Clustering

Prof. Tamara Broderick

Edited From 6.036 Fall21 Offering

Food distribution placement

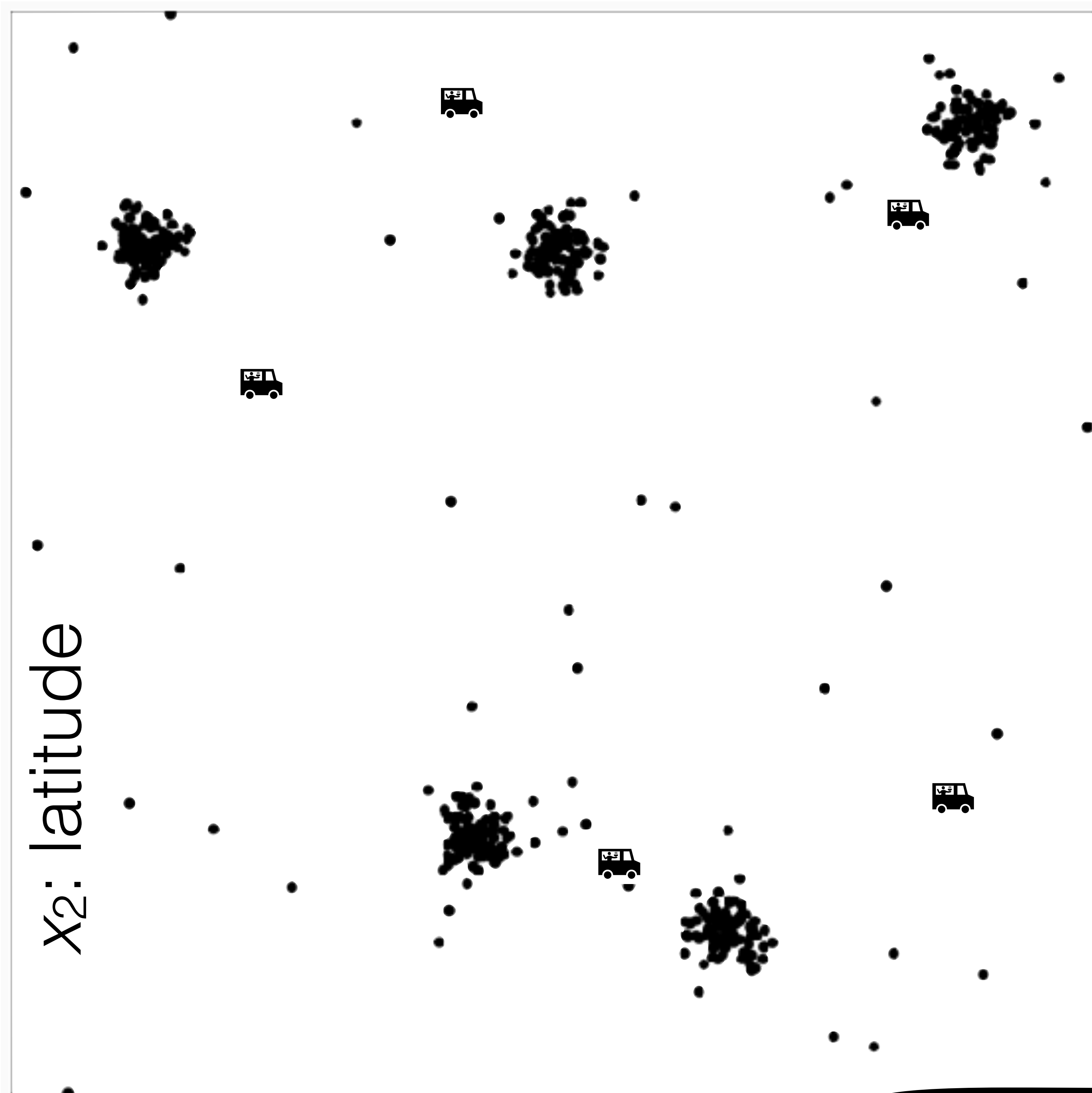
FEEDING AMERICA

MEALS on WHEELS AMERICA

TOGETHER, WE CAN DELIVER.

[https://commons.wikimedia.org/wiki/File:Food_Truck_-_The_Noun_Project.svg; https://www.feedingamerica.org/; https://www.mealsonwheelsamerica.org/]

Food distribution placement



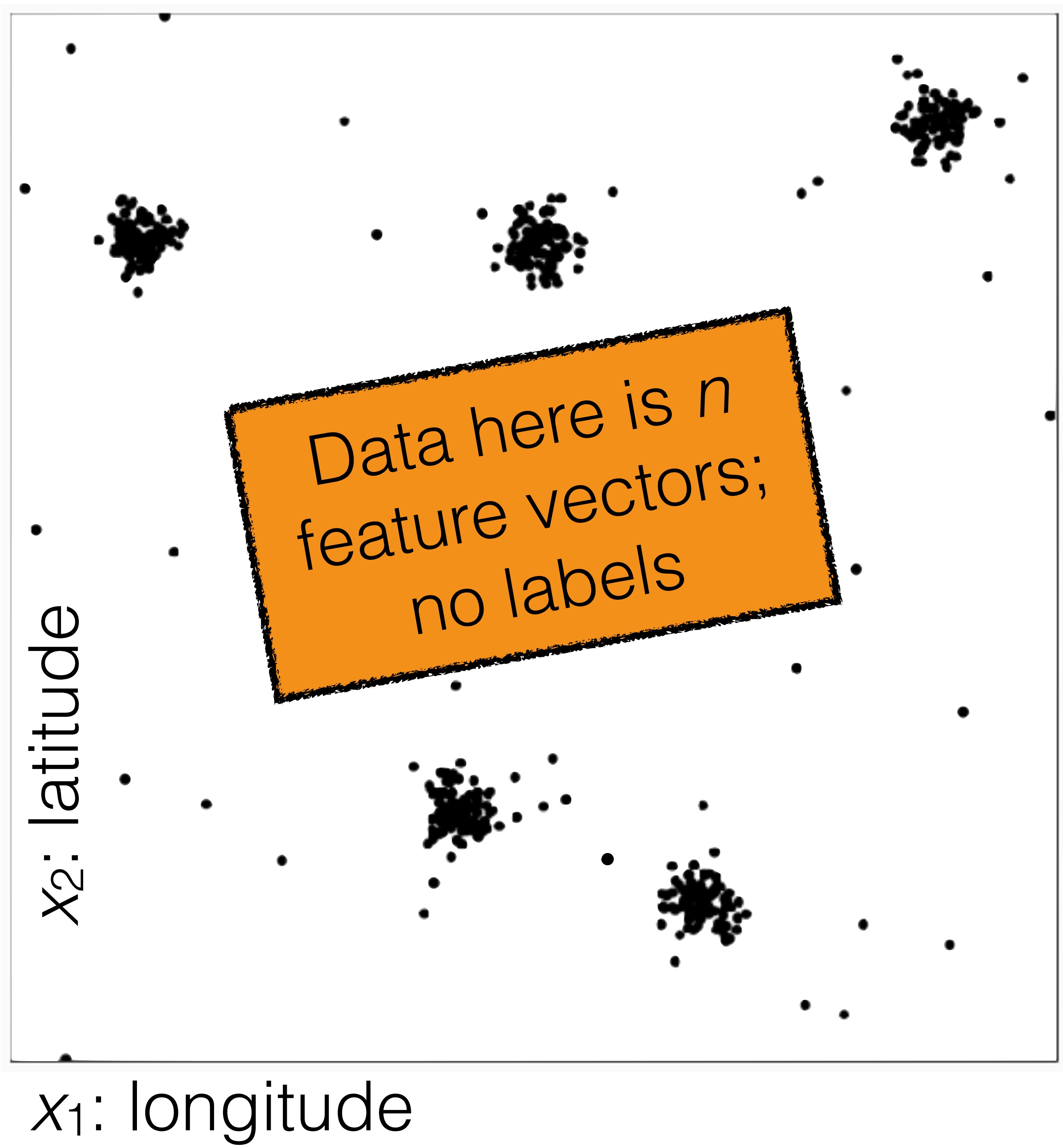
- Where should I have my *k* food trucks park?
- Want to minimize the loss of people we serve
- Person *i* location $x^{(i)}$
- Food truck *j* location $\mu^{(j)}$
- Index of truck where person *i* walks: $y^{(i)}$
- Loss if *i* walks to truck *j*:
$$\|x^{(i)} - \mu^{(j)}\|_2^2$$
- Loss across all people:

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

- a.k.a. *k-means objective*

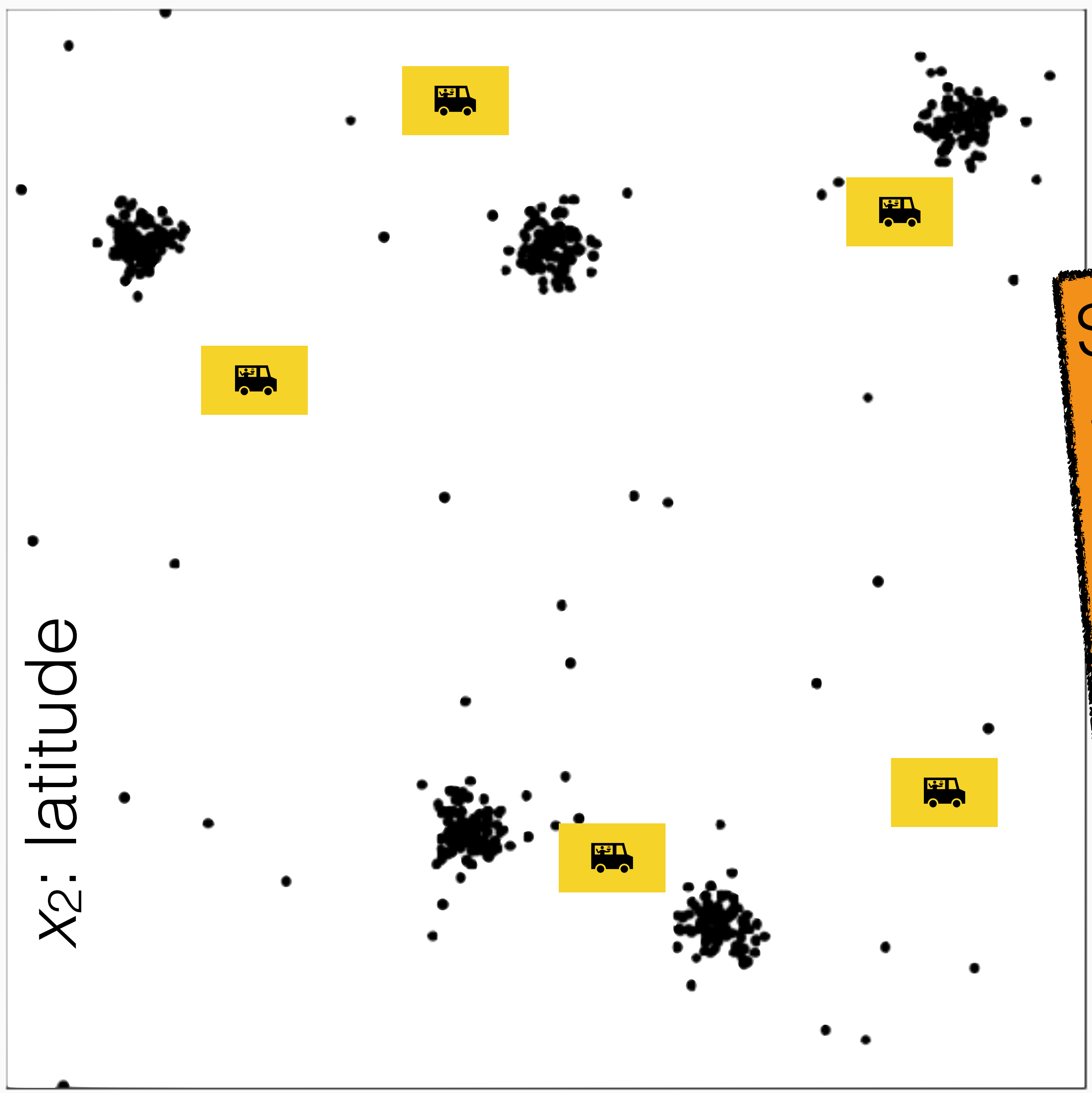
[https://stanford.edu/class/engr108/visualizations/kmeans/kmeans.html]

k-means algorithm



`k-means(k, τ)`

k-means algorithm



x_1: longitude

x_2: latitude

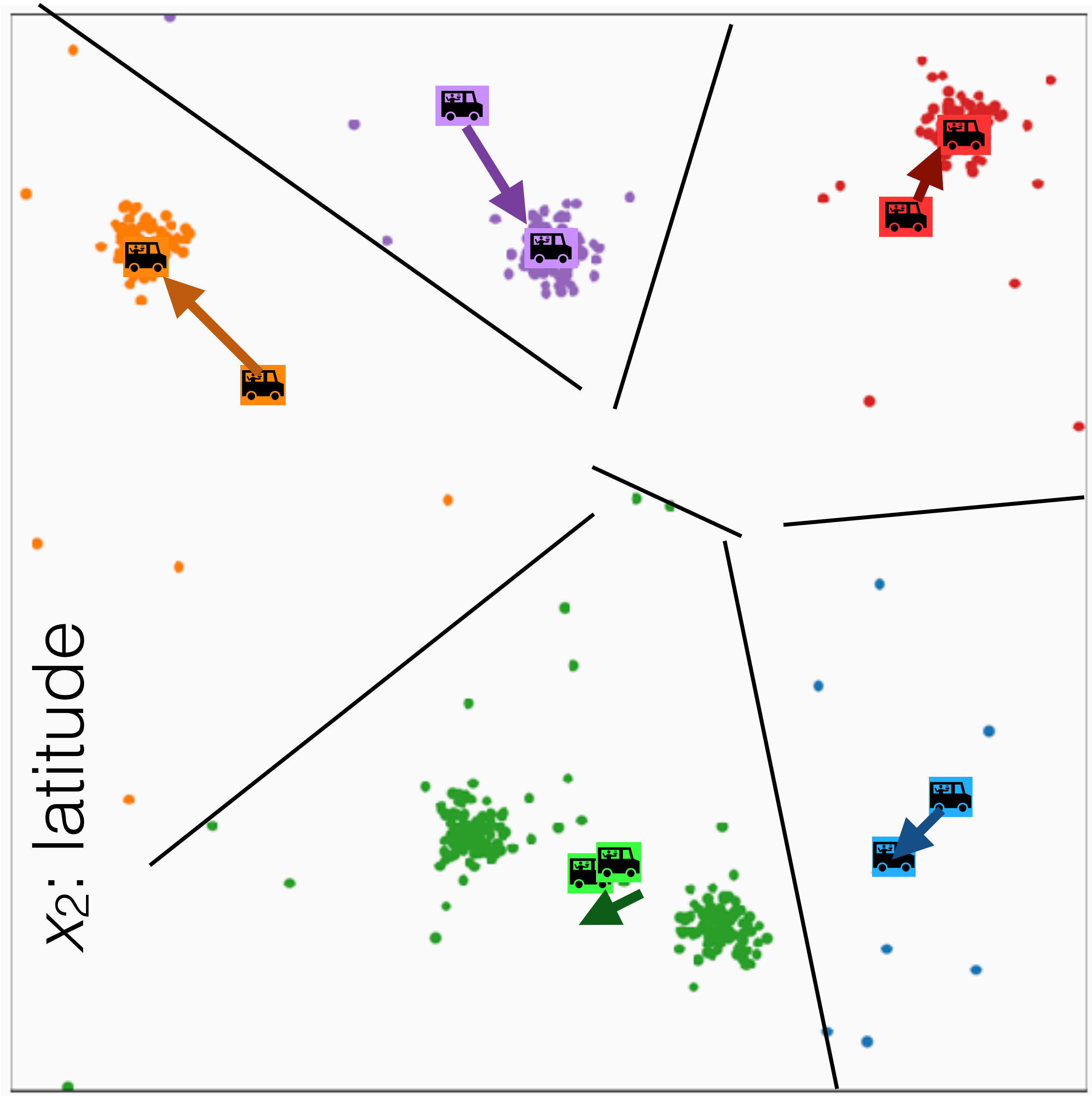
`k-means(k,`τ`)`

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

Some options:

1. Choose *k* data points uniformly at random, *without* replacement
2. Choose uniformly at random within the span of the data

k-means algorithm



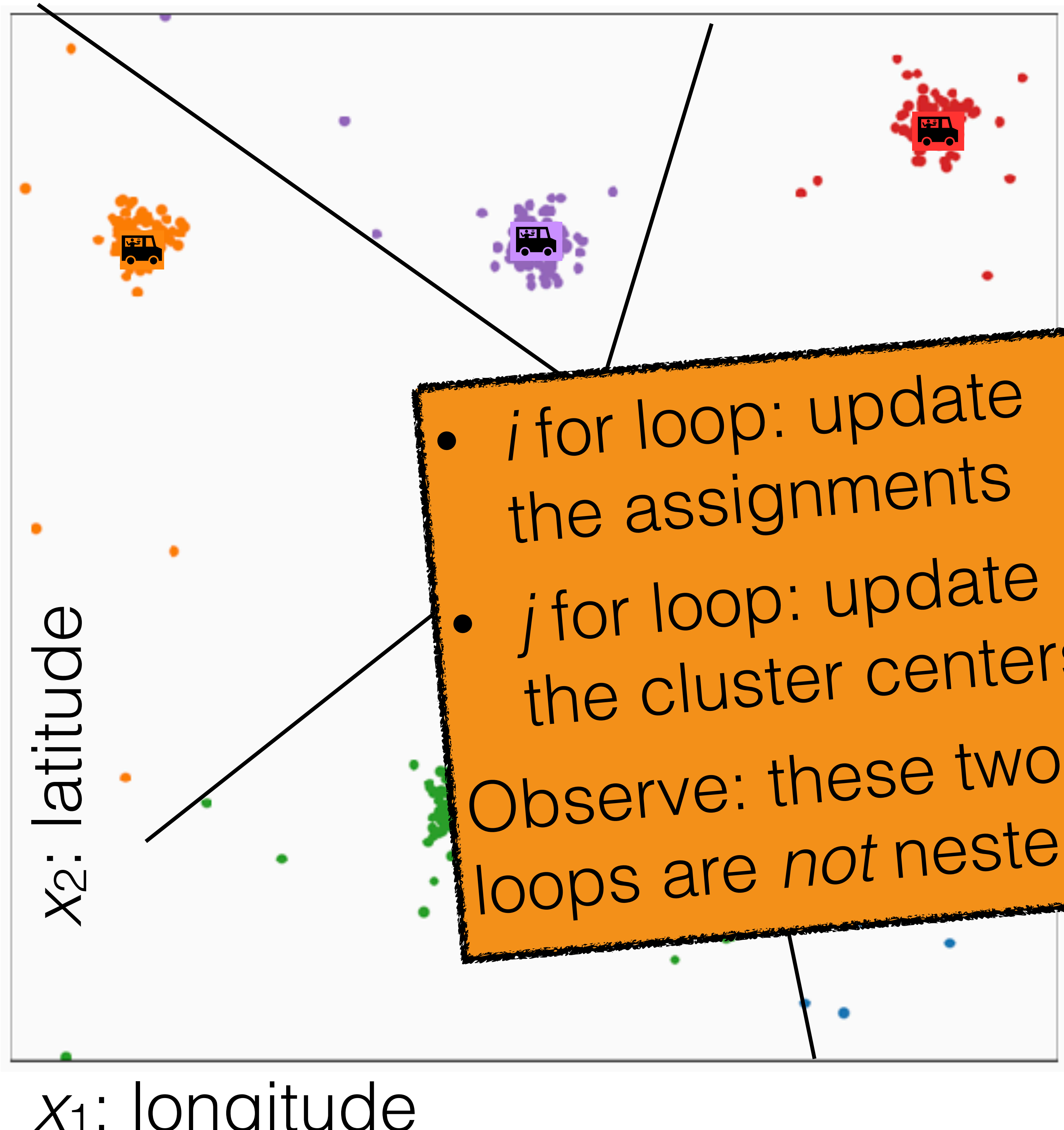

```
k-means(k, τ)
 Init {μ^(j)}_{j=1}^k
 for t = 1 to τ

   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



- *i* for loop: update the assignments
- *j* for loop: update the cluster centers

Observe: these two loops are *not* nested

k-means(k, τ)

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

for t = 1 to τ

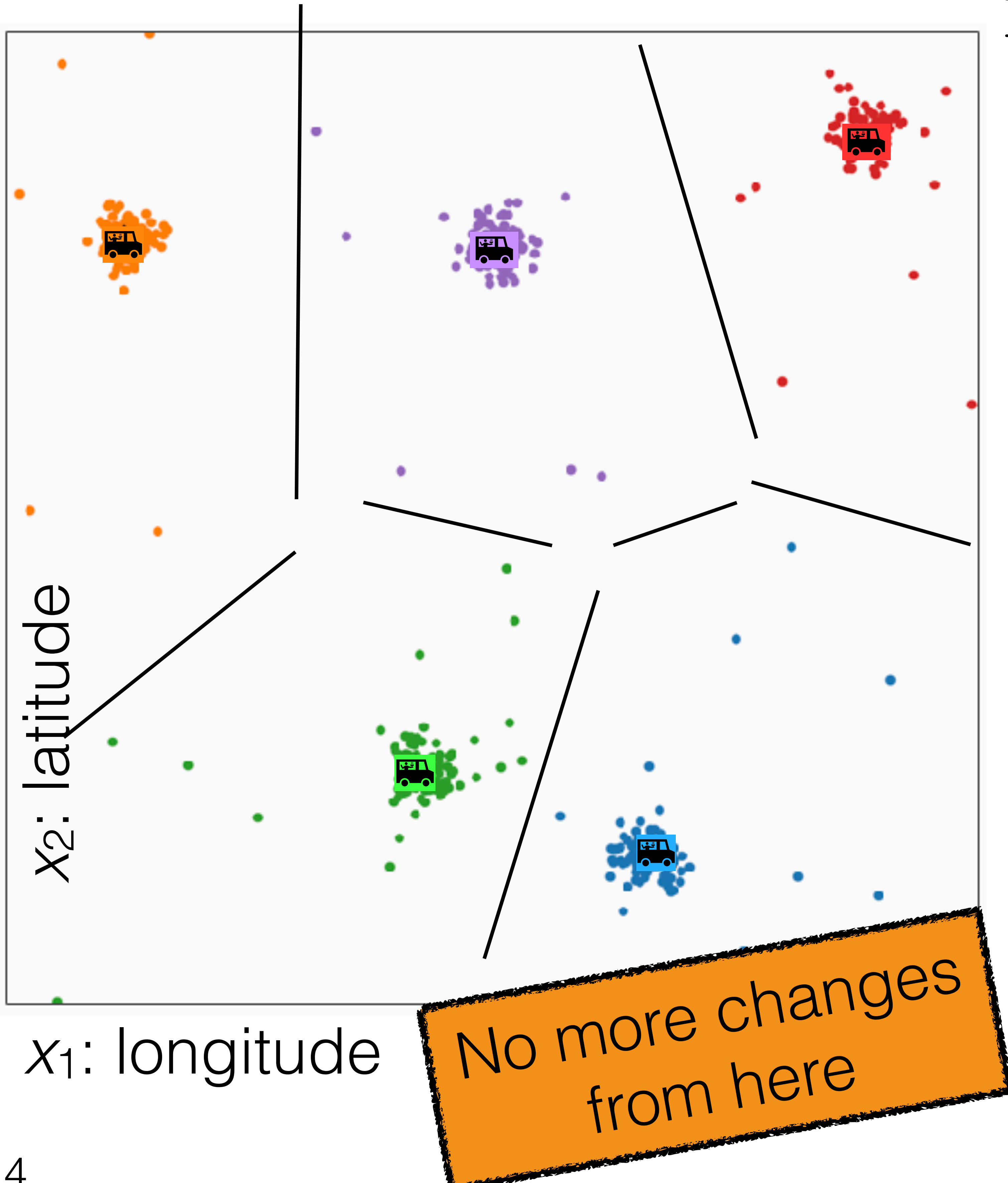
for i = 1 to n

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

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

for t = 1 to τ

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

for i = 1 to n

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

for j = 1 to k

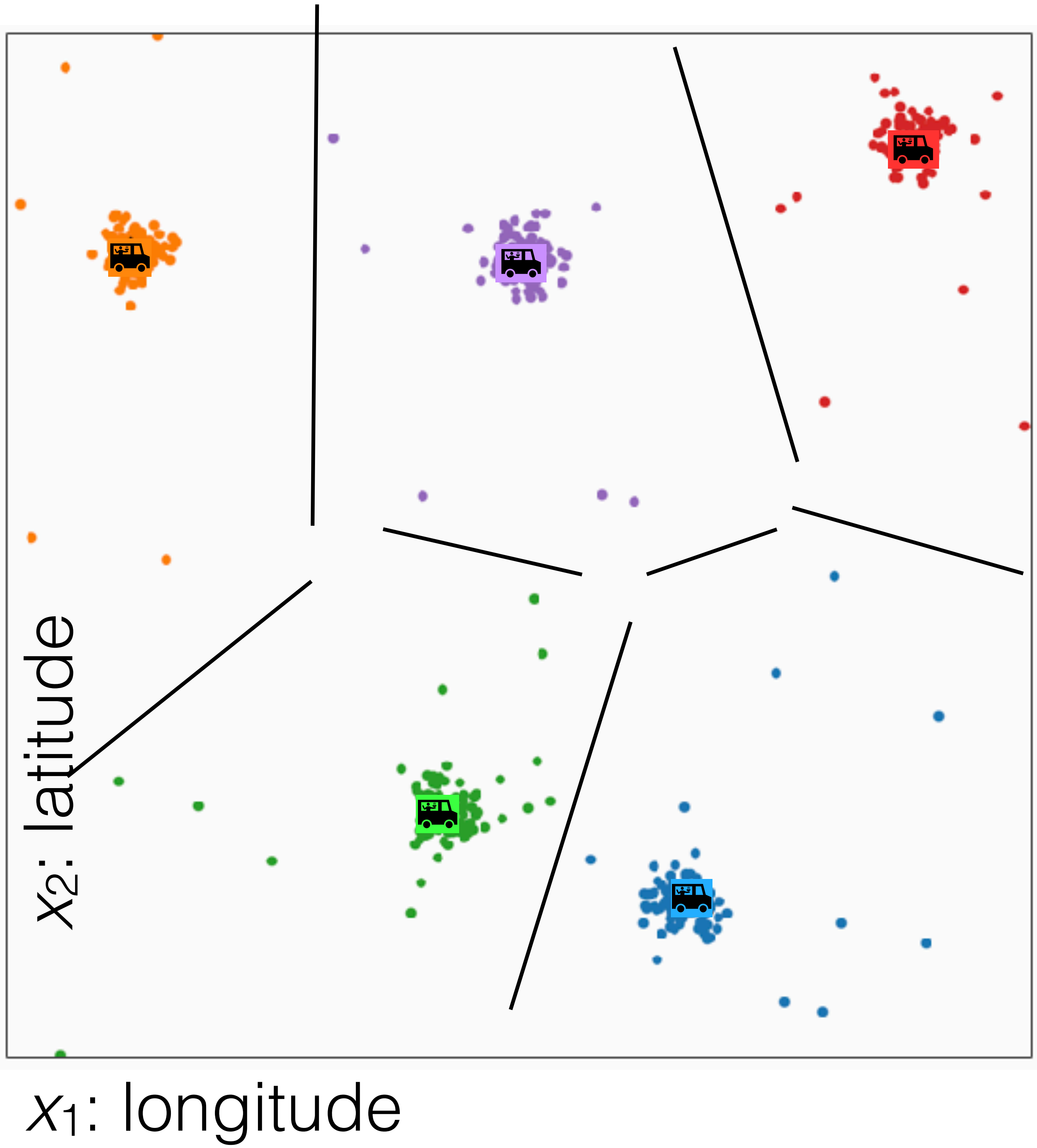
$$\mu^{(j)} = \frac{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{n} \mathbf{1}\{y^{(i)} = j\}}$$

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

break

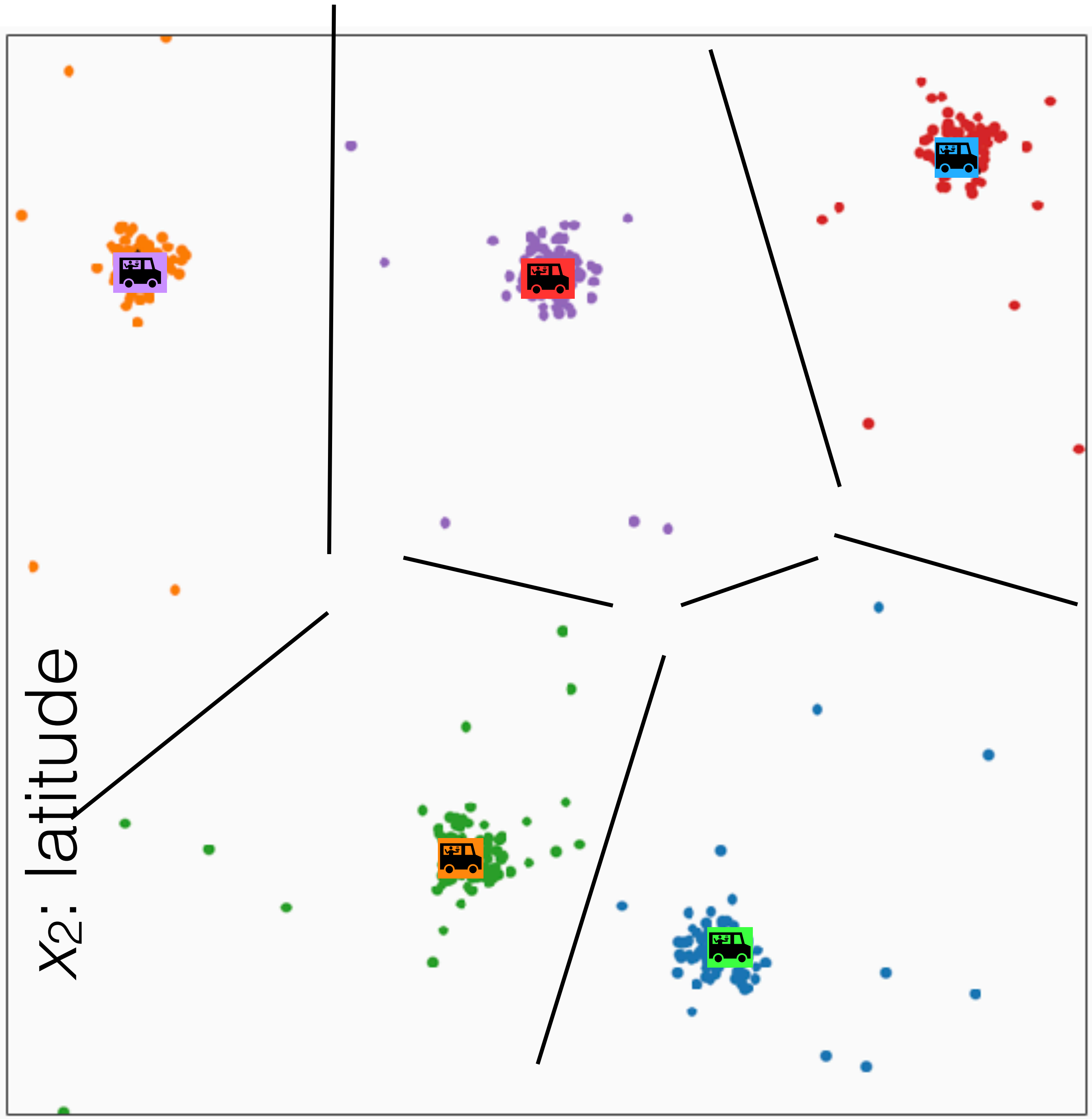
return $\{\mu^{(j)}\}_{j=1}^{k}, \{y^{(i)}\}_{i=1}^{n}$

Compare to classification

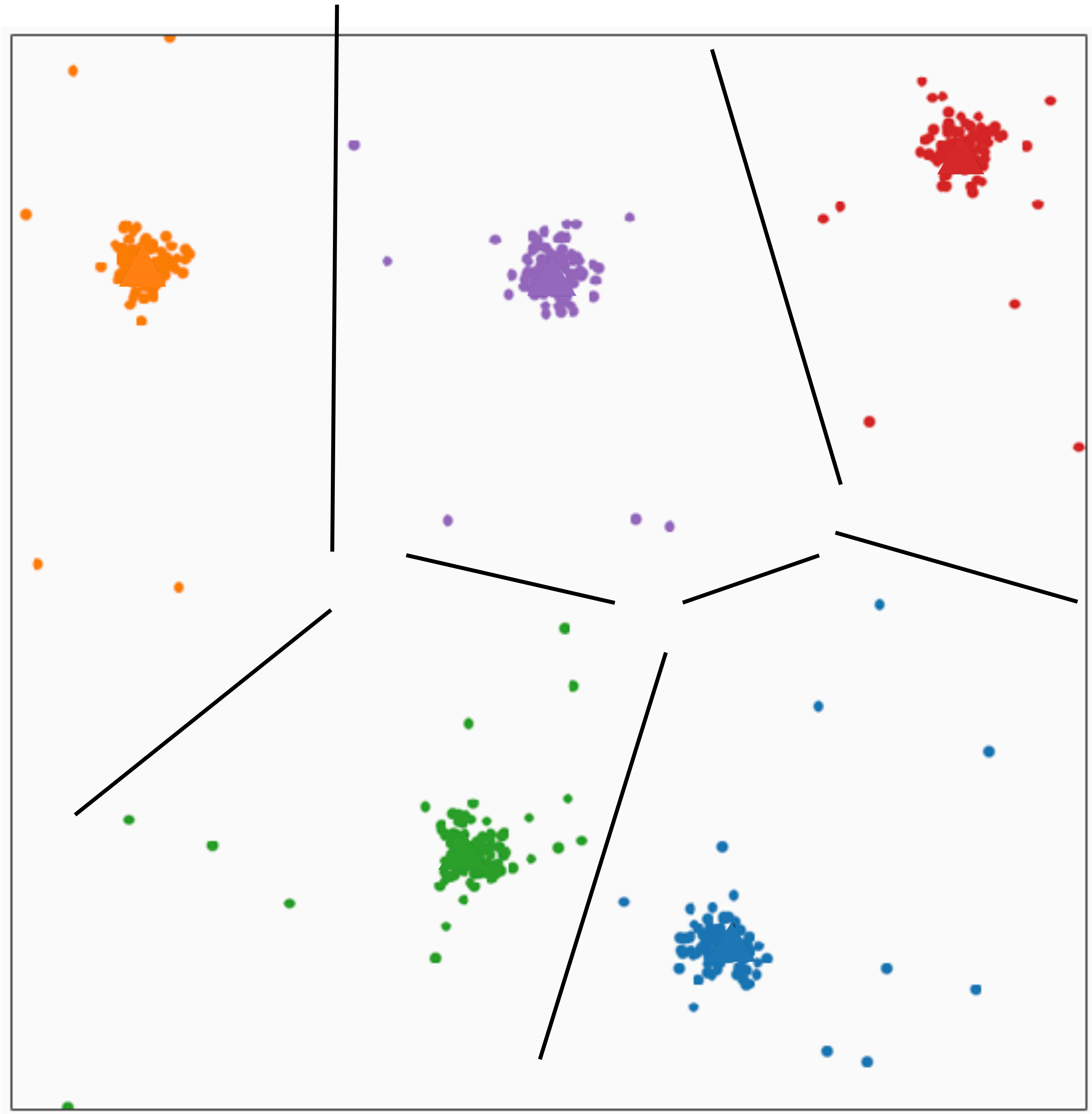


- Did we just do *k*-class classification?
- Looks like we assigned a label $y^{(i)}$, which takes *k* different values, to each feature vector $x^{(i)}$
- But we didn't use any labeled data
- The "labels" here don't have meaning; I could permute them and have the same result

Compare to classification

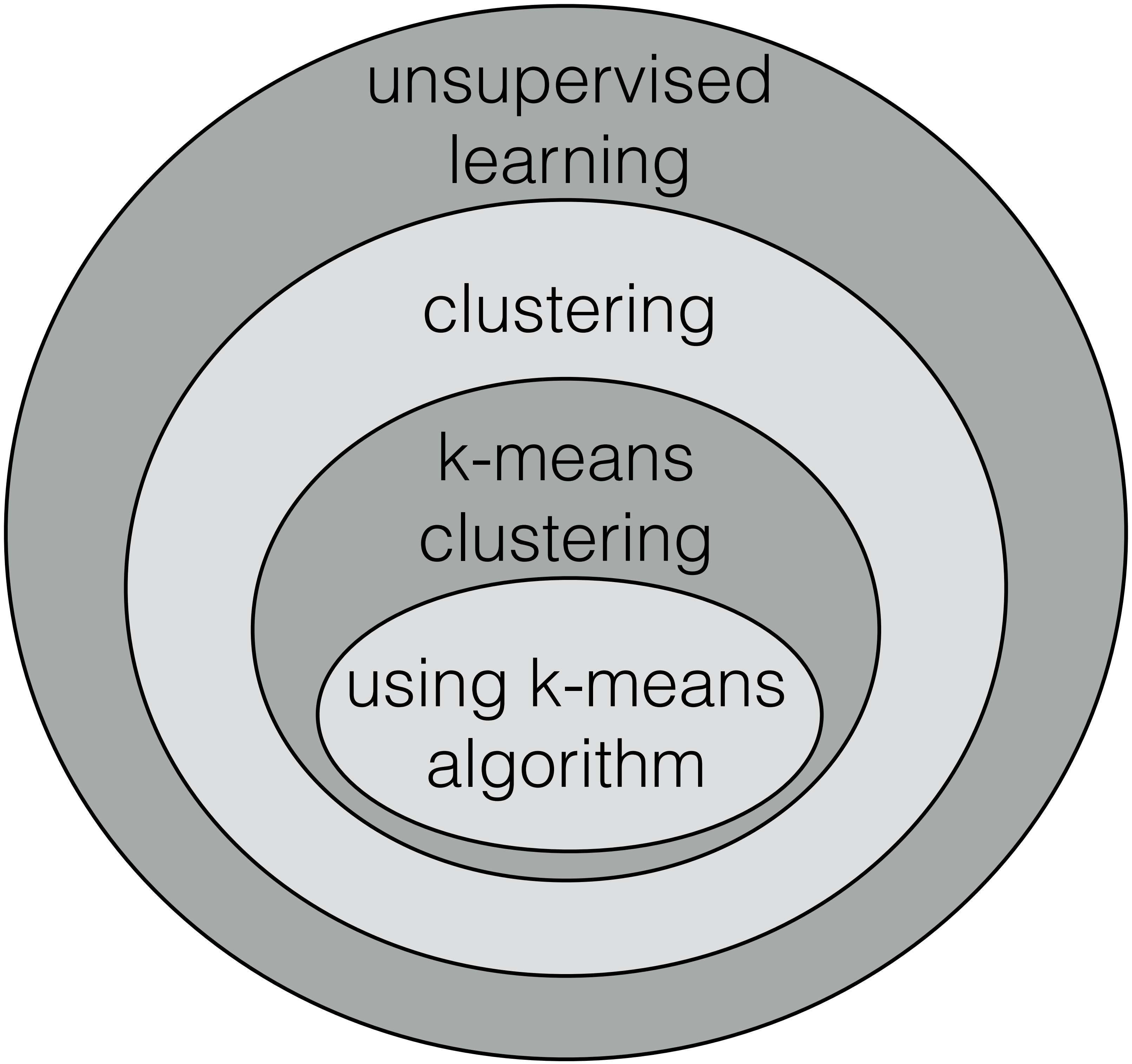


- Did we just do *k*-class classification?
- Looks like we assigned a label $y^{(i)}$, which takes *k* different values, to each feature vector $x^{(i)}$
- But we didn't use any labeled data
- The "labels" here don't have meaning; I could permute them and have the same result
- Output is really a *partition*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



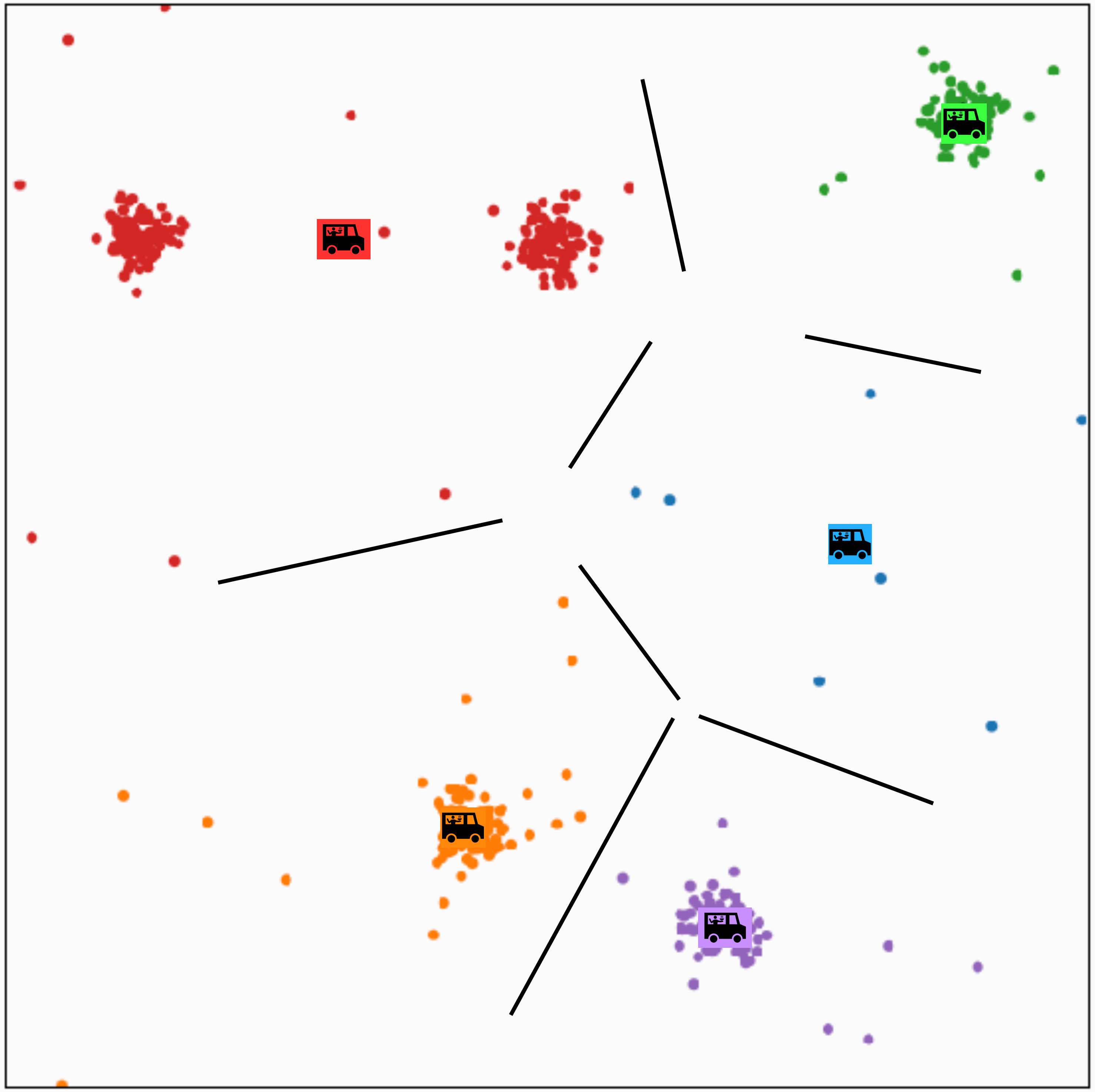
- So what did we do?
- We *clustered* the data: we grouped the data by similarity
  - Why not just plot the data? You should! But also: Precision, big data, high dimensions, high volume
- An example of *unsupervised learning*: no labeled data, & we're finding patterns

Clustering & related



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

k-means algorithm: initialization

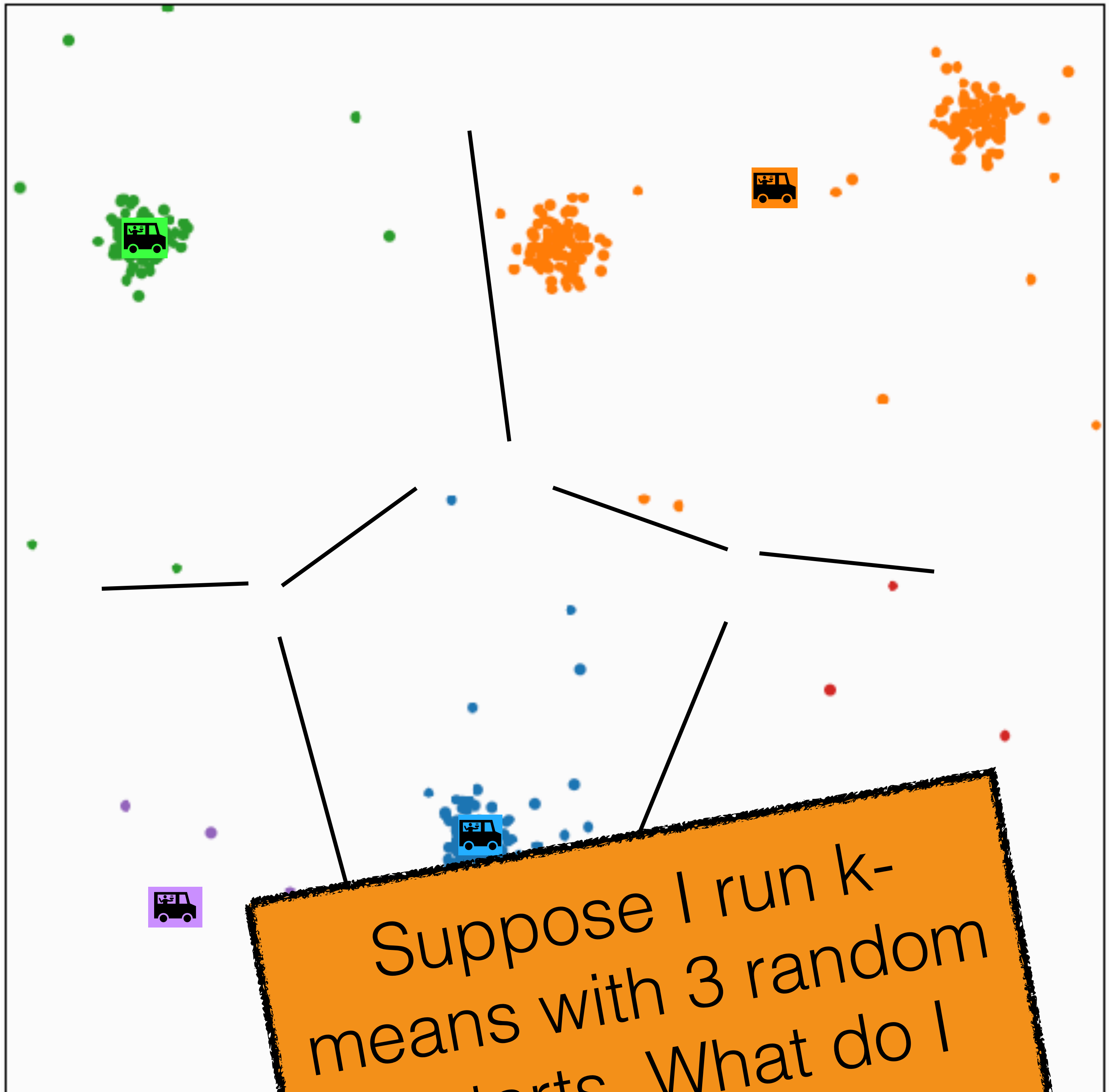


- **Theorem**. If run for enough outer iterations, the k-means algorithm will converge to a local minimum of the k-means objective
- That local minimum could be bad!

Is this clustering worse than the one we found before?

Why or why not?

k-means algorithm: initialization

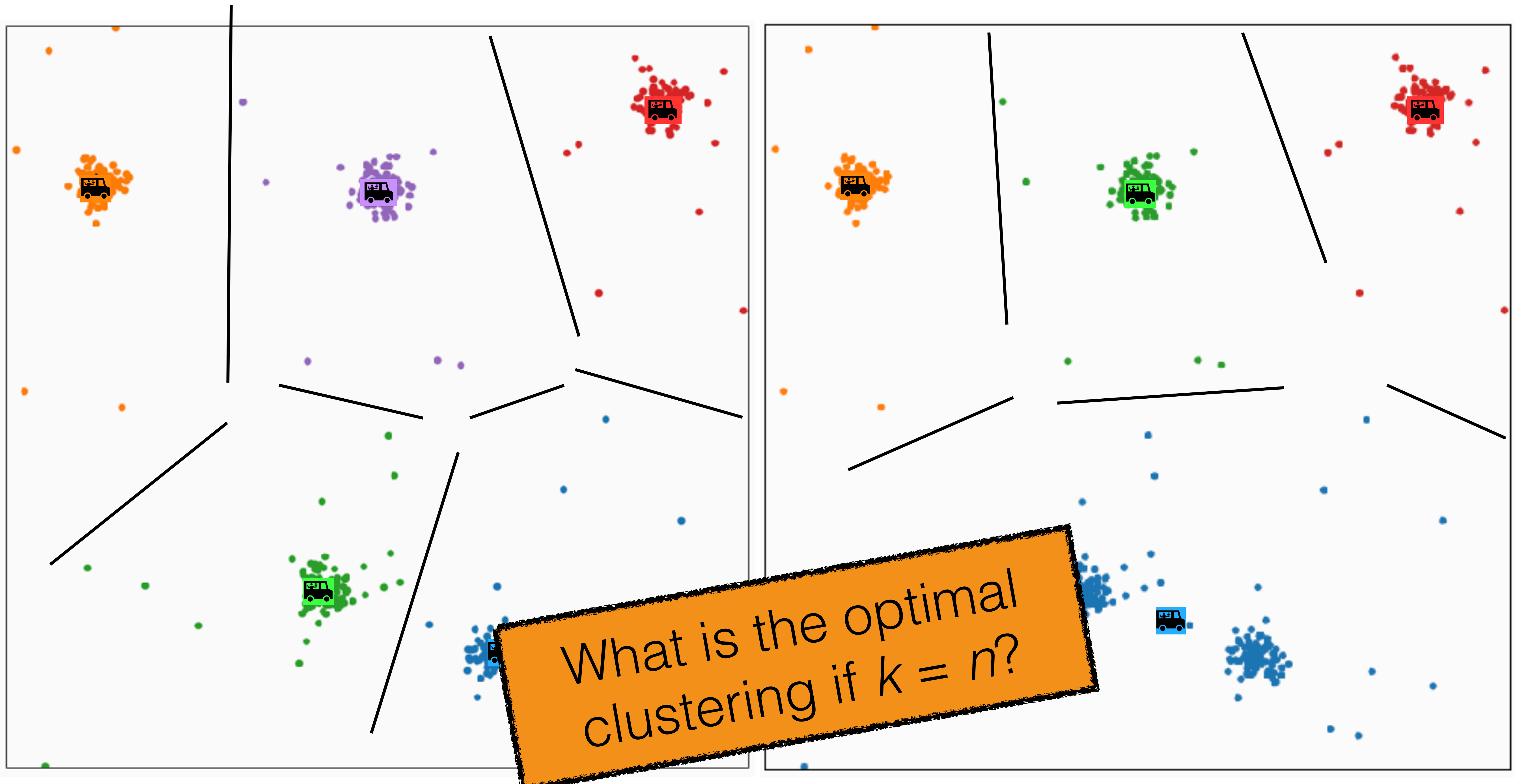


Suppose I run k-means with 3 random restarts. What do I return?

- **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++

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*

- Sometimes we know *k*

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

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

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*



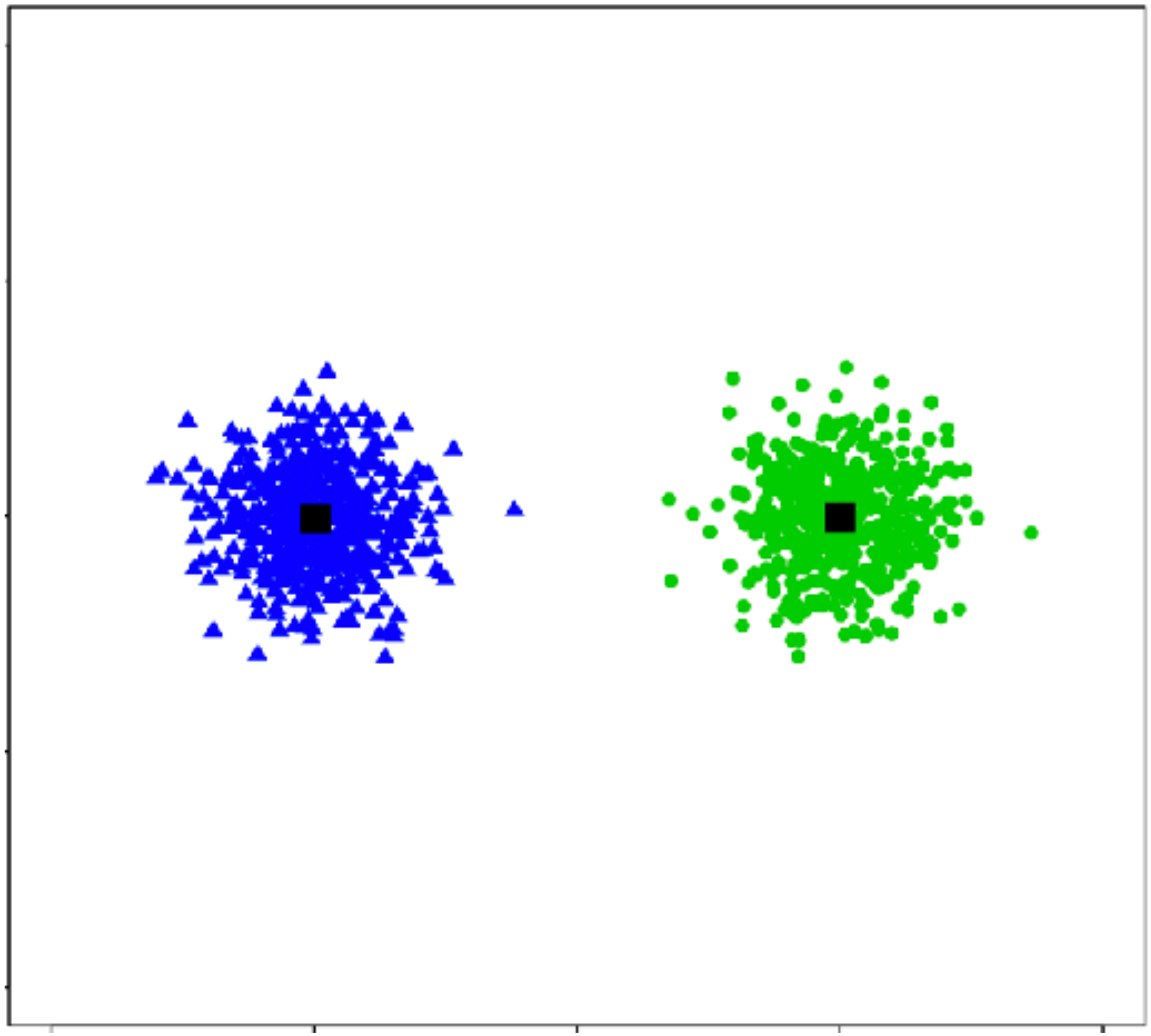

- Sometimes we'd like to choose/learn *k*
    - Can't just minimize the k-means objective over *k* too

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

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

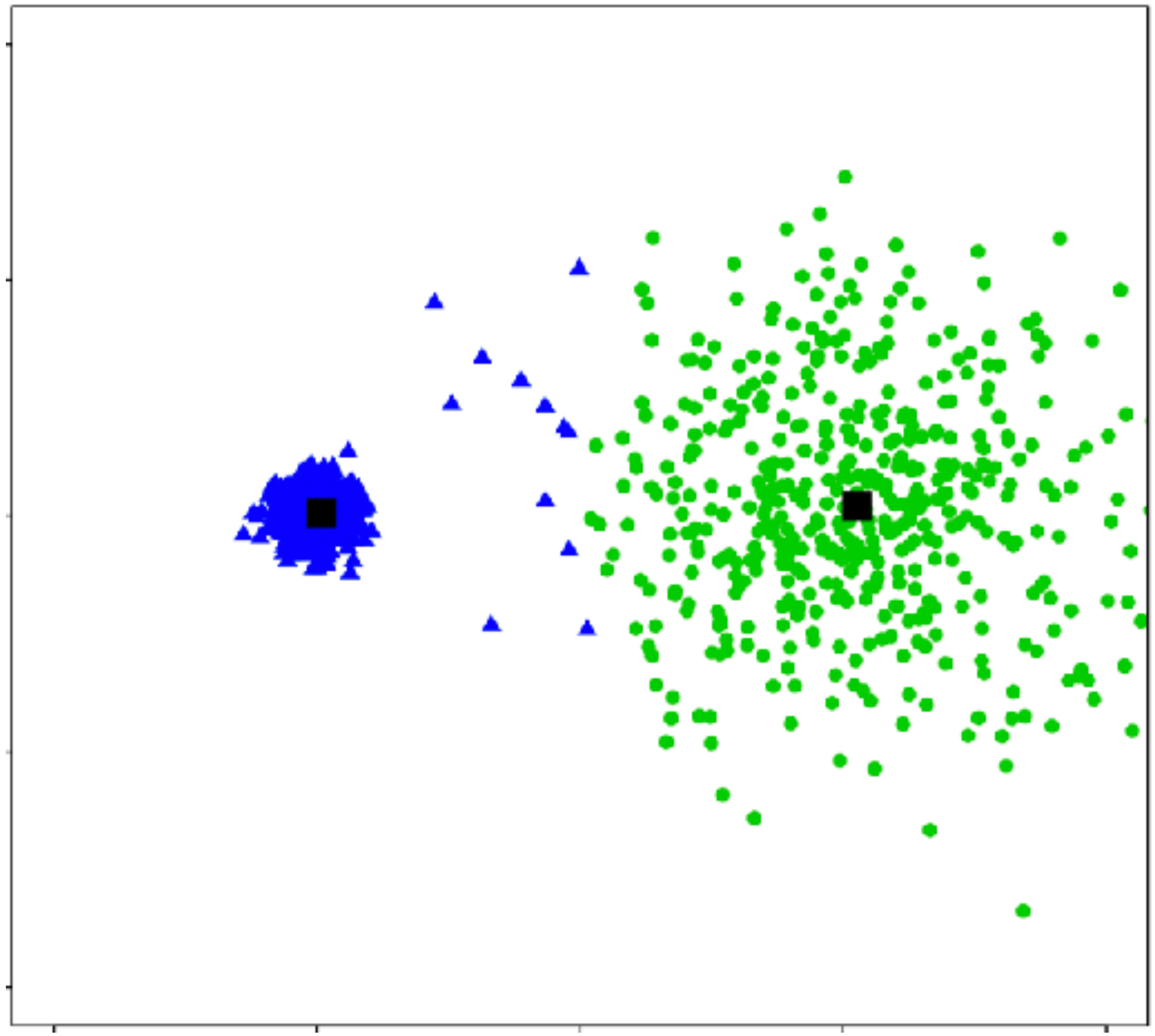
[https://www.billboard.com/articles/columns/hip-hop/9490867/drake-son-adonis-photo-instagram; https://en.wikipedia.org/wiki/Francis_Drake#/media/File:1590_or_later_Marcus_Gheeraerts,_Sir_Francis_Drake_Buckland_Abbey,_Devon.jpg; https://commons.wikimedia.org/wiki/File:Mandarin.duck.arp.jpg; https://en.wikipedia.org/wiki/Drake_University#/media/File:DrakeSeal.png]

Cluster shape



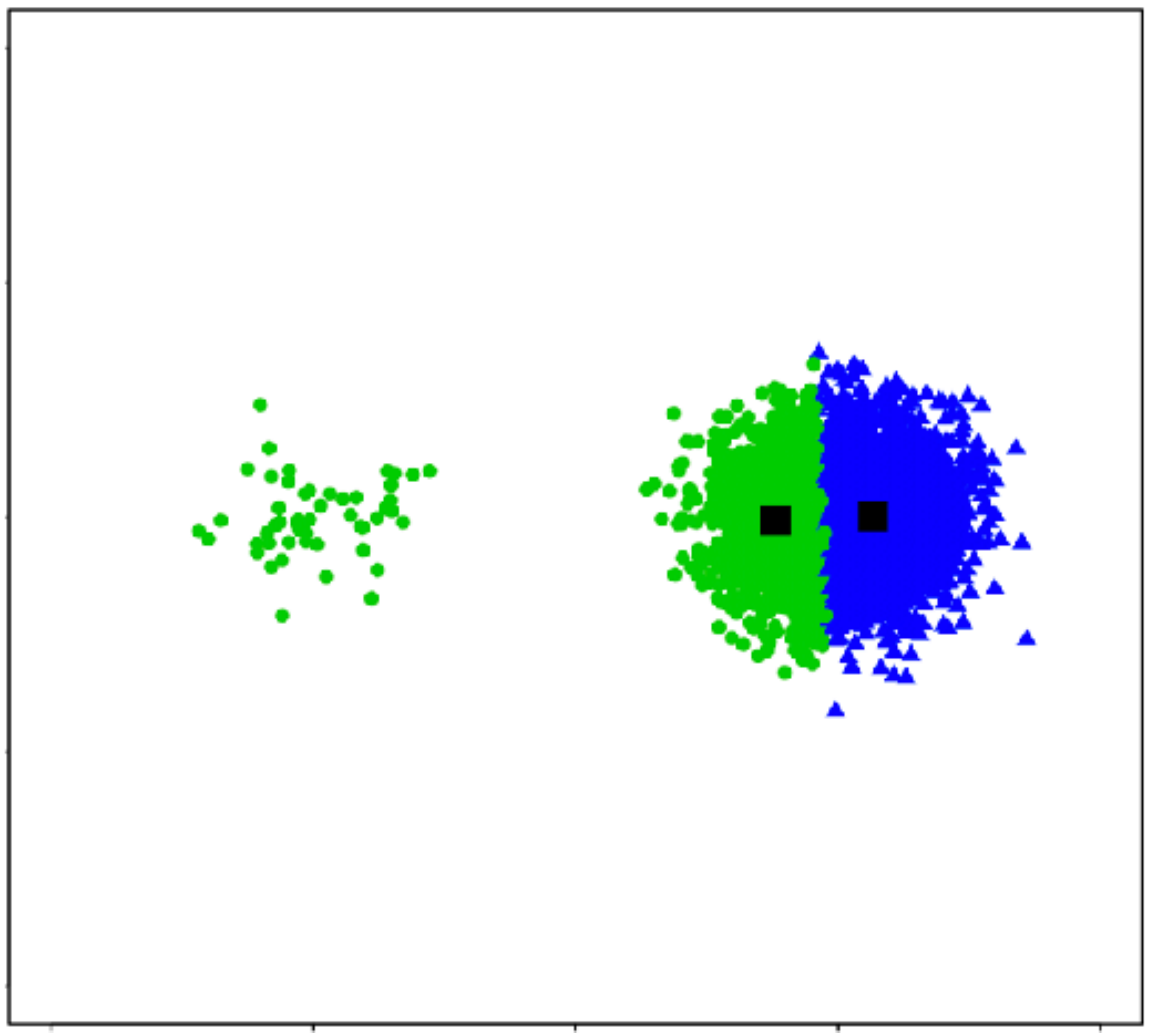
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



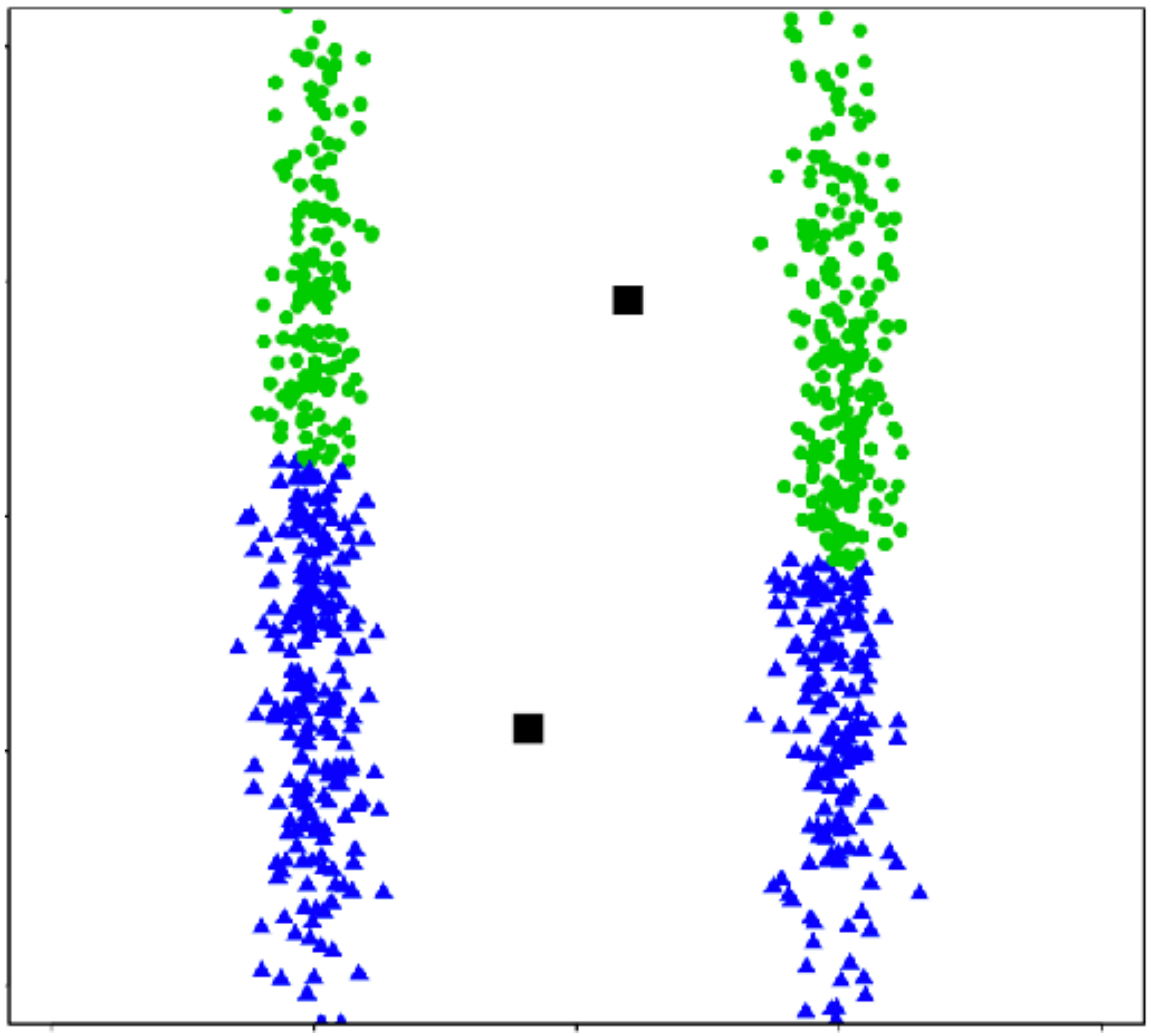
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



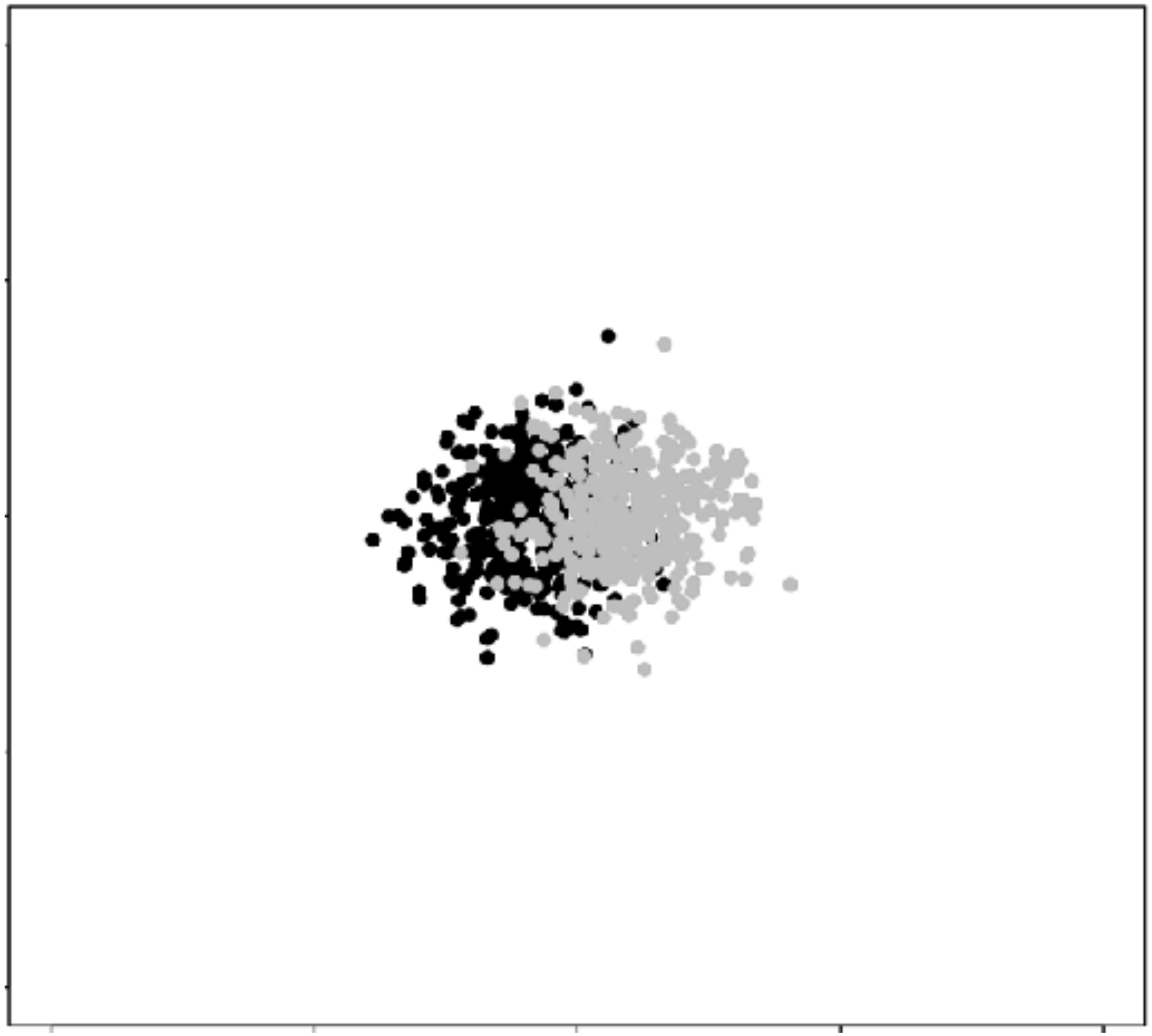
- k-means works well for well-separated circular clusters of the **same size**

Cluster shape



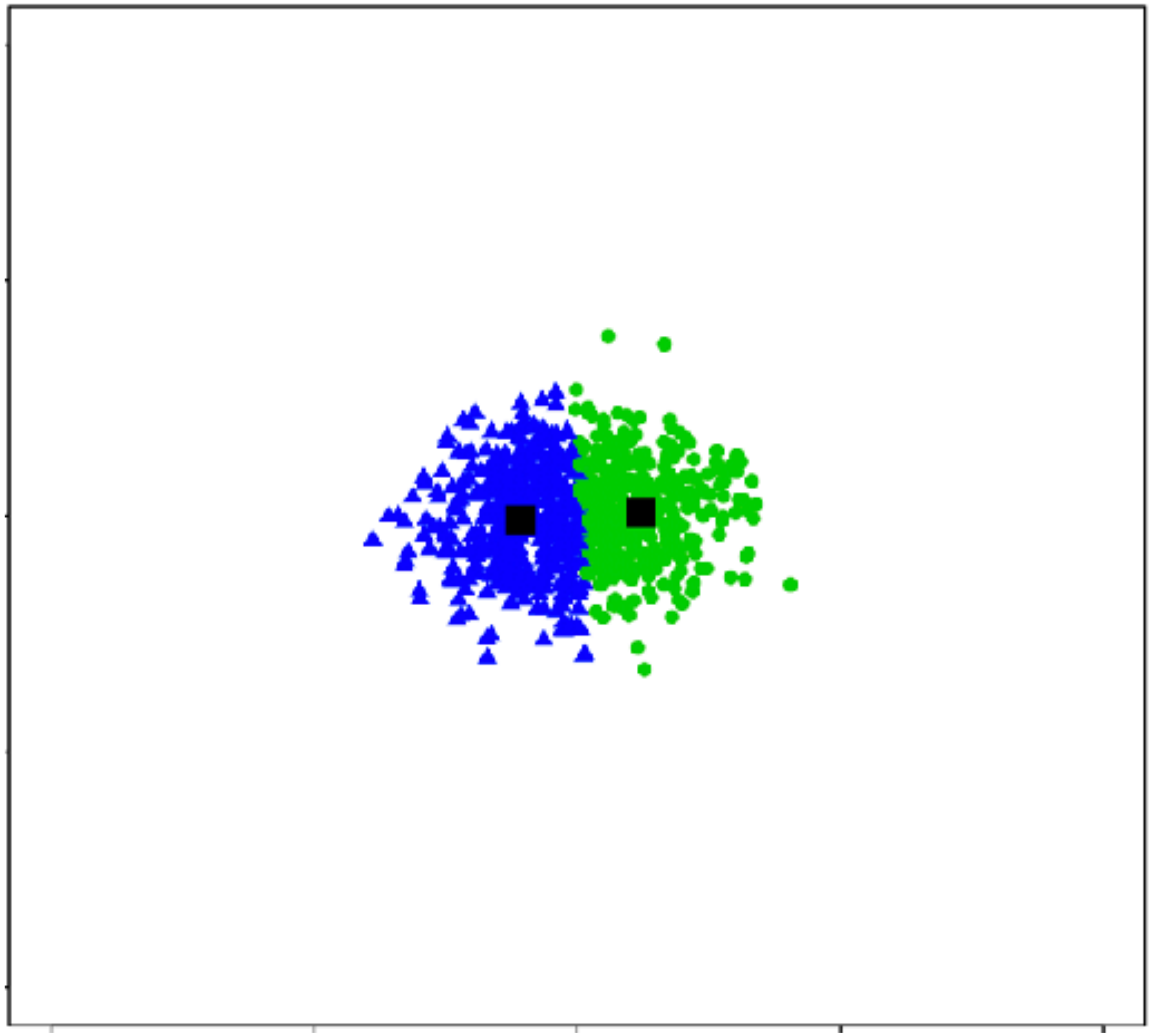
- k-means works well for well-separated **circular** clusters of the same size

Cluster shape



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

Cluster shape



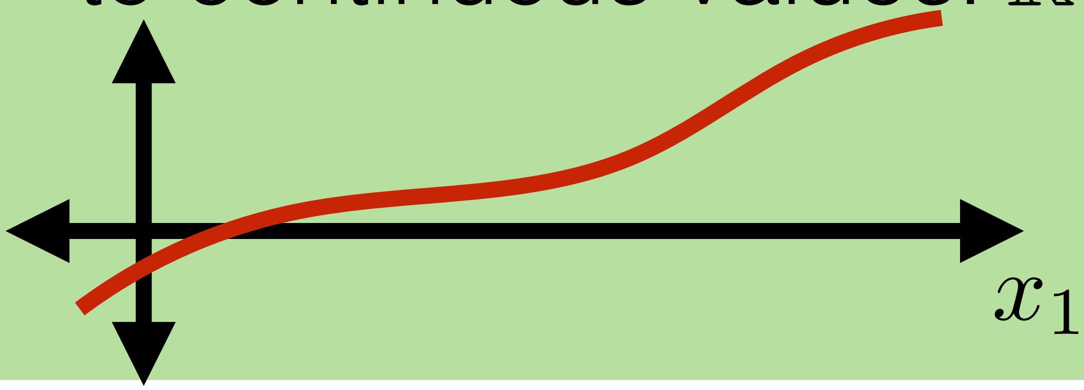
- k-means works well for **well-separated** circular clusters of the same size

Machine Learning Tasks

- **Supervised learning:** Learn a mapping from features to labels

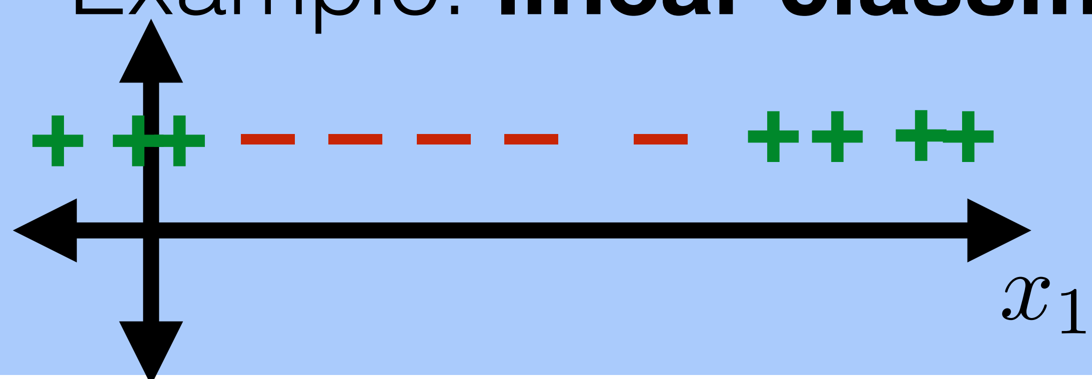
- **Unsupervised learning:** No labels; find patterns

- **Regression:** Learn a mapping to continuous values: $\mathbb{R}^d \rightarrow \mathbb{R}^k$



- **Classification:** Learn a mapping to a discrete set

- **Binary/two-class classification:** Learn a mapping: $\mathbb{R}^d \rightarrow \{-1, +1\}$
  - Example: **linear classification**



- **Multi-class classification:** > 2 label values