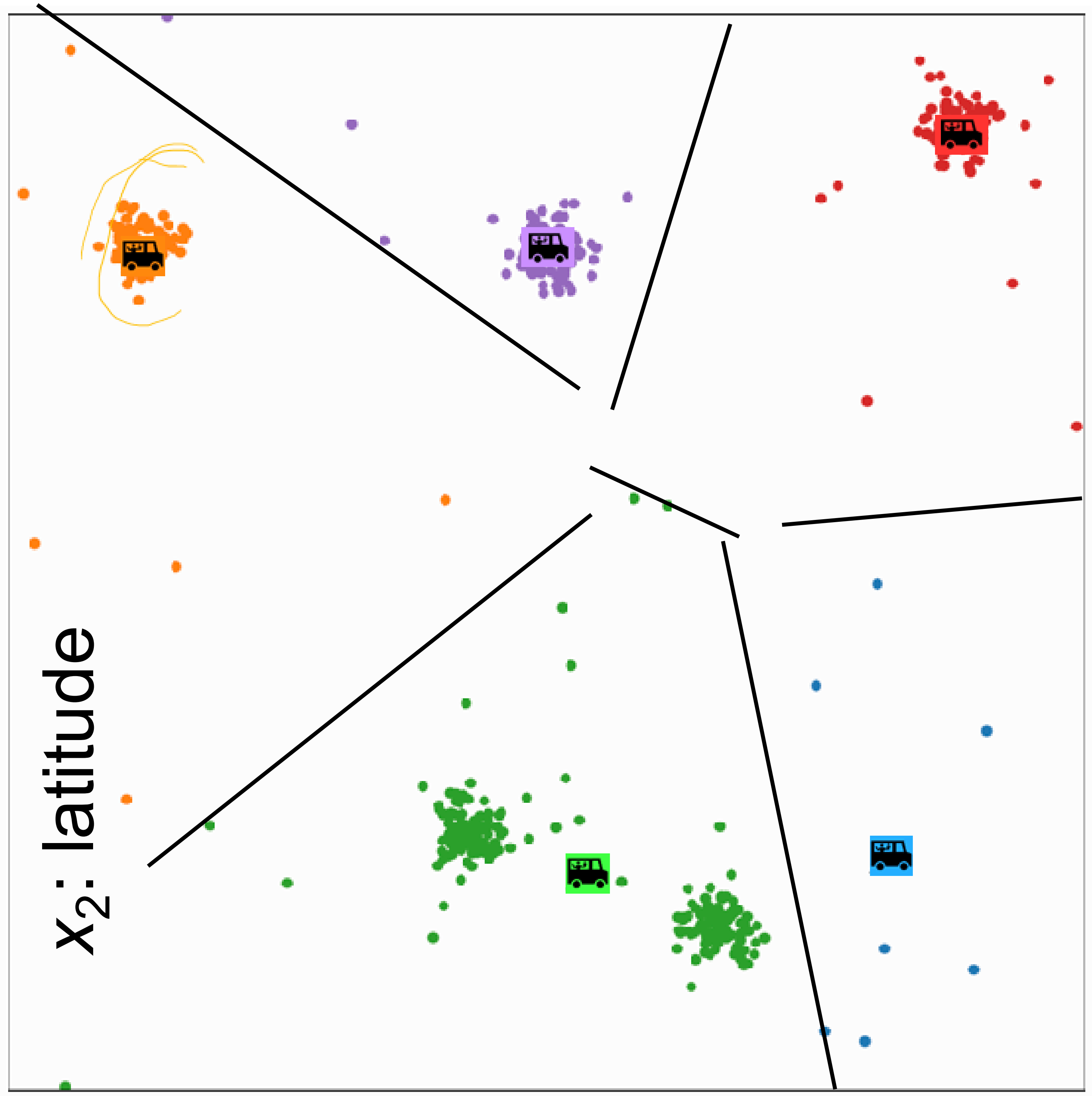

```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k
  for   t = 1 to τ

    for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

```
    for j = 1 to k
```

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

k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for  t = 1 to τ

   for i = 1 to n
     y^(i) = argmin_j ||x^(i) − μ^(j)||_2^2
   for j = 1 to k
     μ^(j) = Σ_{i=1}^n 1{y^(i) = j} x^(i) / Σ_{i=1}^n 1{y^(i) = j}
```

k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for $t = 1$ to τ

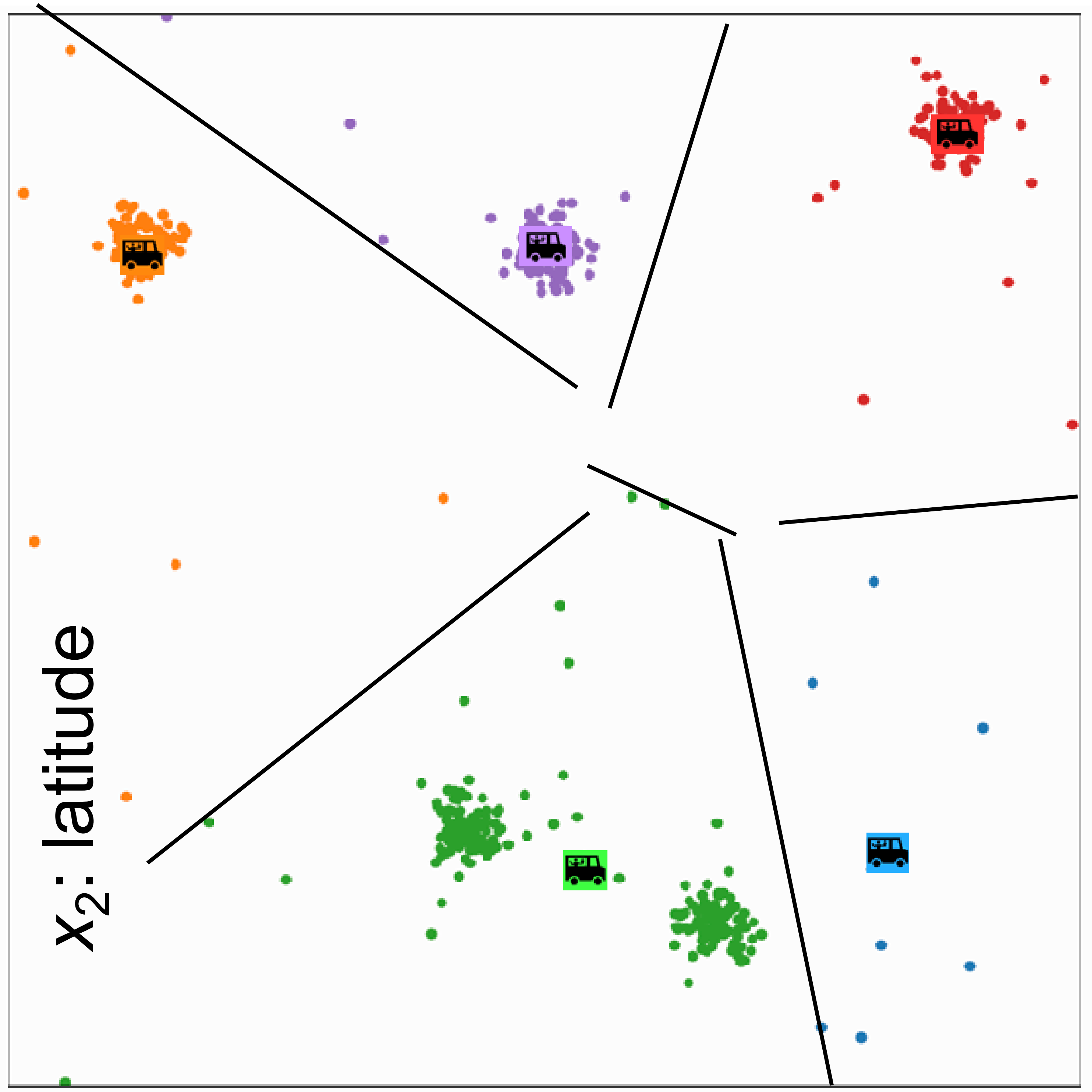
for $i = 1$ to n

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

for $j = 1$ to k

$$\boxed{\mu^{(j)}} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\}}$$

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

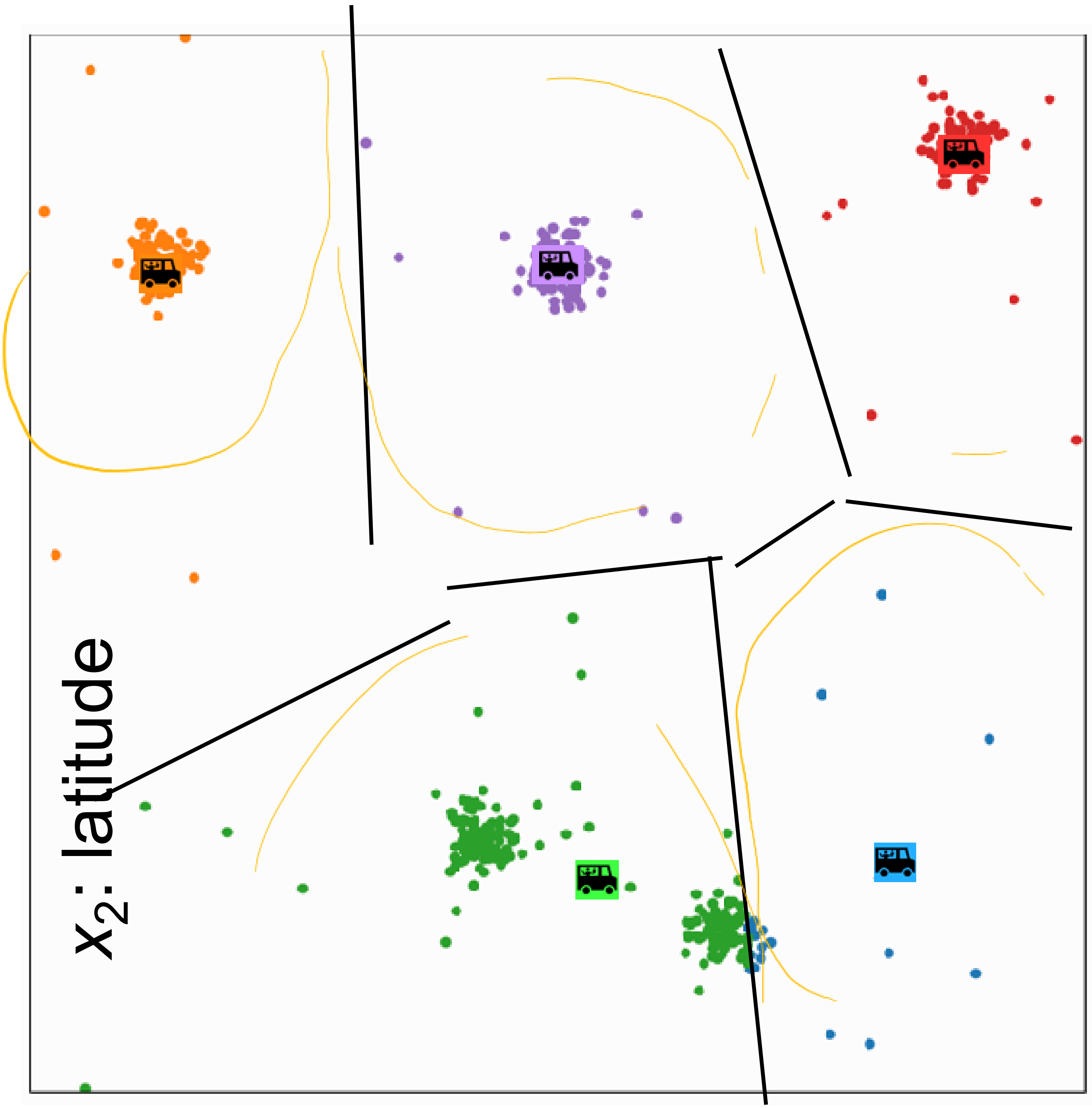
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

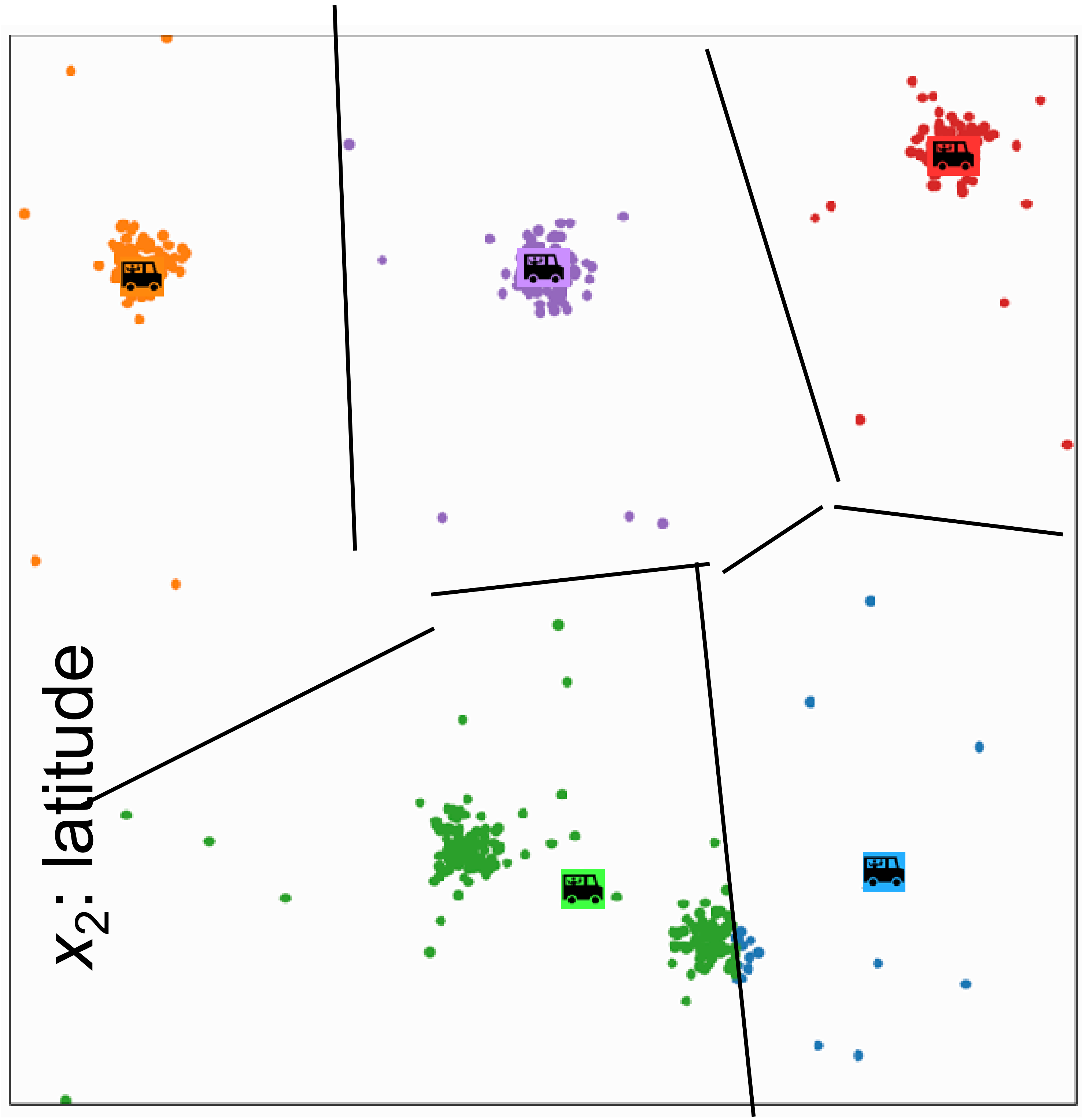
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm



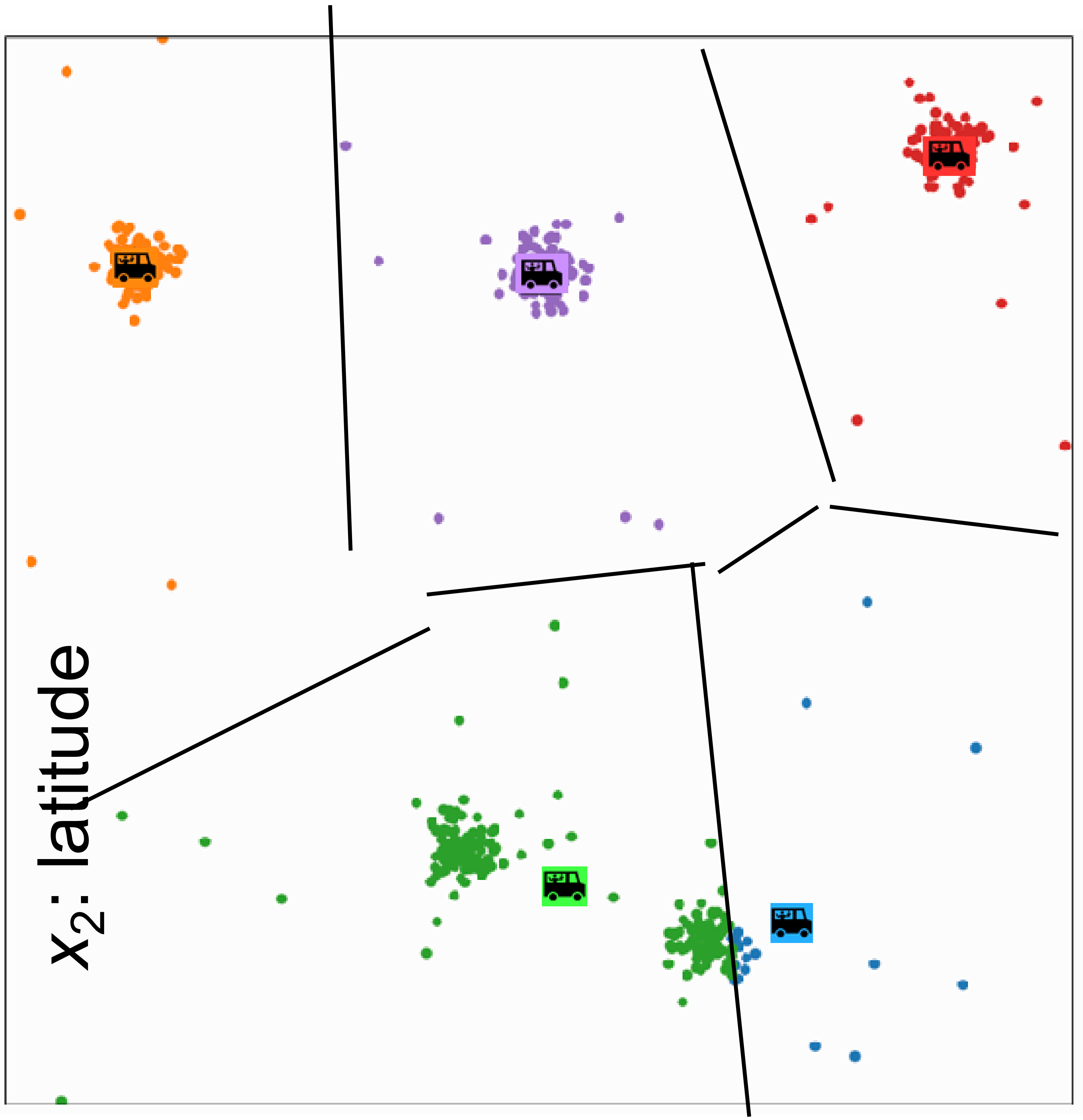

```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k
  for   t = 1 to τ
     for i = 1 to n
       y^(i) = argmin_j ||x^(i) − μ^(j)||_2^2
     for j = 1 to k
       μ^(j) = Σ_{i=1}^n 1{y^(i) = j} x^(i) / Σ_{i=1}^n 1{y^(i) = j}
```

$$\text{Init } \{\mu^{(j)}\}_{j=1}^{k}$$

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

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

k-means algorithm

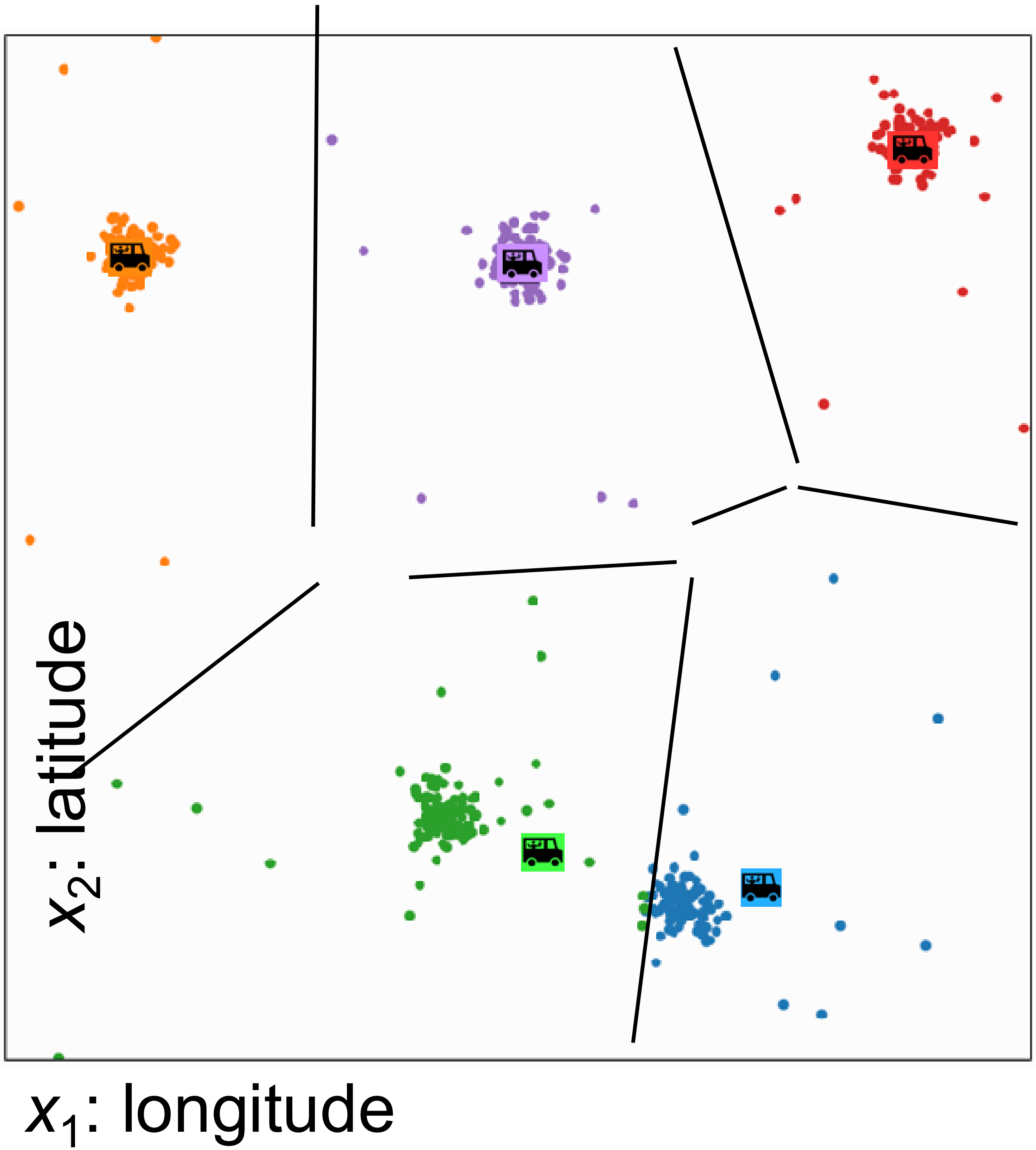



```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for   t = 1 to τ

   for i = 1 to n
     y^(i) = argmin_j ||x^(i) − μ^(j)||_2^2
   for j = 1 to k
     μ^(j) = (Σ_{i=1}^n 1{y^(i) = j} x^(i)) / (Σ_{i=1}^n 1{y^(i) = j})
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

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

k-means algorithm



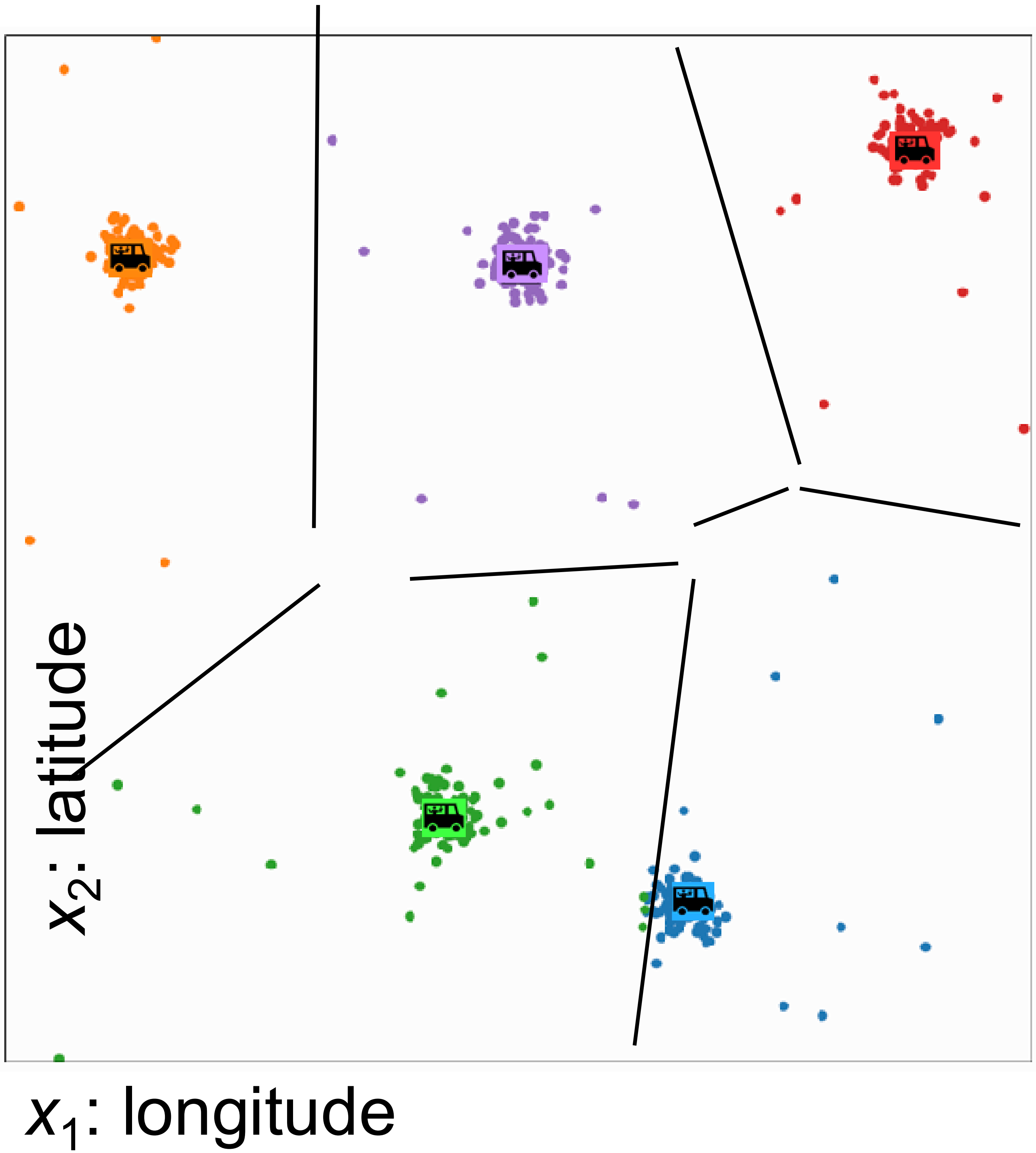

```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for   t = 1 to τ

    for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

```
    for j = 1 to k
```

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

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

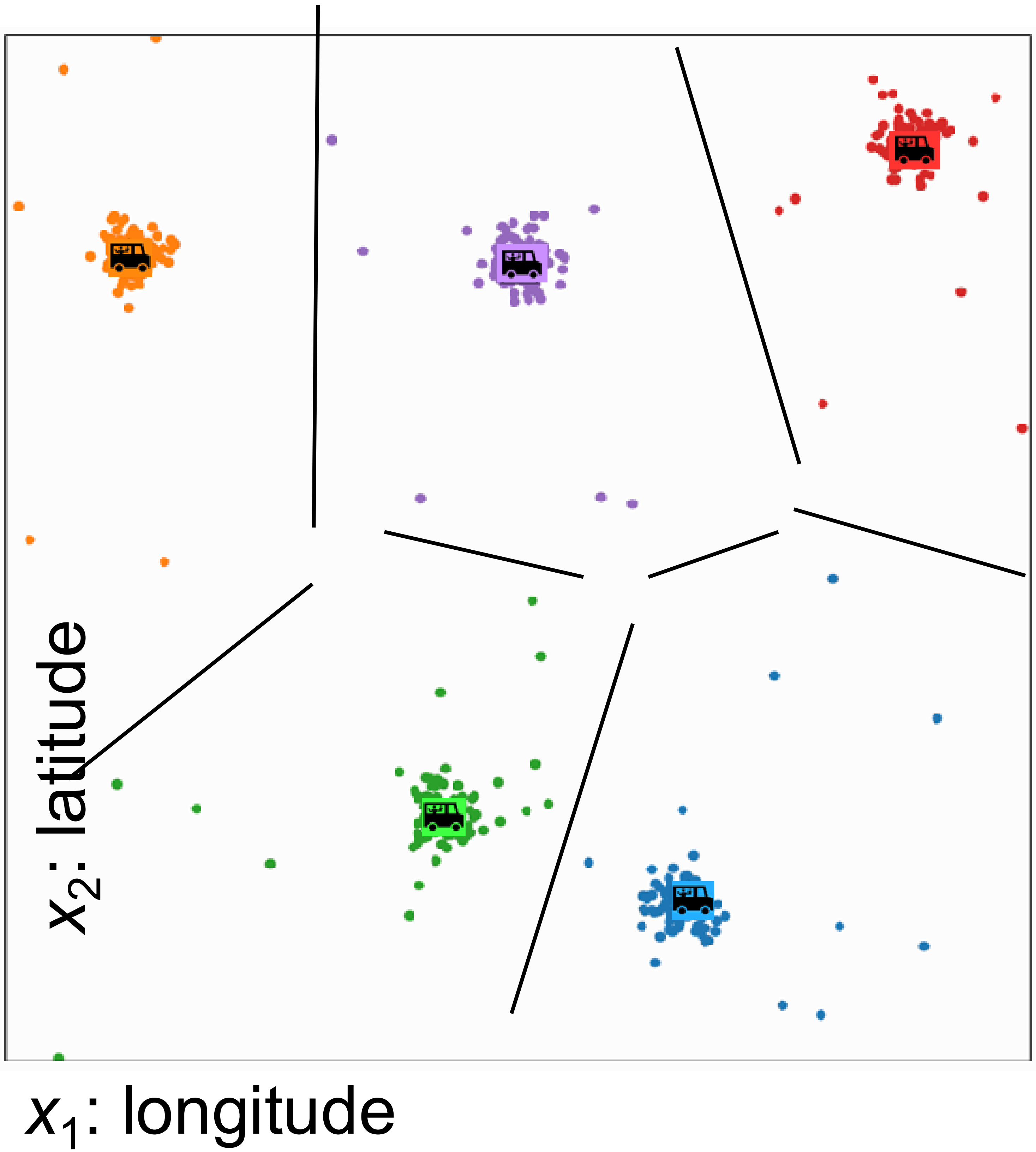
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

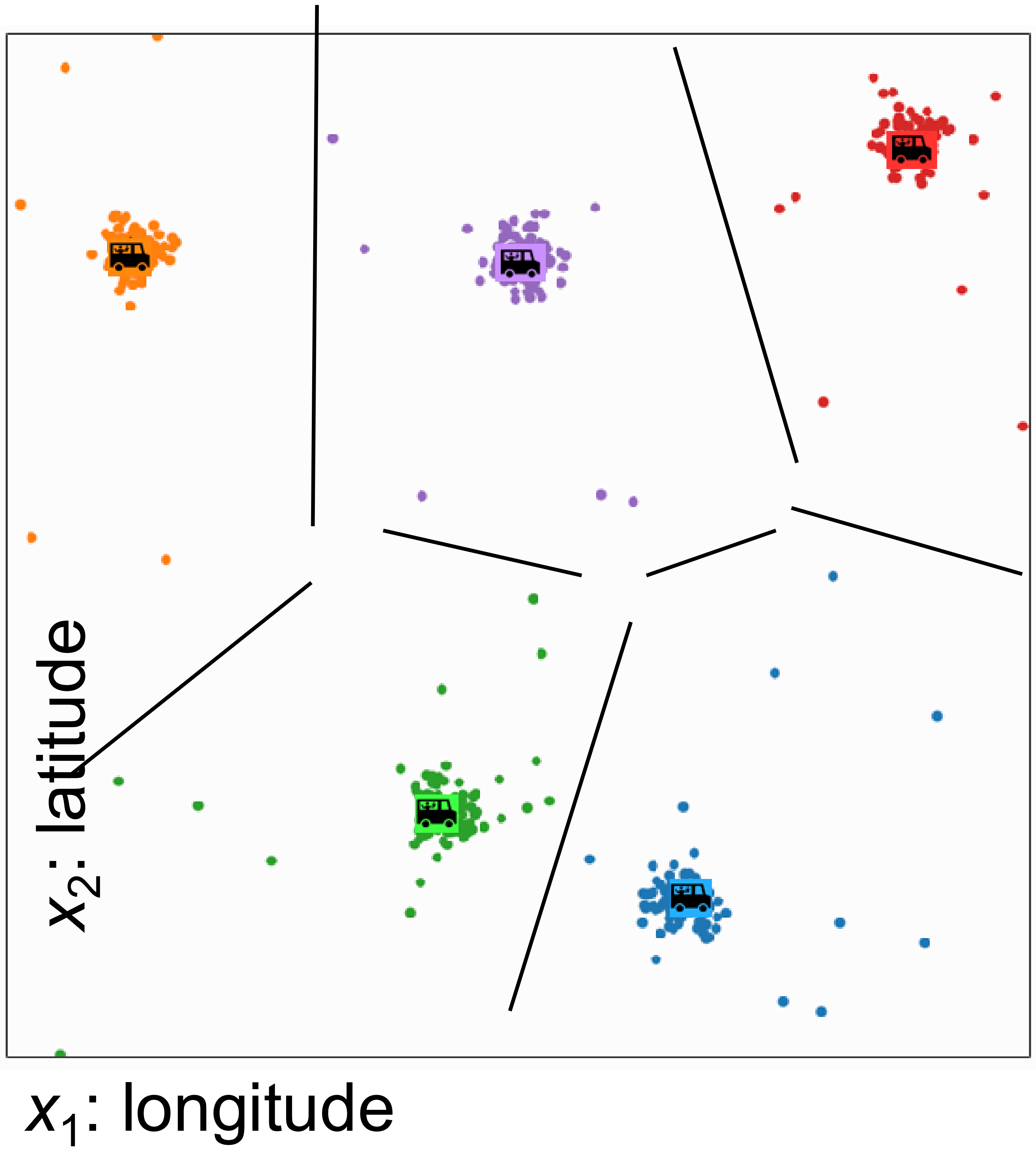
for i = 1 to n

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

for j = 1 to k

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

k-means algorithm



k-means(k, τ)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

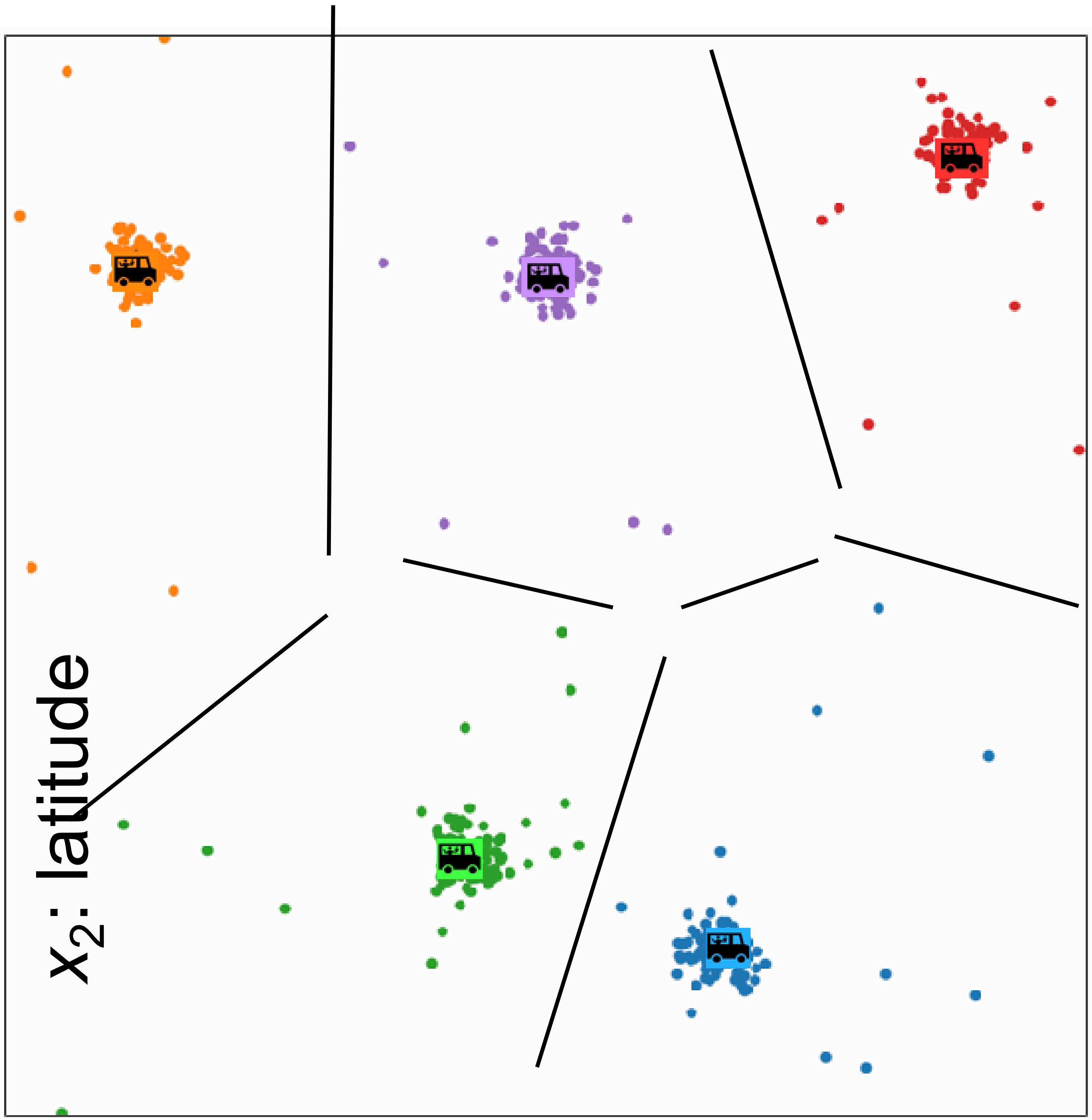
for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

k-means algorithm




```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for  t = 1 to τ
```

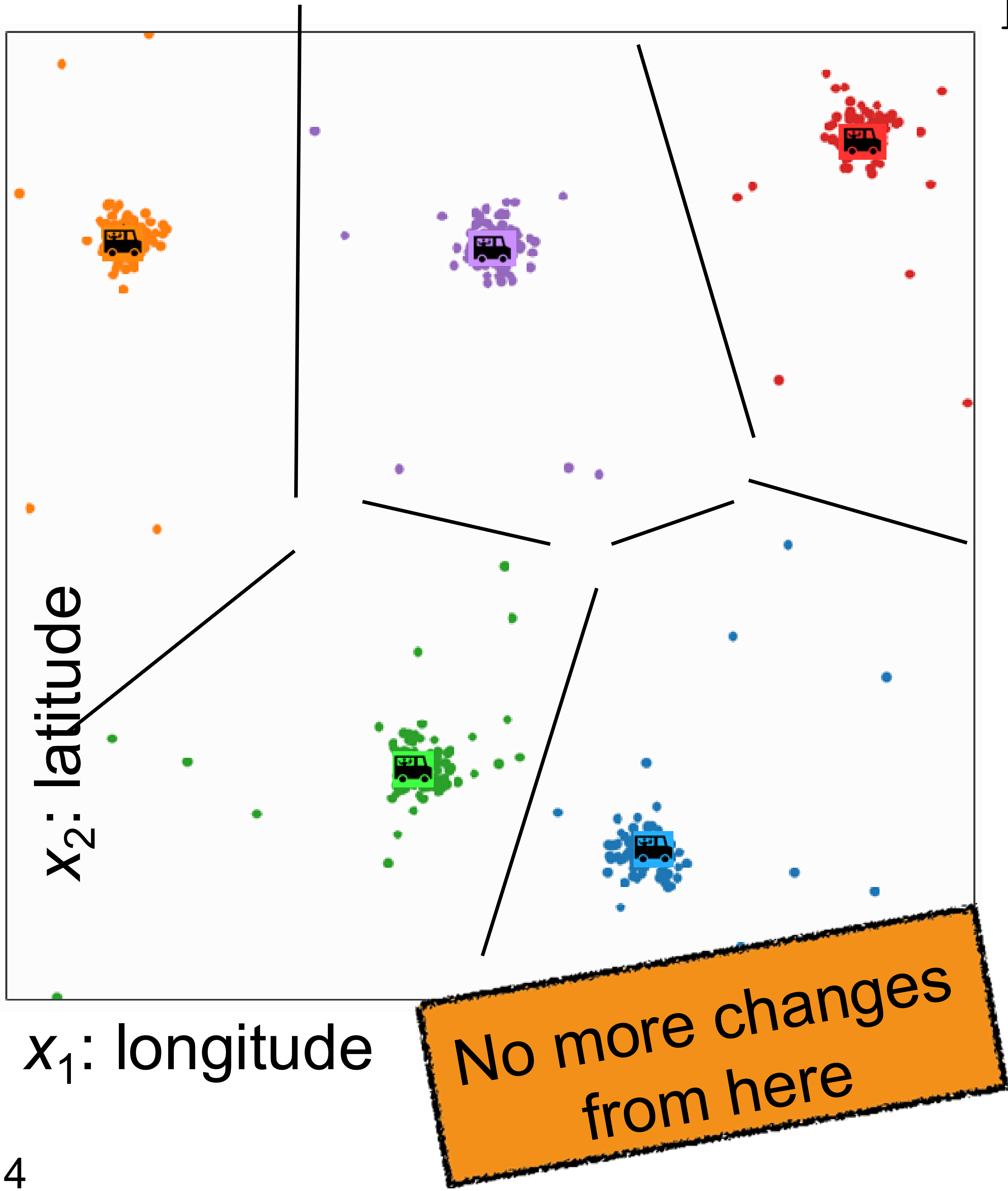
  for i = 1 to n

  $y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

  for j = 1 to k

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

k-means algorithm



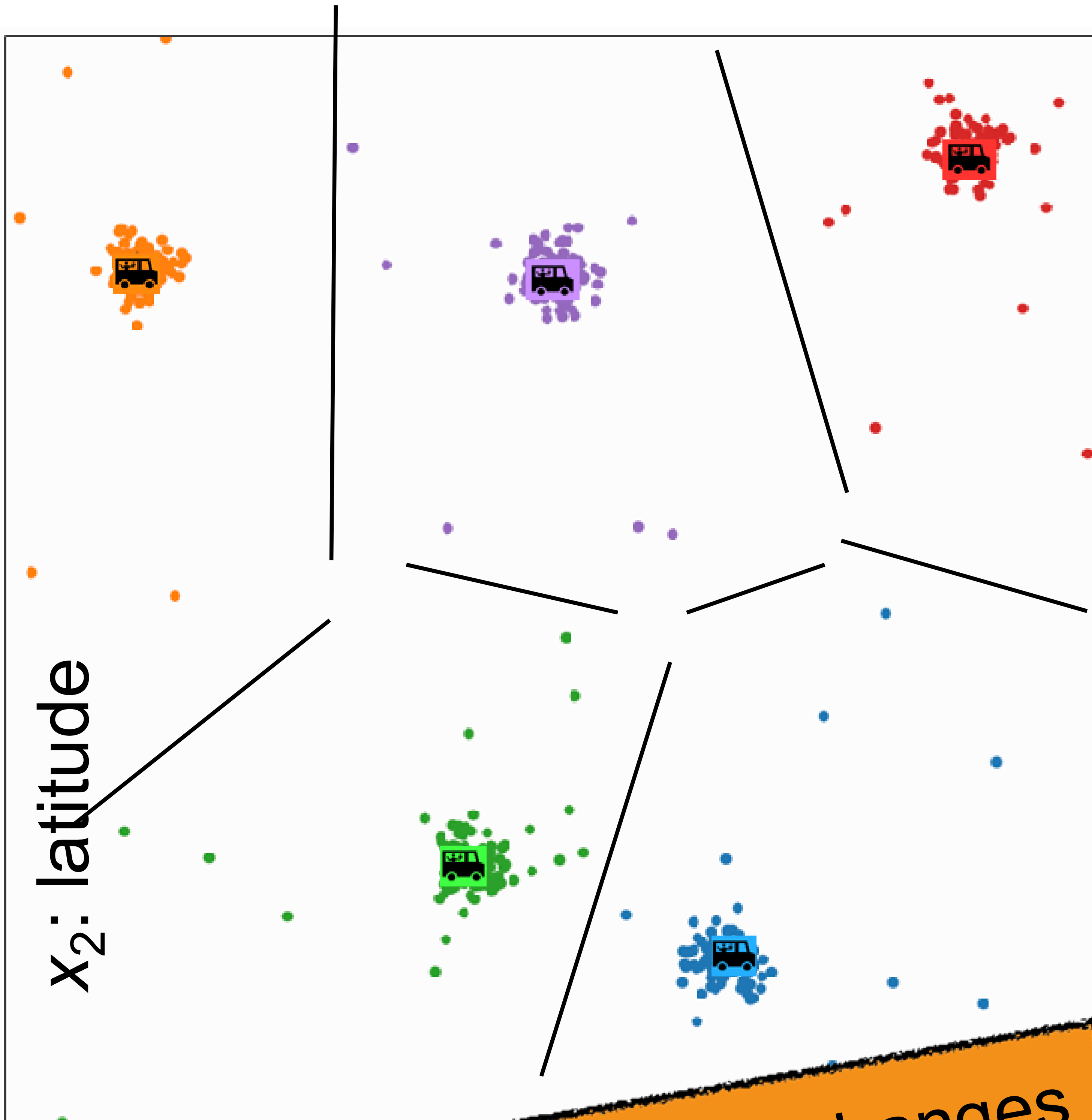

```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k
  for   t = 1 to τ

    for i = 1 to n
```

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

```
    for j = 1 to k
```

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

k-means algorithm



k-means(k, τ) 3 onwards

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to τ

for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

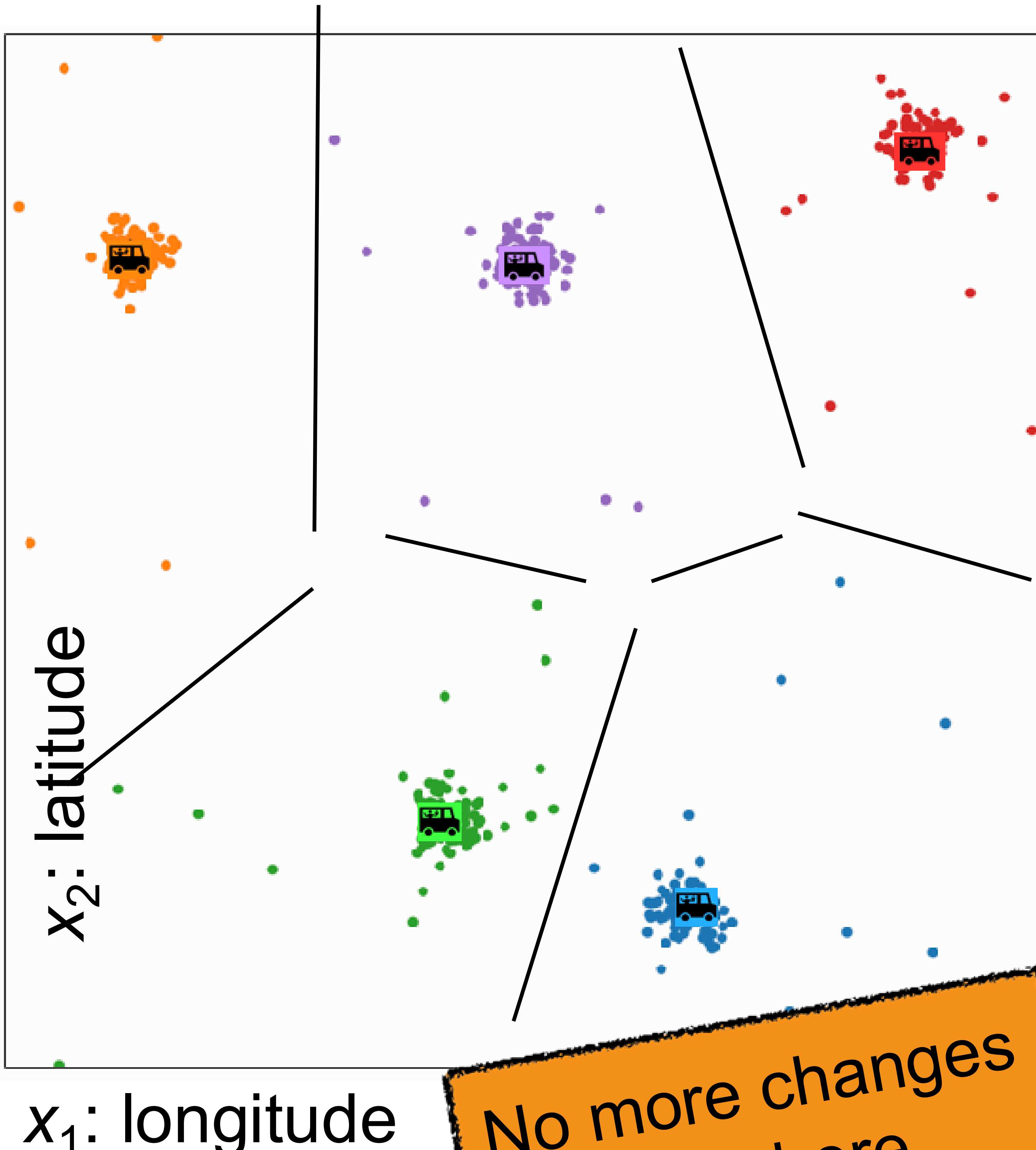
for j = 1 to k

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

No more changes from here

How can I be so sure?

k-means algorithm



k-means(k, $\boldsymbol{\tau}$)

Init $\{\mu^{(j)}\}_{j=1}^{k}$

for t = 1 to $\boldsymbol{\tau}$

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

for i = 1 to n

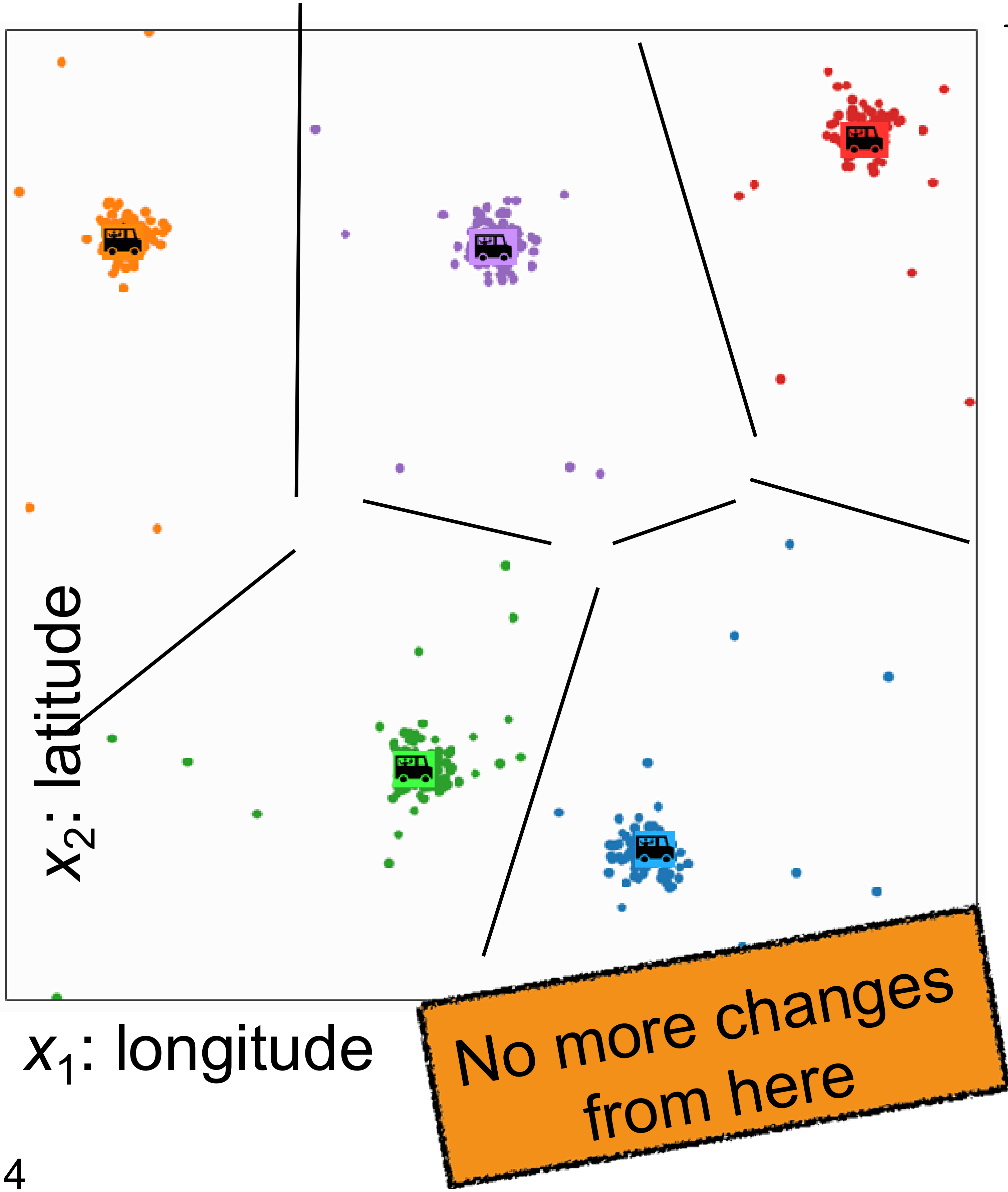
$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

for j = 1 to k

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

No more changes from here

k-means algorithm




```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k , y
  for t = 1 to τ
    y_old = y
    for i = 1 to n
      y^(i) = argmin_j ||x^(i) − μ^(j)||_2^2
    for j = 1 to k
      μ^(j) = Σ_{i=1}^n 1{y^(i) = j} x^(i) / Σ_{i=1}^n 1{y^(i) = j}
    if y = y_old
      break
```

$$\text{Init } \{\mu^{(j)}\}_{j=1}^{k},\ y$$

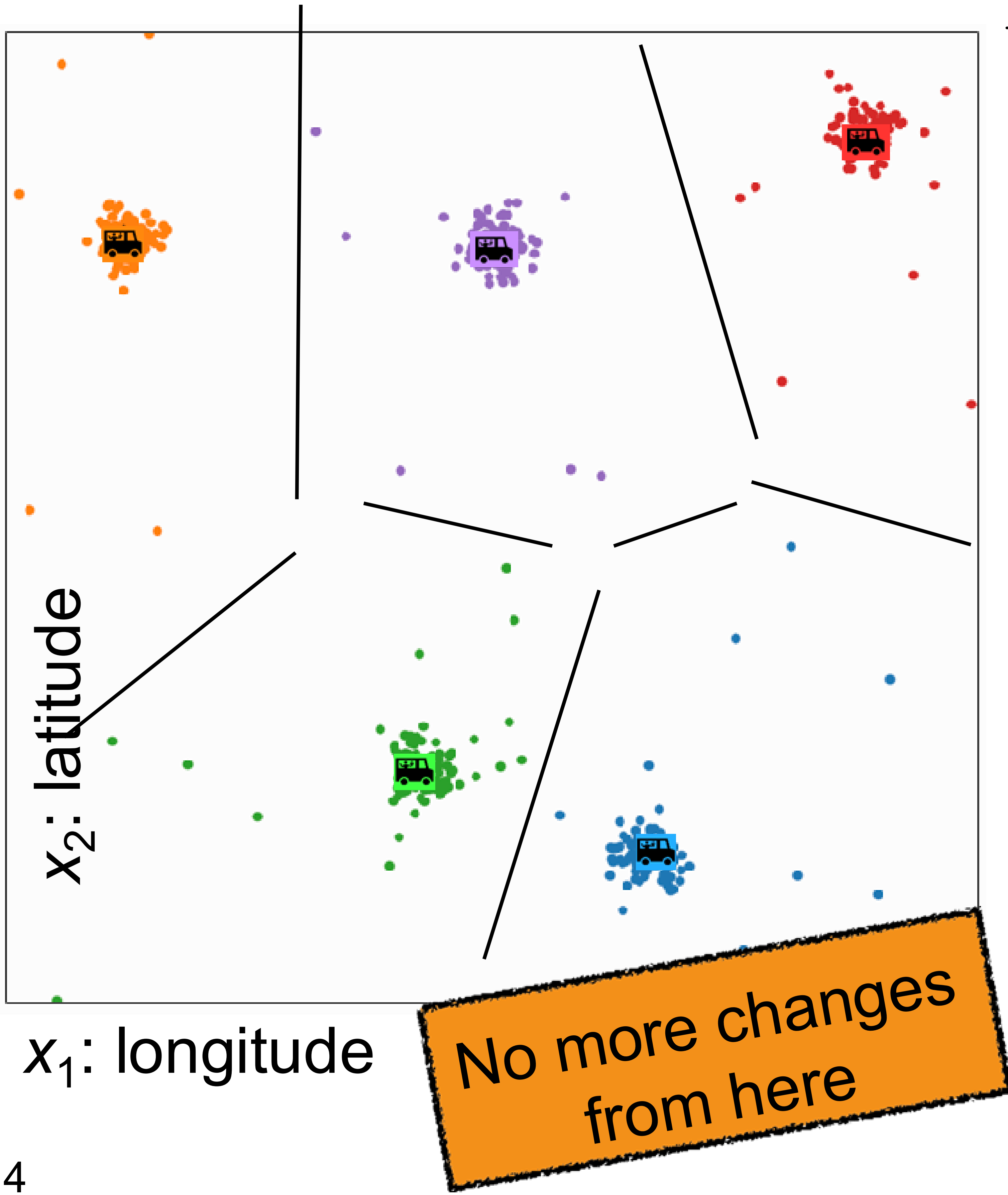
$$y_{\text{old}} = y$$

$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$

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

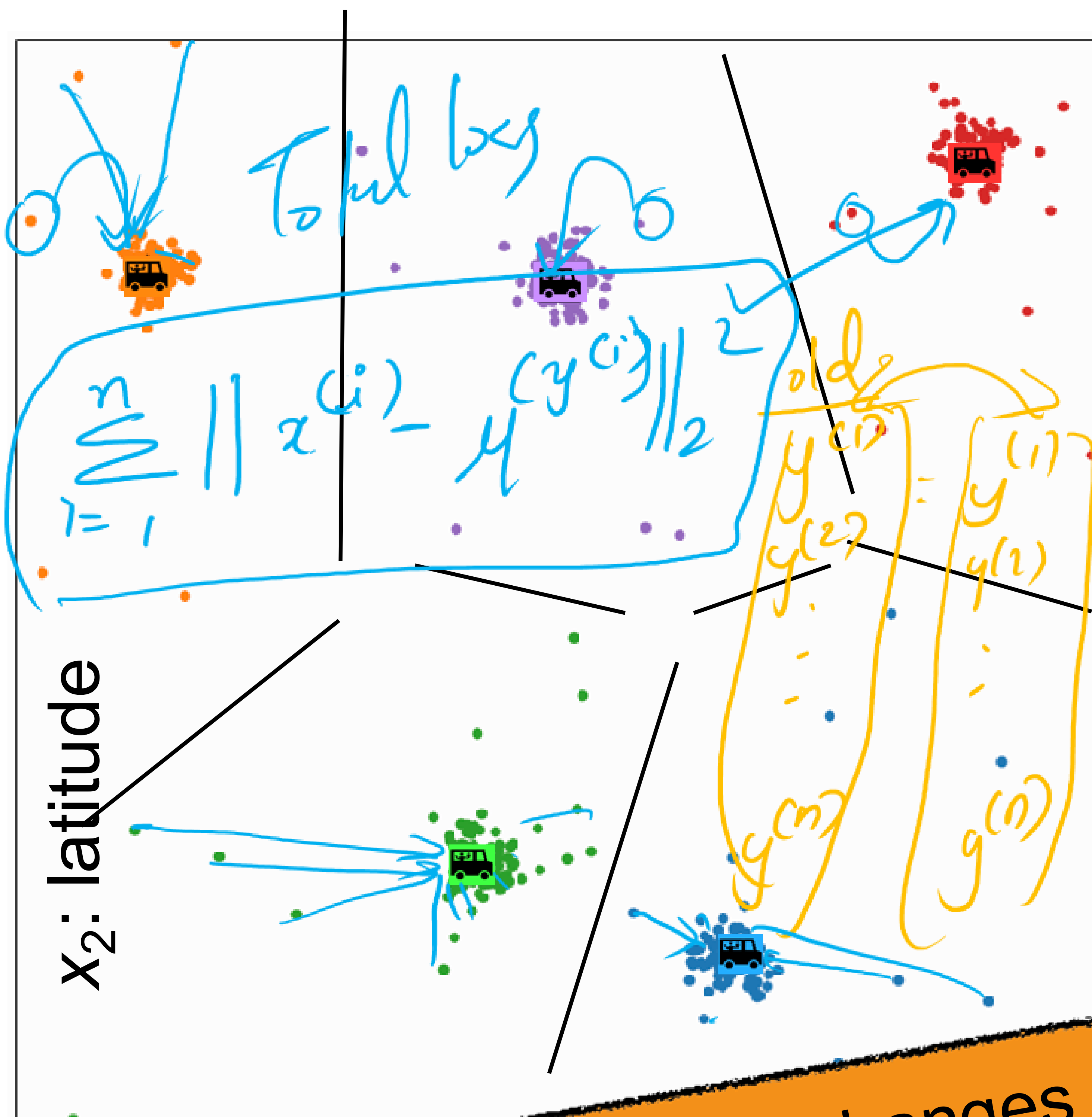
if $y = y_{\text{old}}$ break

k-means algorithm




```
k-means(k, τ)
  Init {μ^(j)}_{j=1}^k, {y^(i)}_{i=1}^n
  for t = 1 to τ
    y_old = y
    for i = 1 to n
```
$$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$$
```
    for j = 1 to k
```
$$\mu^{(j)} = \frac{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^n \mathbf{1}\{y^{(i)} = j\}}$$
```
    if y = y_old
      break
```

k-means algorithm



x_1: longitude

No more changes from here

k-means(k, $\boldsymbol{\tau}$) 10

Init $\{\mu^{(j)}\}_{j=1}^{k}, \{y^{(i)}\}_{i=1}^{n}$

for t = 1 to $\boldsymbol{\tau}$

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

for i = 1 to n

$y^{(i)} = \arg\min_j \|x^{(i)} - \mu^{(j)}\|_2^2$

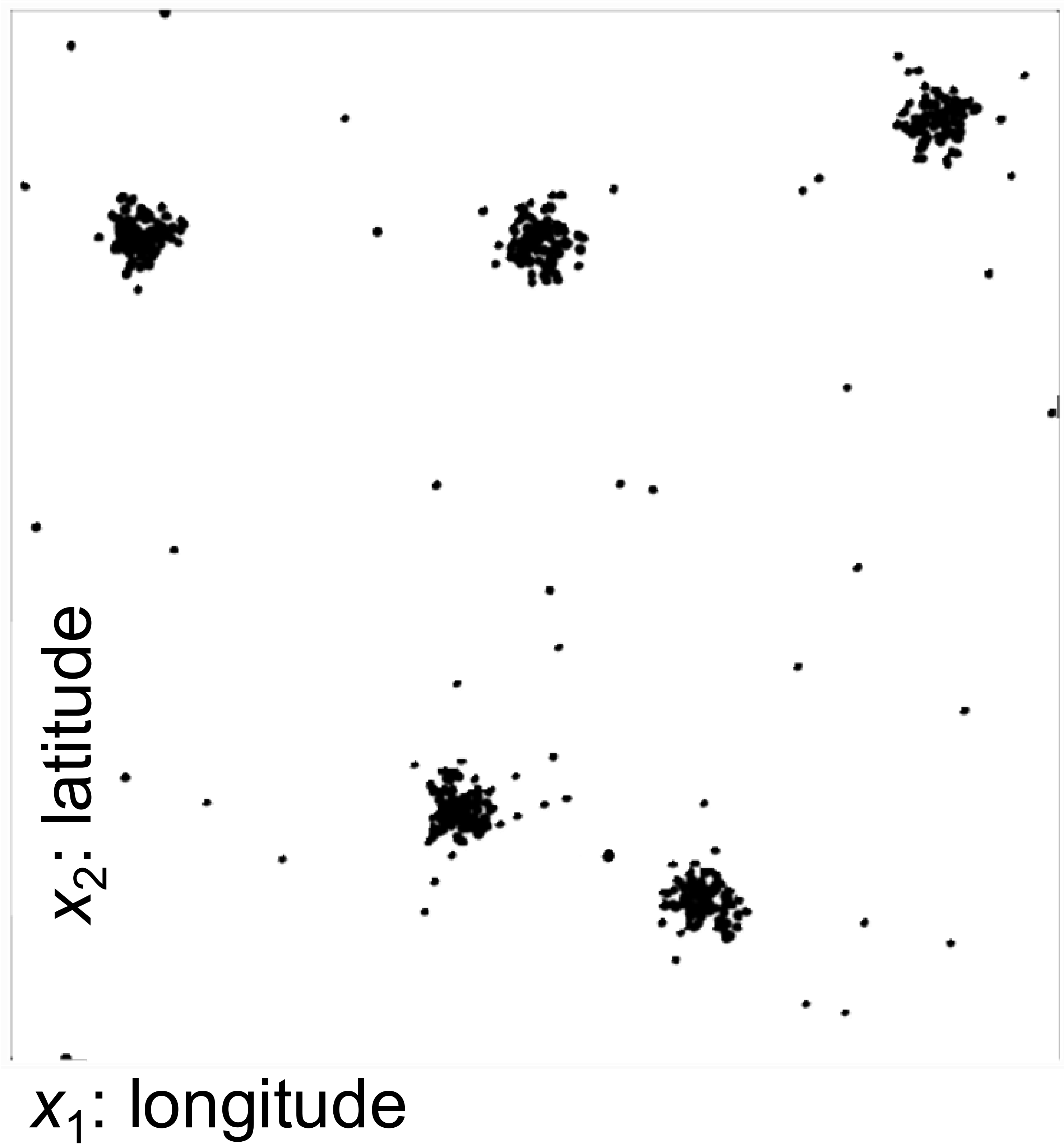
for j = 1 to k

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

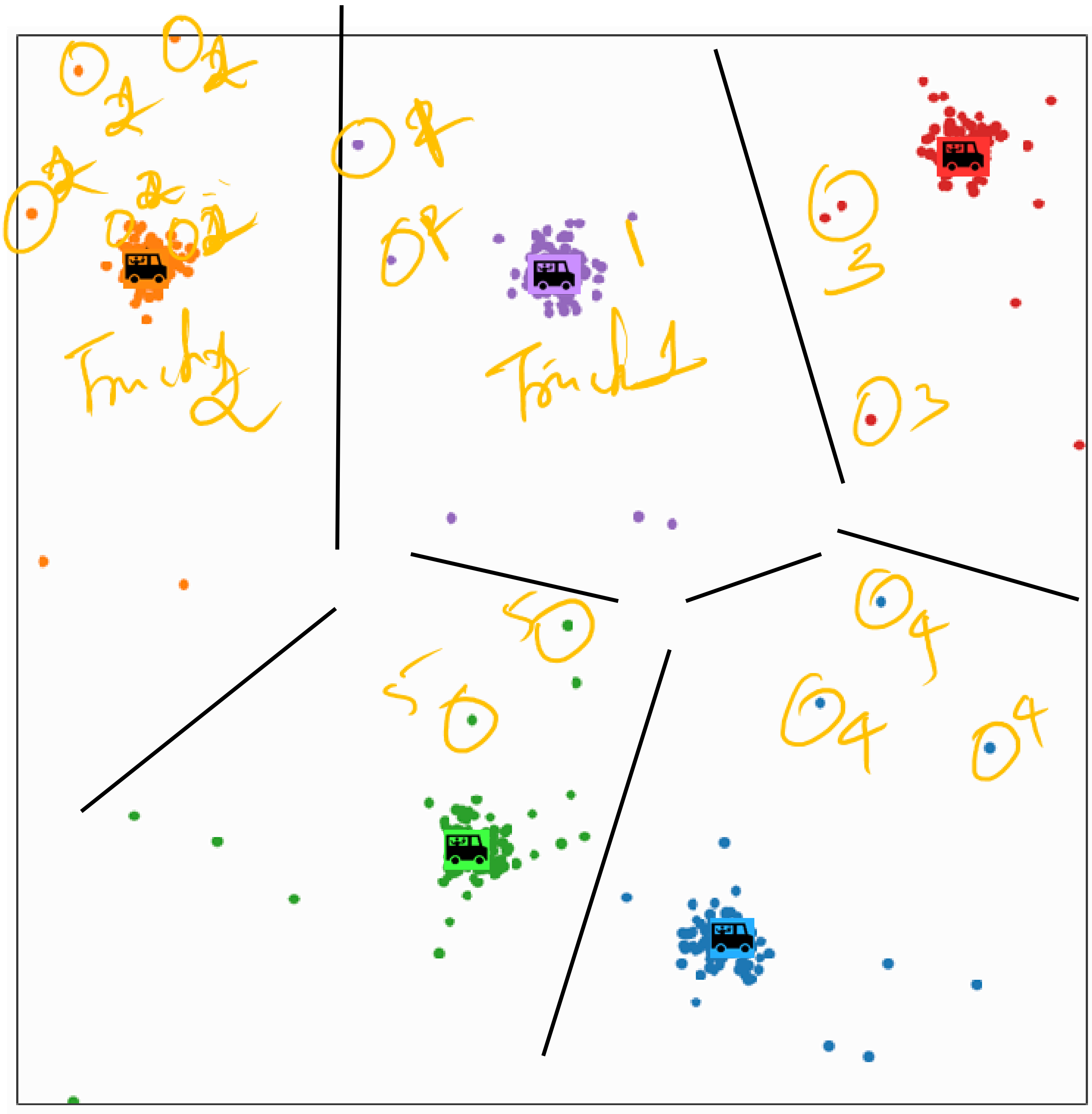
if $y = y_{\text{old}}$ vector

break

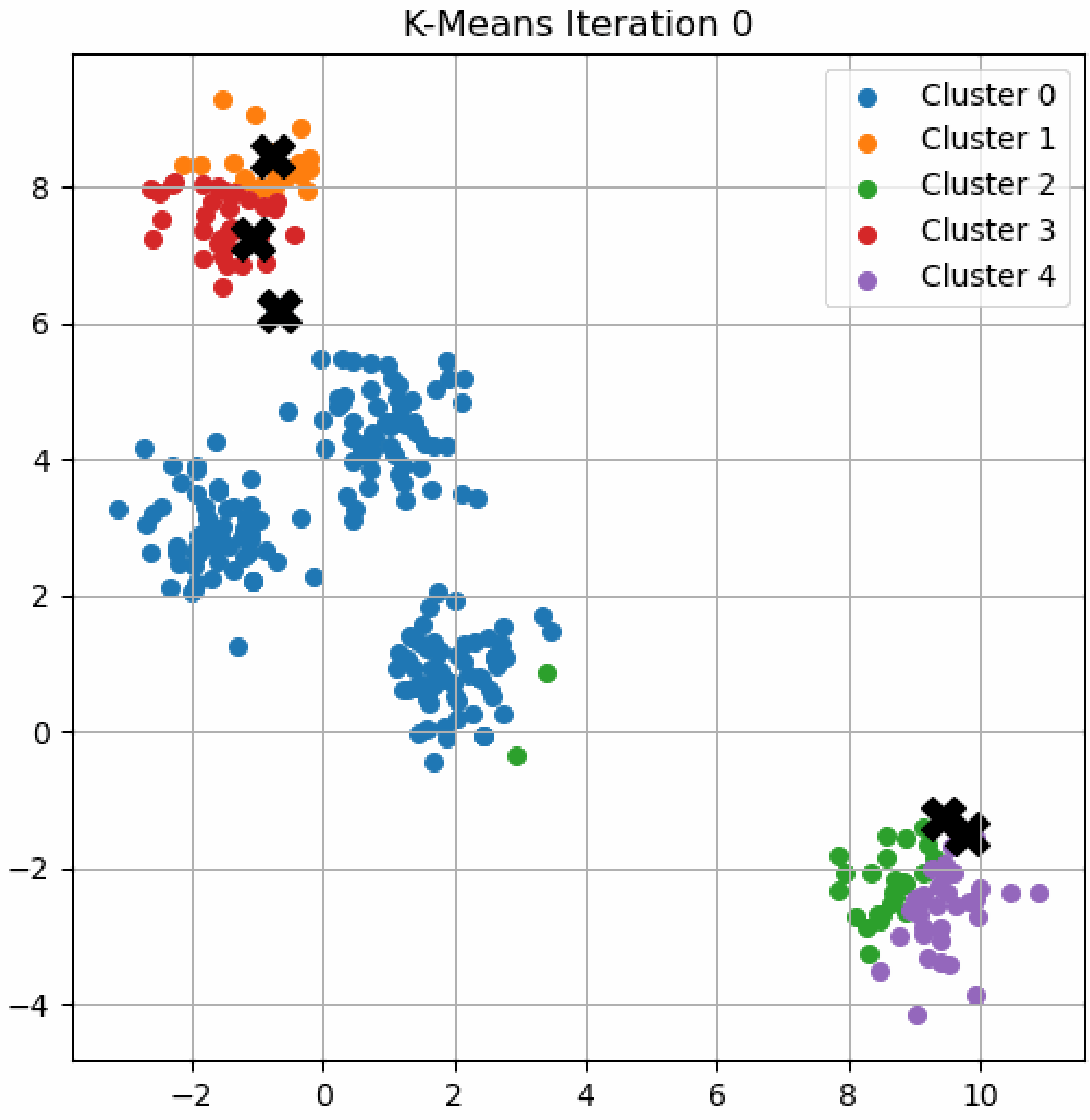
return $\{\mu^{(j)}\}_{j=1}^{k}, \{y^{(i)}\}_{i=1}^{n}$



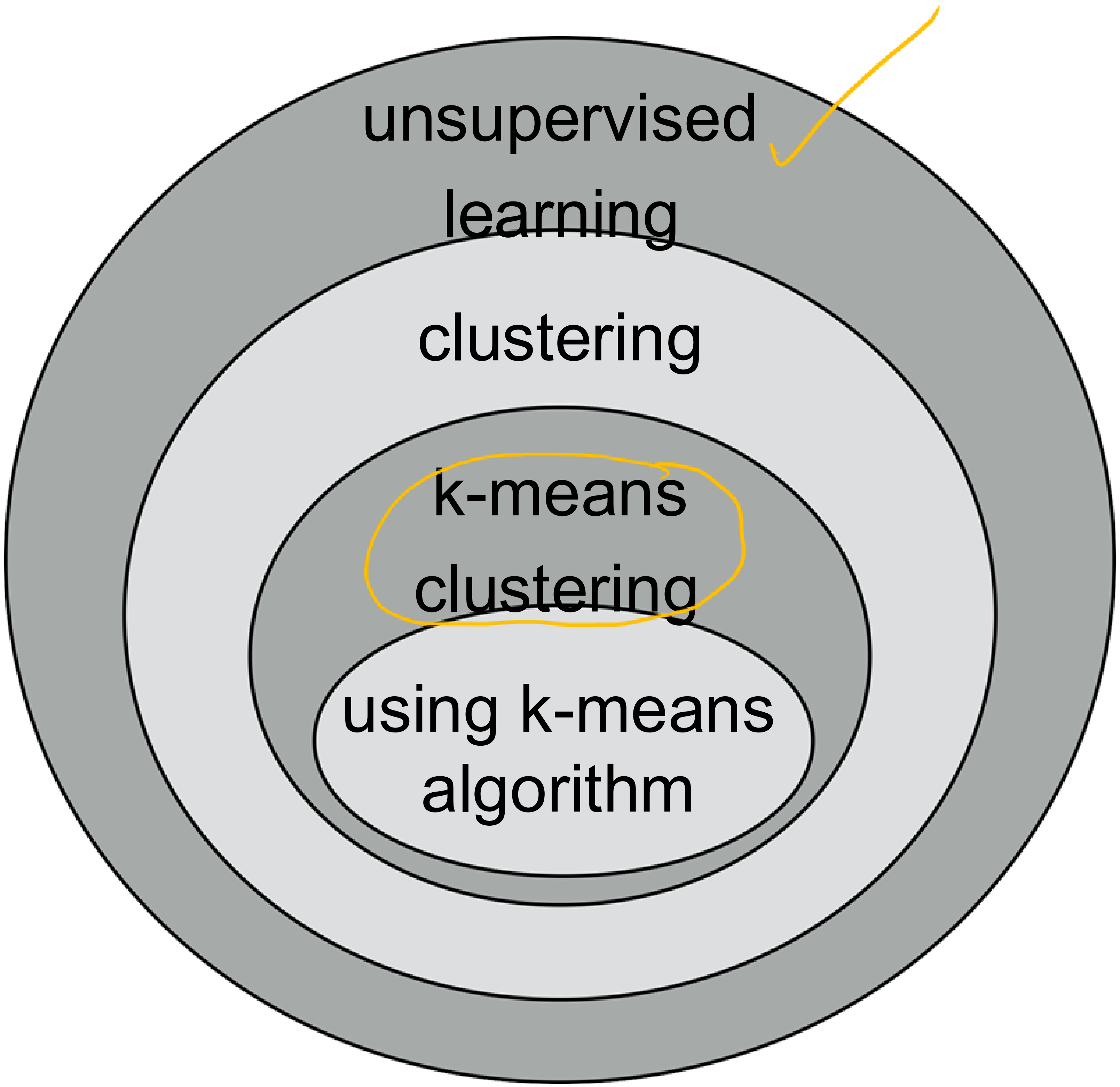
- So what did we do?
- We *clustered* the data: we grouped the **data** by similarity



- So what did we do?
- We *clustered* the data: we **grouped** the data by similarity

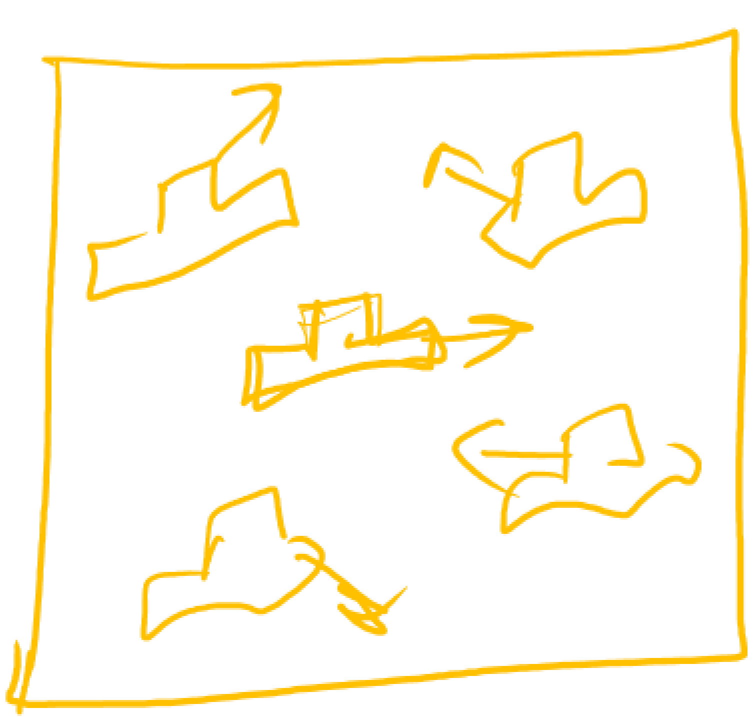

K-Means Iteration 0
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
8
6
4
2
0
−2
−4
−2
0
2
4
6
8
10

Clustering & related

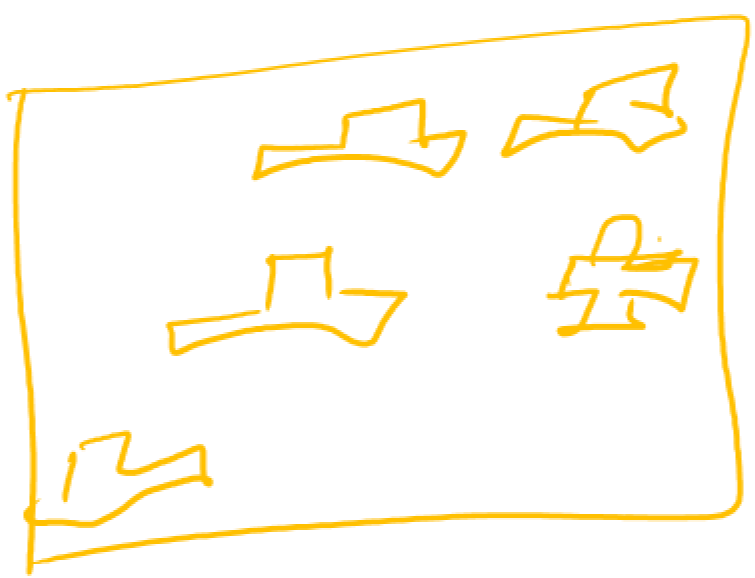


k-means algorithm: initialization

k-means algorithm: initialization



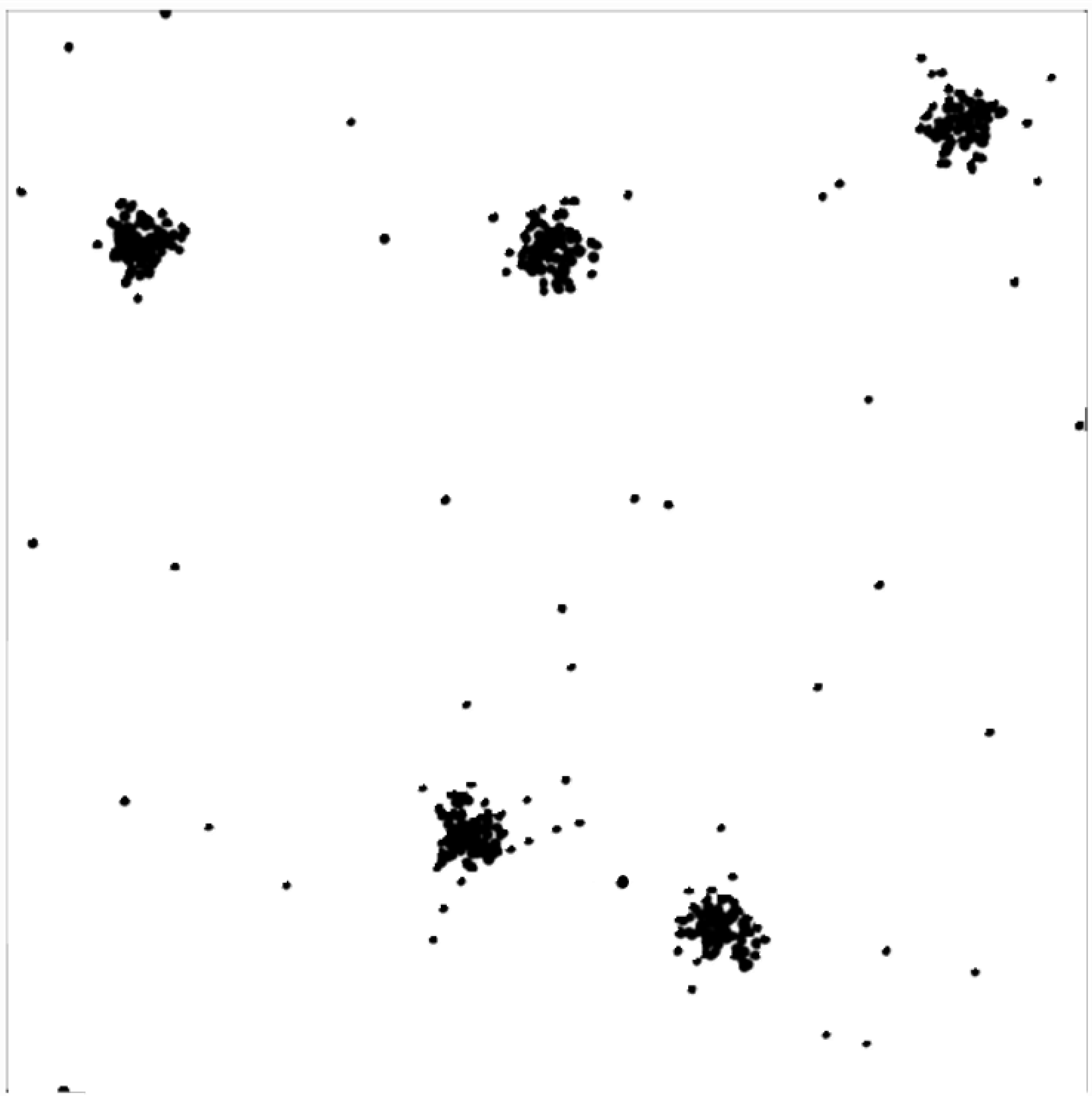
Restart 1



Restart 2

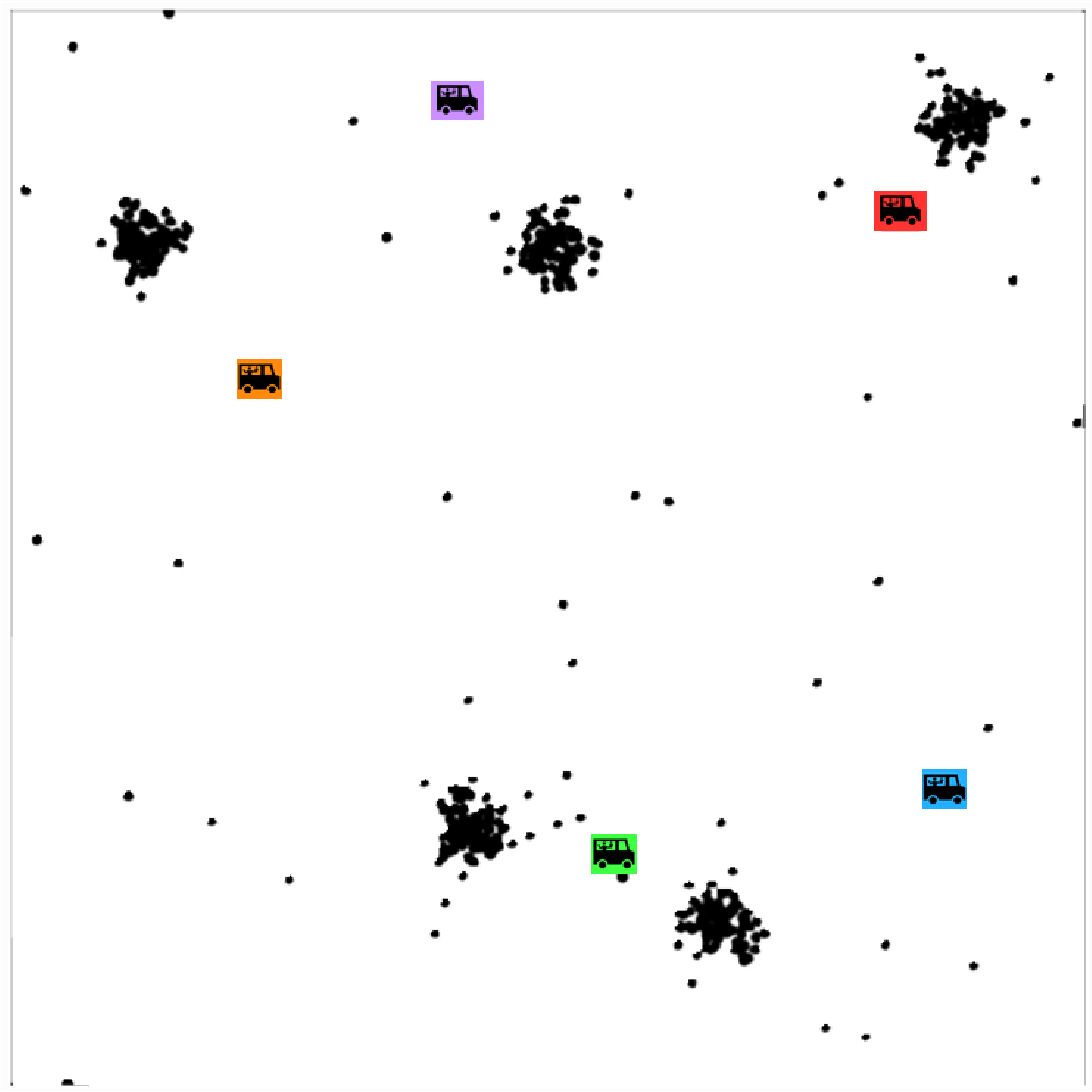
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective

k-means algorithm: initialization



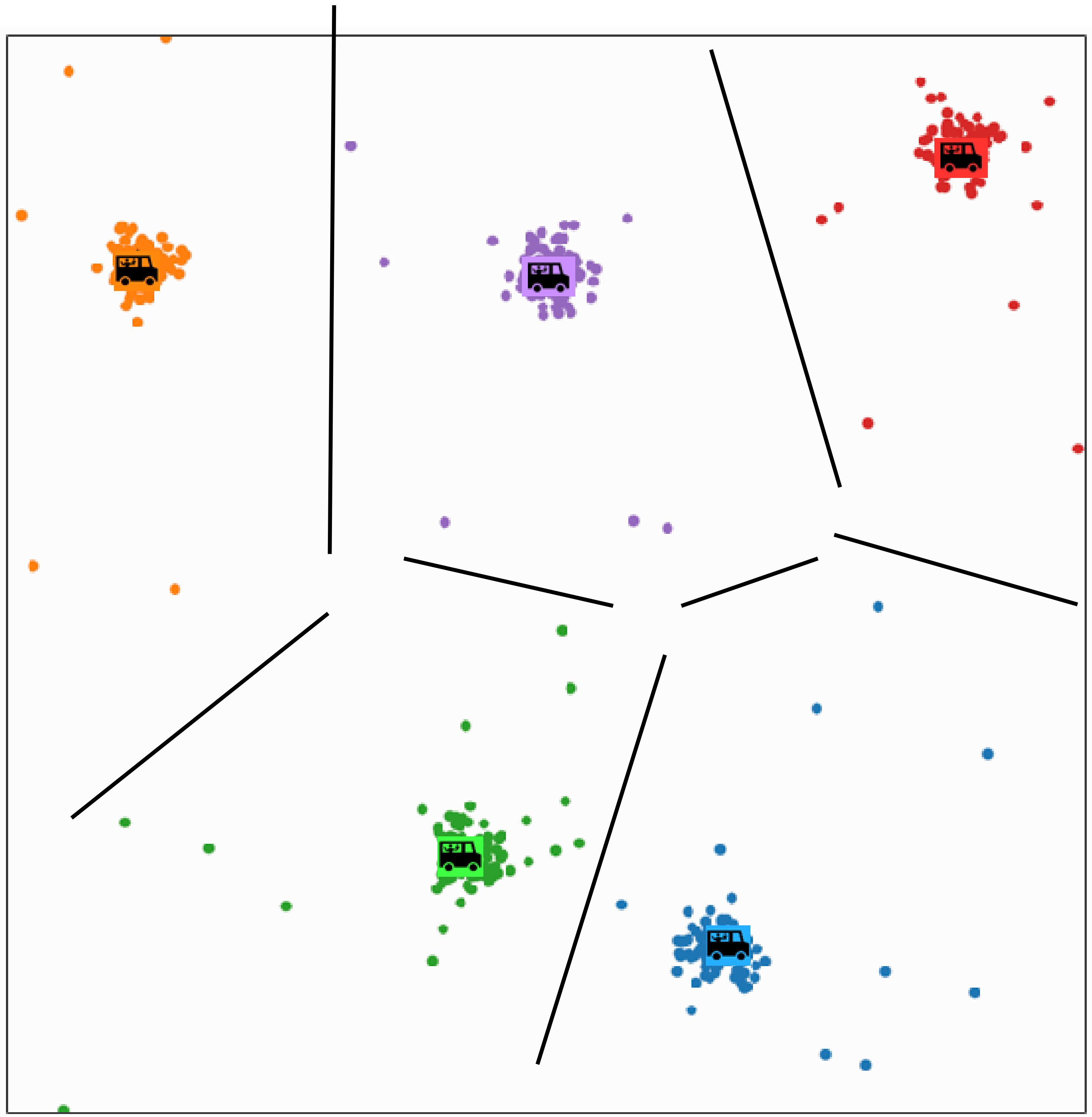
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective

k-means algorithm: initialization



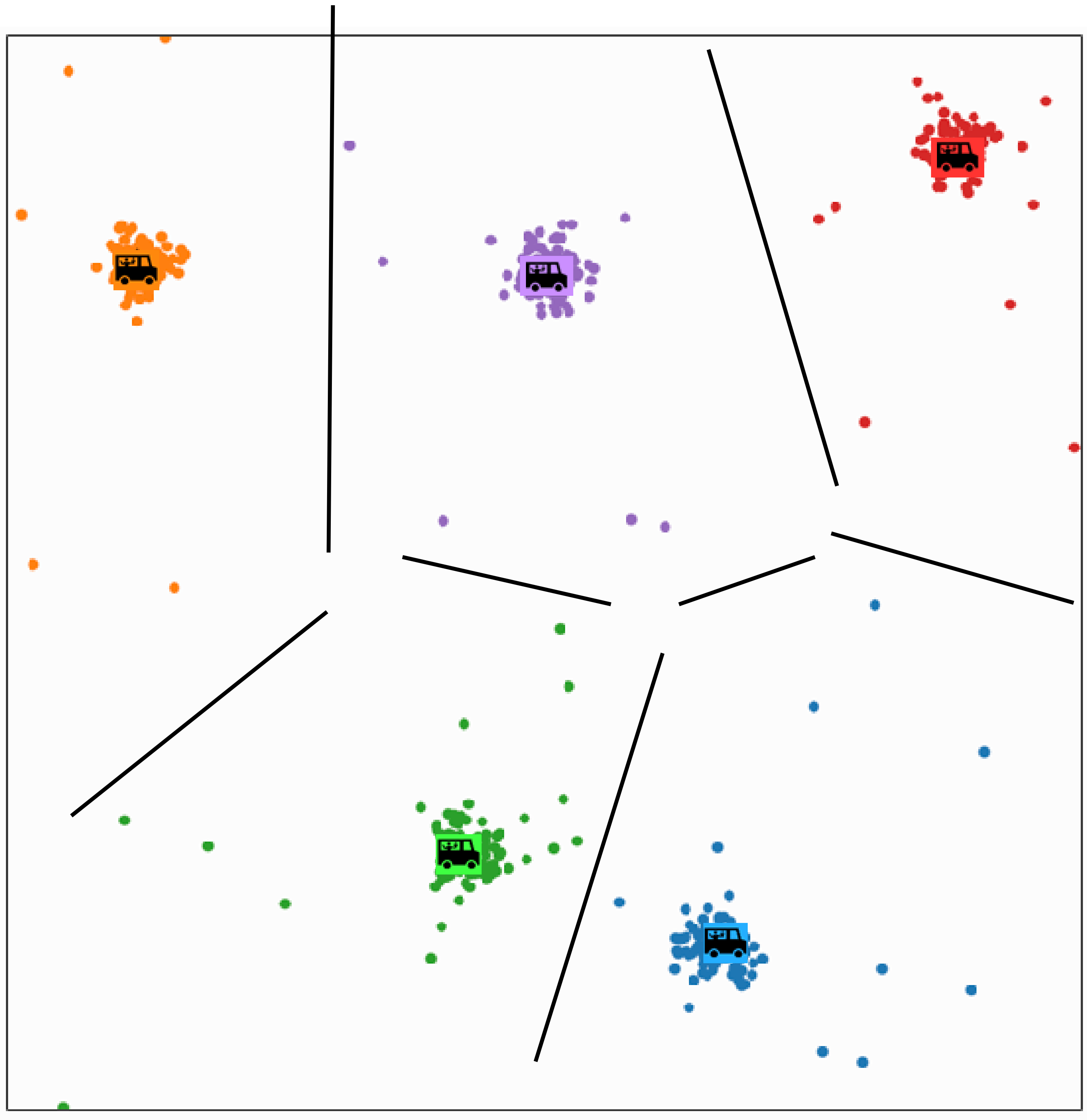
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective

k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective

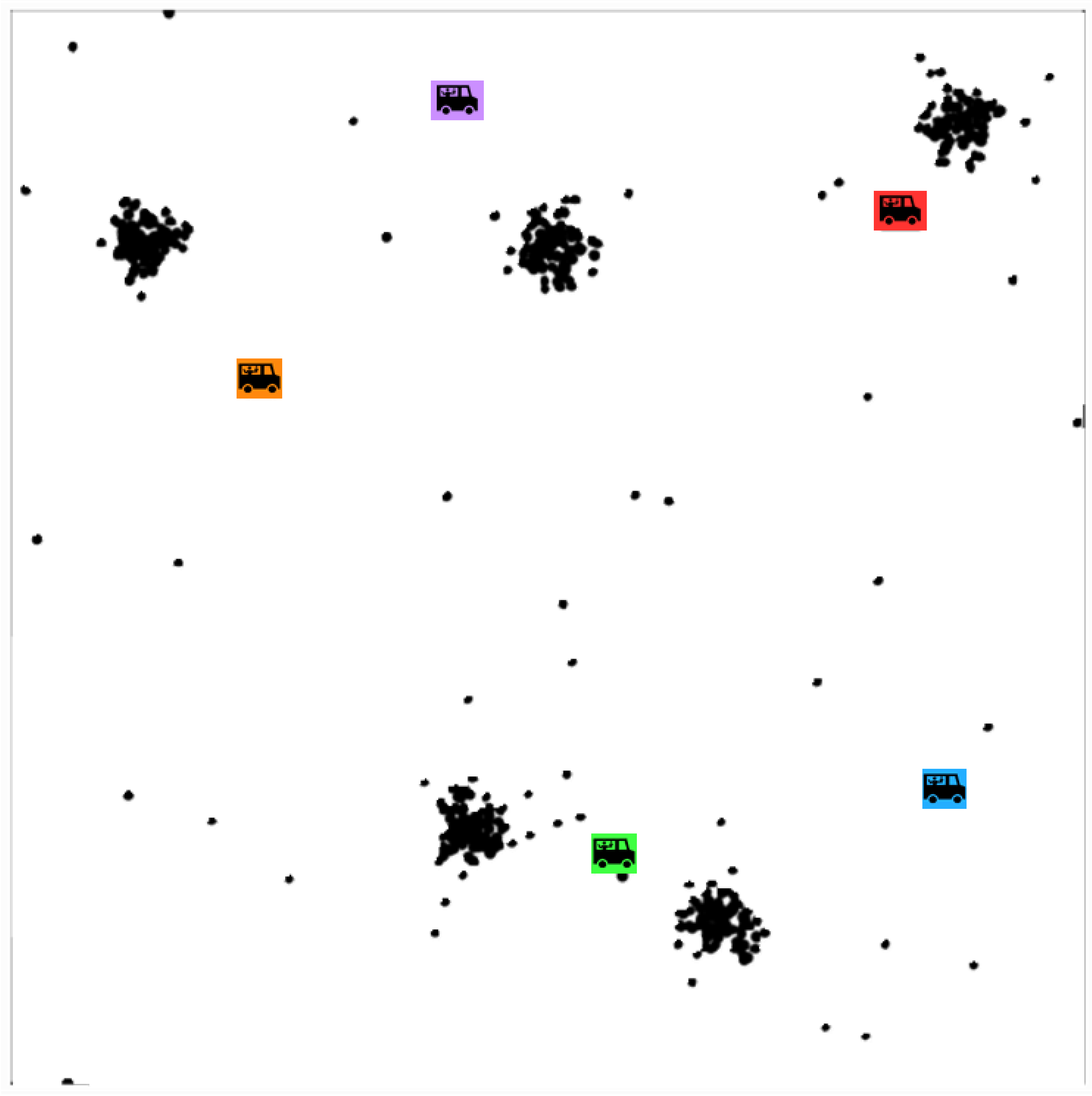
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could be bad!

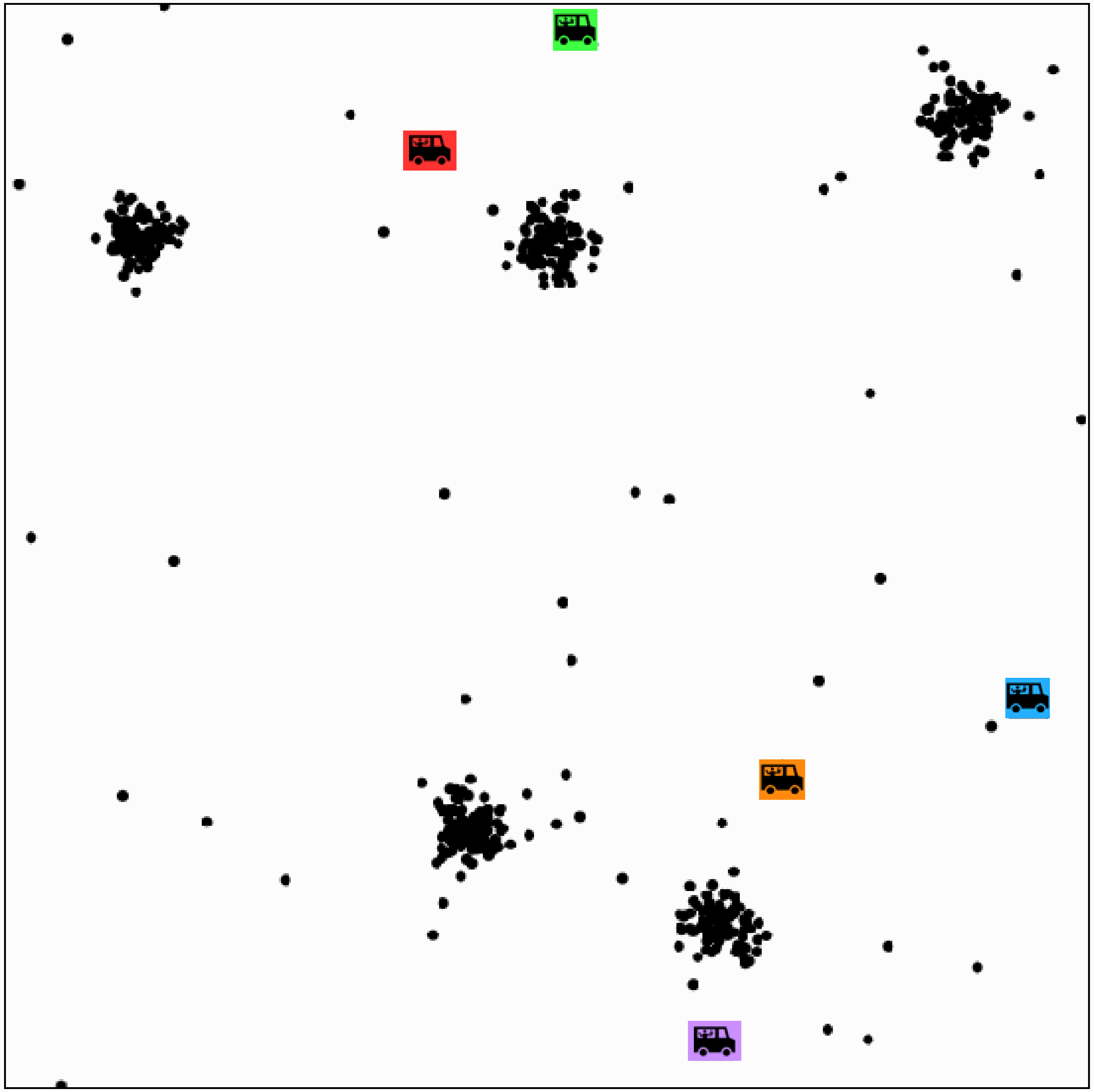
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could be bad!

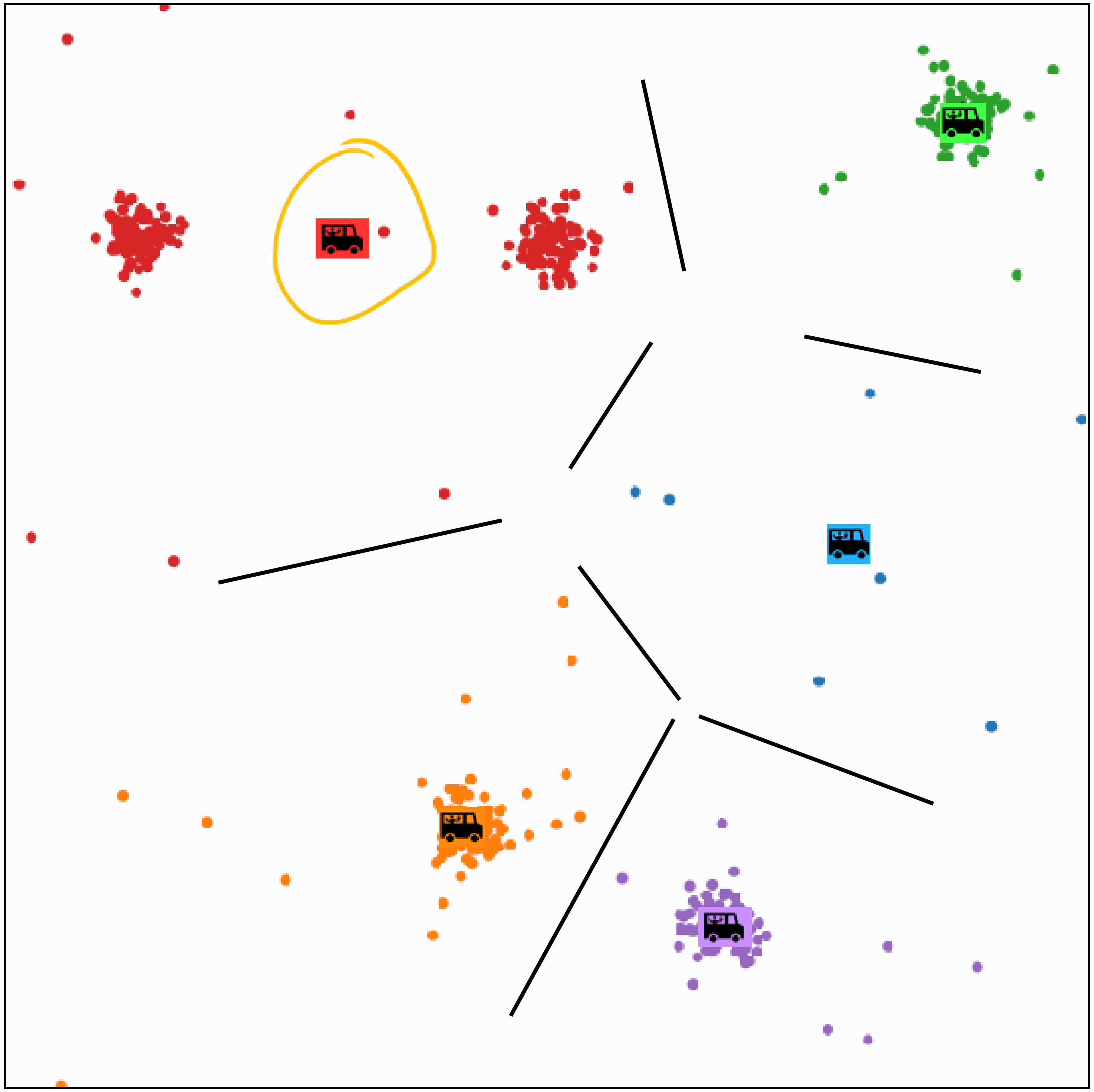
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could be bad!

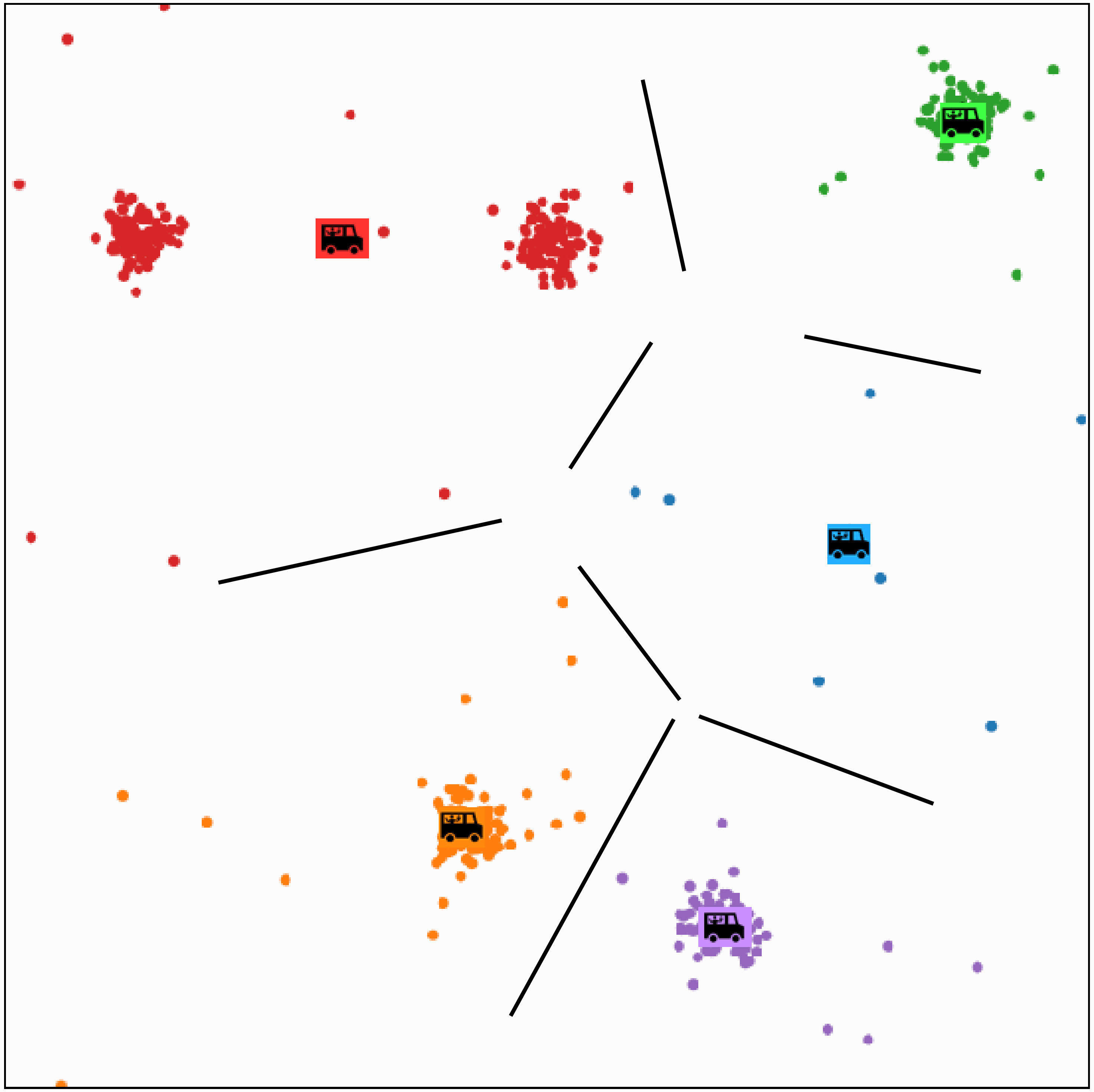
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could be bad!

k-means algorithm: initialization

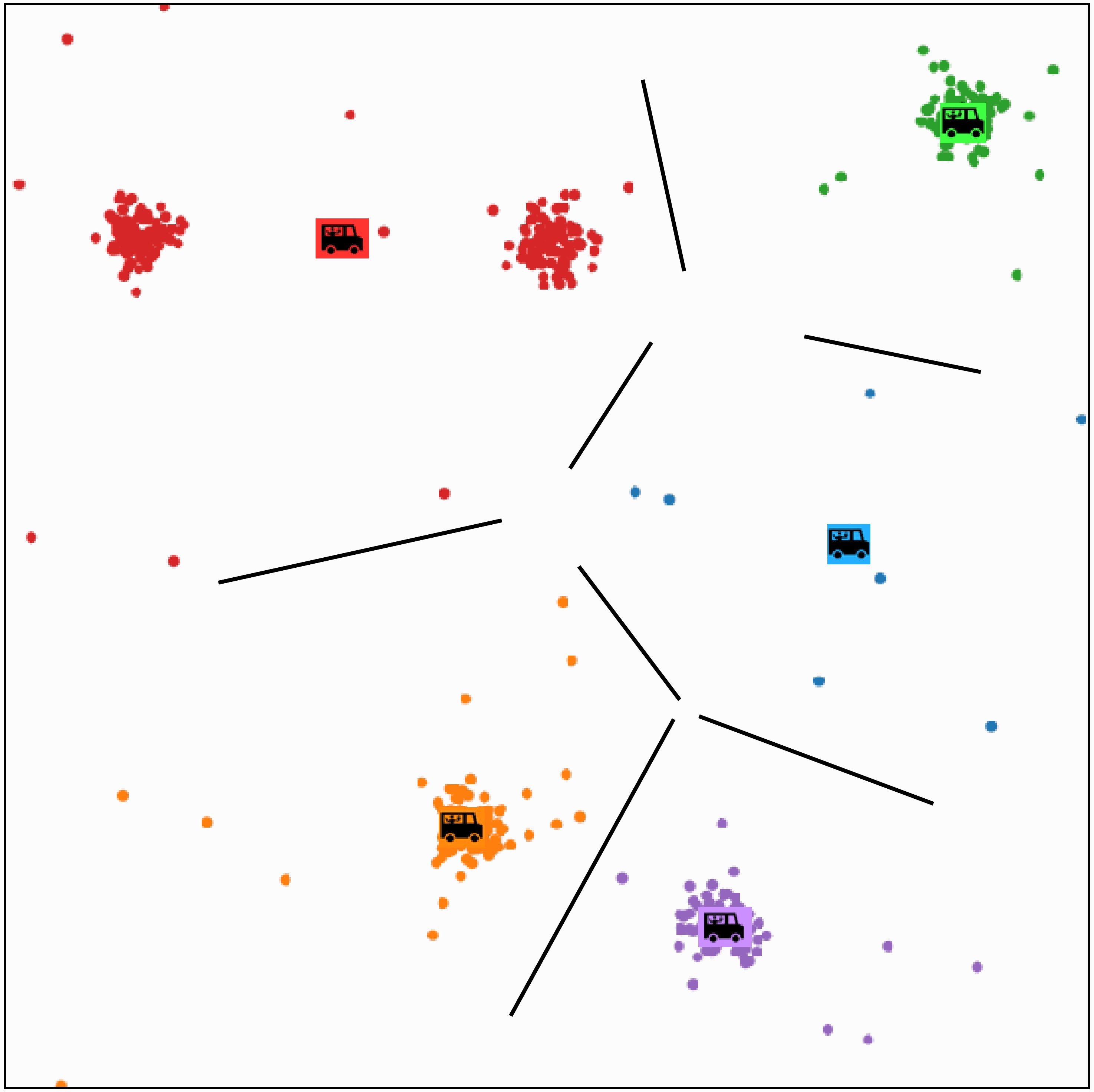


- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum cou[...]

Is this clustering worse than the one we found before?

k-means algorithm: initialization



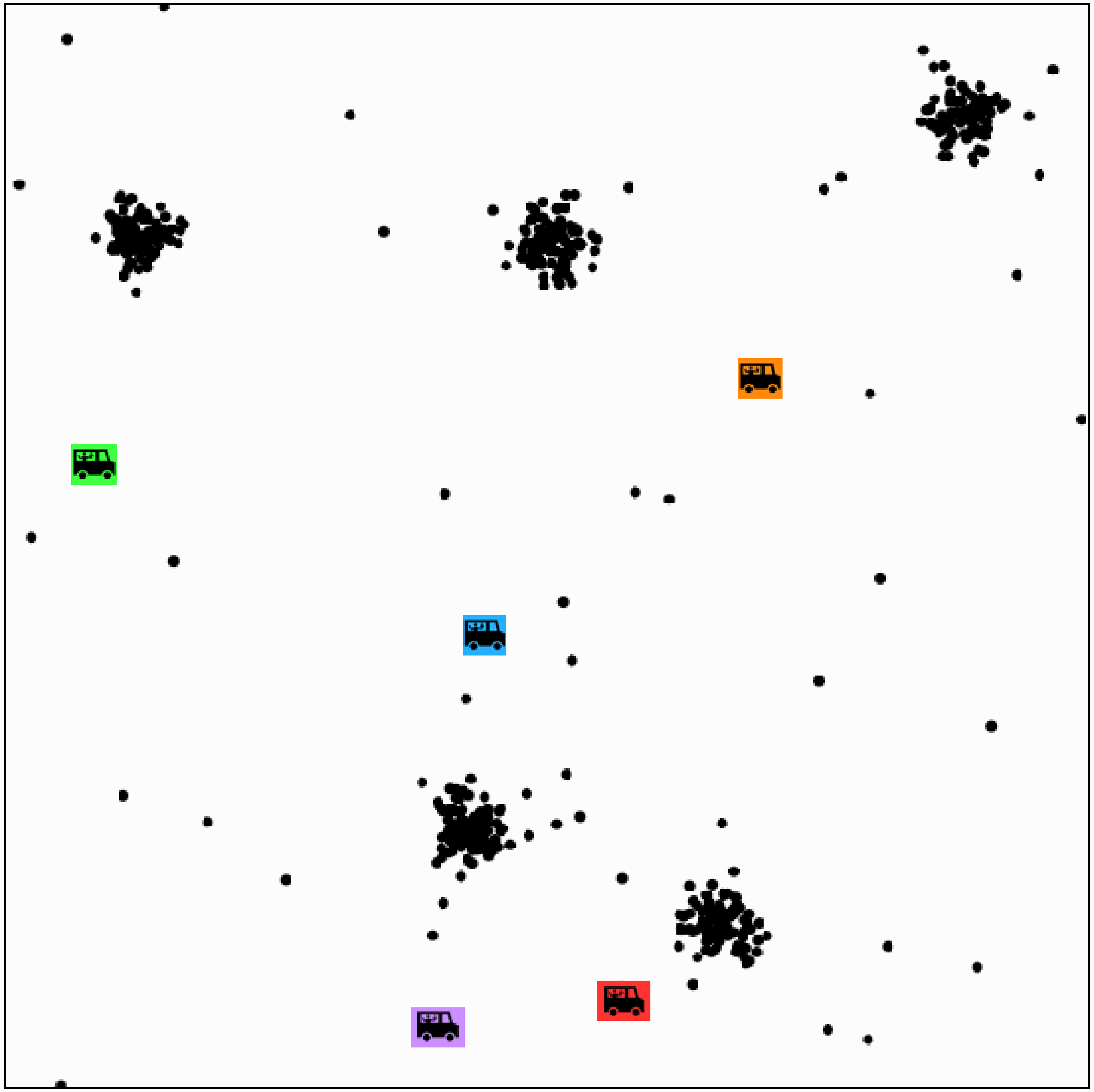
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could ...

Is this clustering worse than the one we found before?

Why or why not?

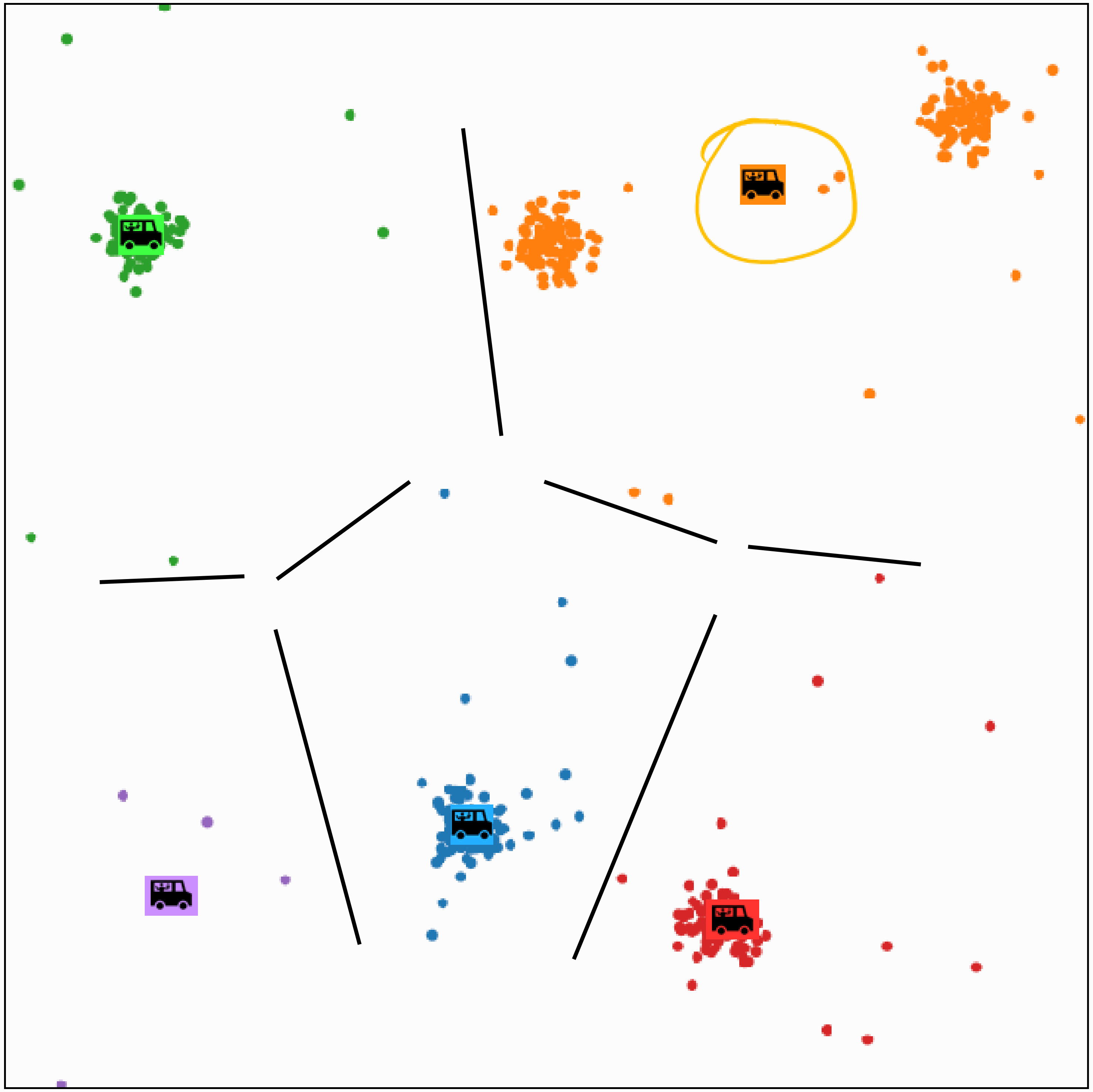
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

That local minimum could be bad!

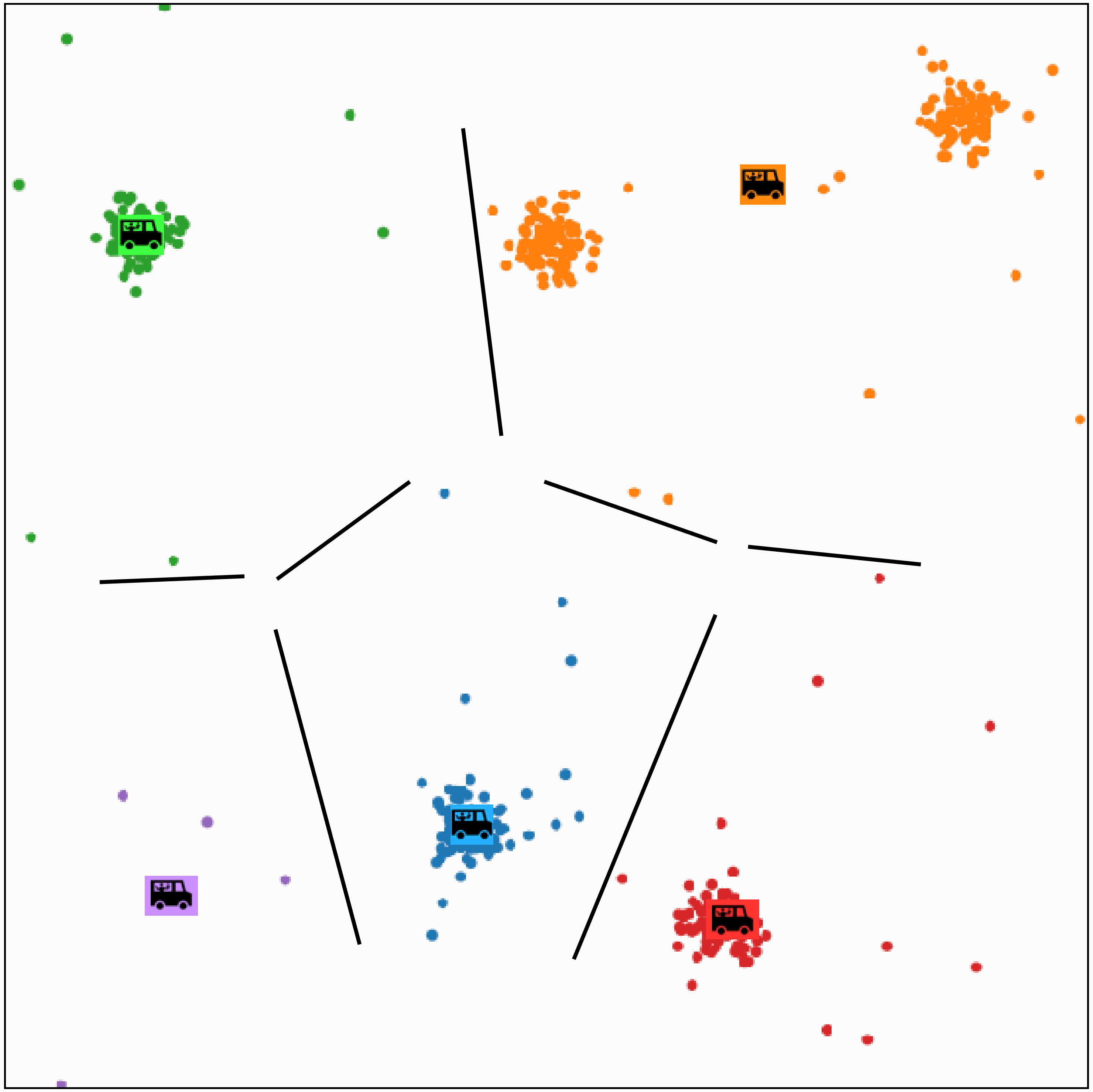
k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-
- means objective

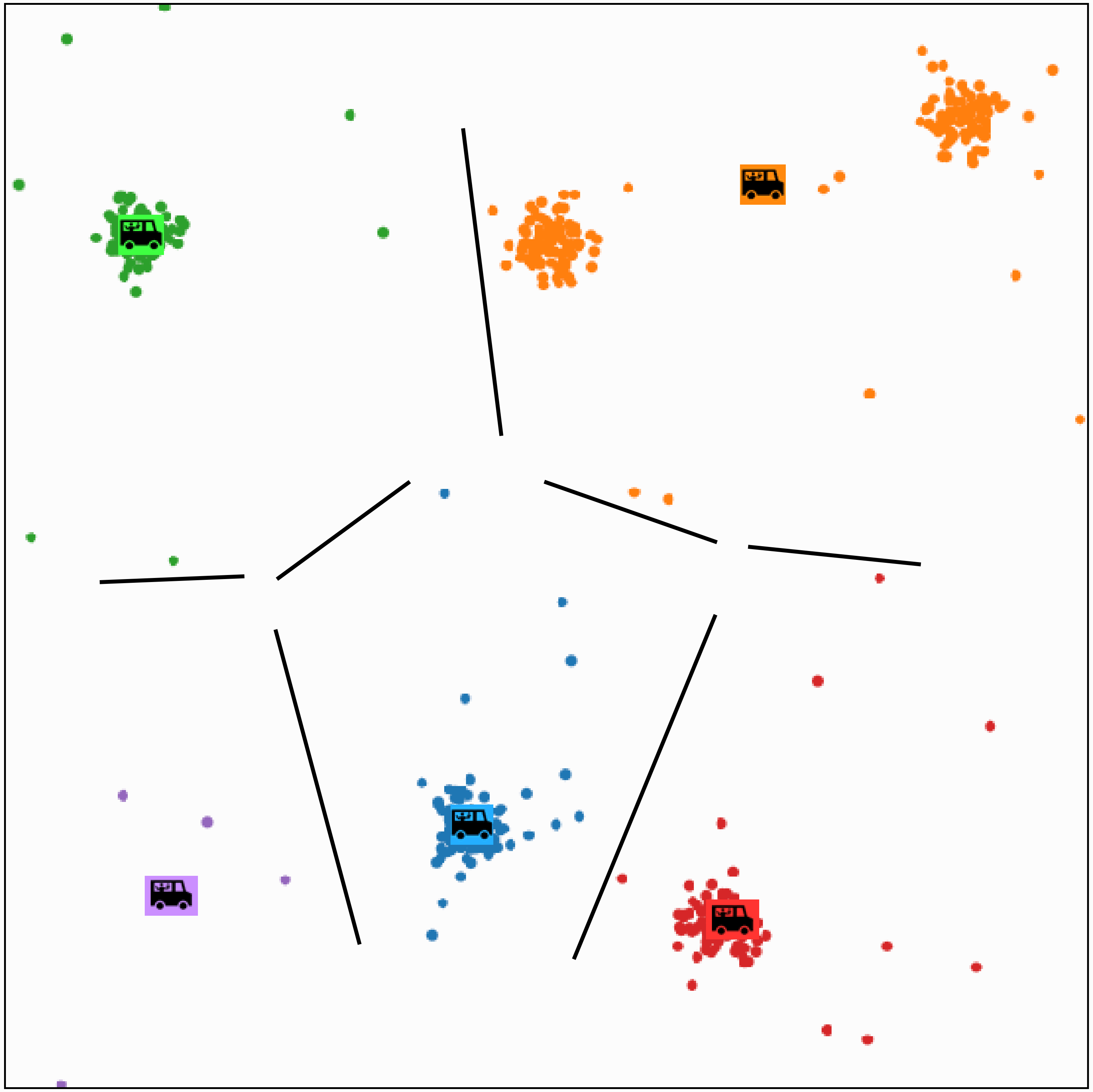
That local minimum could be bad!

k-means algorithm: initialization



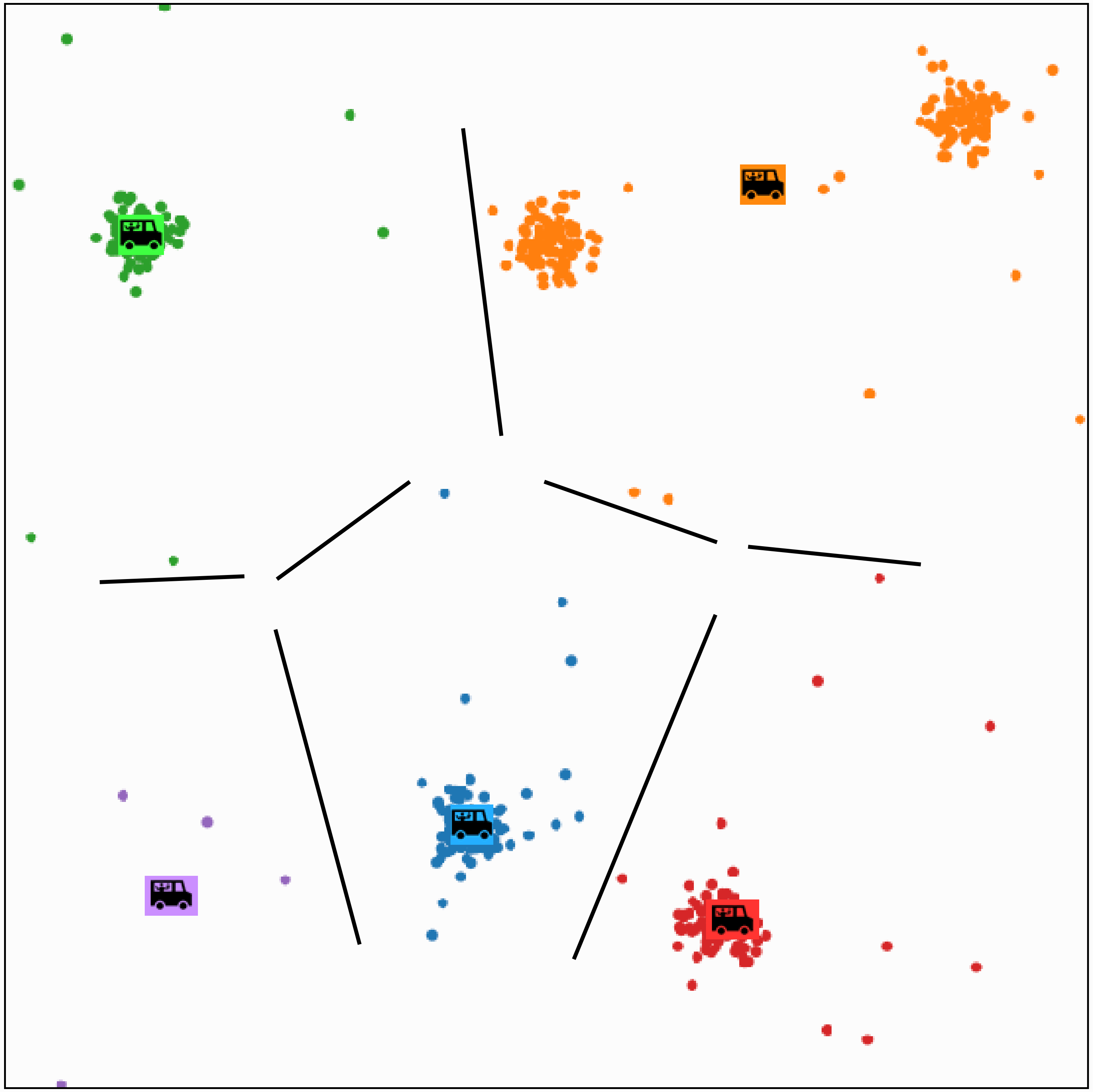
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective
- That local minimum could be bad!
- The initialization can make a big difference

k-means algorithm: initialization



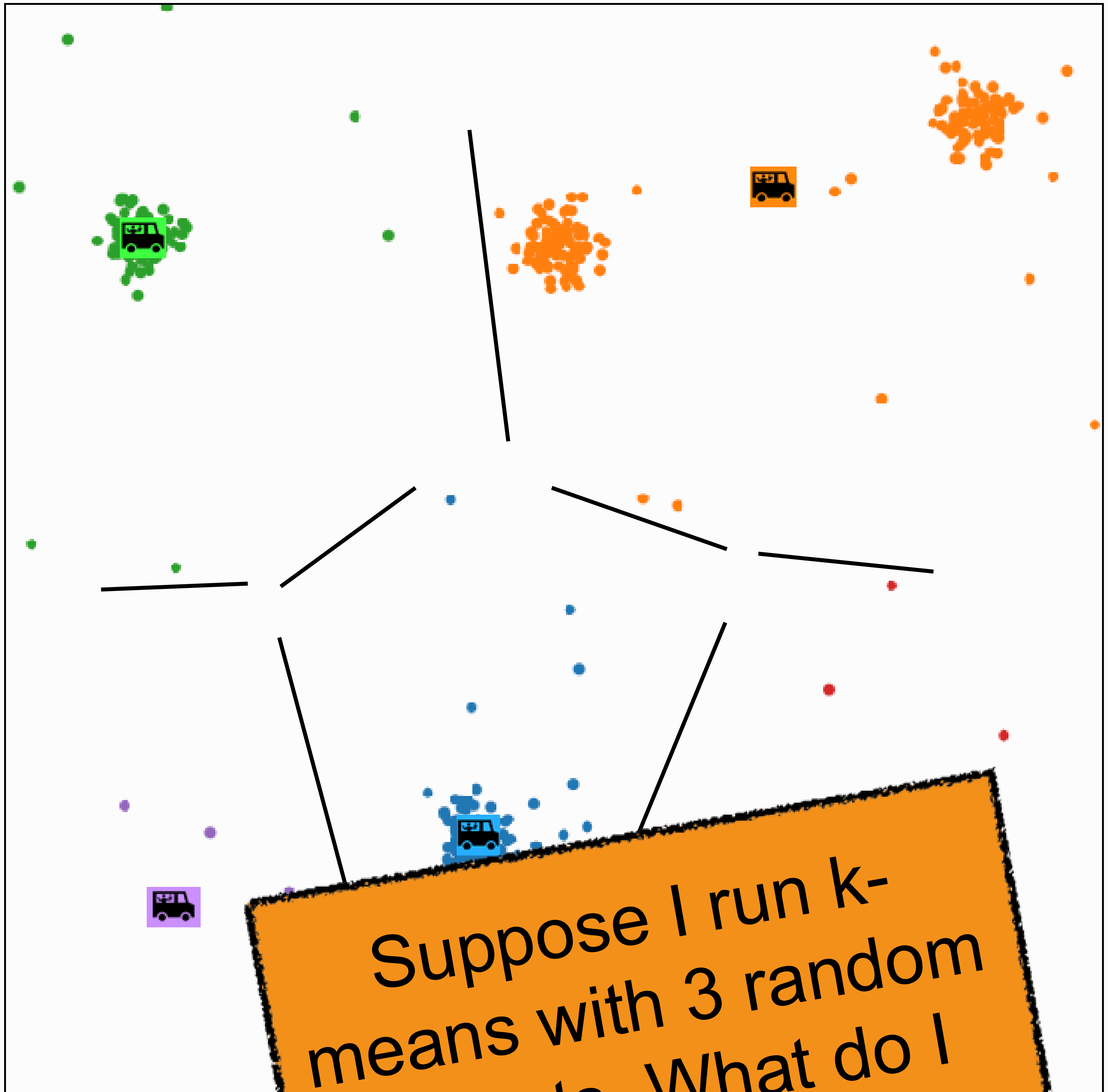
- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective
- That local minimum could be bad!
- The initialization can make a big difference
- Some options:

k-means algorithm: initialization



- **Theorem**. If run for enough outer iterations, 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

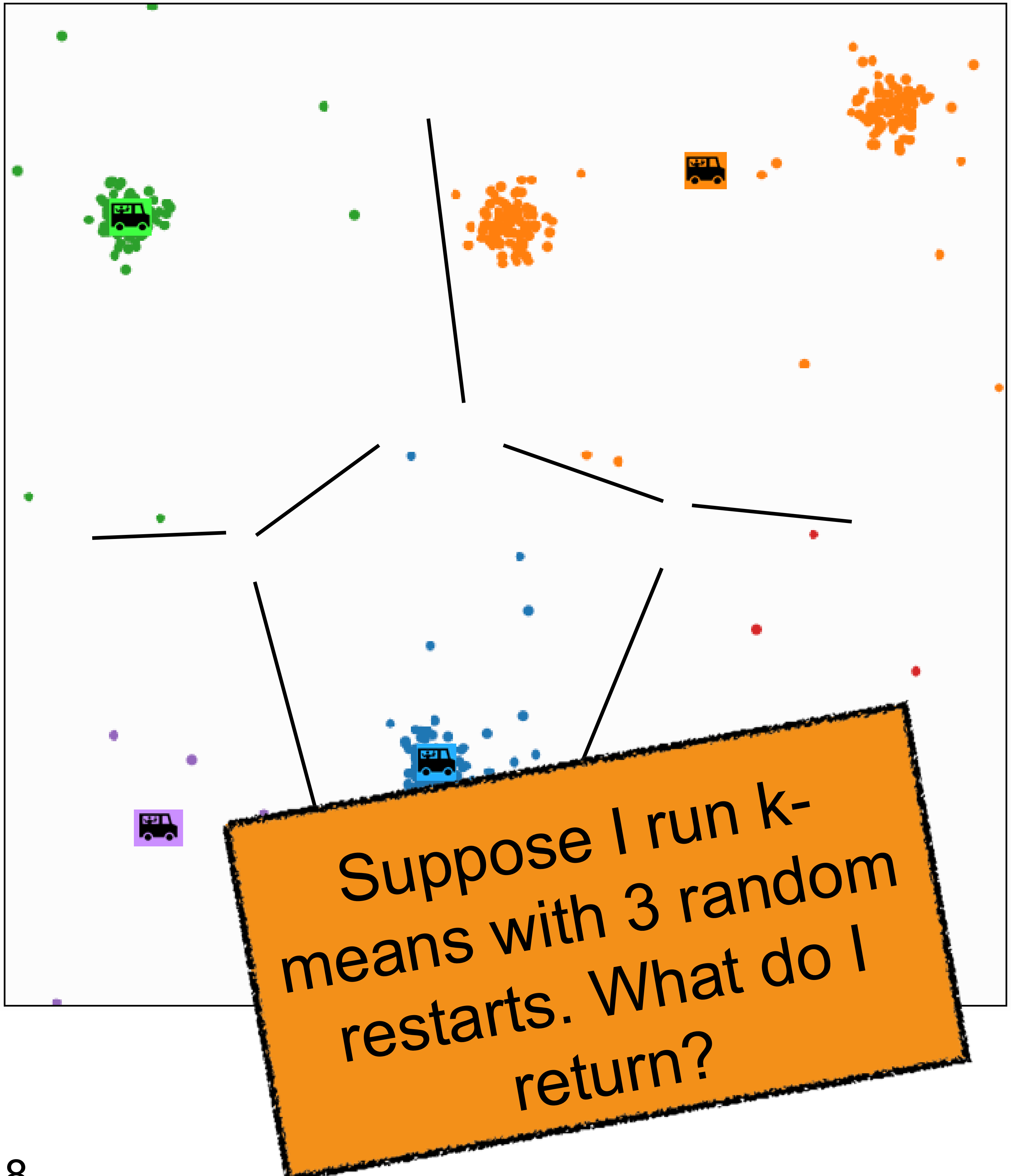
k-means algorithm: initialization



Suppose I run k-means with 3 random restarts. What do I return?

- **Theorem**. If run for enough outer iterations, 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

k-means algorithm: initialization

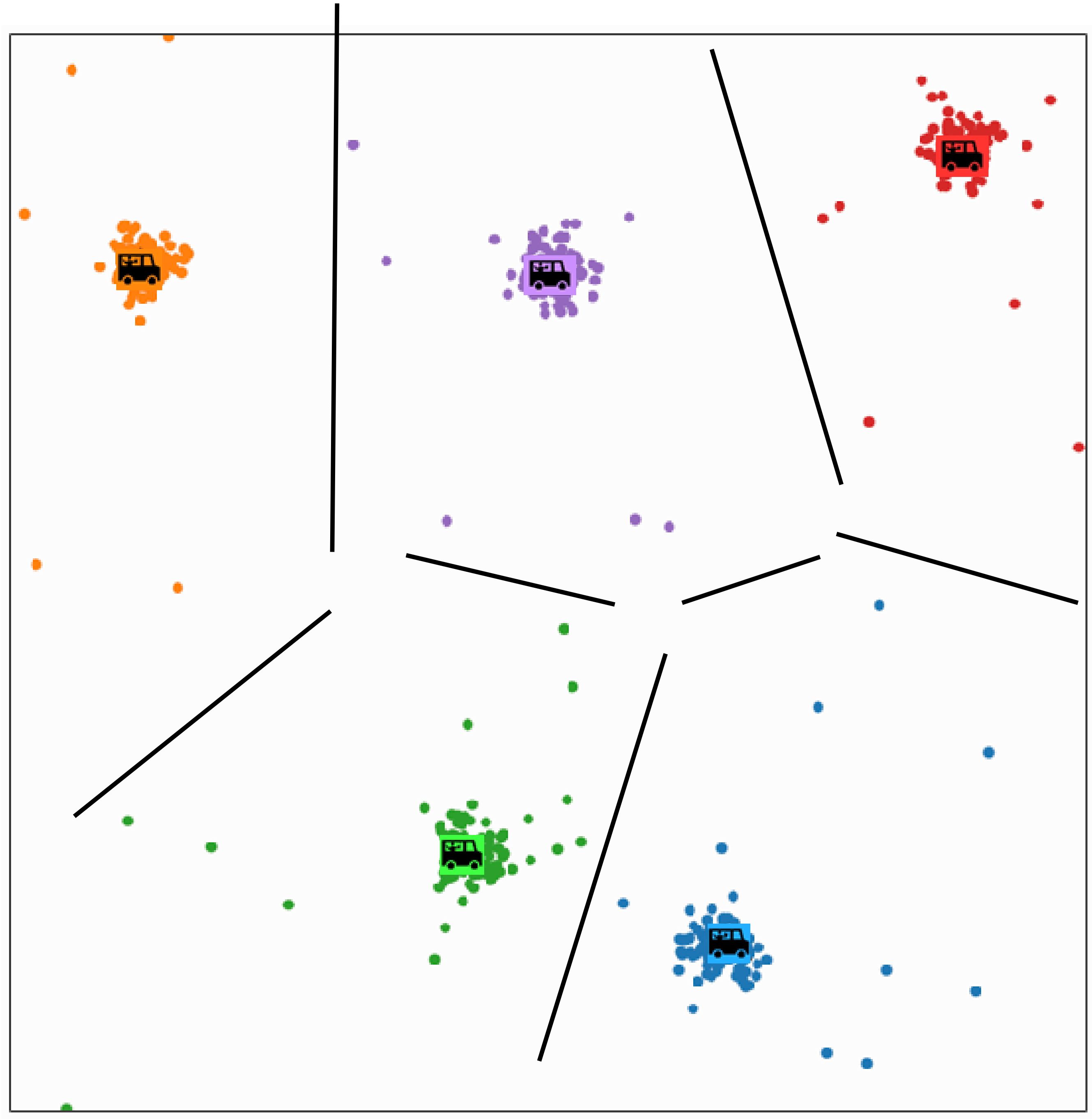


- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective
- That local minimum could be bad!
- The initialization can make a big difference
- Some options: random restarts, k-means++

k-means algorithm: effect of *k*

[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

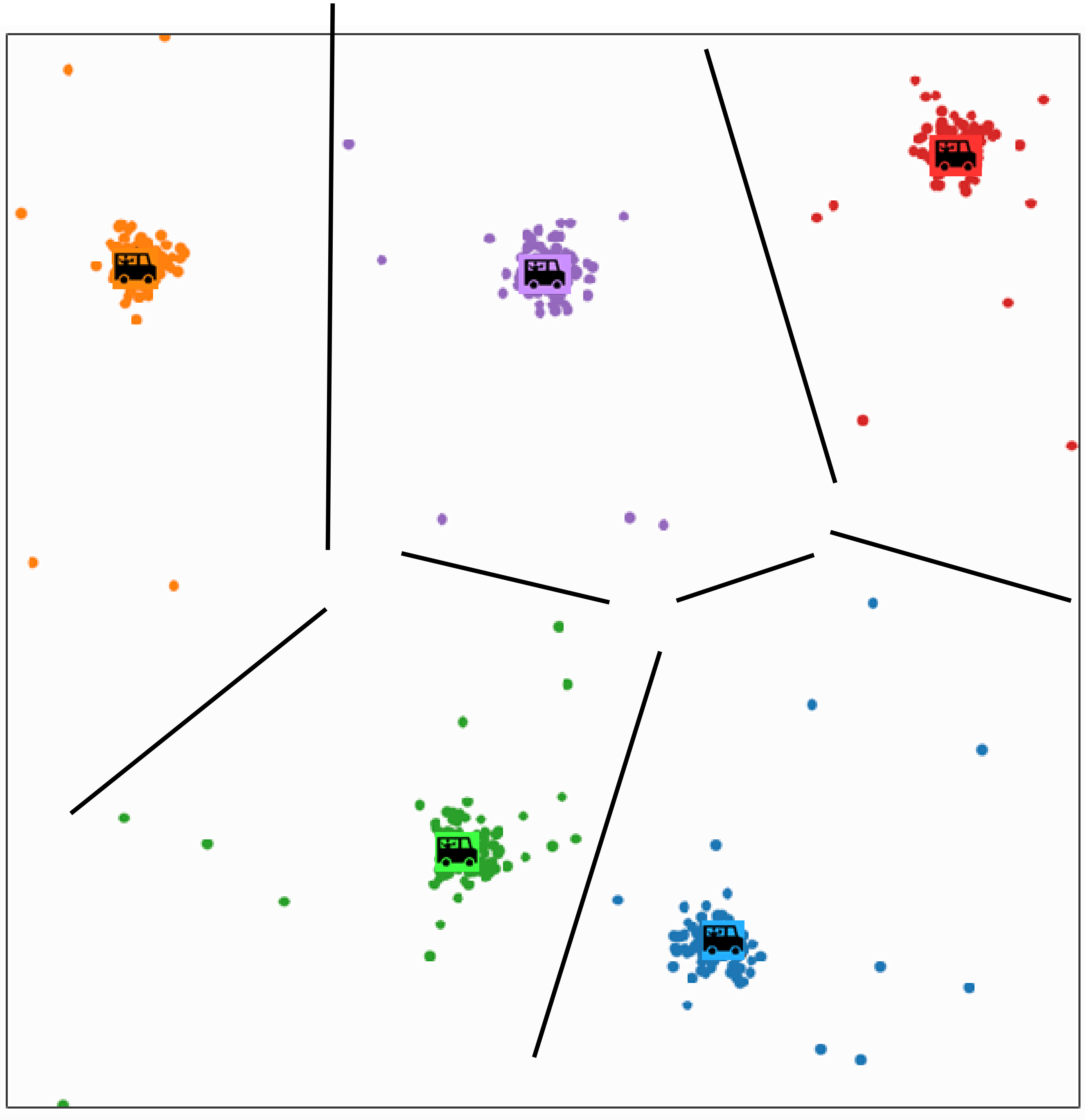
k-means algorithm: effect of *k*



[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm: effect of *k*

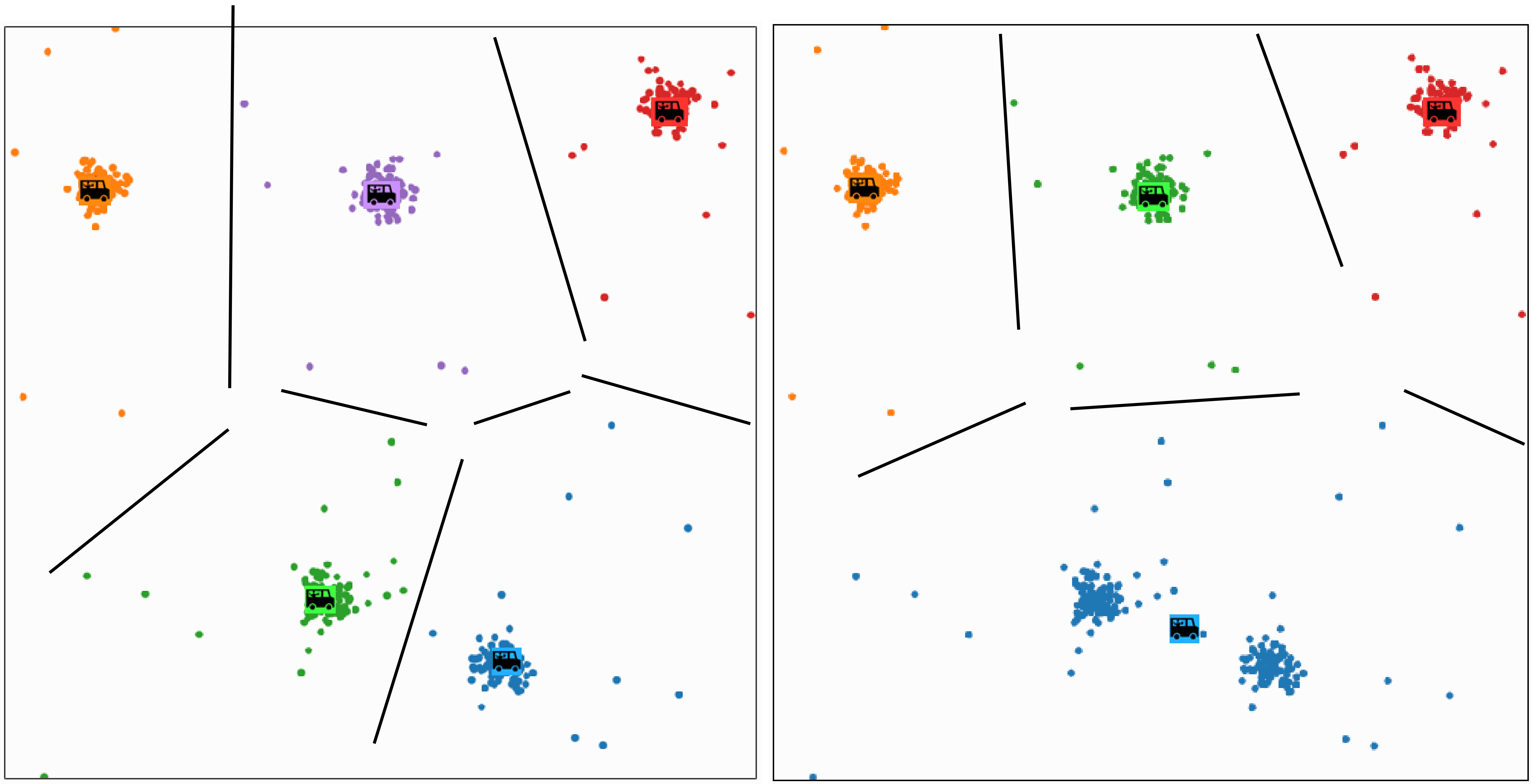
- Different *k* will give us different results



[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm: effect of *k*

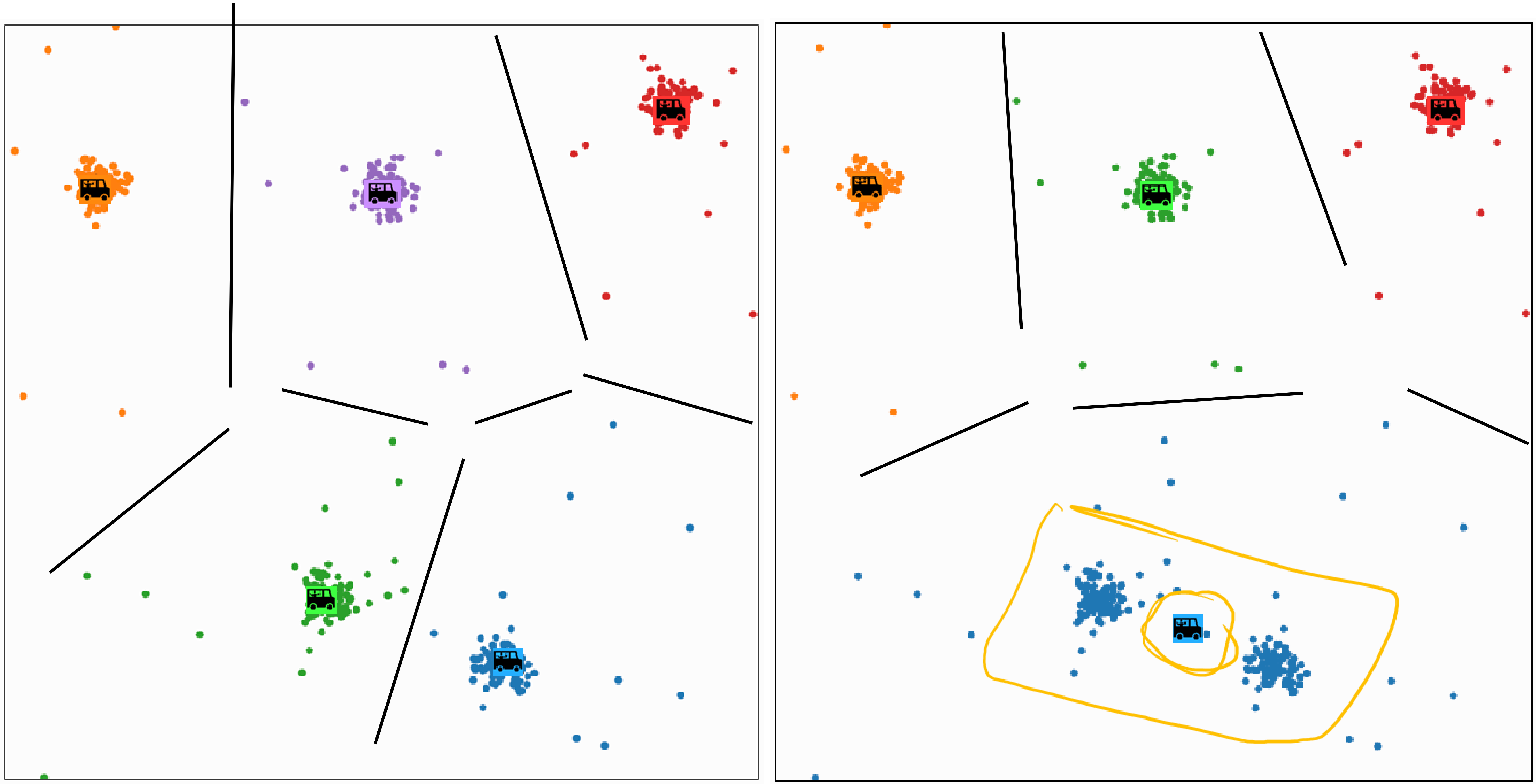
- Different *k* will give us different results



[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm: effect of *k*

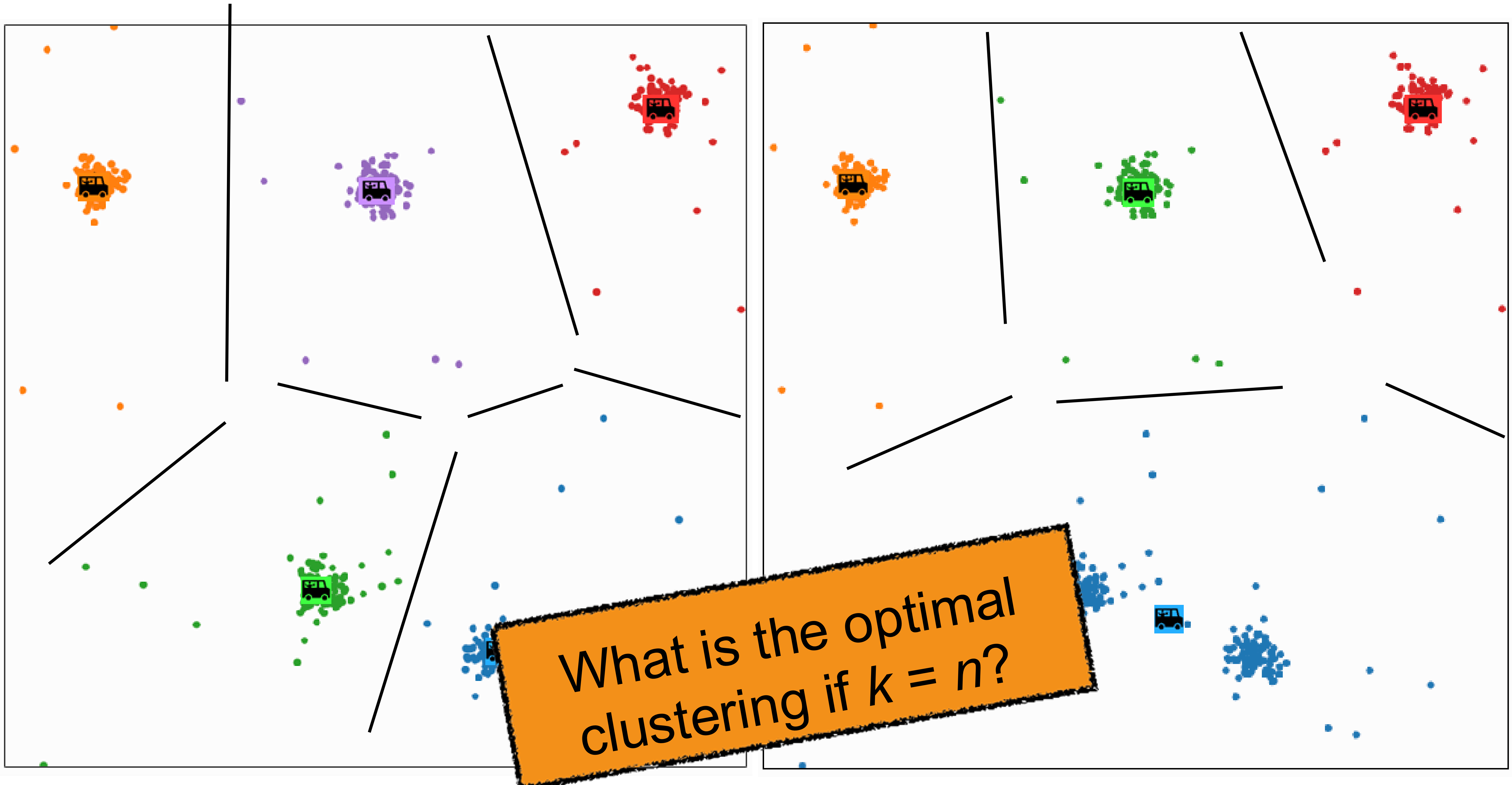
- Different *k* will give us different results
- Larger *k* gets trucks closer to people



[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm: effect of *k*

- Different *k* will give us different results
- Larger *k* gets trucks closer to people



[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm: choosing *k*

k-means algorithm: choosing *k*

- Sometimes we know *k*

k-means algorithm: choosing *k*

- Sometimes we know *k*

k-means algorithm: choosing k

- Sometimes we know k

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram]

k-means algorithm: choosing *k*

- Sometimes we know *k*

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg]

k-means algorithm: choosing *k*

- Sometimes we know *k*




[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg]

k-means algorithm: choosing *k*

- Sometimes we know *k*

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

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

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*




- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

$$\arg\min_{y,\mu} \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\mathbf{1}\{y^{(i)} = j\}} \|x^{(i)} - \mu^{(j)}\|_2^2$$

loss function

$$\arg\min_{y,\mu} \sum_{i=1}^{n} \sum_{j=1}^{k} \underbrace{1\{y^{(i)} = j\}} \underbrace{\|x^{(i)} - \mu^{(j)}\|}_2^2$$

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

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

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

$$\arg\min_{y,\mu,\boxed{k}} \sum_{j=1}^{k} \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} \|x^{(i)} - \mu^{(j)}\|_2^2$$

Why not?

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

$$\arg\min_{y,\mu,\boxed{k}} \sum_{j=1}^{k} \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} \|x^{(i)} - \mu^{(j)}\|_2^2$$

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*




- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

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

- How to choose *k* depends on what you'd like to do

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

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

- How to choose *k* depends on what you'd like to do
  - E.g. cost-benefit trade-off

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*

- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

$$\arg\min_{y,\mu,k} \sum_{j=1}^{k} \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} \|x^{(i)} - \mu^{(j)}\|_2^2 + \mathrm{cost}(k)$$

- How to choose *k* depends on what you'd like to do
  - E.g. cost-benefit trade-off

[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

k-means algorithm: choosing *k*

- Sometimes we know *k*




- Sometimes we'd like to choose/learn *k*
  - Can't just minimize the k-means objective over *k* too

$$\arg\min_{y,\mu,k} \sum_{j=1}^{k} \sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} \|x^{(i)} - \mu^{(j)}\|_2^2 + \underline{\mathrm{cost}(k)}$$

learn k

- How to choose *k* depends on what you'd like to do
  - E.g. cost-benefit trade-off
  - Often no single "right answer"

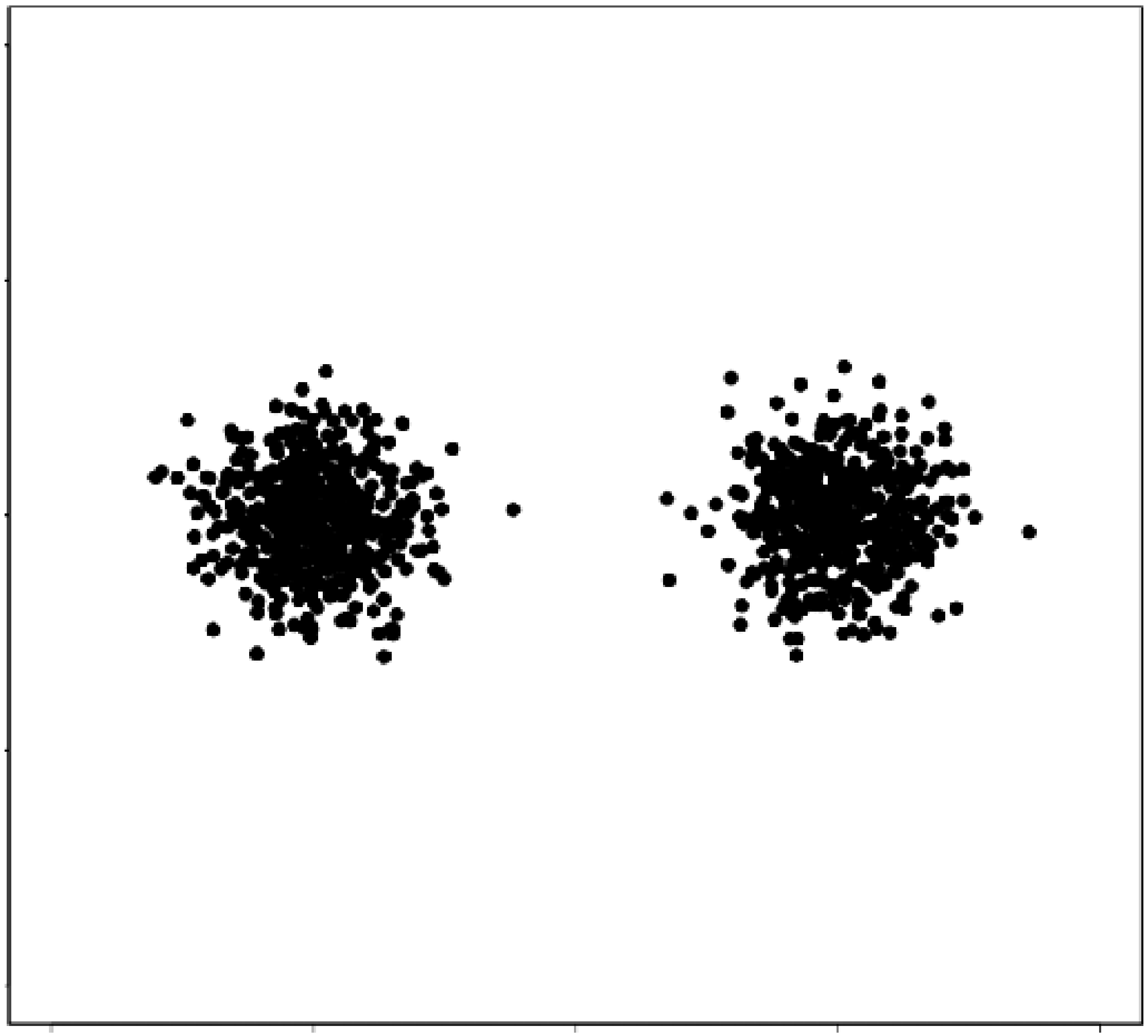
[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

Cluster shape

Cluster shape

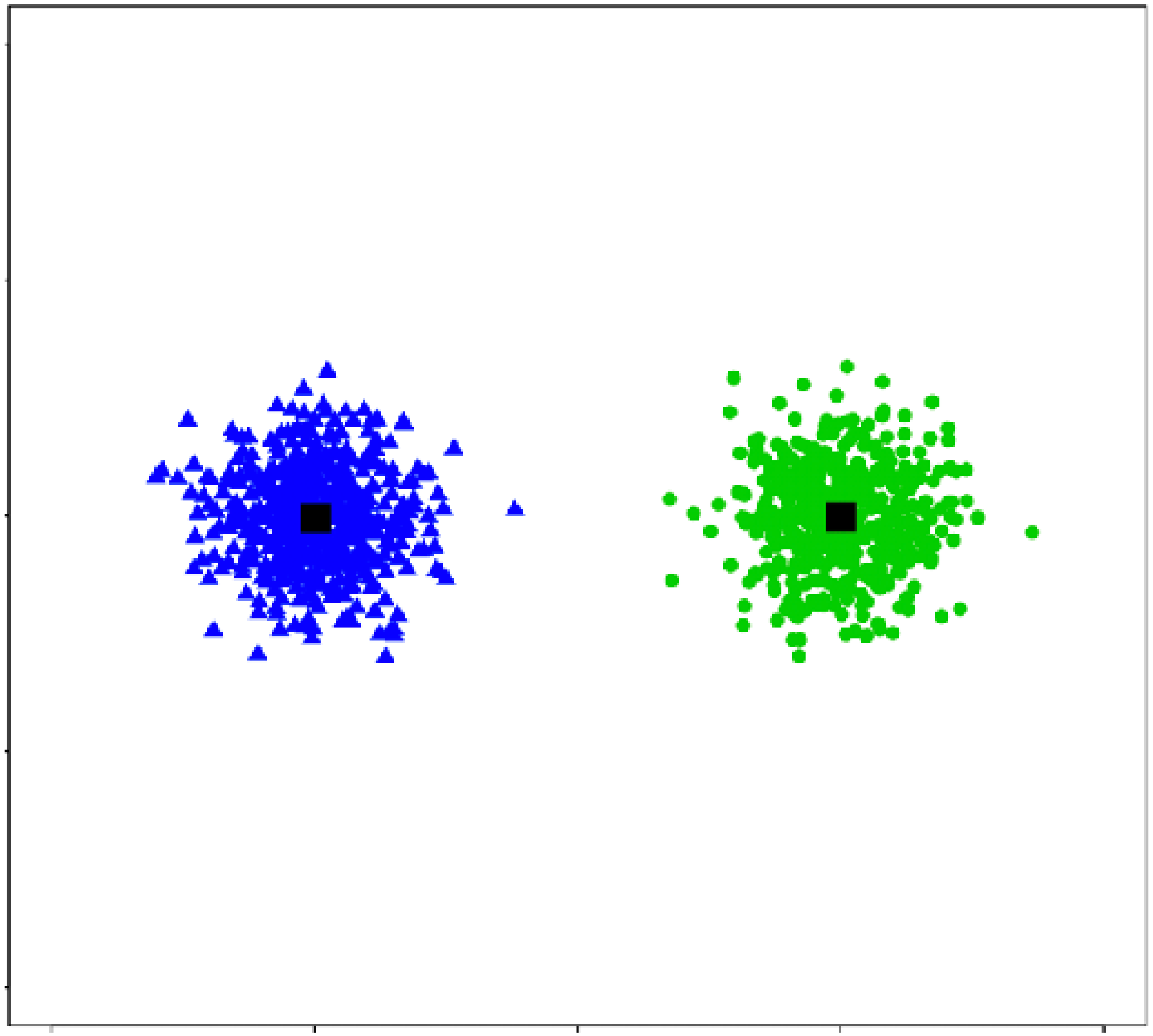
- k-means works well for well-separated circular clusters of the same size

Cluster shape



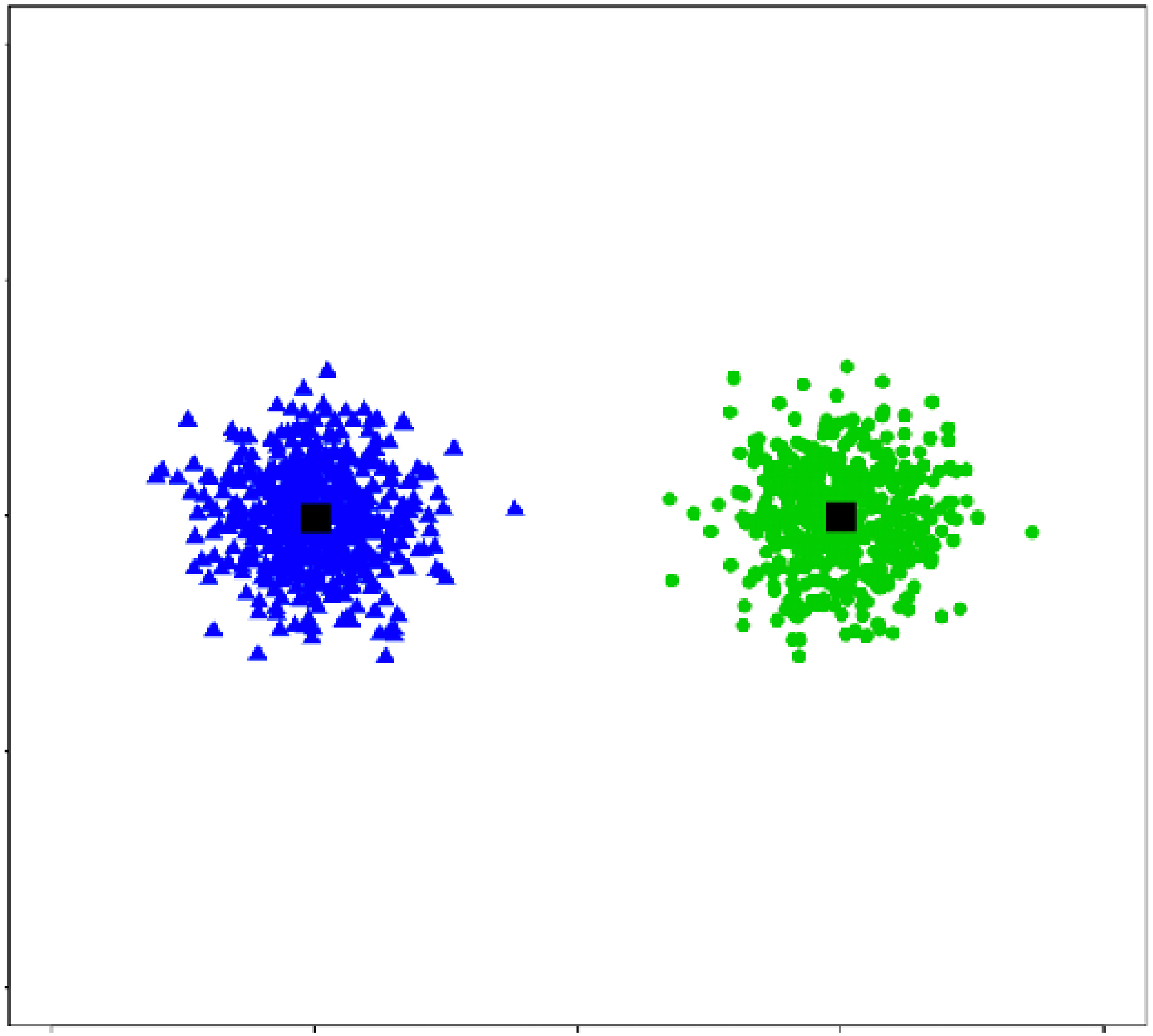
- k-means works well for well-separated circular clusters of the same size

Cluster shape



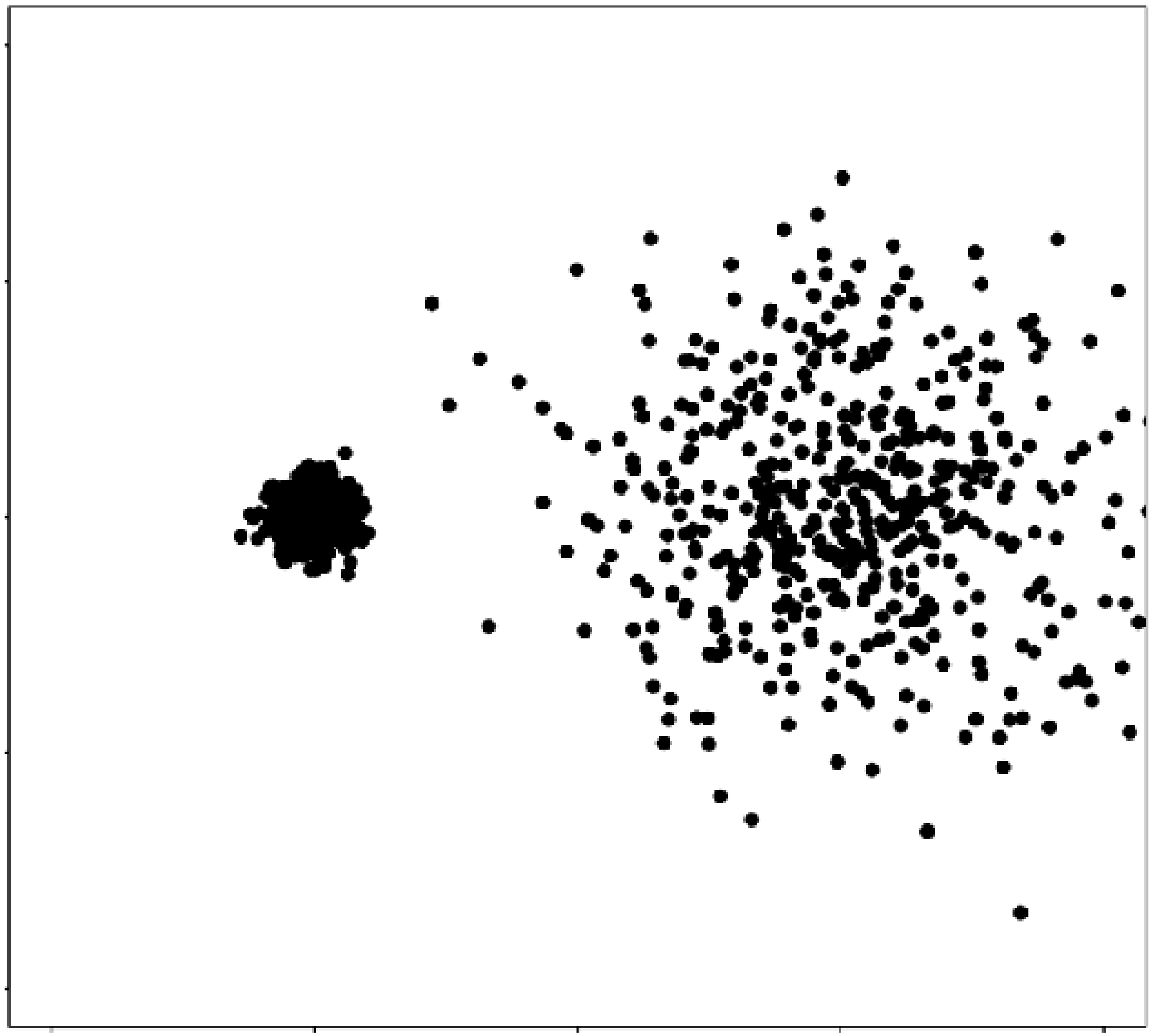
- k-means works well for well-separated circular clusters of the same size

Cluster shape



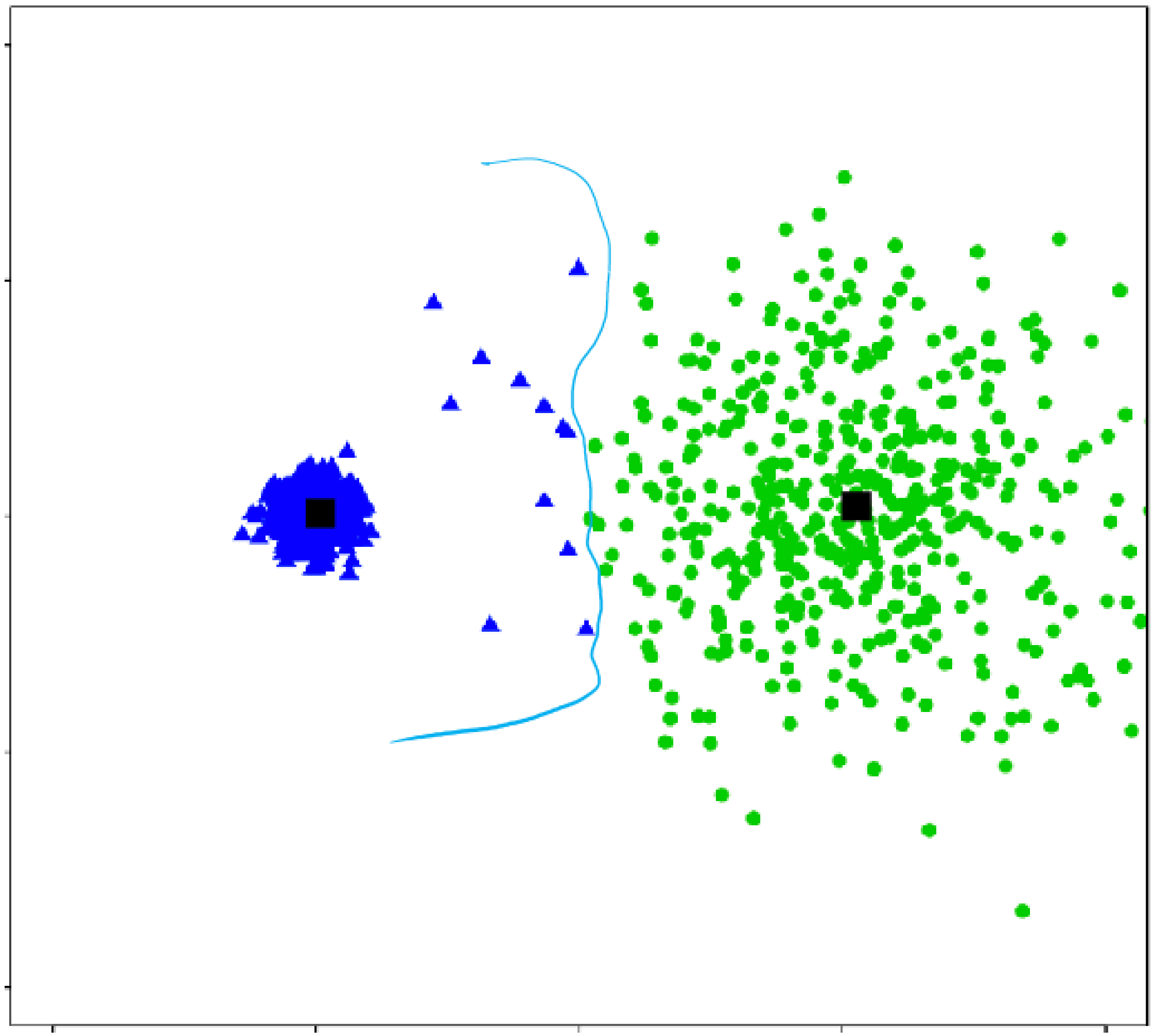
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



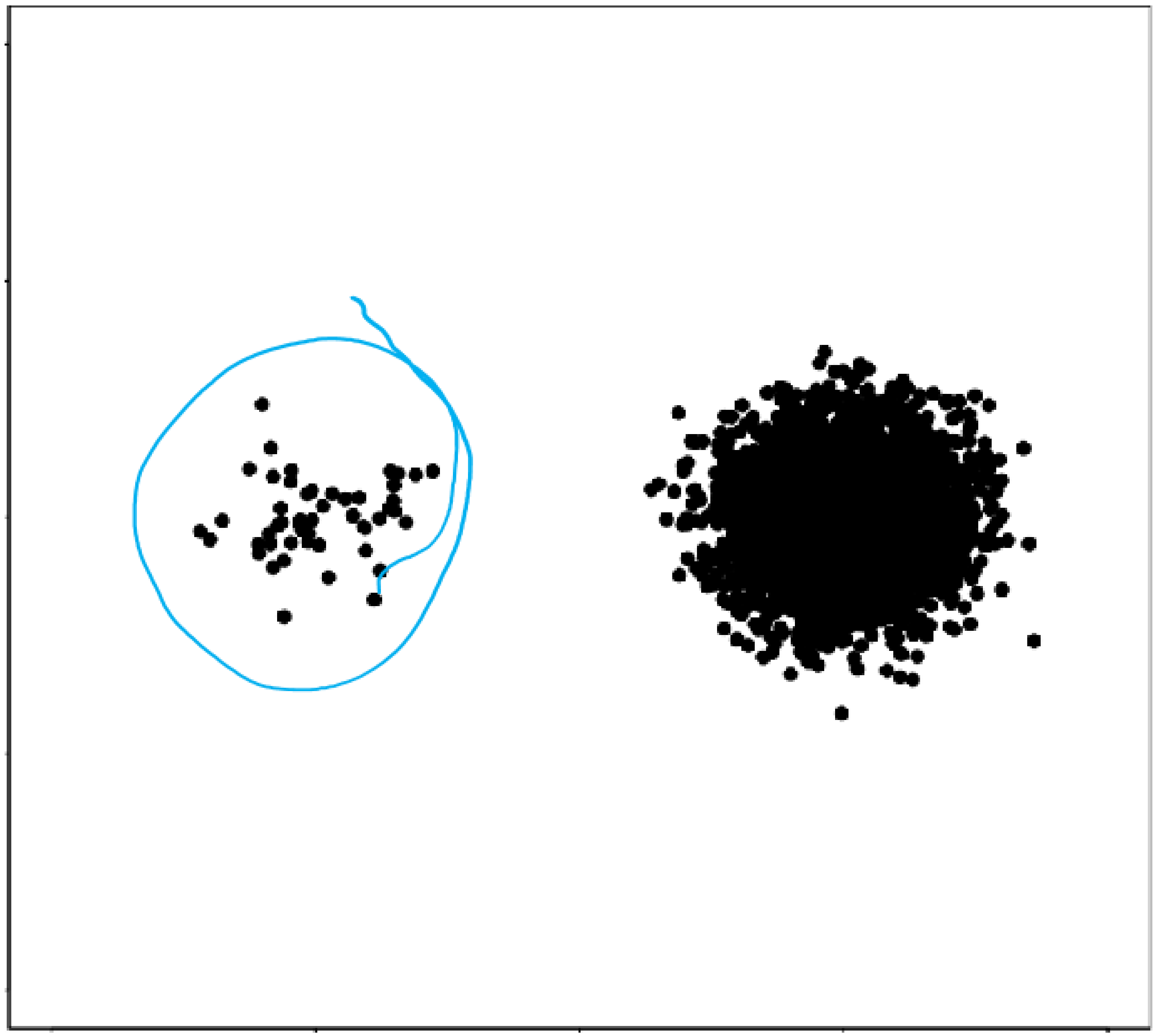
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



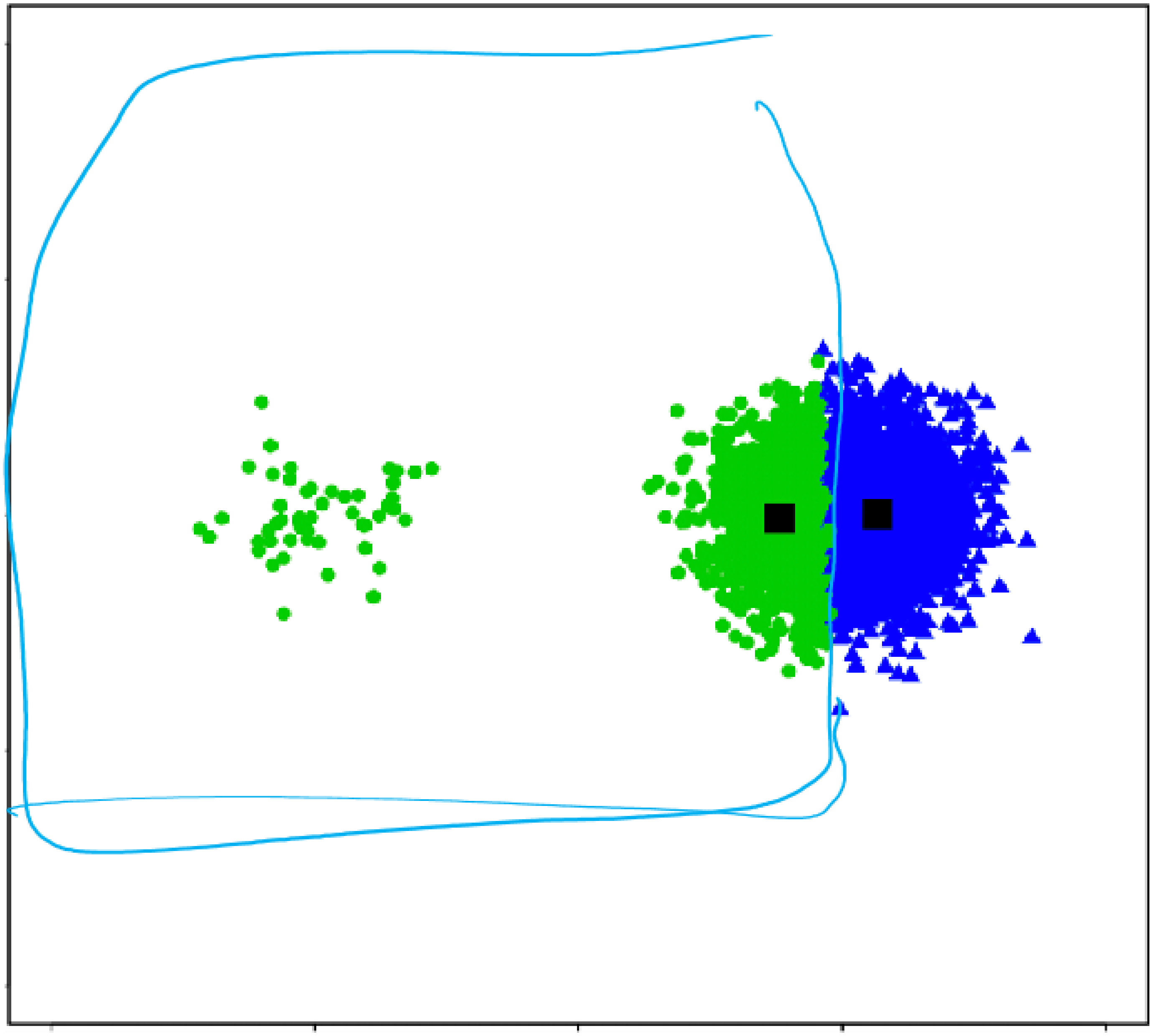
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



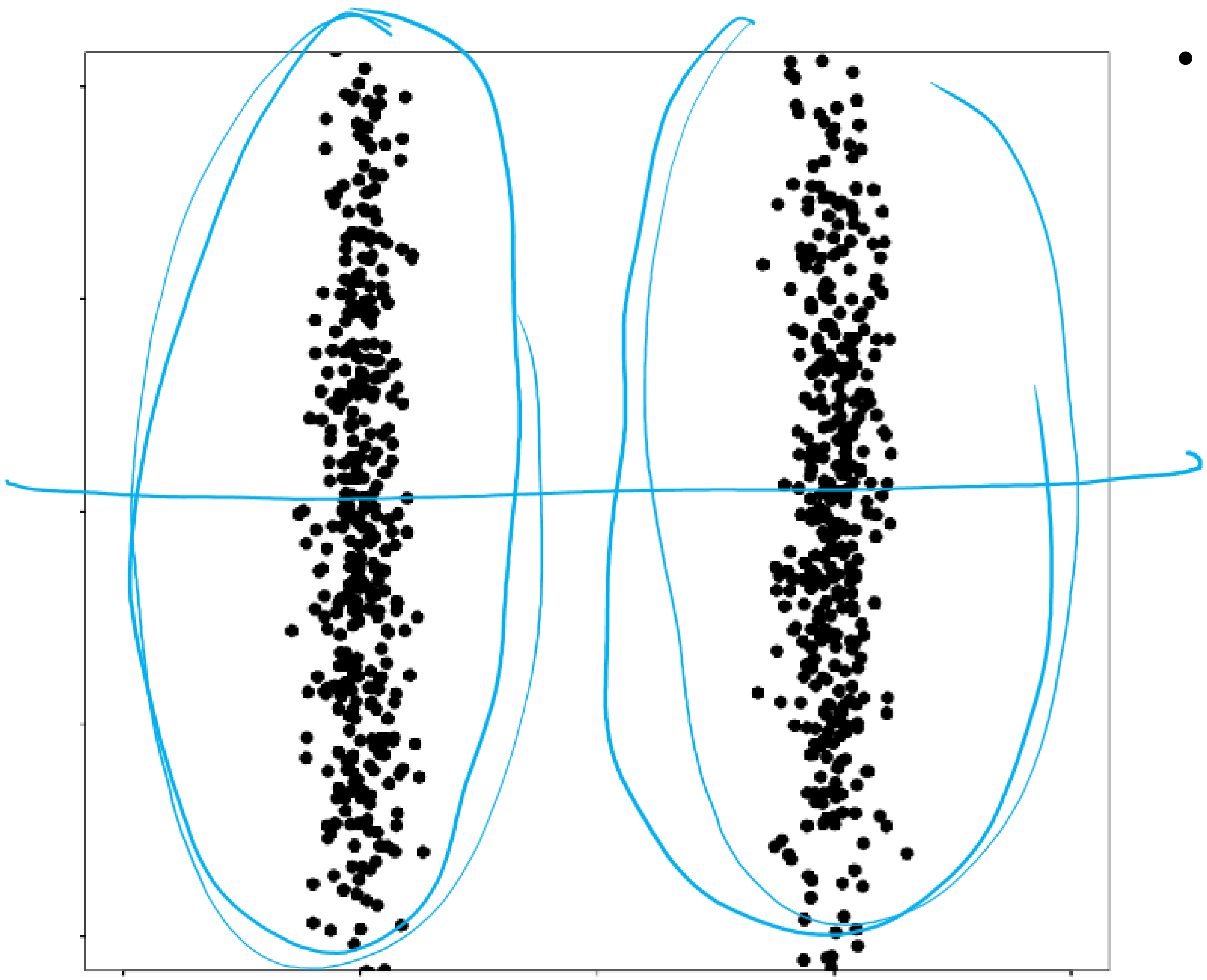
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



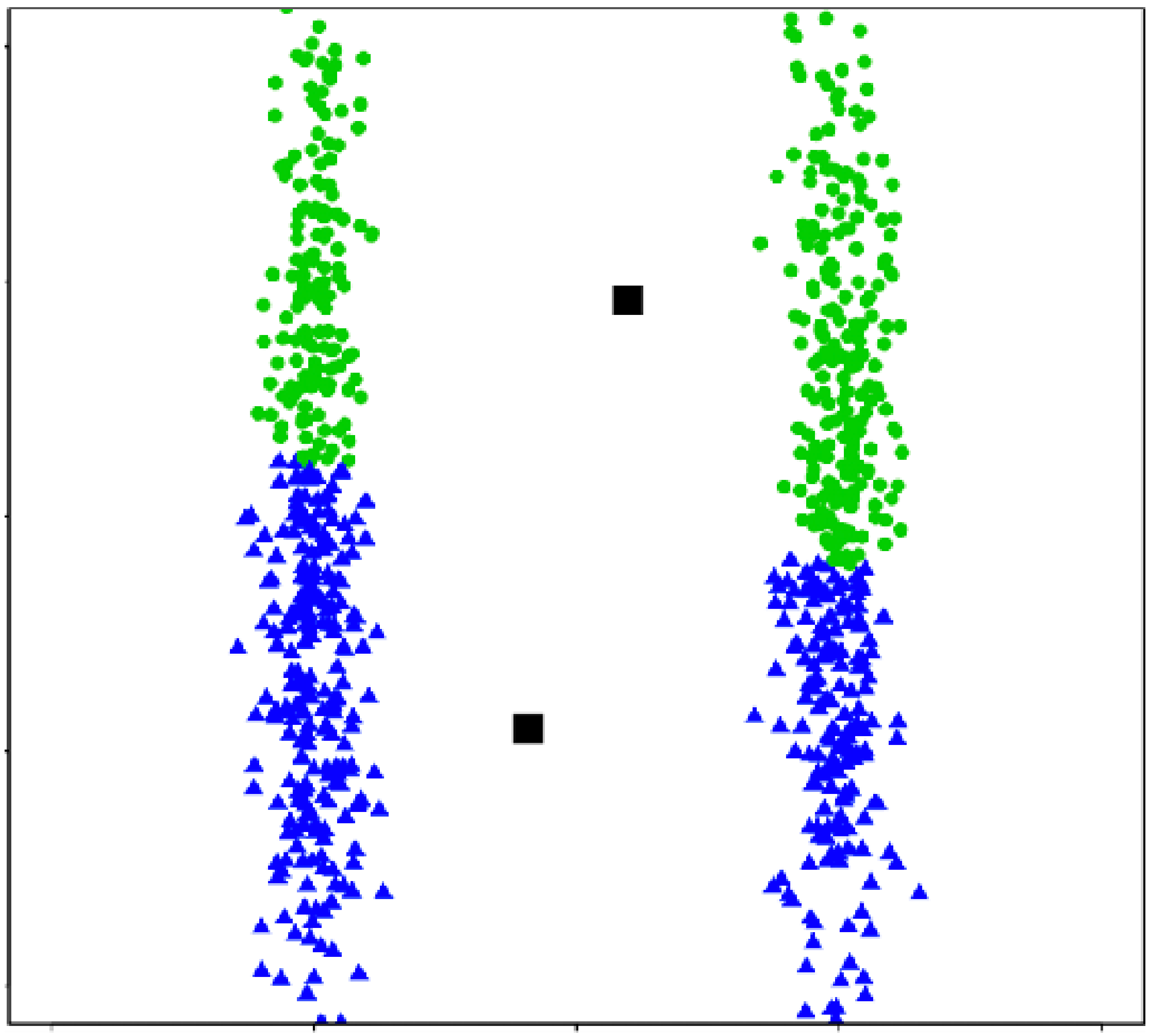
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



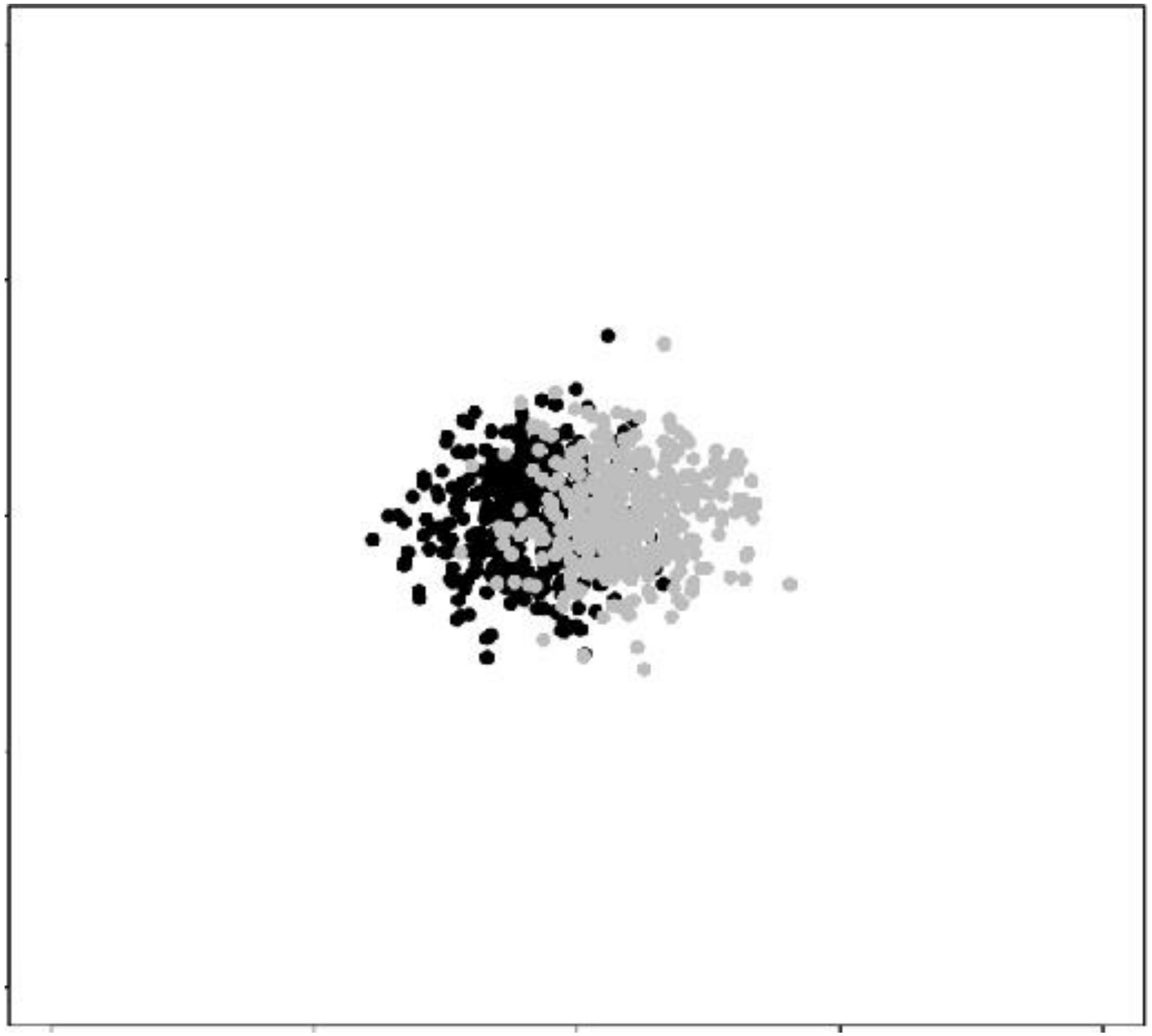
- k-means works well for well-separated **circular** clusters of the same size

Cluster shape



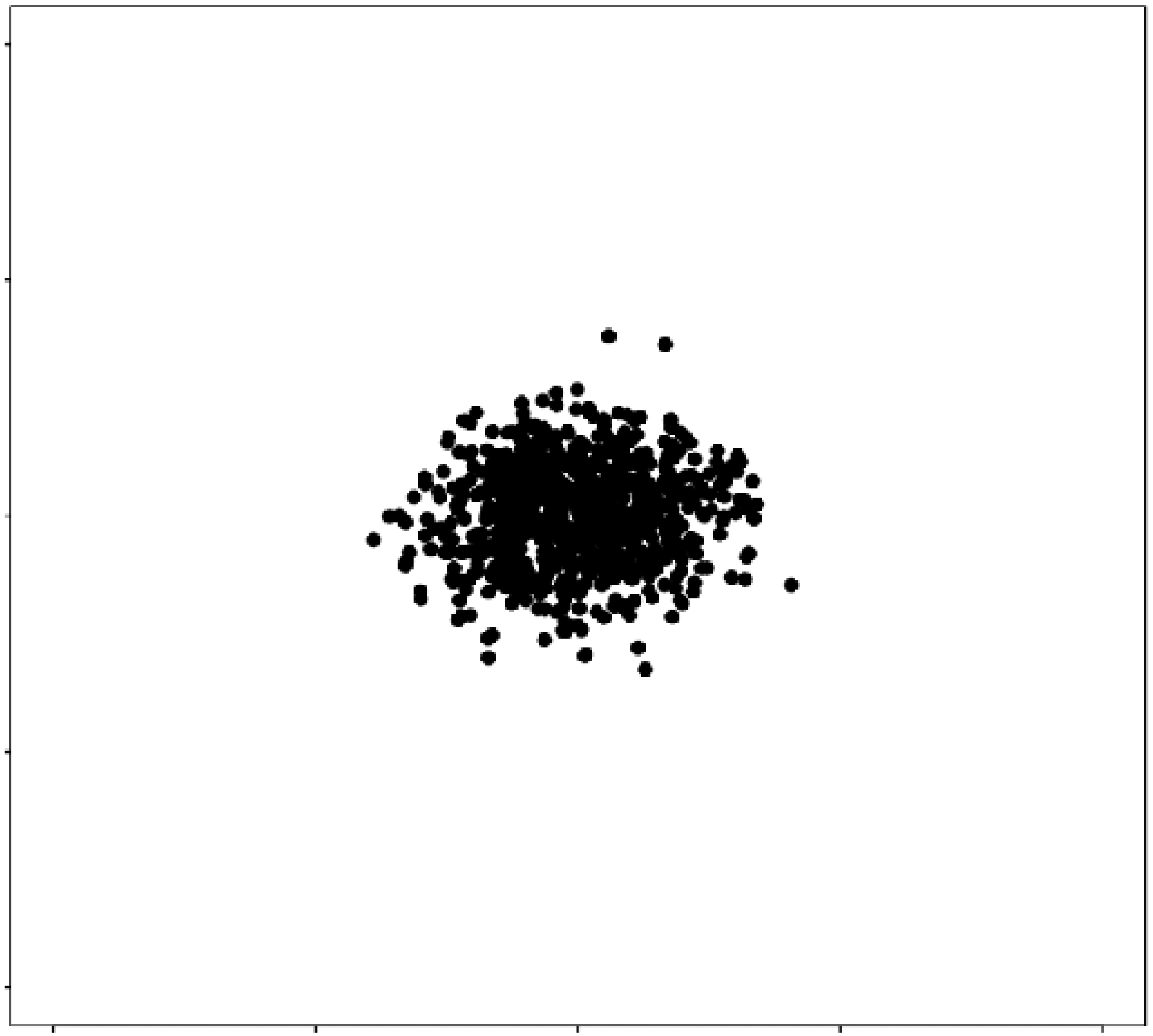
- k-means works well for well-separated **circular** clusters of the same size

Cluster shape



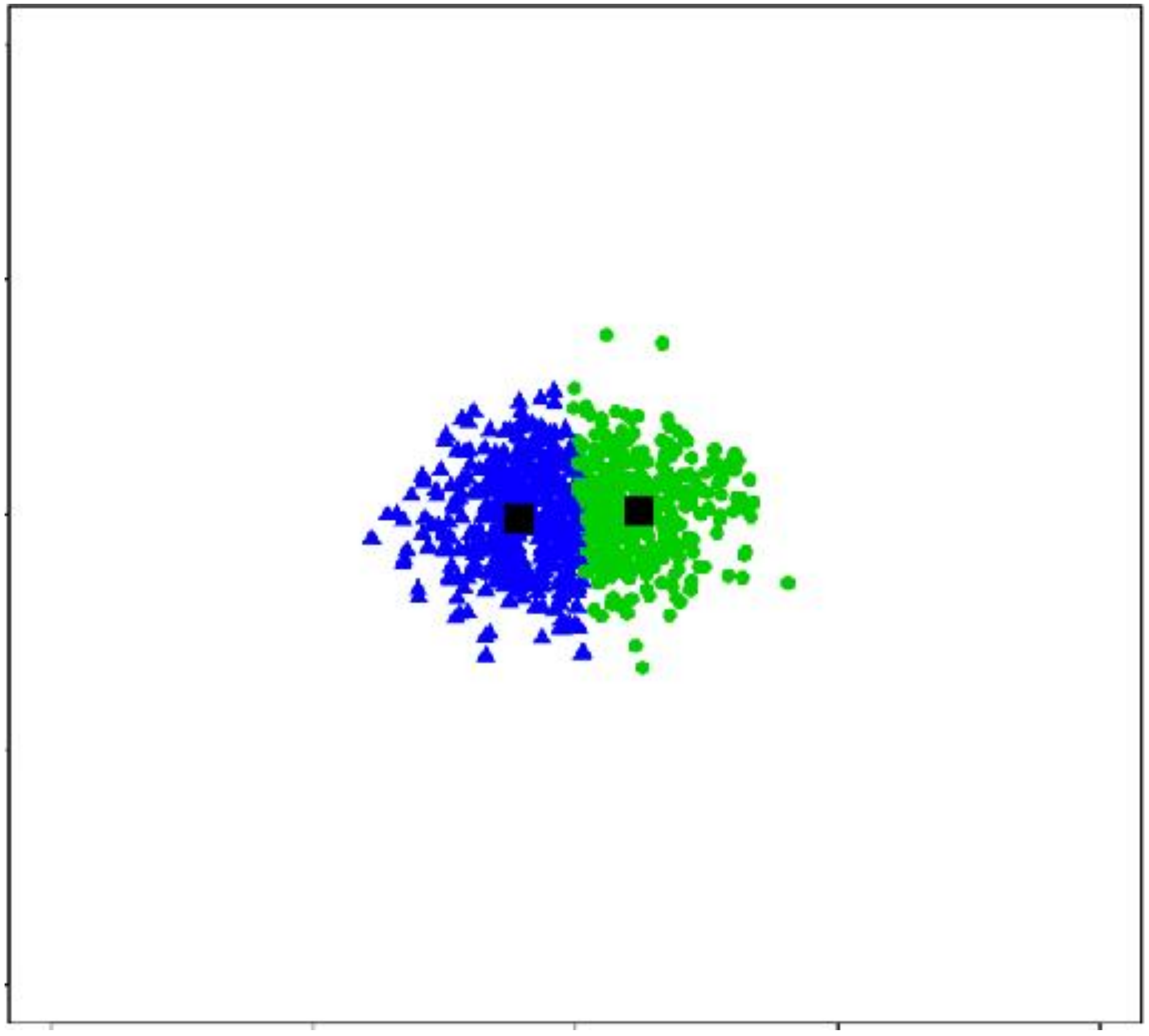
- k-means works well for **well-separated** circular clusters of the same size

Cluster shape



- k-means works well for **well-separated** circular clusters of the same size

Cluster shape



- k-means works well for **well-separated** circular clusters of the same size

Machine learning (ML): why & what

k Nearest Neighbors
Linear Regression & Ridge Regression
Decision Tree, Random Forest
(Bagging)

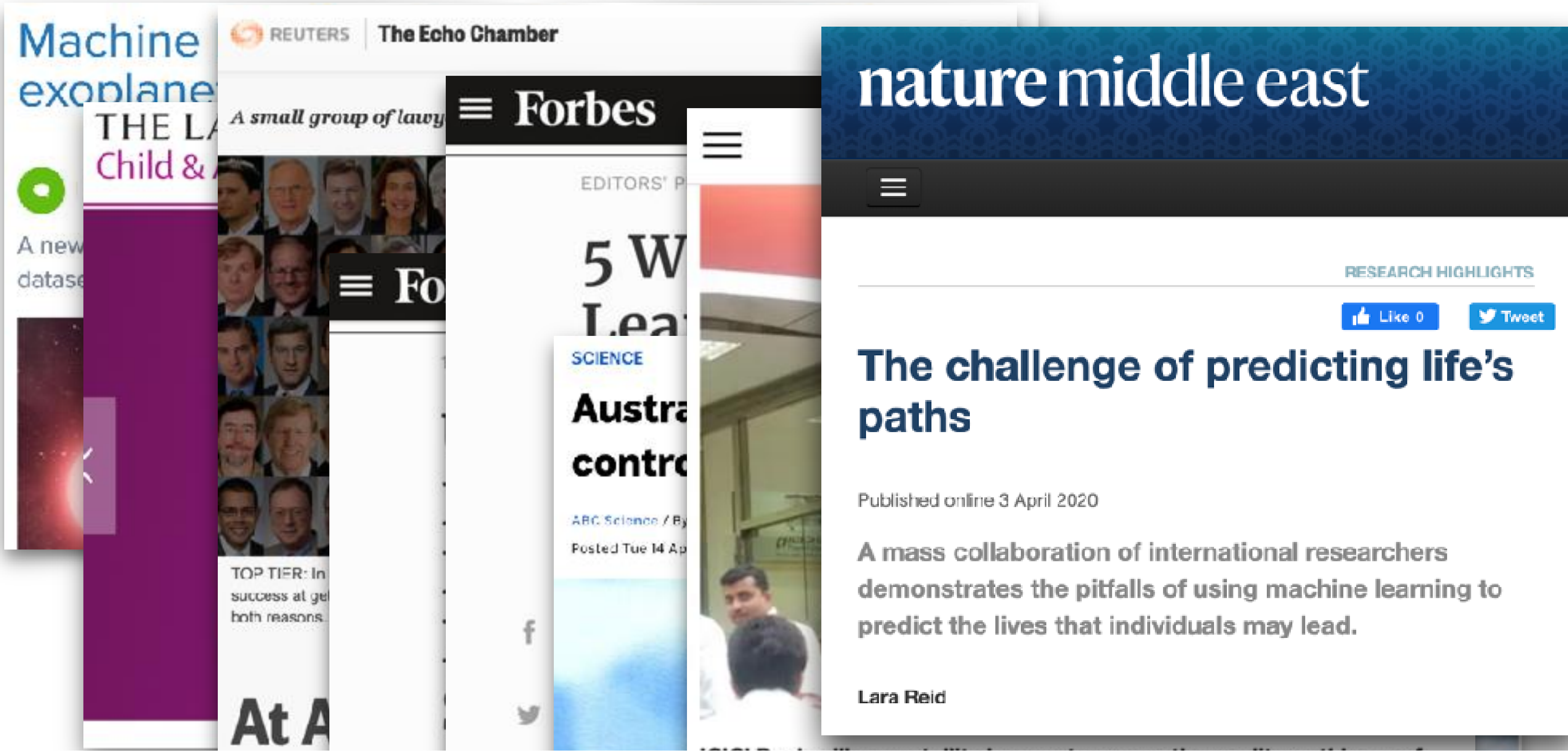
Logistic Regression
Naive Bayes
SVM
Perceptrons
Bias & Variance Trade off
Kmeans Clustering

PCA, Linear Algebra
(Principal Component Analysis)

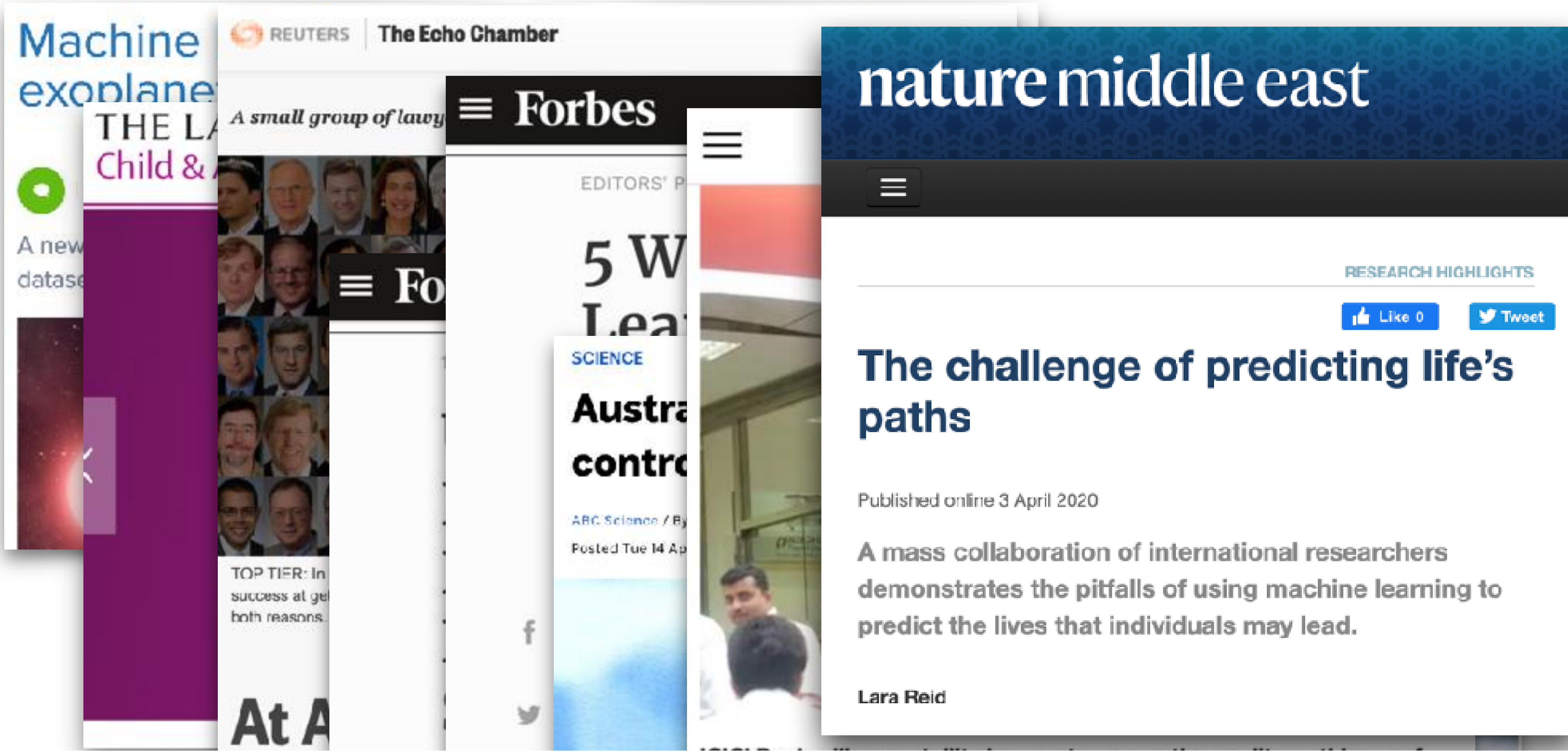
Basics of Info. Theory
Accuracy, Precision, Recall, F1 score, confusion matrix
Class Imbalance
Encoding schemes
~~Bia~~

$f: X \rightarrow Y$

Machine learning (ML): why & what

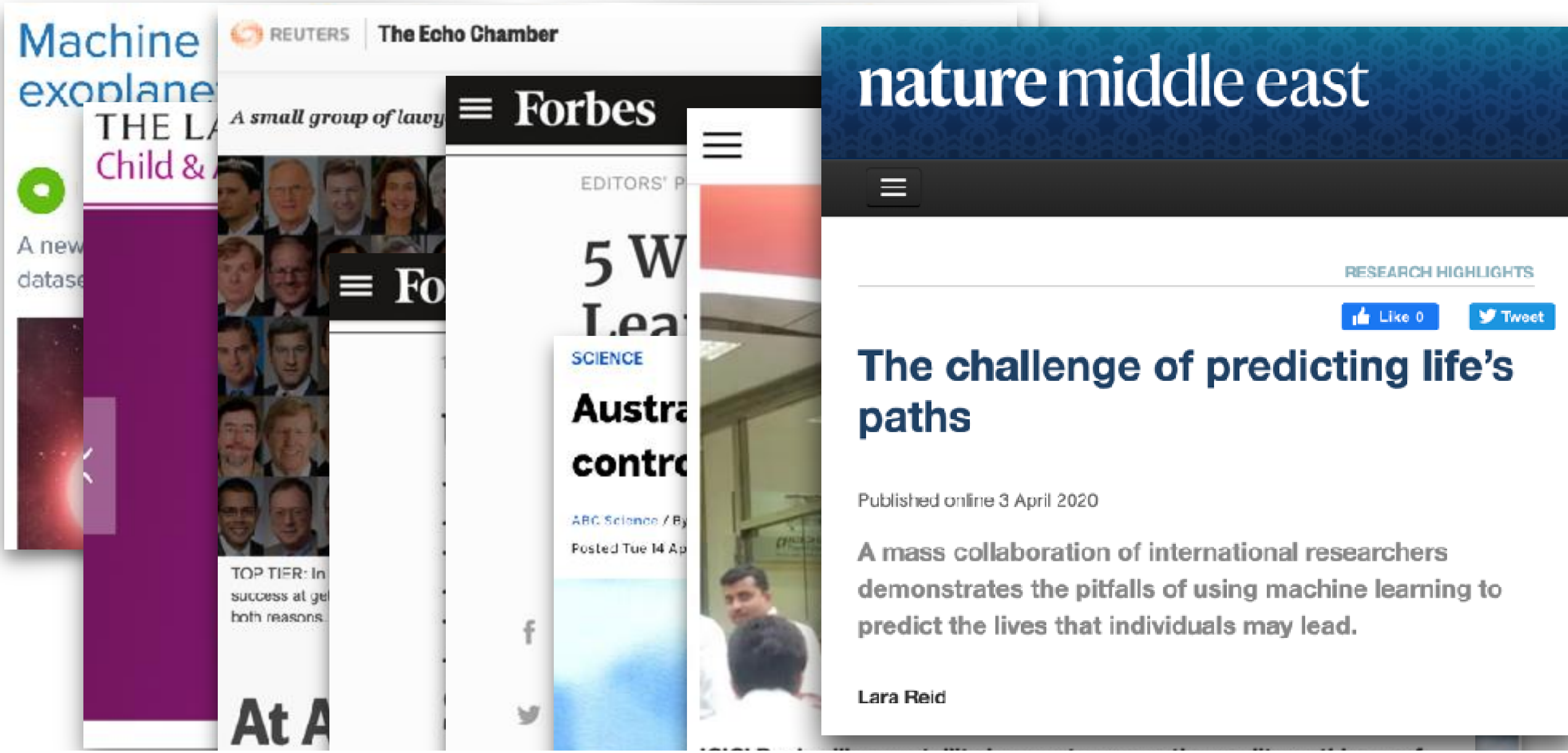


Machine learning (ML): why & what



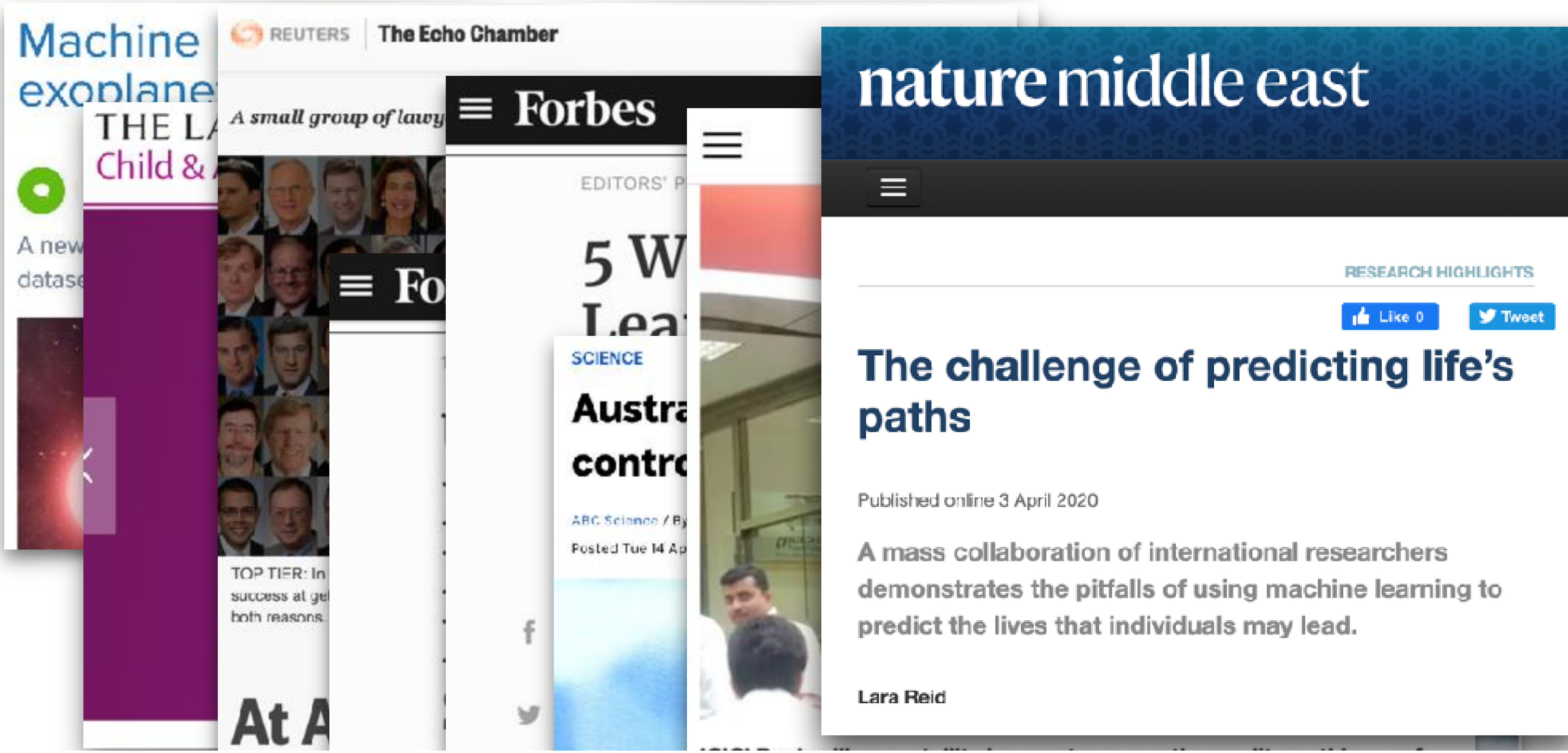
- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)

Machine learning (ML): why & what



- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate

Machine learning (ML): why & what



- **What is ML?** A set of methods for making decisions from data. (See the rest of the course!)
- **Why study ML?** To apply; to understand; to evaluate
- **Notes:** ML is not magic. ML is built on math.