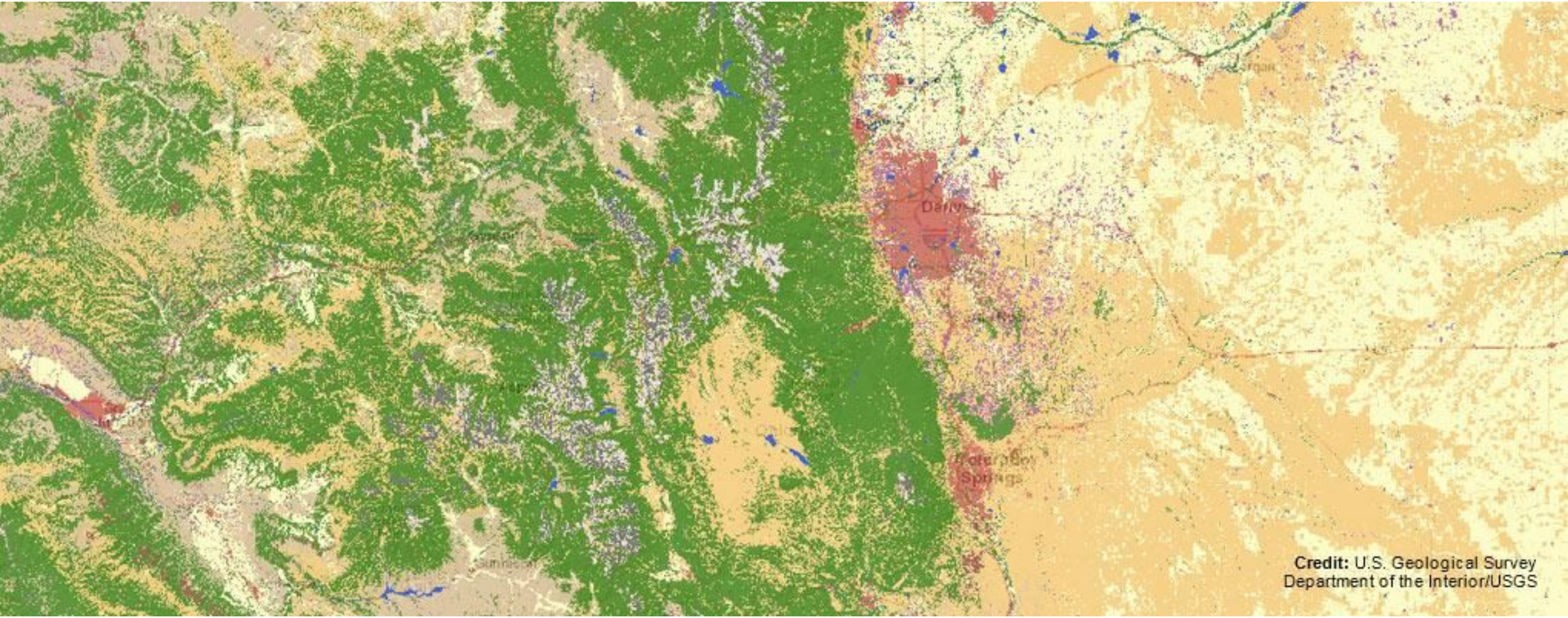
EDS 223: Geospatial Analysis & Remote Sensing
Week 7

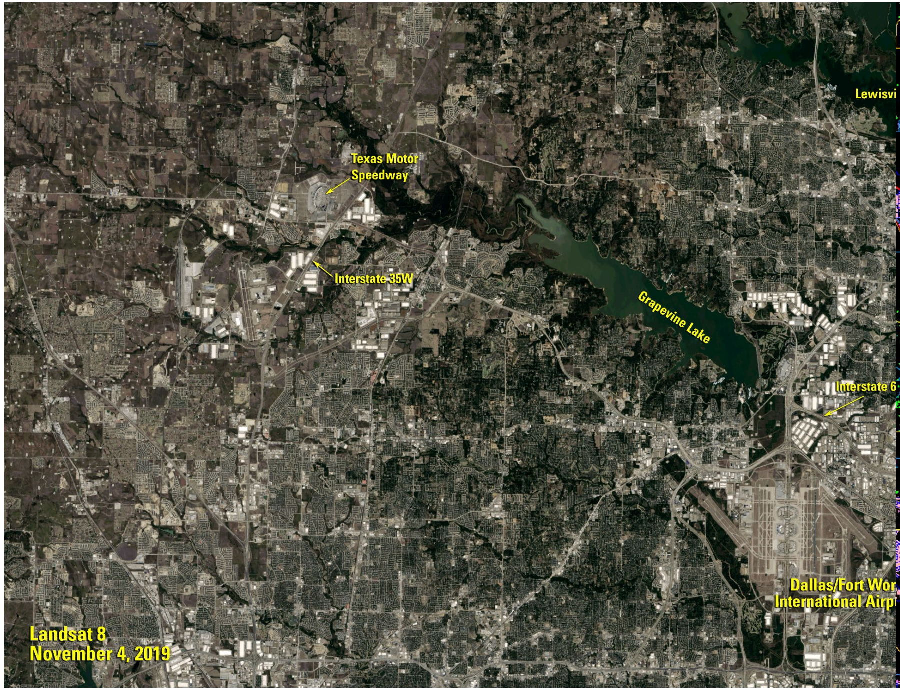


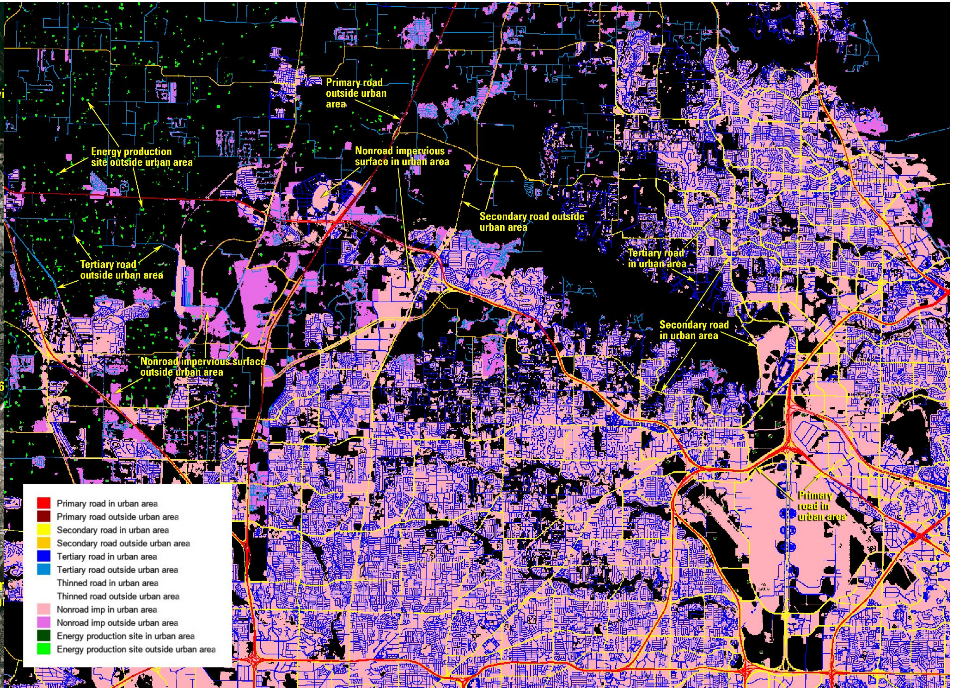
Reminders

- End of class Survey #7 (due tonight)
- Homework Assignment # 4 (due 11/26)
- Final Project (due 12/06)

Week	Tuesday	Thursday
7	No Class	Landcover Classification
8	Landcover Classification Lab	Active RS
9	Active RS Lab	No Class
10	Drone Activity	Drone Lab

Land cover classification





Source: USGS

Image classification

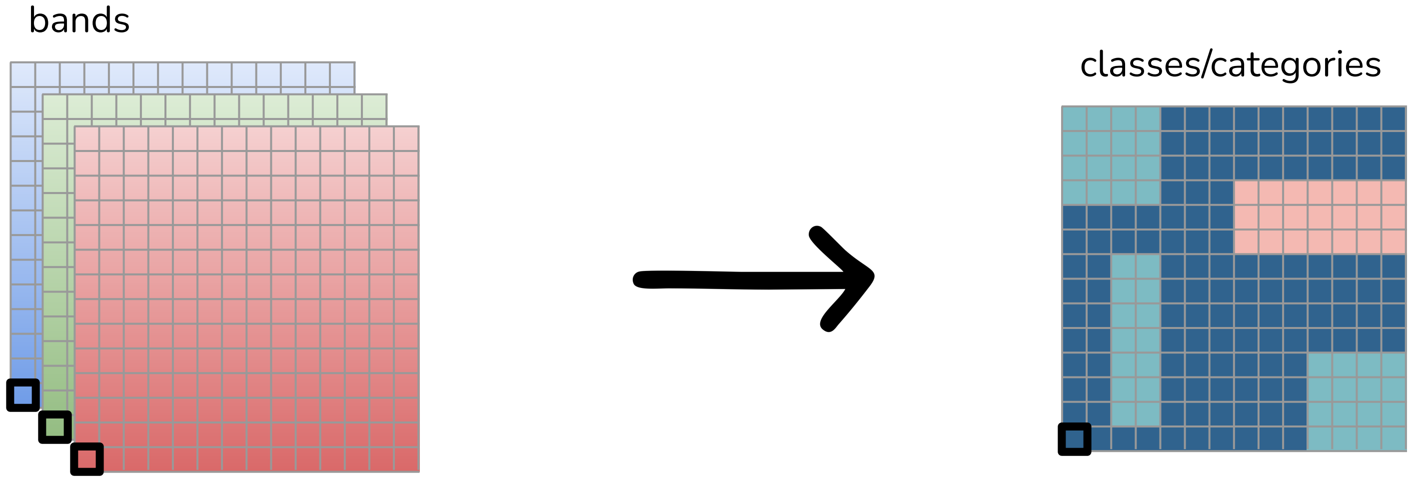
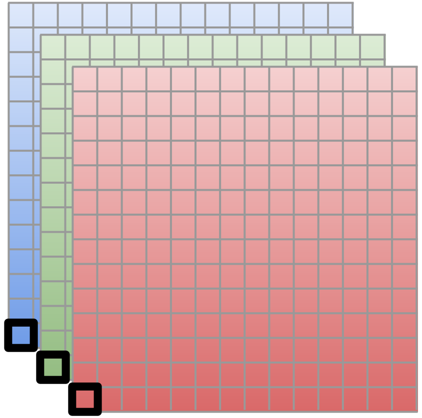
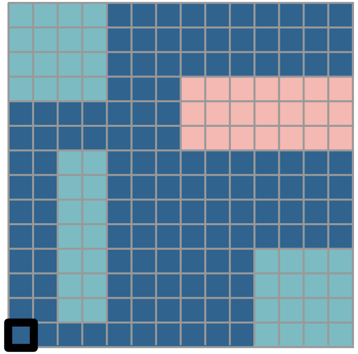


Image classification

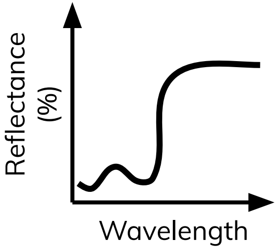
bands



classes/categories

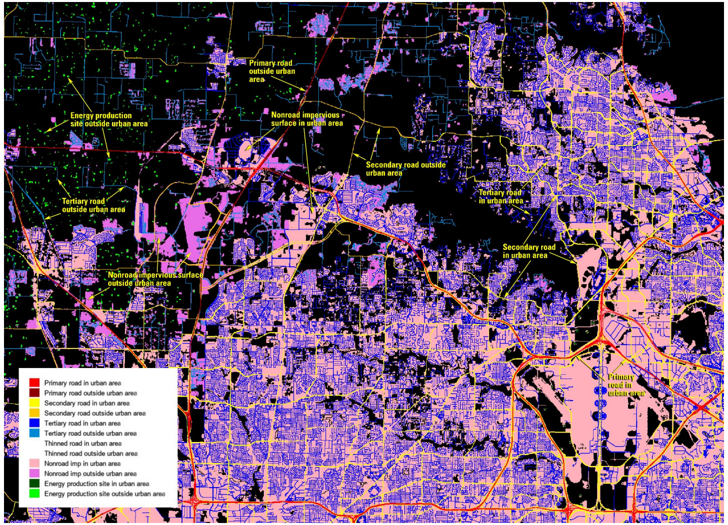


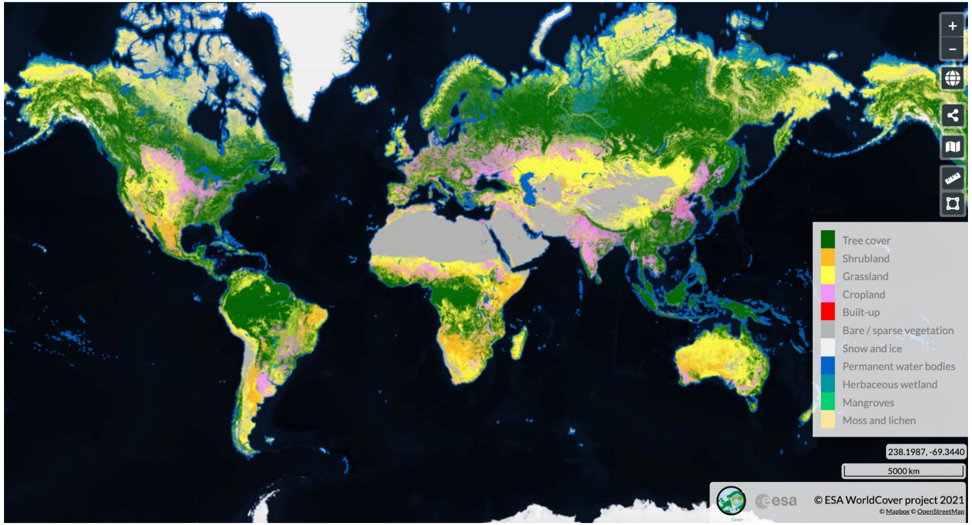
reflectance spectra



finite number of classes

Classification across scales

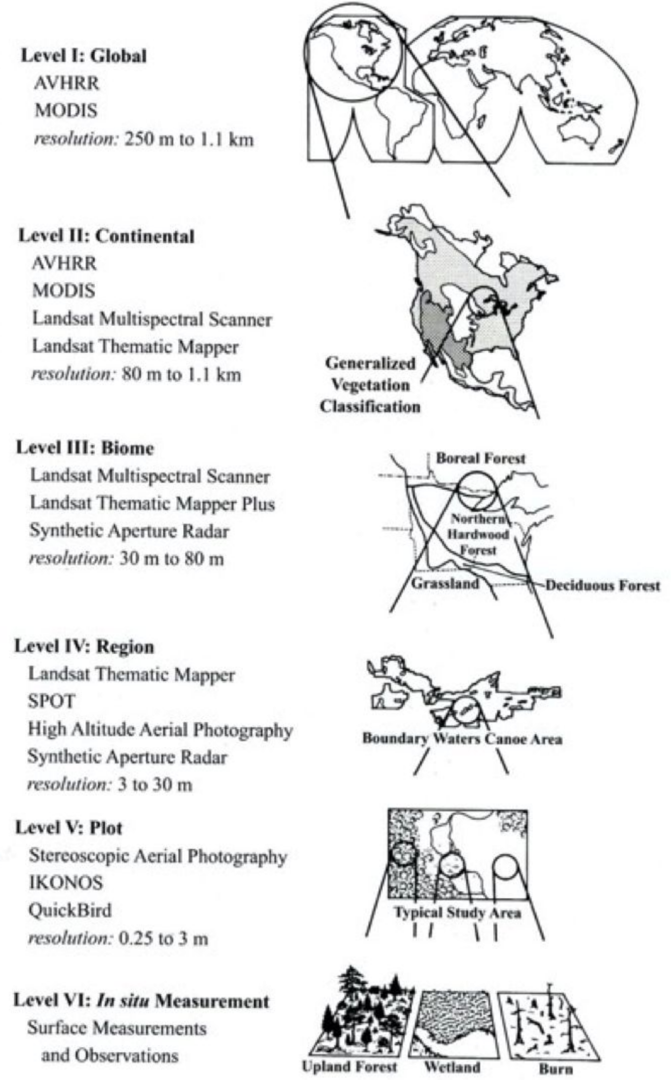




Source: USGS

Classification across scales

When selecting data for land cover classification, you should consider the spatial, temporal, and spectral resolutions necessary to detect the categories of interest.



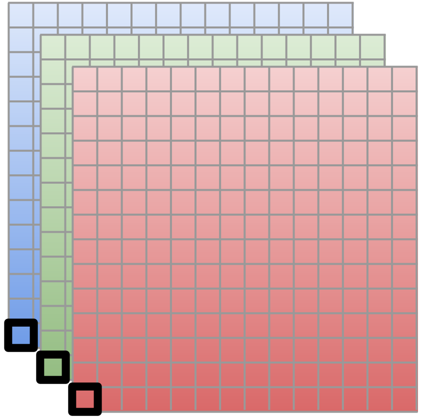
Source: Jensen 2007

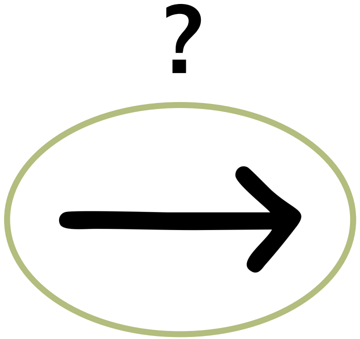
Land cover vs. land use

Land cover	Land use
Refers to the type of natural and artificial materials present on a landscape	Refers to the human use of landscapes
E.g. forest, sand, water, cement	E.g. protected area, industrial, residential, agricultural
Able to observe	Abstract/intangible, requires deductive reasoning

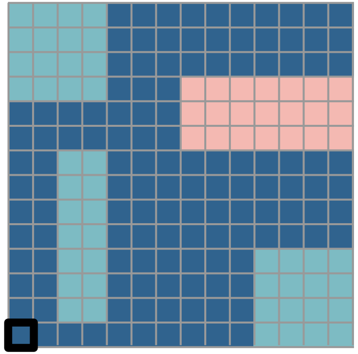
Land cover classification

bands

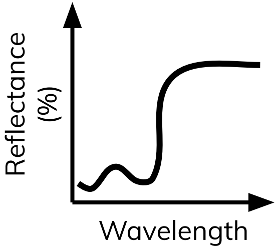


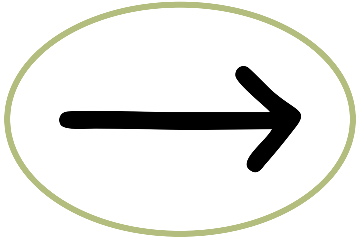


classes/categories



reflectance spectra

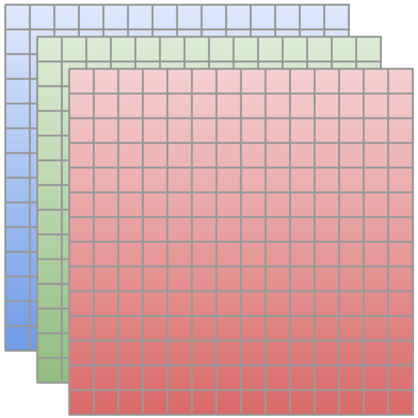




finite number of classes

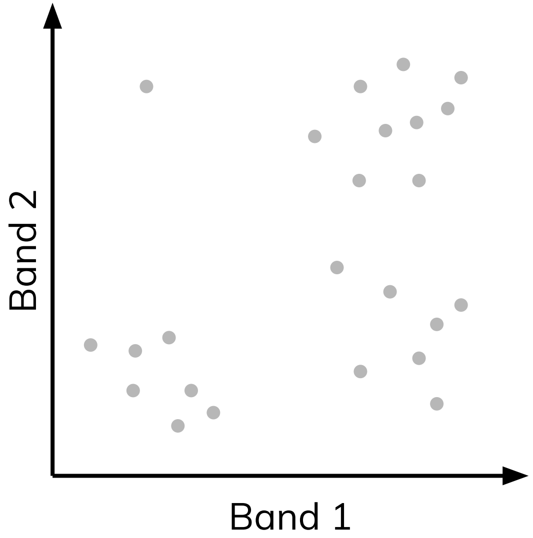
Land cover classification

Geographic space



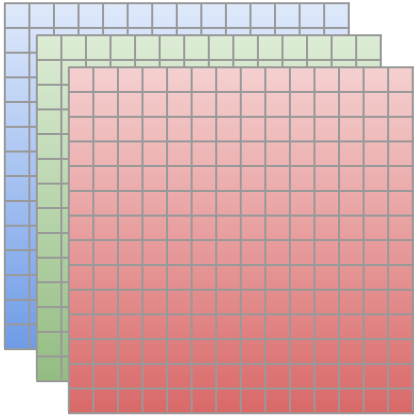
Feature space

Points are pixels



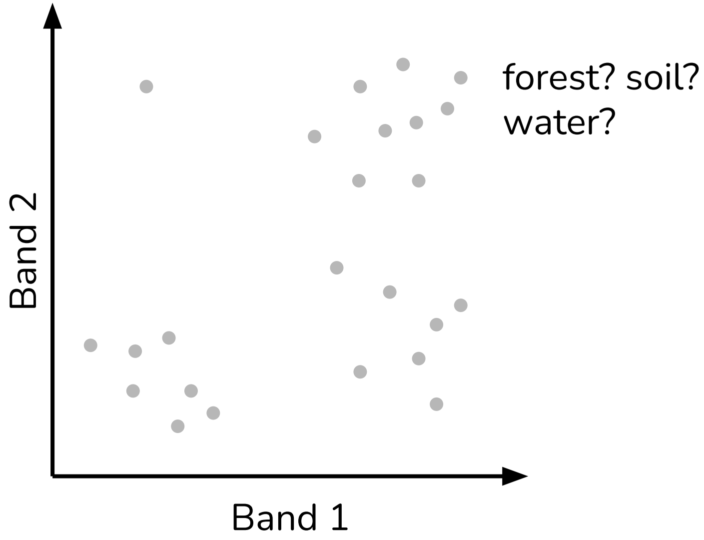
Land cover classification

Geographic space



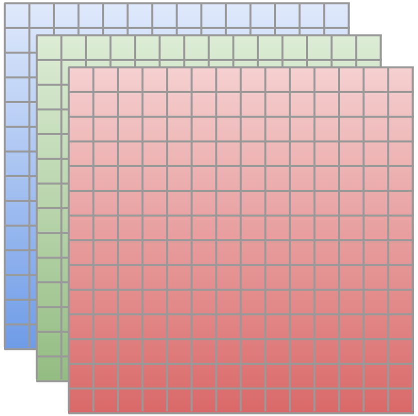
Feature space

Points are pixels



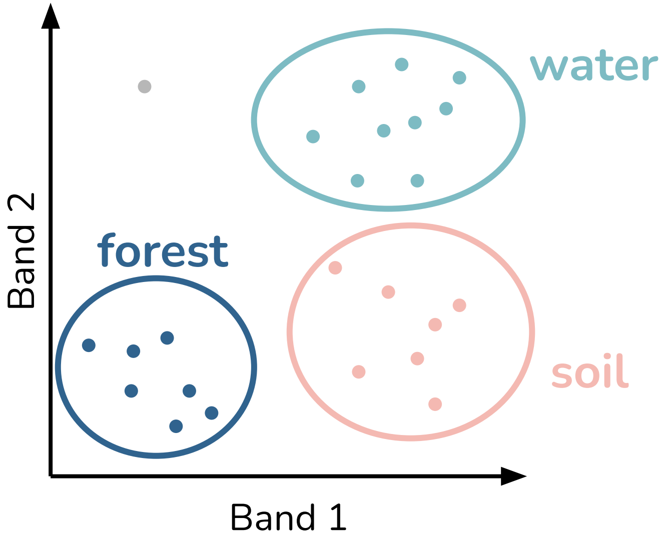
Land cover classification

Geographic space



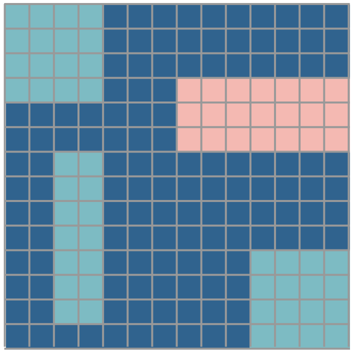
Feature space

Points are pixels



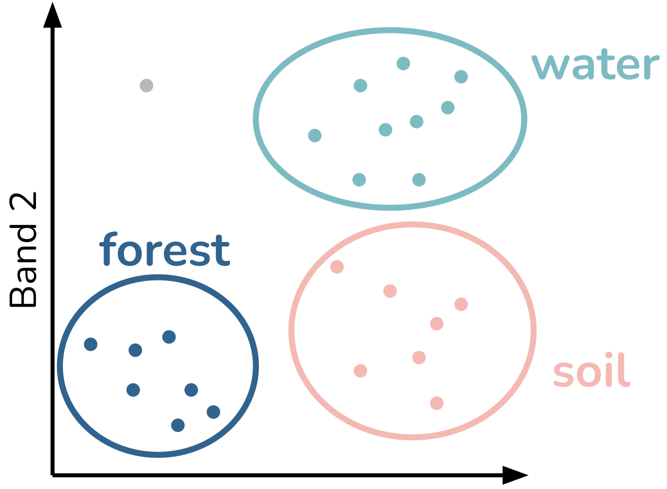
Land cover classification

Geographic space



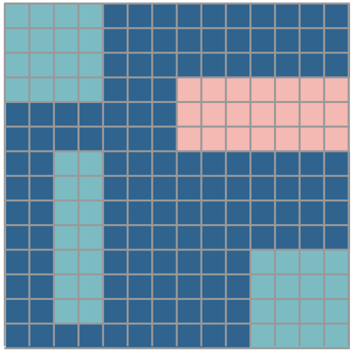
Feature space

Points are pixels



Land cover classification

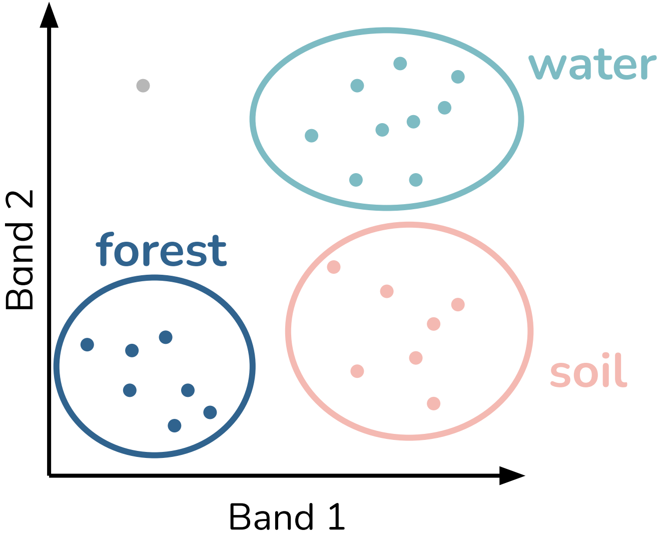
Geographic space



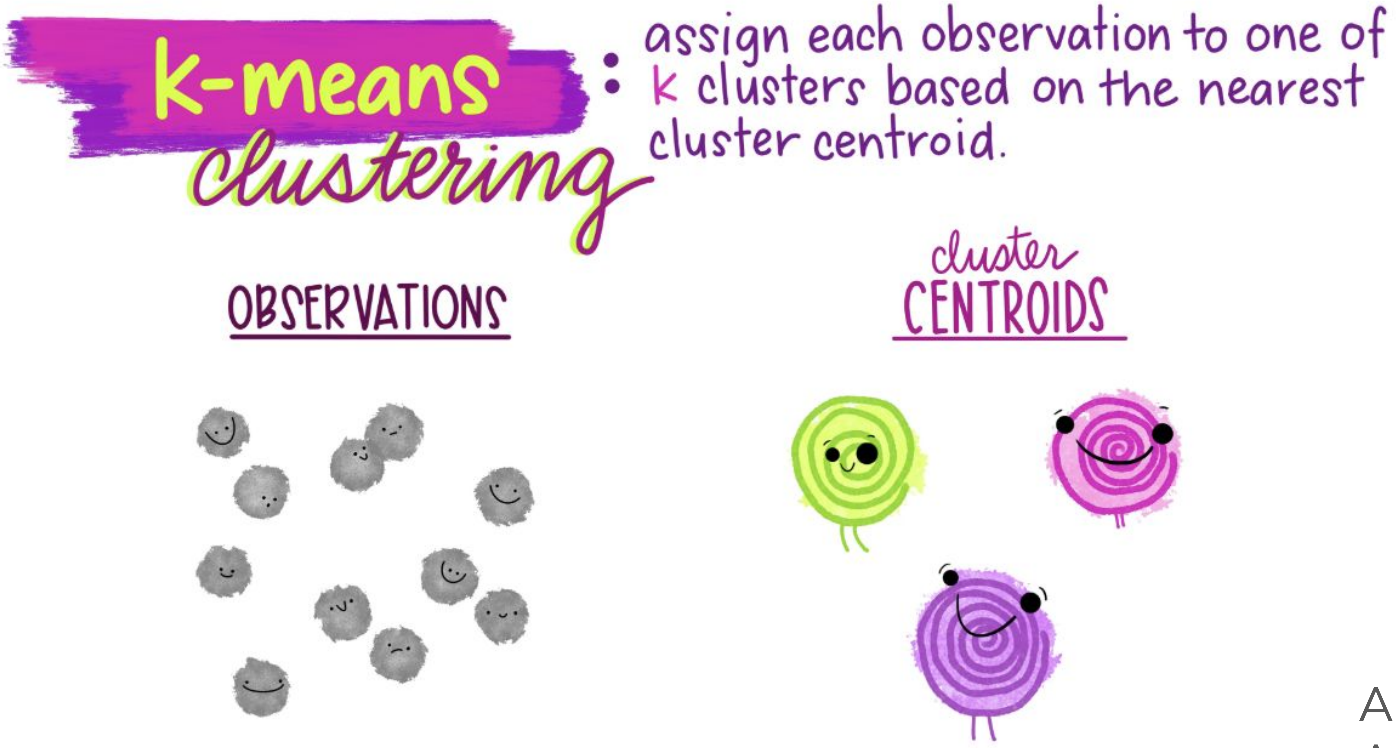
Lots of ways to assign pixels to groups!

Feature space

Points are pixels



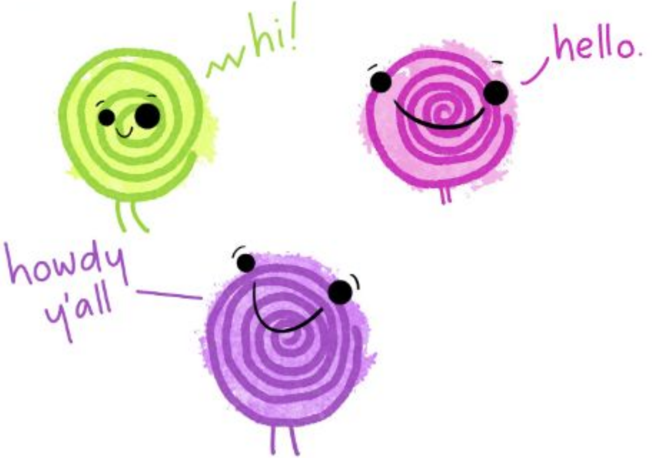
k-means clustering



Artwork by Allison Horst

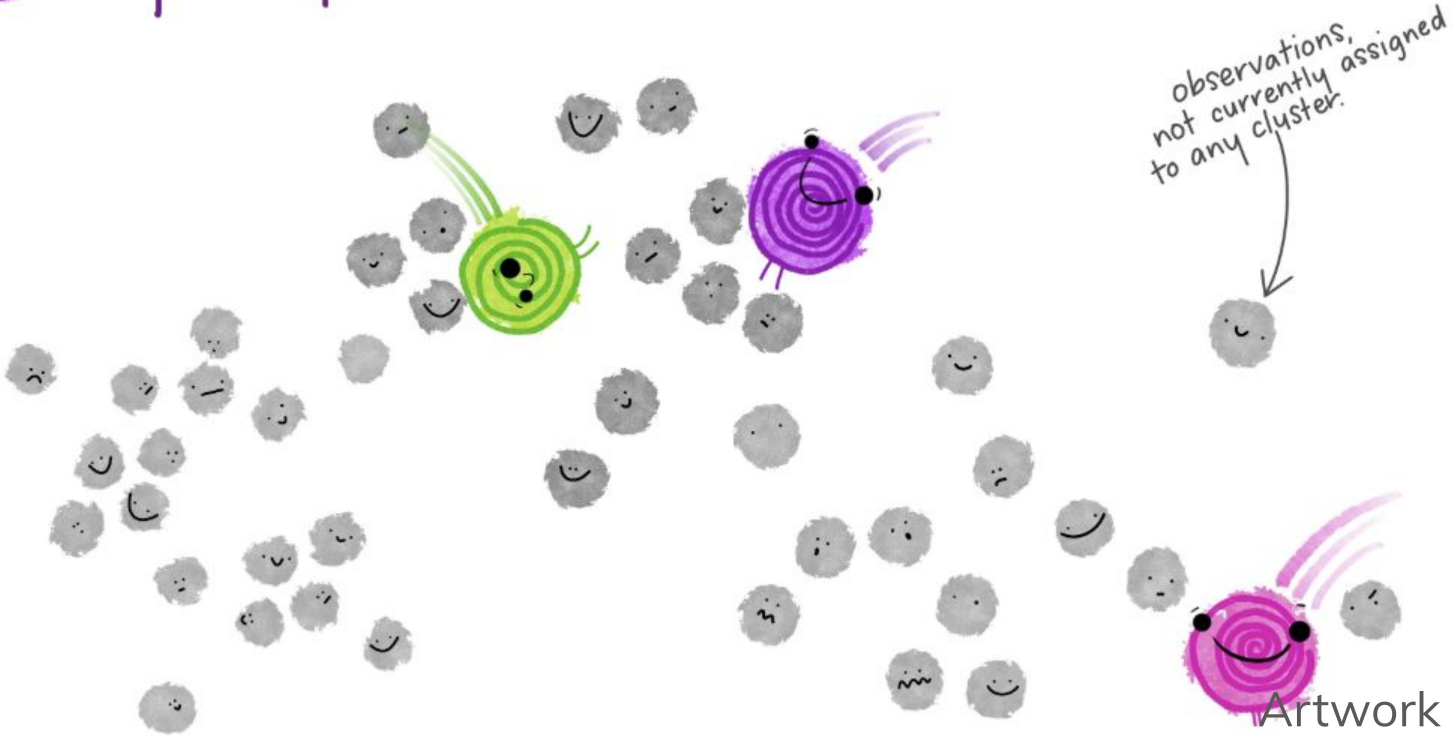
① Specify the number of clusters (in this example, k = 3).

Then imagine k cluster centroids are created.



Artwork by
Allison Horst

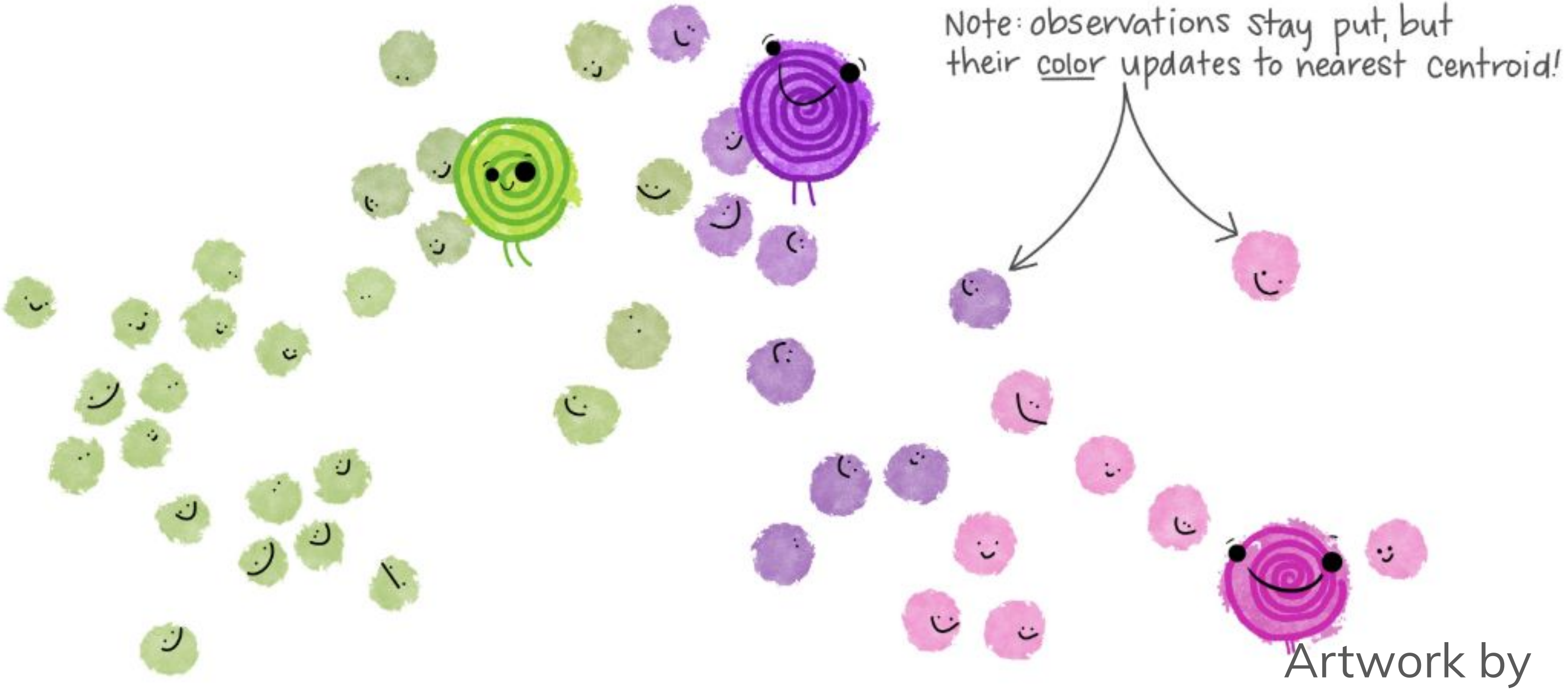
② Those k centroids get randomly placed in your space.



Artwork by Allison Horst

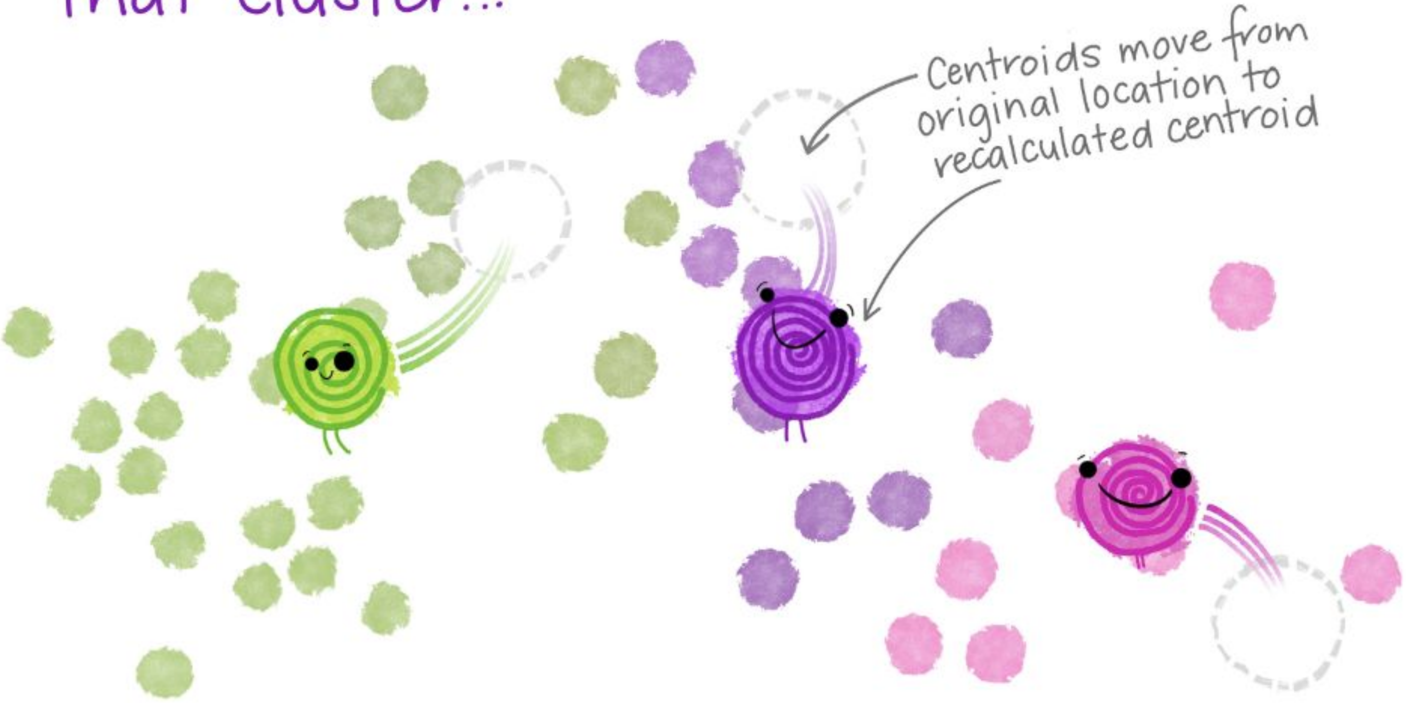
3 Each observation gets temporarily "assigned" to its closest centroid.

(e.g. by Euclidean distance)



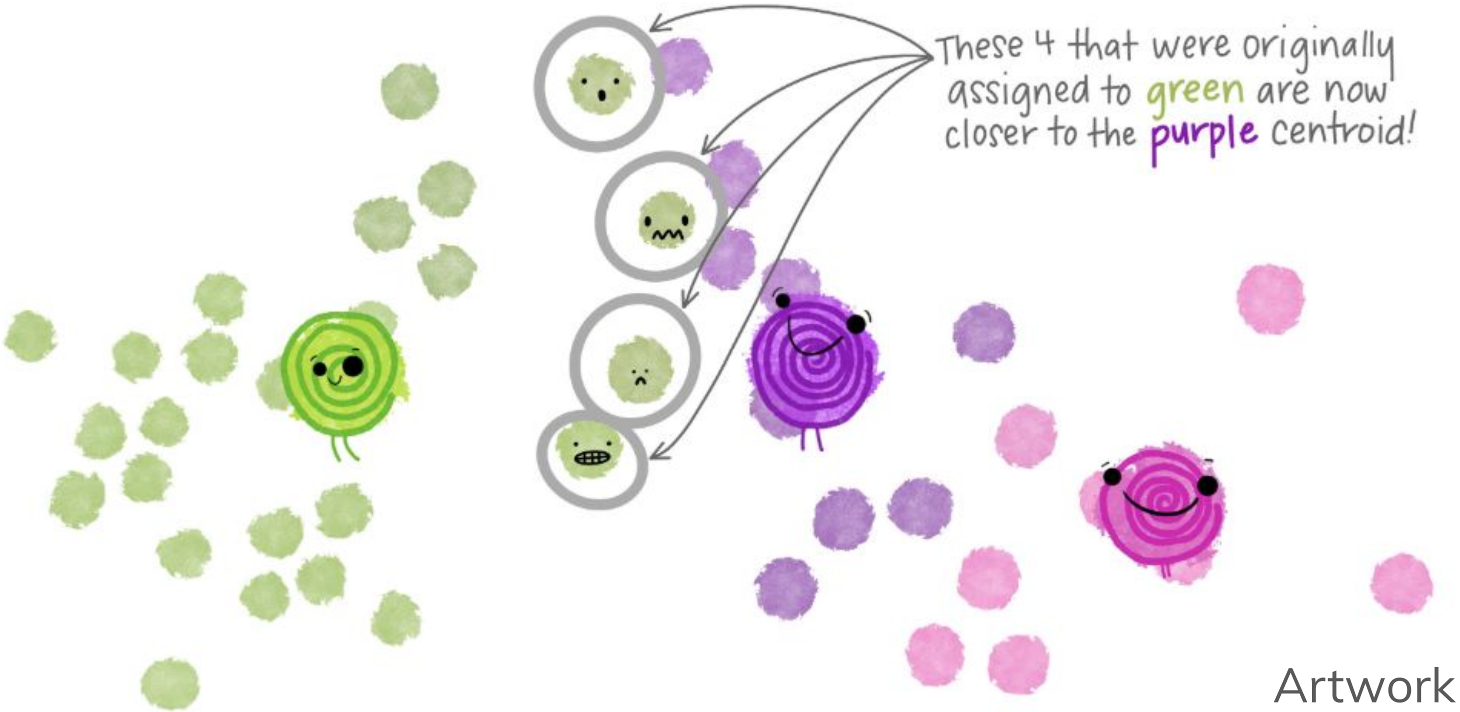
Artwork by Allison Horst

④ Then the centroid of each cluster is calculated based on all observations assigned to that cluster...



Artwork by
Allison Horst

UH OH. Now that the cluster centroids have moved, some of the observations are now closer to a different centroid!



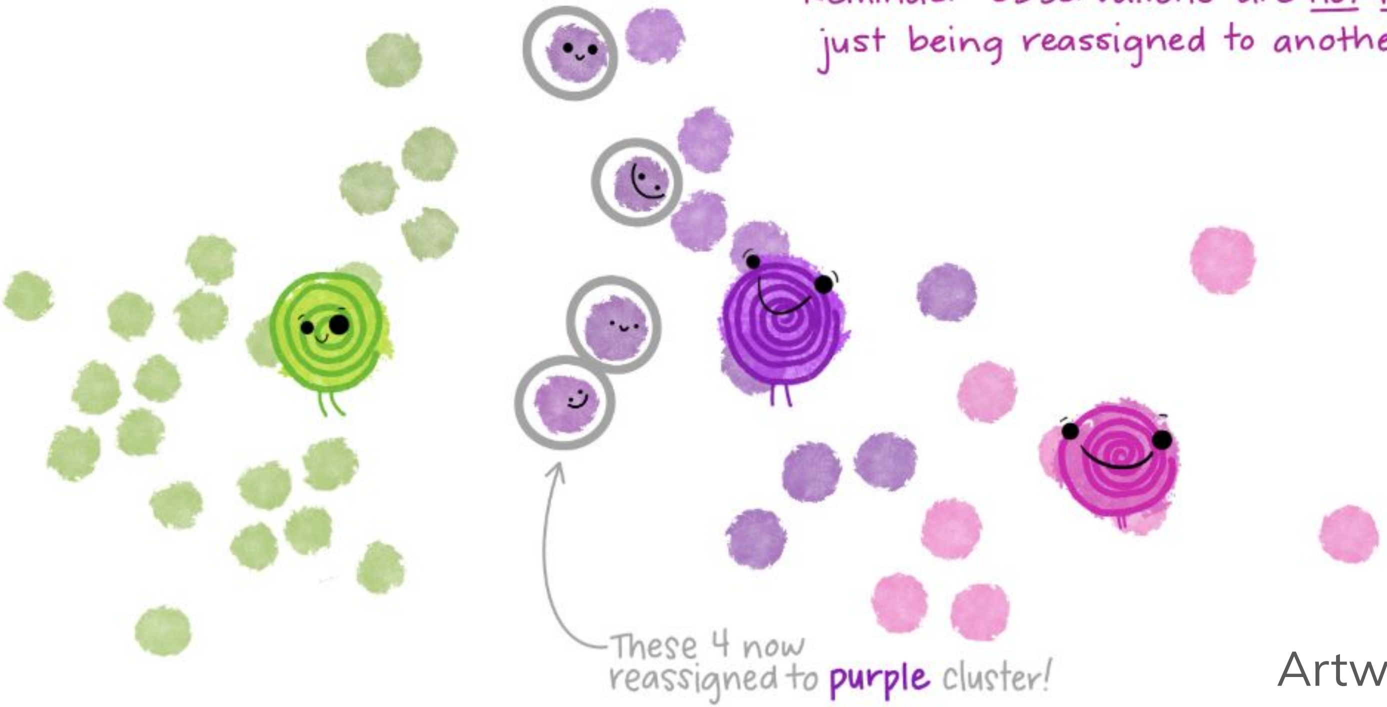
Artwork by
Allison Horst

5

NO PROBLEM!

Observations get reassigned* to a different cluster based on the recalculated centroid.

*Reminder: observations are not moving, just being reassigned to another cluster.

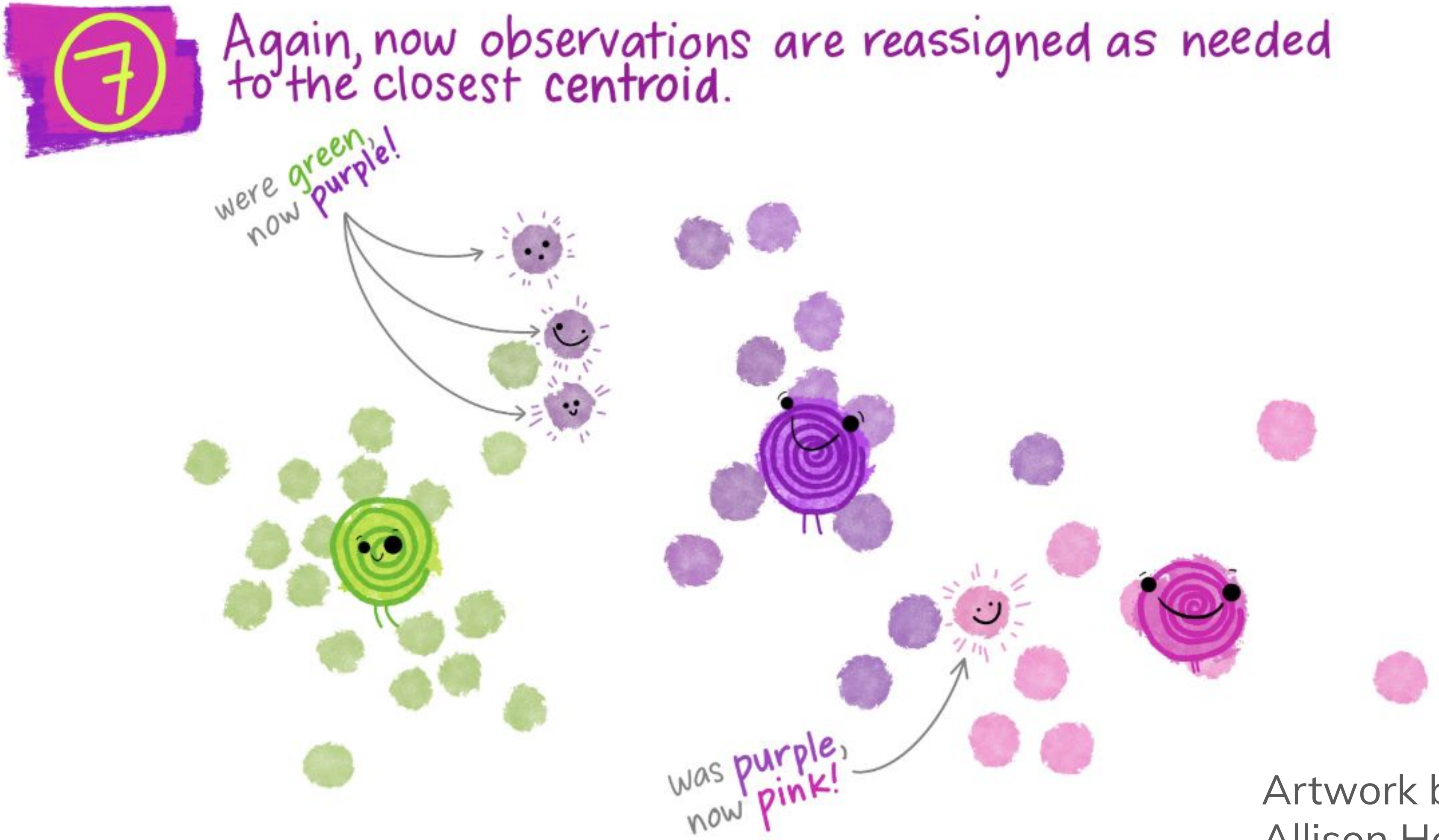


Artwork by Allison Horst

6 But now that observations have been reassigned, the centroids need to move again [recalculate centroids from updated clusters]

Artwork by
Allison Horst

Again, now observations are reassigned as needed to the closest centroid.



Artwork by Allison Horst

Then the centroid for each cluster is recalculated...

...which means observations will be reassigned...

That iterative process of

Recalculate cluster centroids

→ Reassign observations to nearest centroid

→ Recalculate cluster centroids

→ Reassign observations to nearest centroid

→ Recalculate cluster centroids

→ Reassign observations to nearest centroid

↓

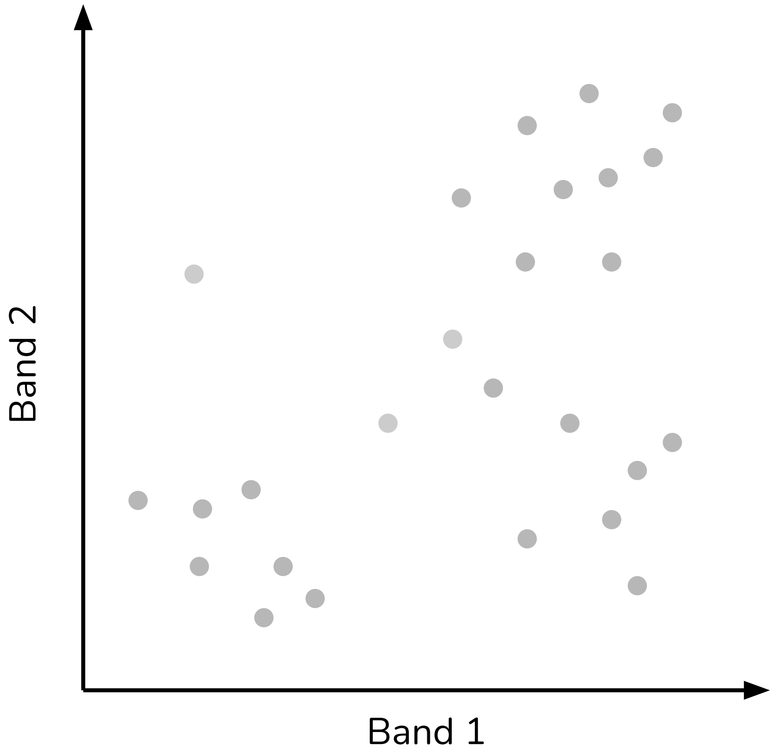
Continues until nothing is moving or being reassigned anymore!

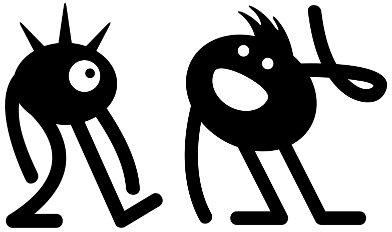
fin

Which means the iteration is done and each observation is assigned to its final cluster.

Artwork by Allison Horst

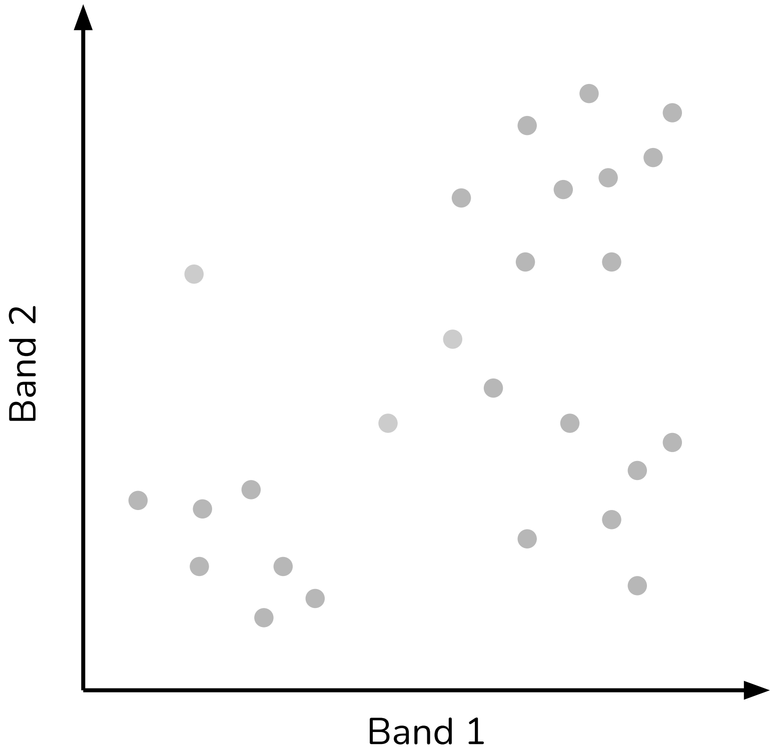
How to group pixels into land cover types





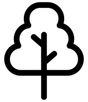
How to group pixels into land cover types

- Pick a number of groups



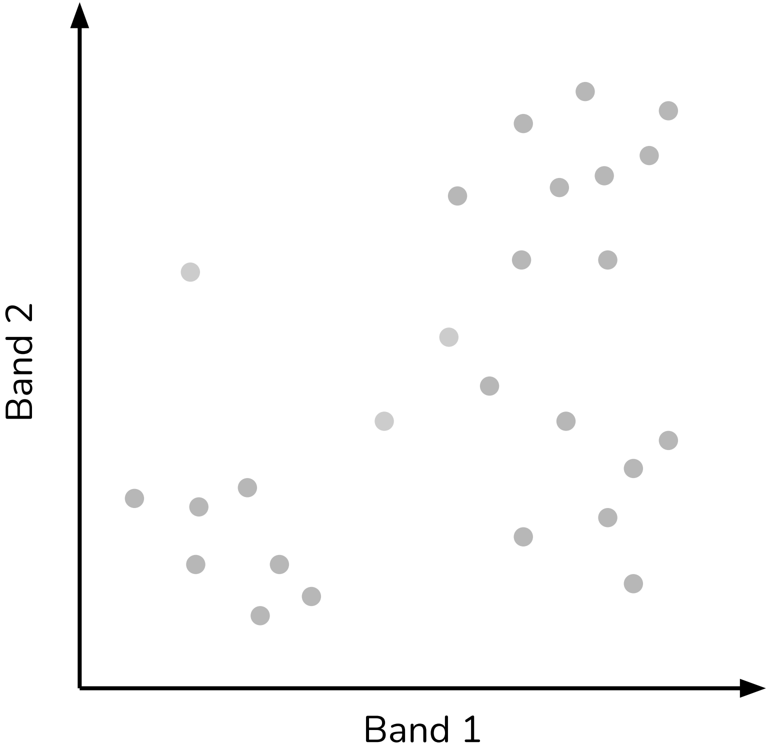
How to group pixels into land cover types

- Pick a number of groups



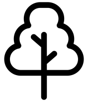
or

or

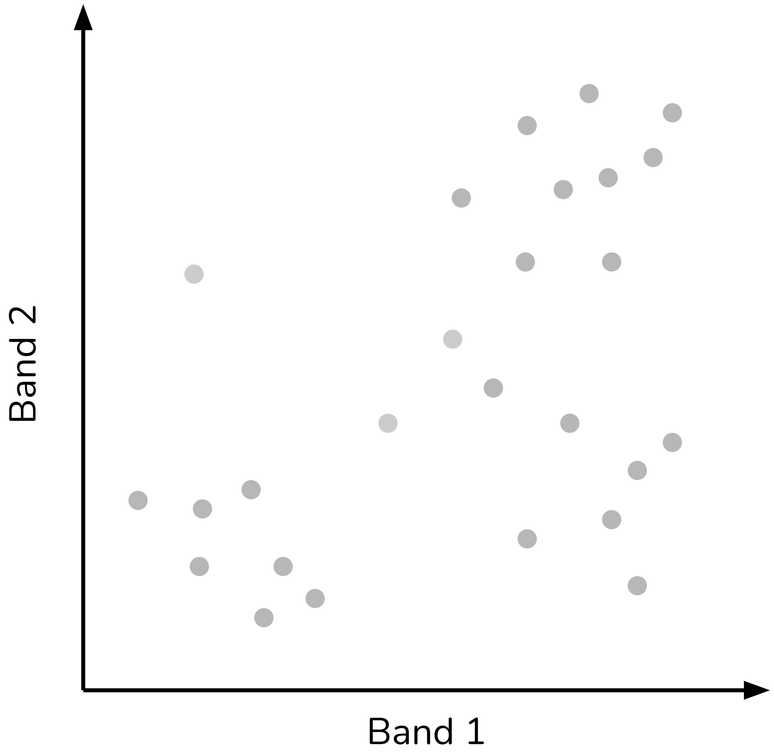


How to group pixels into land cover types

- Pick a number of groups

 or  or 

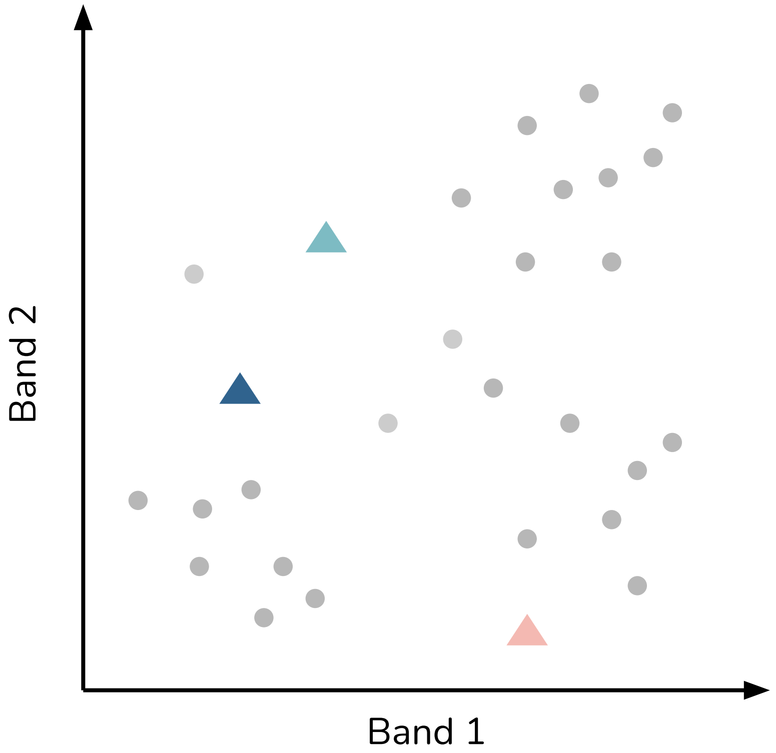
- Make a guess about where those groups are in feature space



How to group pixels into land cover types

- Pick a number of groups

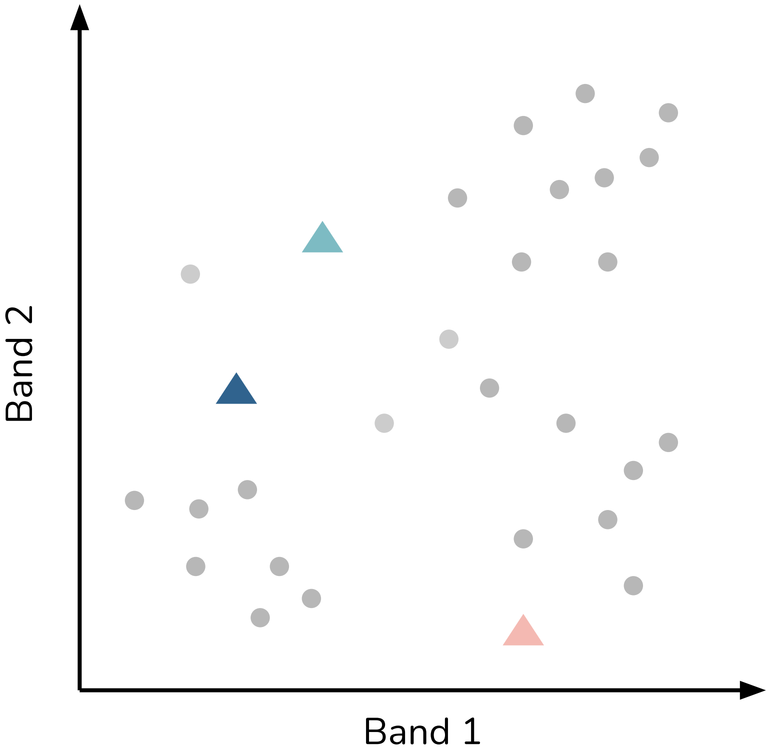
- Make a guess about where those groups are in feature space



How to group pixels into land cover types

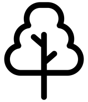
- Pick a number of groups

- Make a guess about where those groups are in feature space
- Assign each point to the closest group

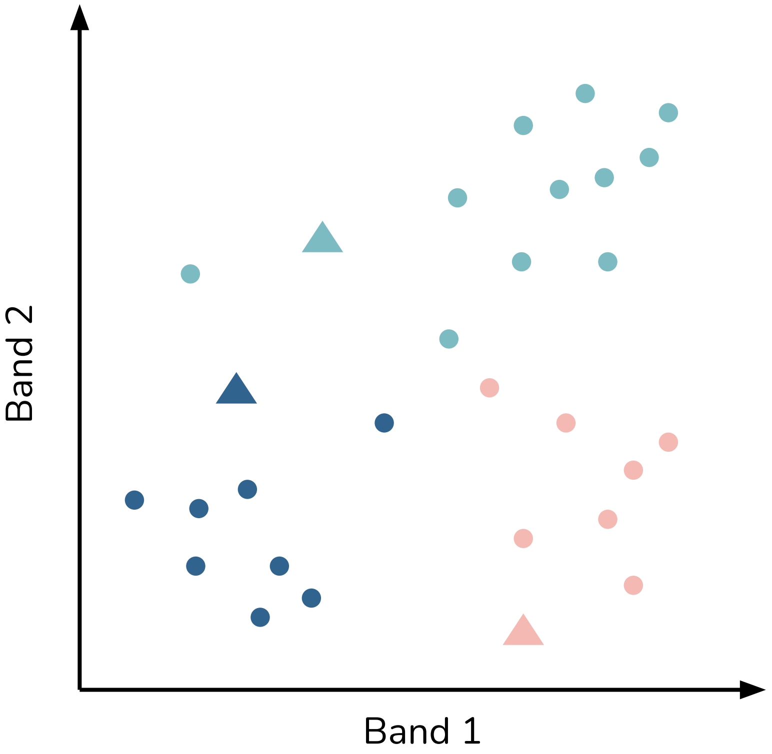


How to group pixels into land cover types

- Pick a number of groups

 or  or 

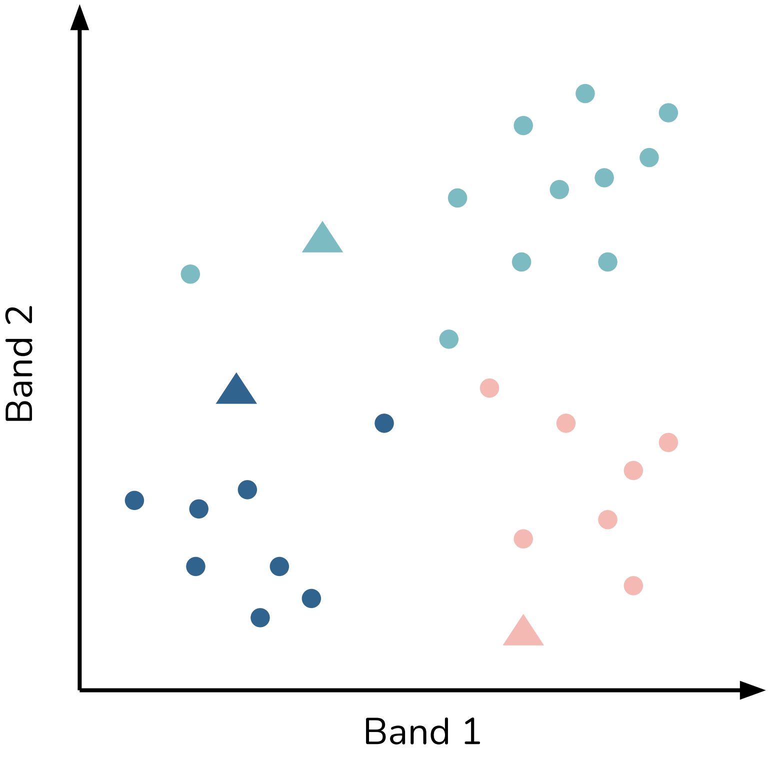
- Make a guess about where those groups are in feature space
- Assign each point to the closest group



How to group pixels into land cover types

- Pick a number of groups

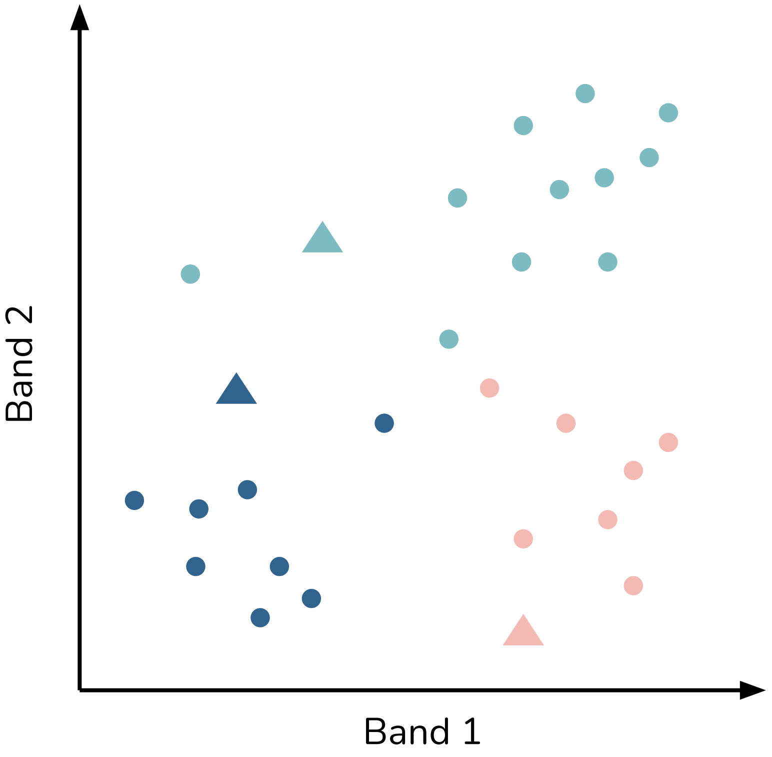
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups



How to group pixels into land cover types

- Pick a number of groups

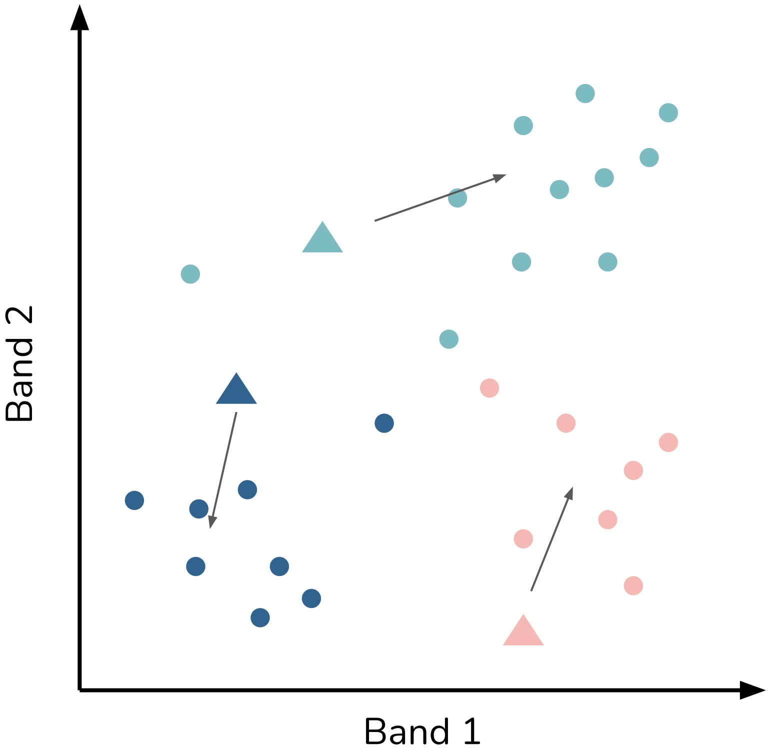
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
    - Use the mean!



How to group pixels into land cover types

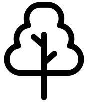
- Pick a number of groups

- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
    - Use the mean!

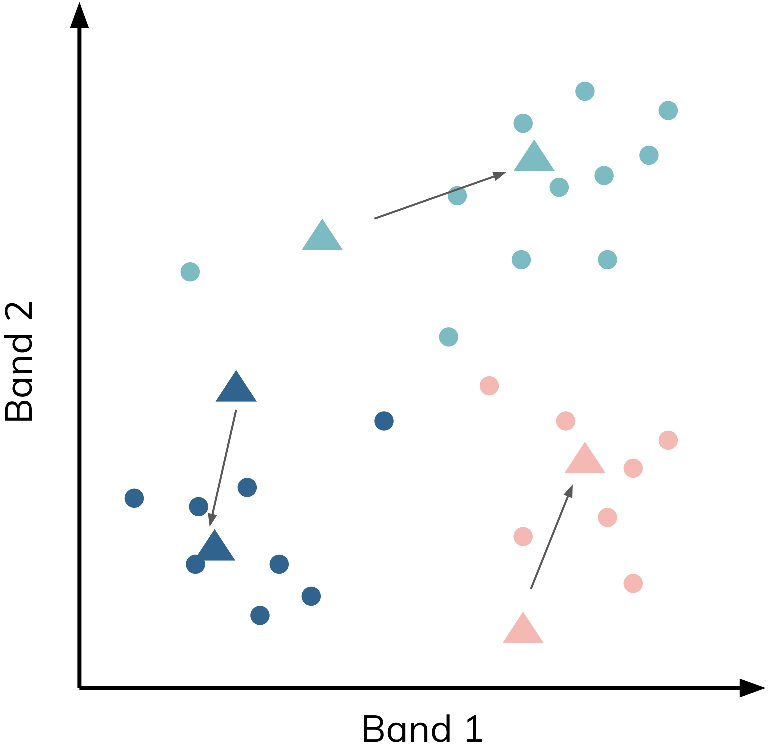


How to group pixels into land cover types

- Pick a number of groups

 or  or 

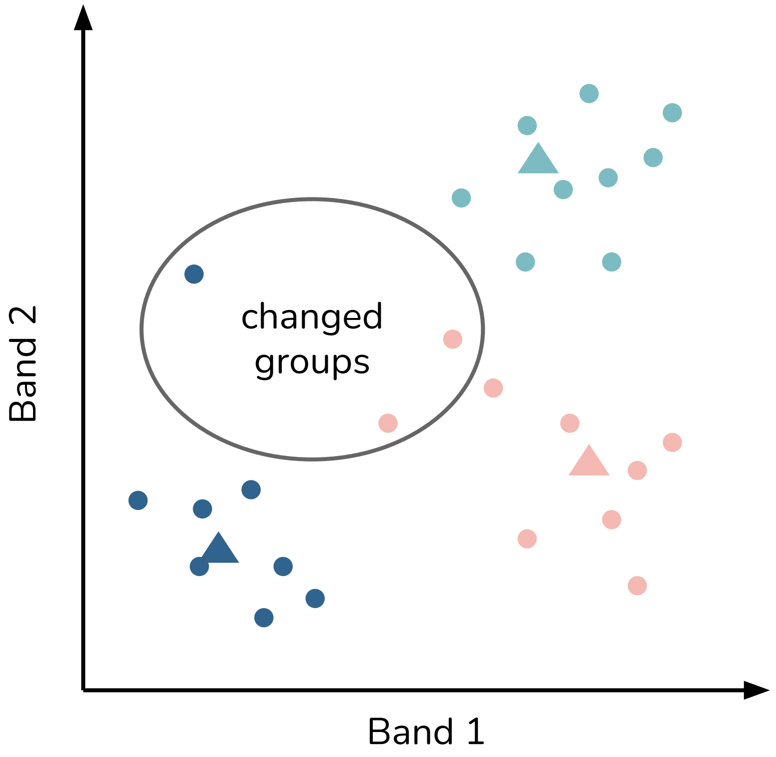
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!



How to group pixels into land cover types

- Pick a number of groups

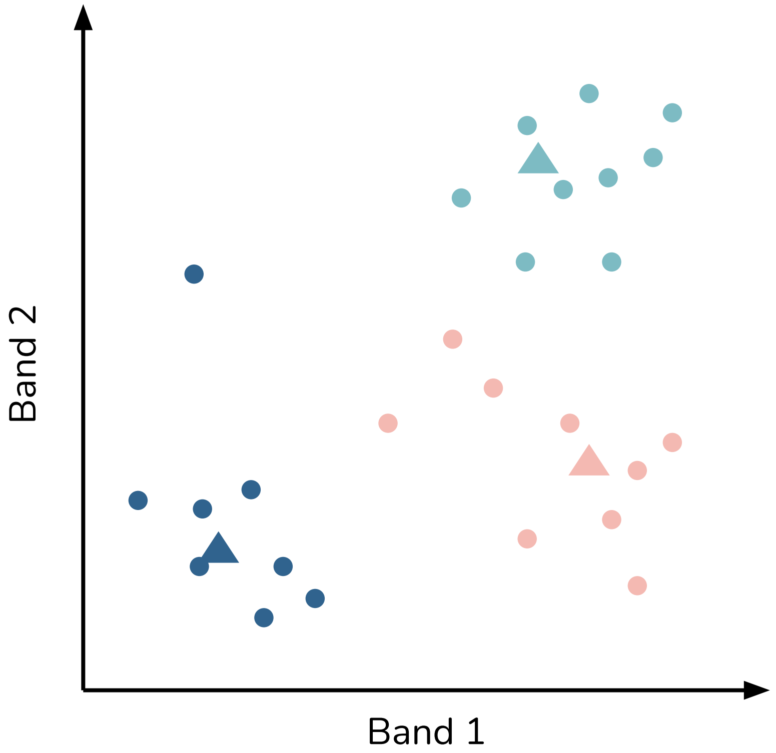
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
    - Use the mean!
- Update groups



How to group pixels into land cover types

- Pick a number of groups

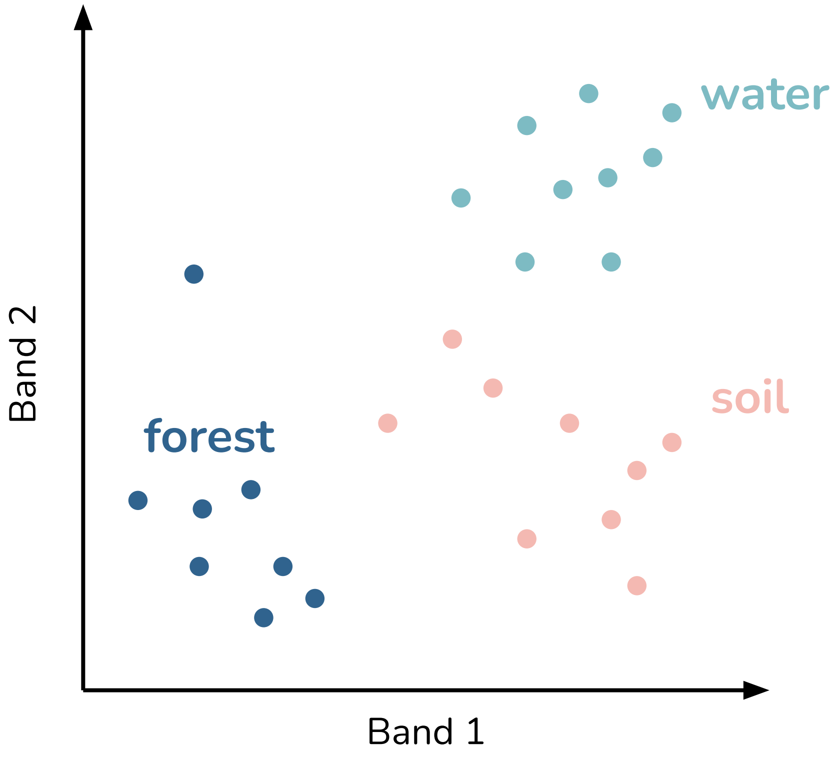
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!
- Update groups
- Keep going until distances are minimized



How to group pixels into land cover types

- Pick a number of groups

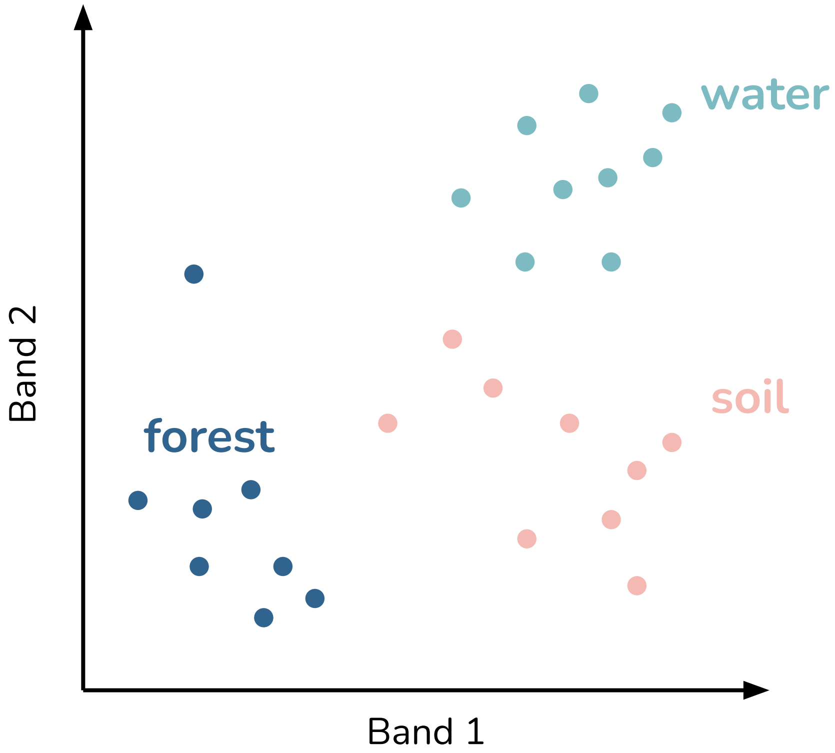
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are



k-means clustering

- Pick a number of groups

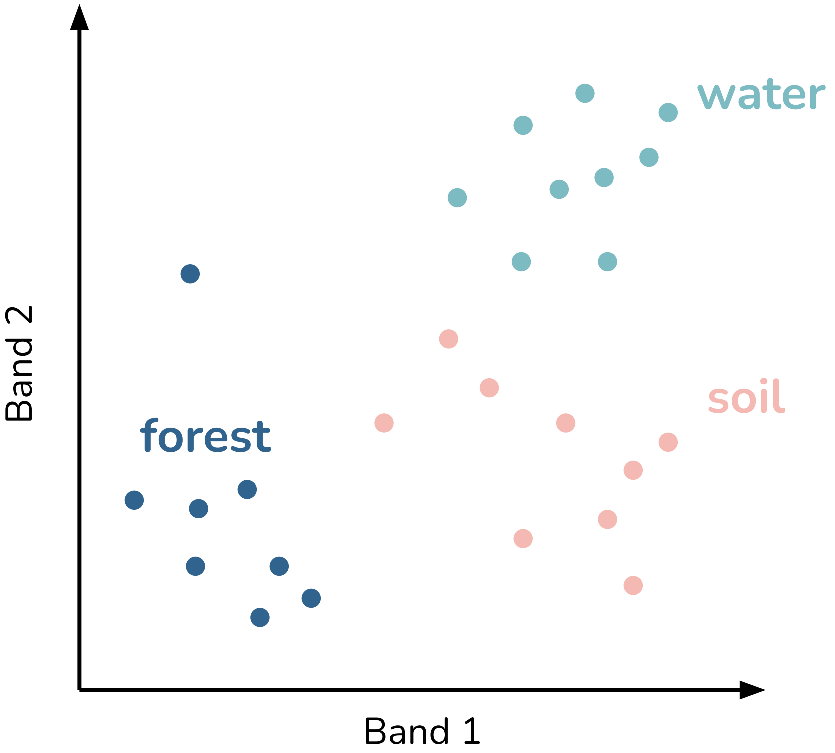
- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
    - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are



k-means clustering

- Pick a number of groups

- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are

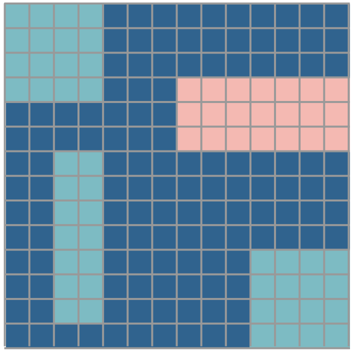


k is the number of groups (or clusters)

clusters are based on the group mean

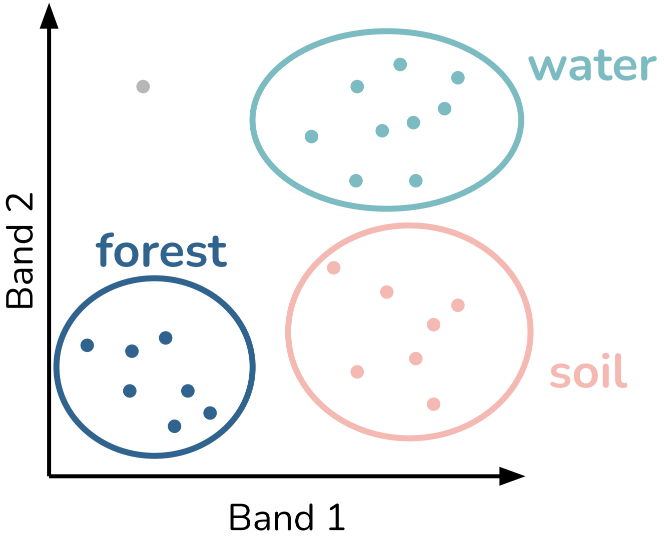
k-means clustering

Geographic space



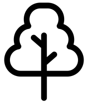
Feature space

Points are pixels

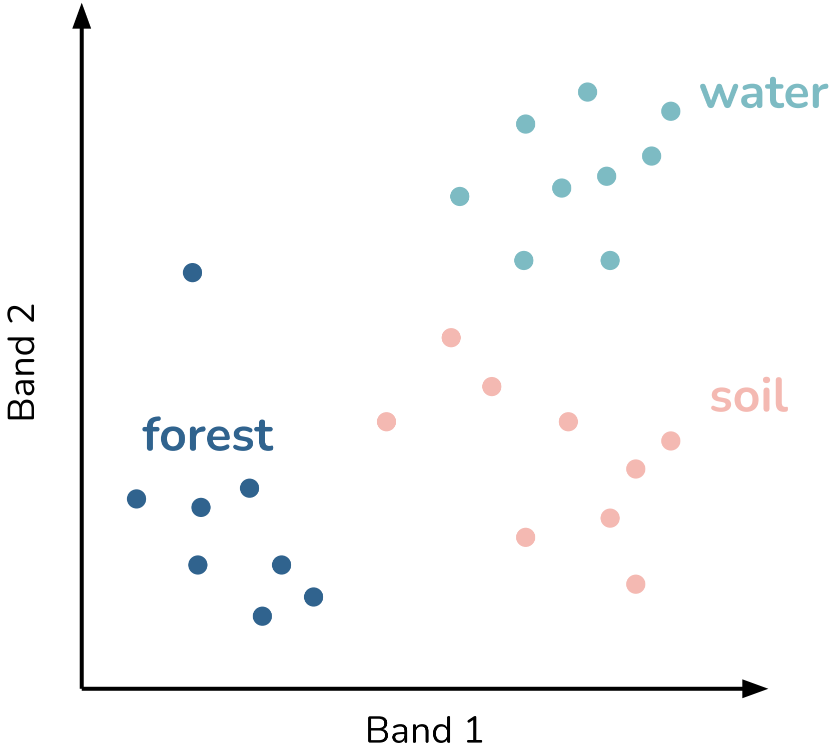


k-means clustering

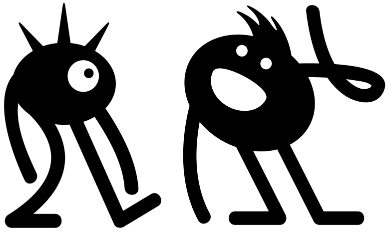
- Pick a number of groups

 or  or 

- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are

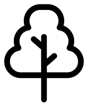


Pros/Cons

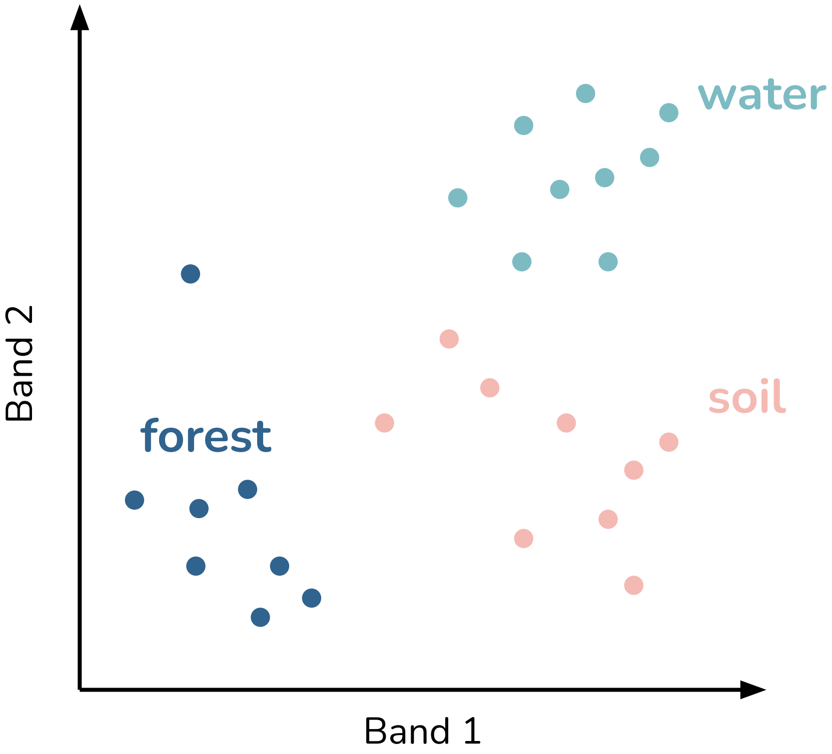


k-means clustering

- Pick a number of groups

 or  or 

- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
    - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are



Pros

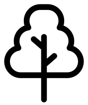
- Only needed remote sensing data
- Explored how similar different areas are

Cons

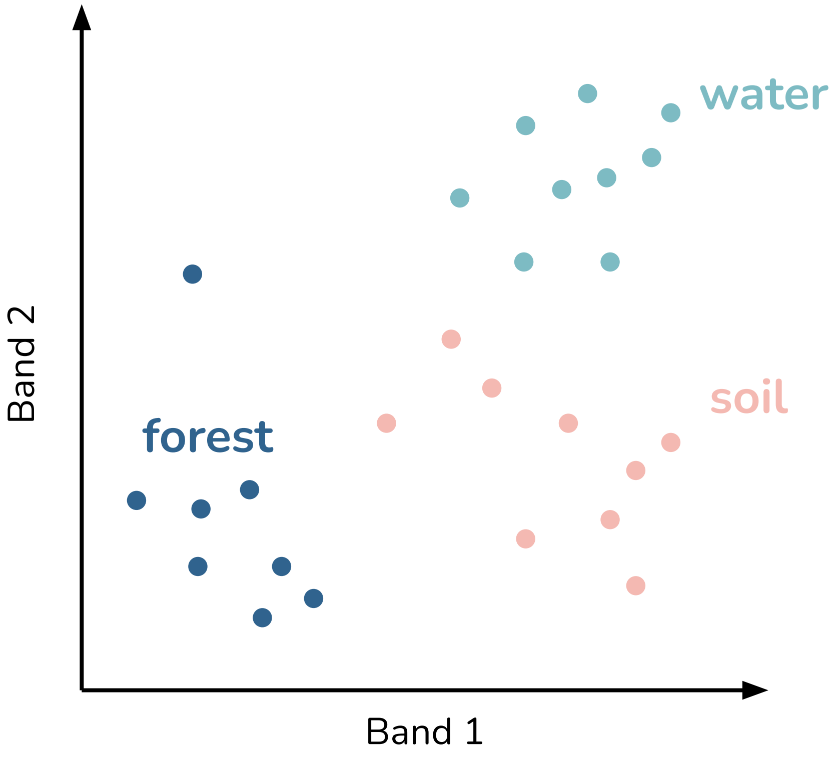
- Clusters might not always work out this well
- Number of groups was arbitrary
- Starting guess on clusters might impact results
- Needed to figure out what the clusters meant

k-means clustering

- Pick a number of groups

 or  or 

- Make a guess about where those groups are in feature space
- Assign each point to the closest group
- Move group centers to better represent groups
  - Use the mean!
- Update groups
- Keep going until distances are minimized
- Figure out what the groups are



Pros

- **Only needed remote sensing data**
- Explored how similar different areas are

Cons

- Clusters might not always work out this well
- Number of groups was arbitrary
- Starting guess on clusters might impact results
- **Needed to figure out what the clusters meant**

Image classification

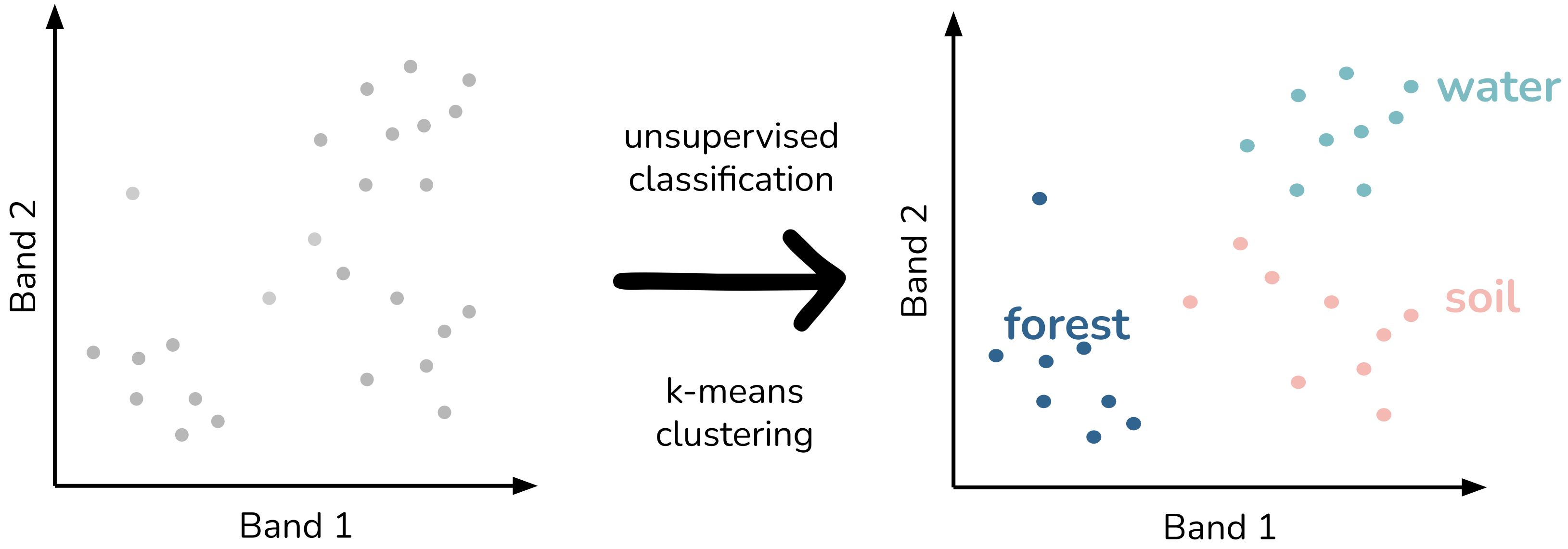


Image classification

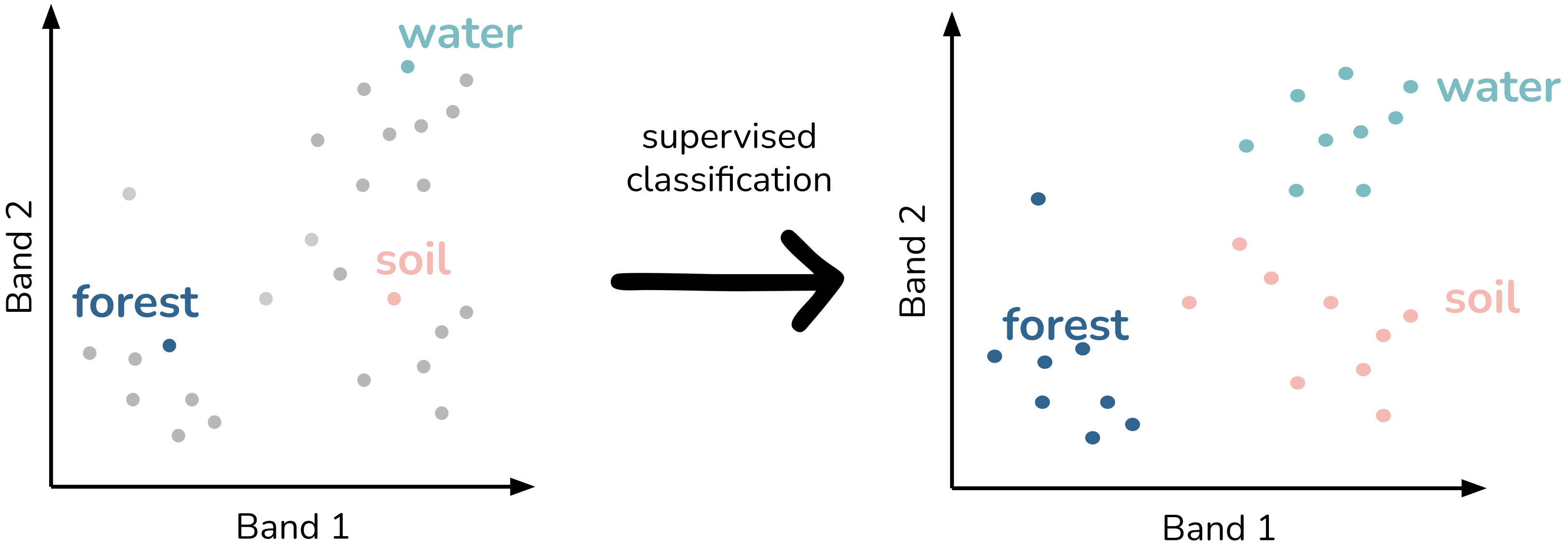
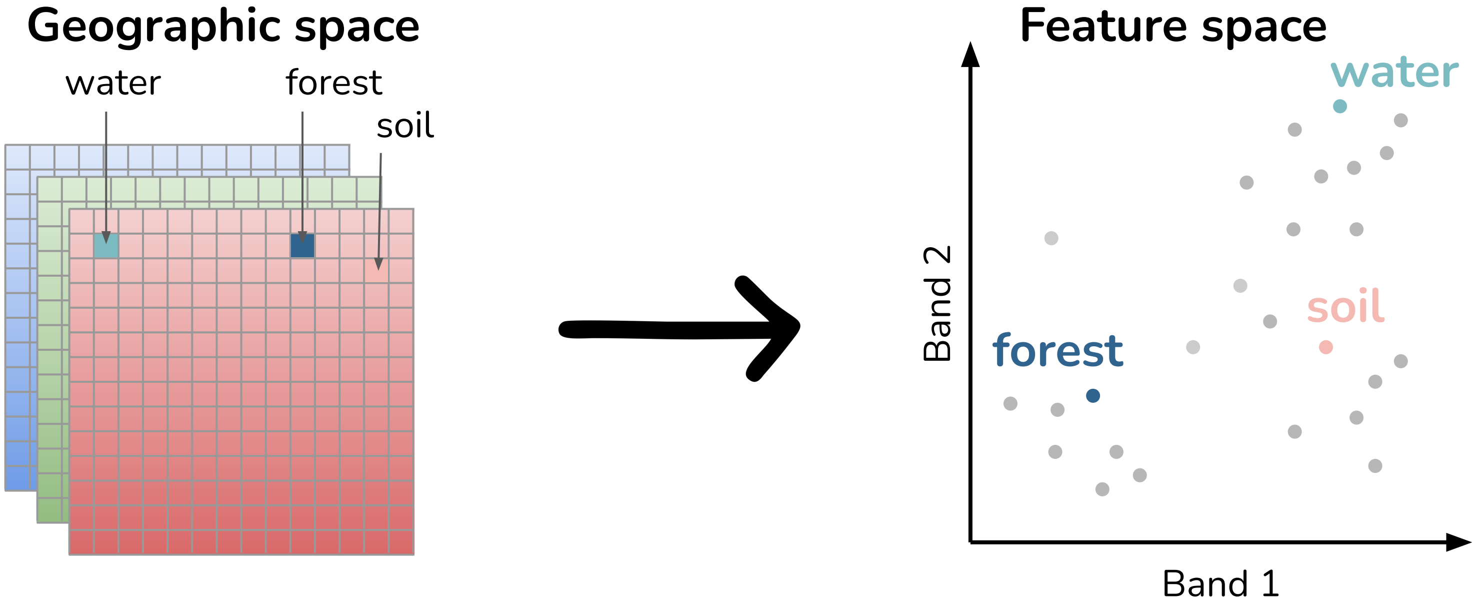
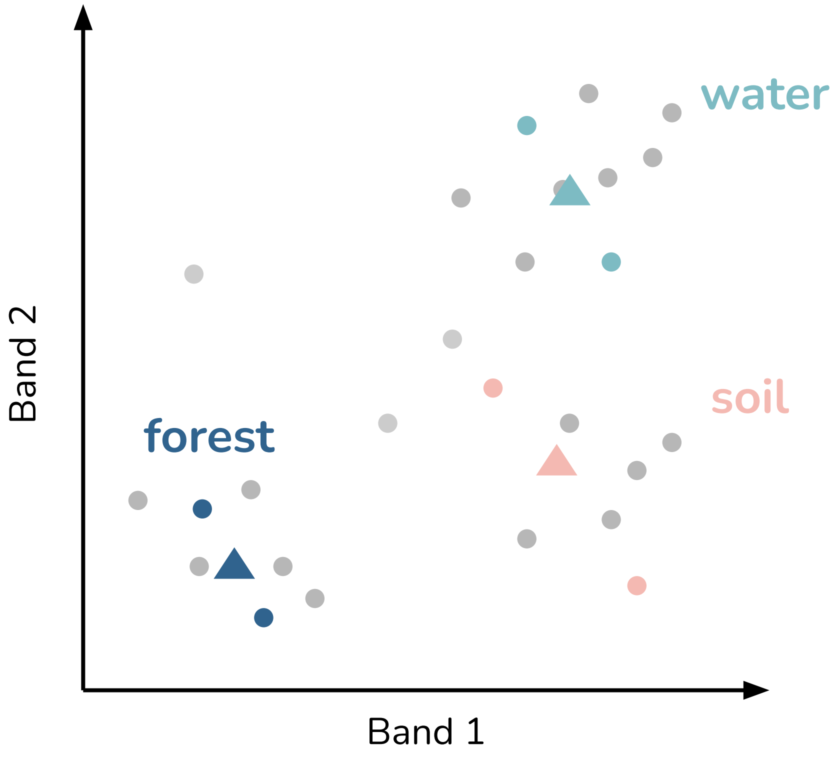
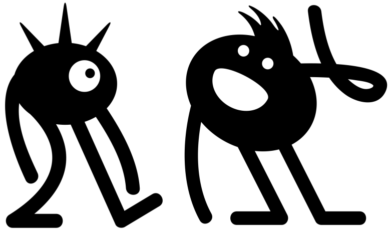


Image classification



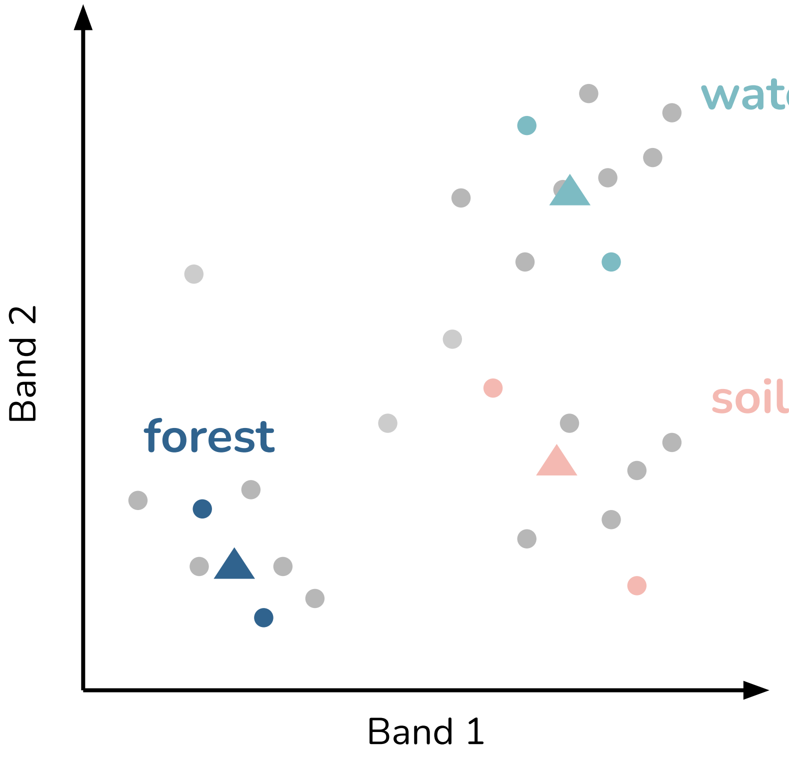
Supervised classification: Minimum distance to mean algorithm

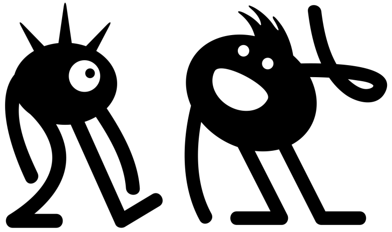




Supervised classification: Minimum distance to mean algorithm

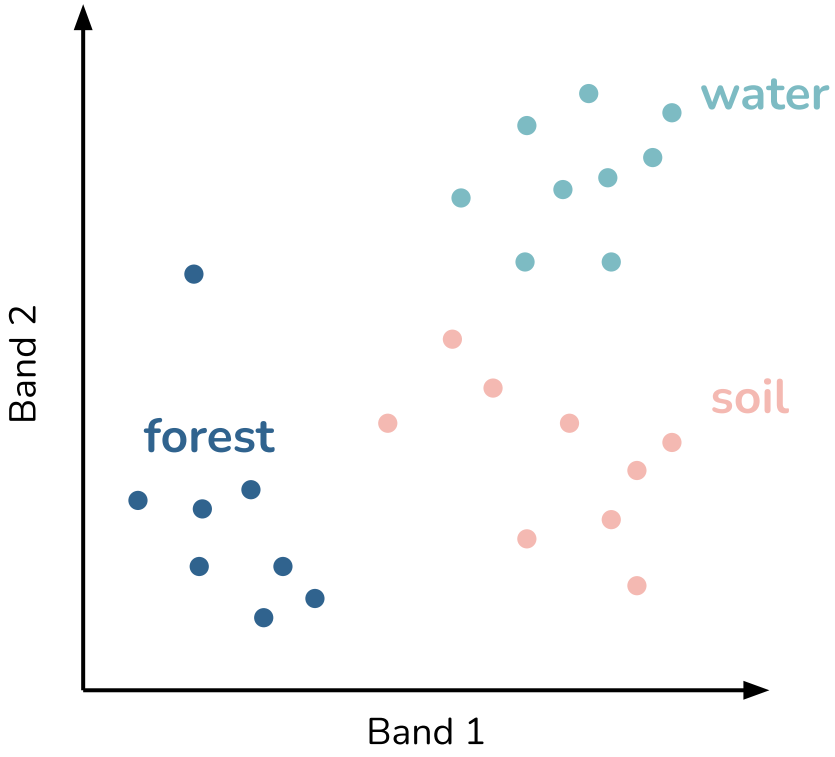
- Find means for each group based on known points





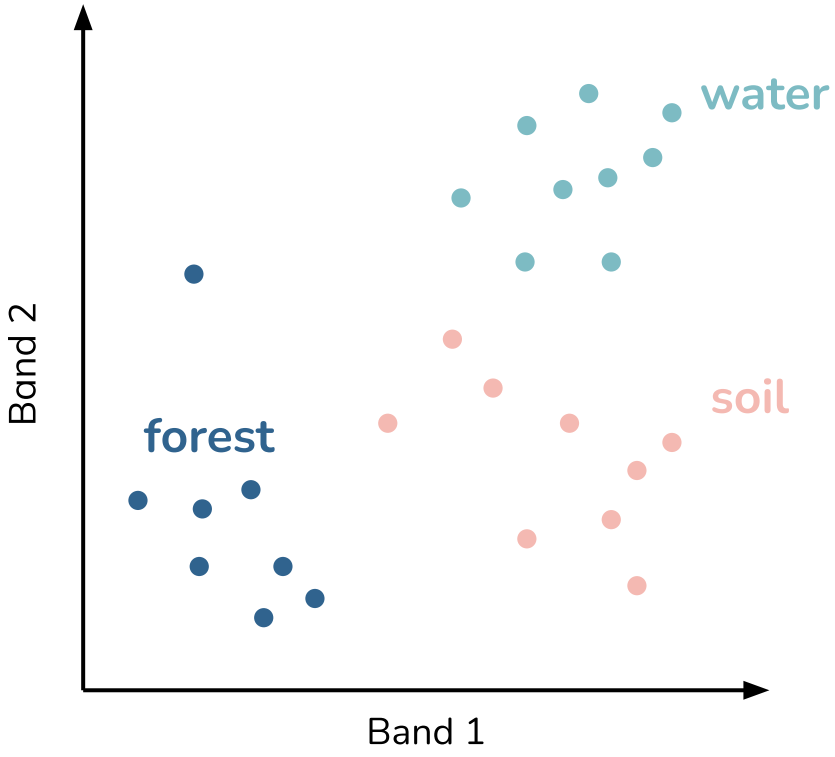
Supervised classification: Minimum distance to mean algorithm

- Find means for each group based on known points
- Assign each point to the closest group



Supervised classification: Minimum distance to mean algorithm

- Find means for each group based on known points
- Assign each point to the closest group



Pros

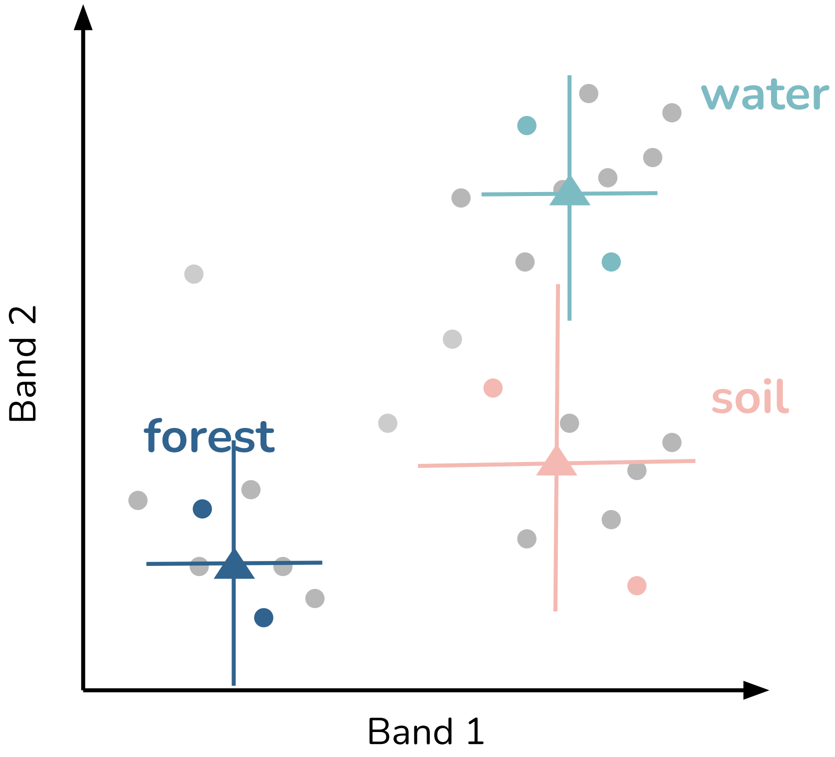
- fast/easy

Cons

- only uses means, not other statistical differences between classes

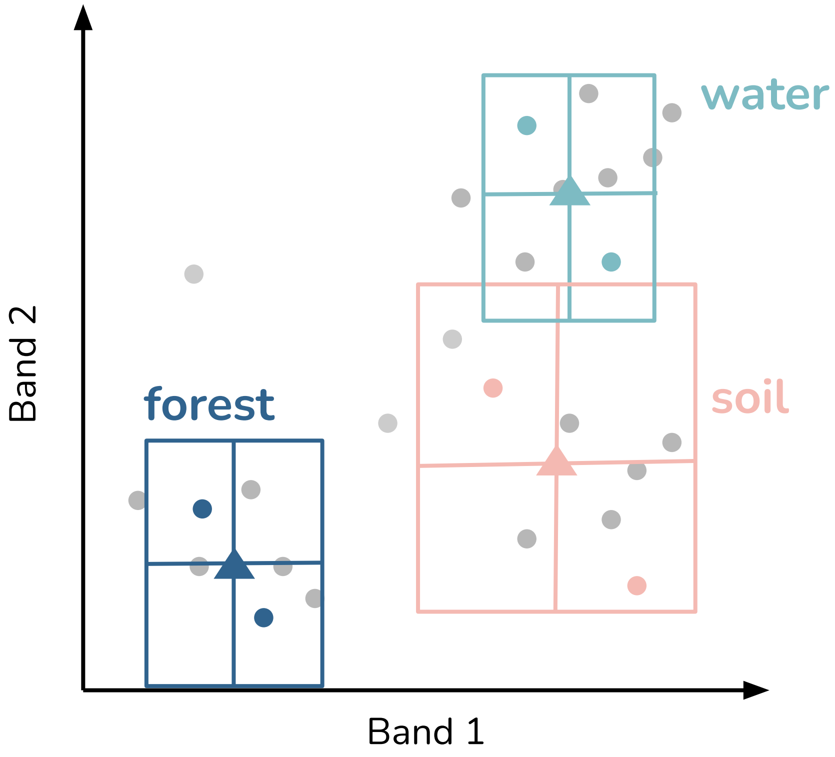
Supervised classification: Parallelipiped

- Find means and standard deviations for each group based on known points



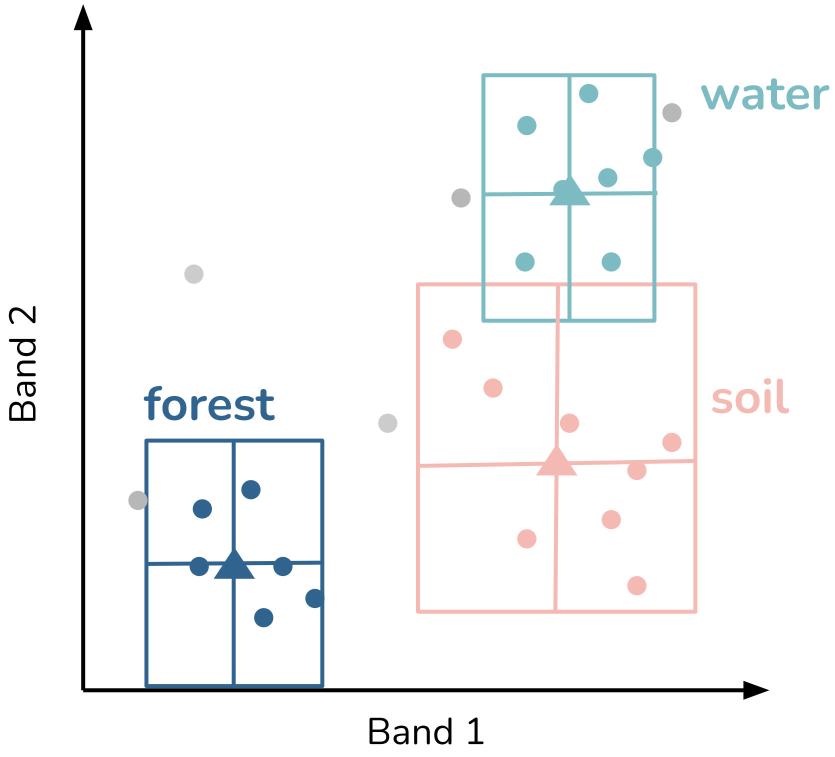
Supervised classification: Parallelipiped

- Find means and standard deviations for each group based on known points



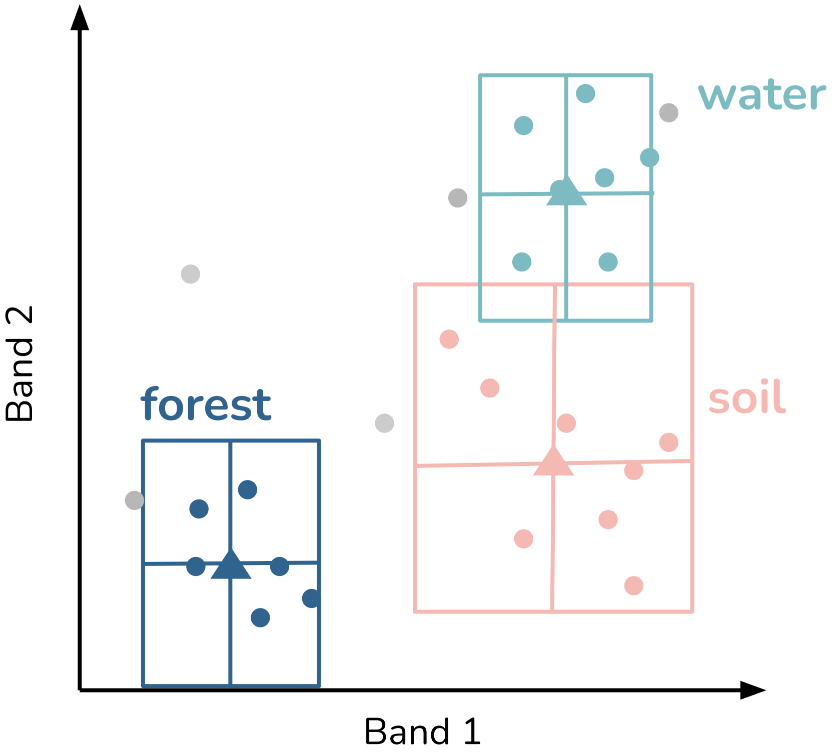
Supervised classification: Parallelipiped

- Find means and standard deviations for each group based on known points
- Assign points to groups



Supervised classification: Parallelipiped

- Find means and standard deviations for each group based on known points
- Assign points to groups



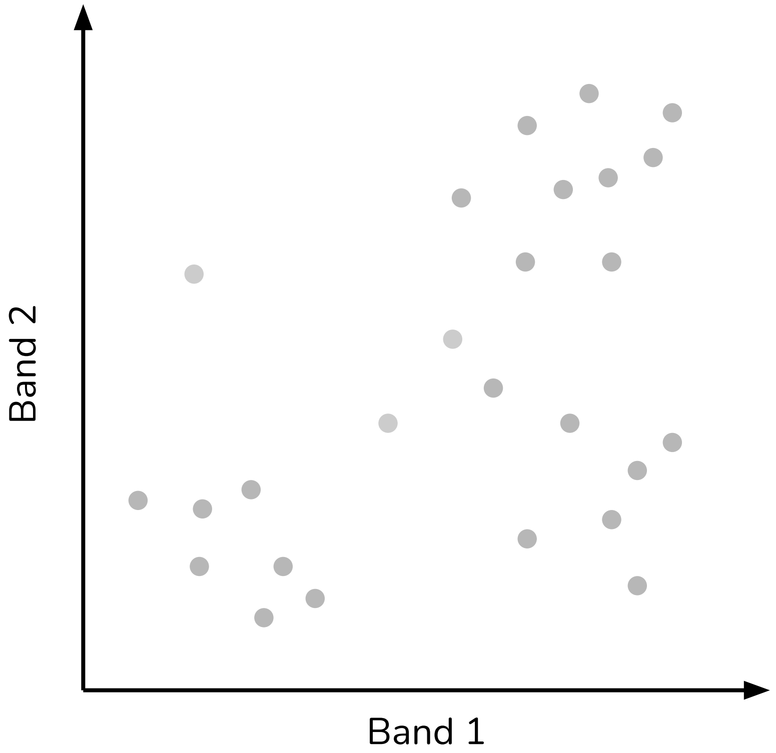
Pros

- fast/easy
- More realistic than just using the mean

Cons

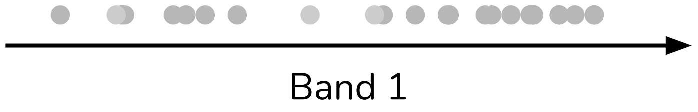
- Unclassified pixels
- Overlapping classes

Supervised classification



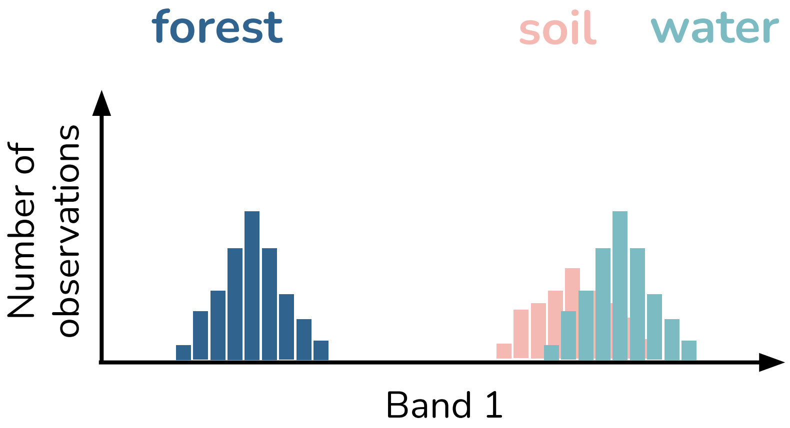
Supervised classification

Unknown pixels:

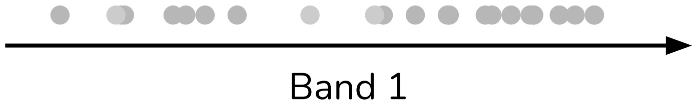


Supervised classification

Known pixels:

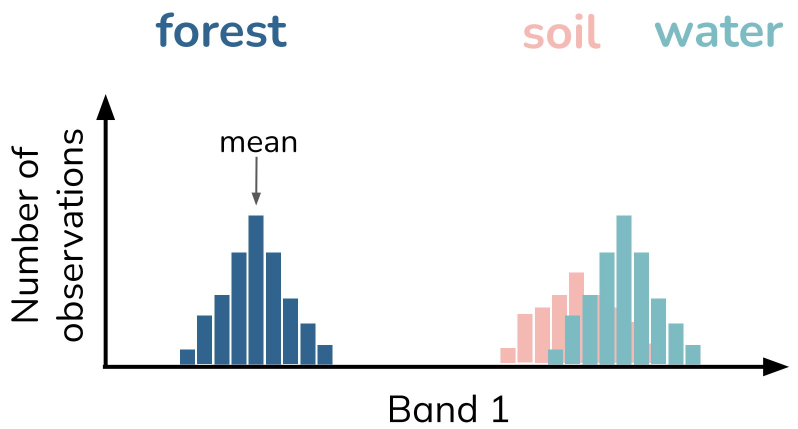


Unknown pixels:

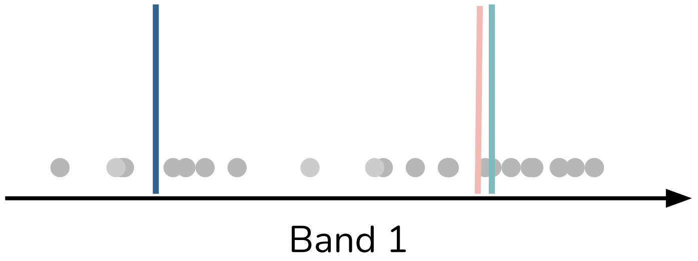


Supervised classification

Known pixels:



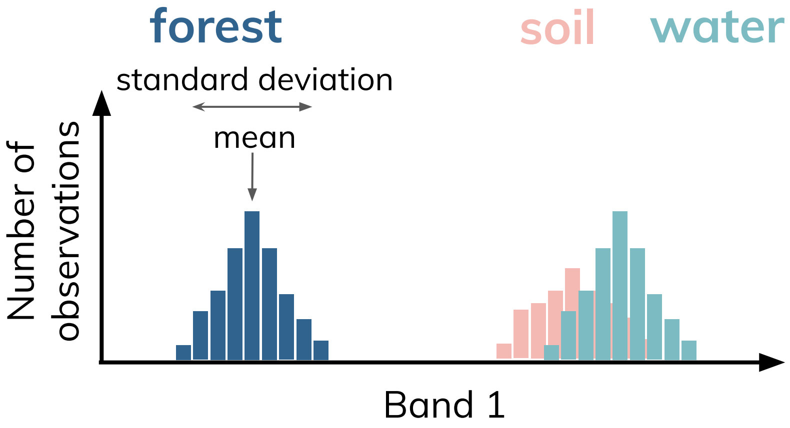
Unknown pixels:



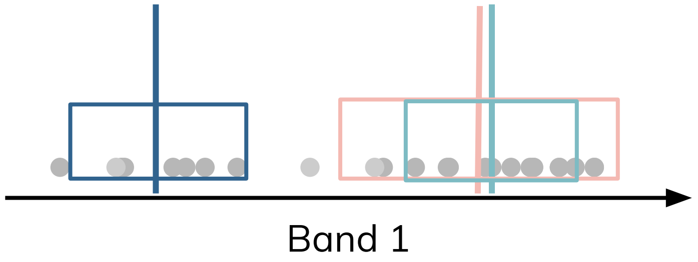
minimum distance to mean

Supervised classification

Known pixels:



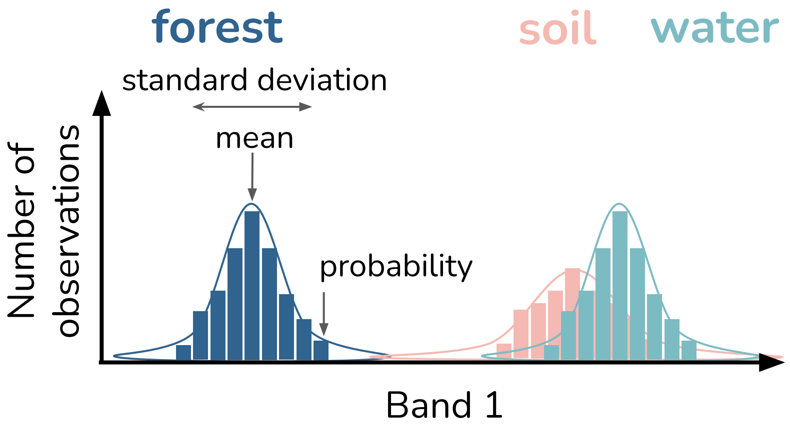
Unknown pixels:



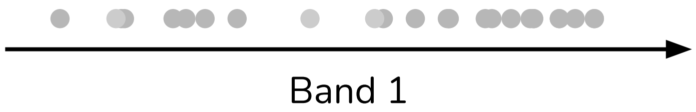
parallelipiped

Supervised classification

Known pixels:

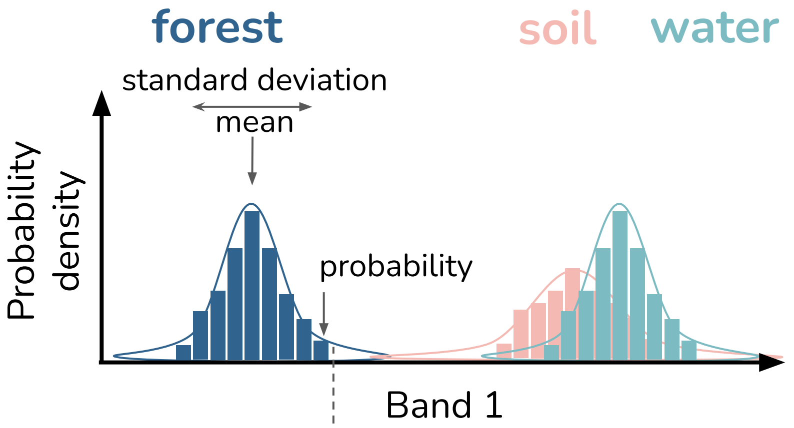


Unknown pixels:

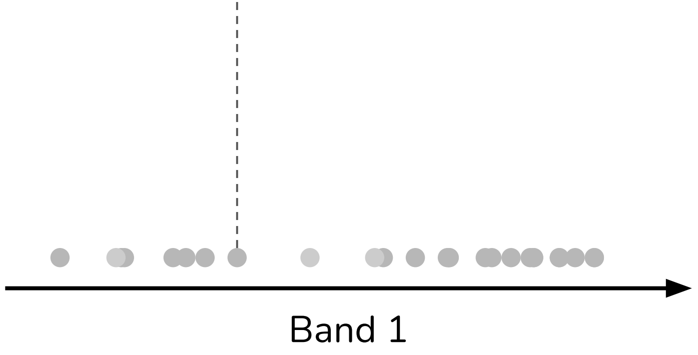


Supervised classification

Known pixels:



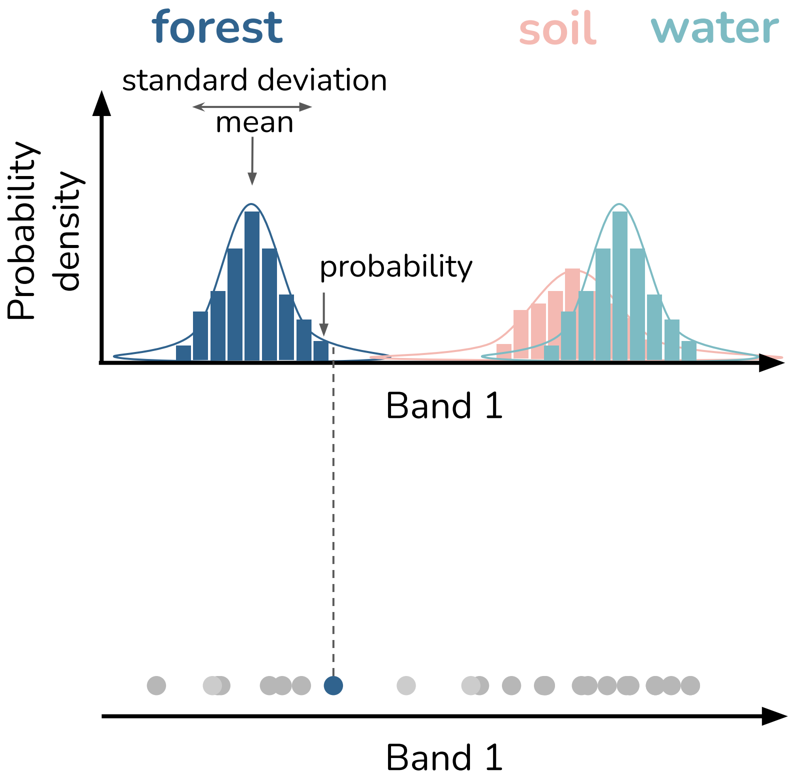
Unknown pixels:



Supervised classification

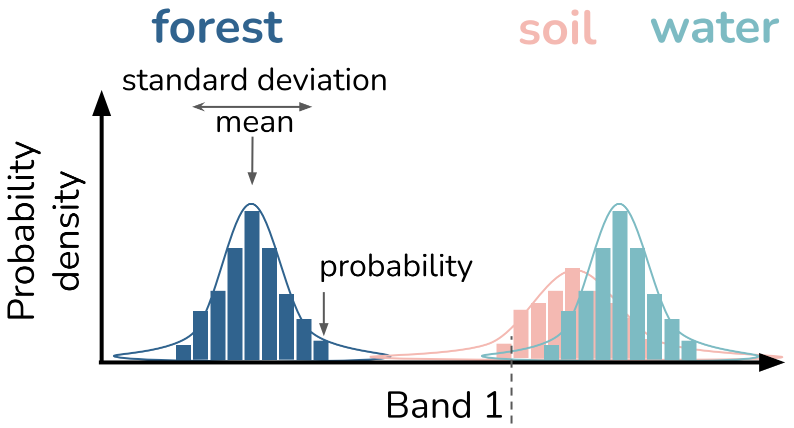
Known pixels:

Unknown pixels:

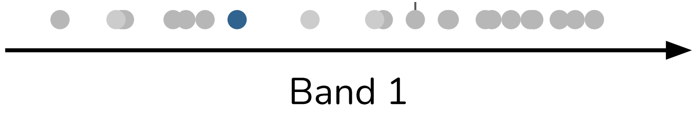


Supervised classification

Known pixels:



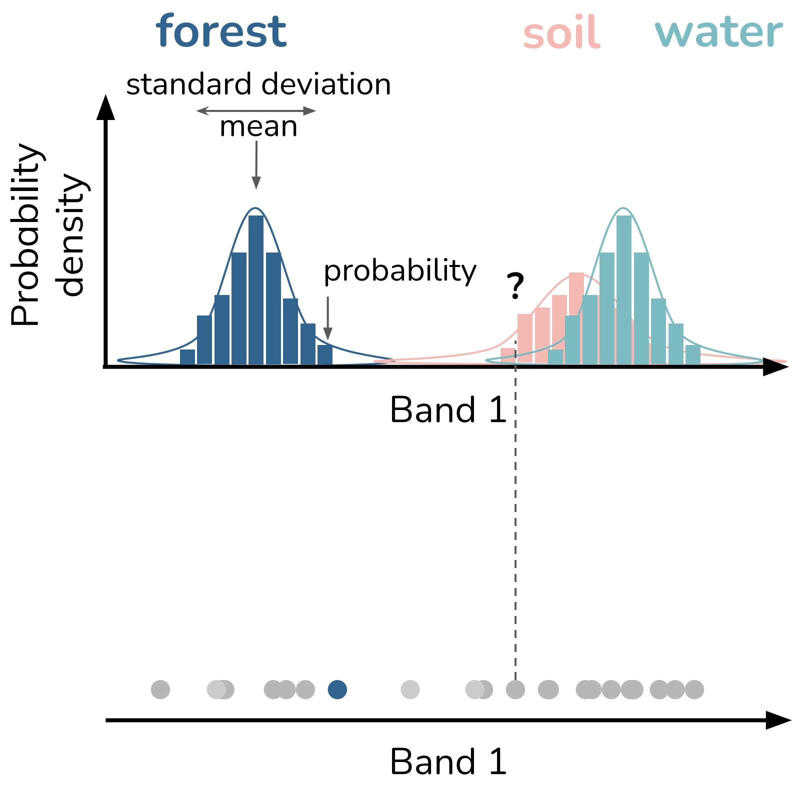
Unknown pixels:



Supervised classification

Known pixels:

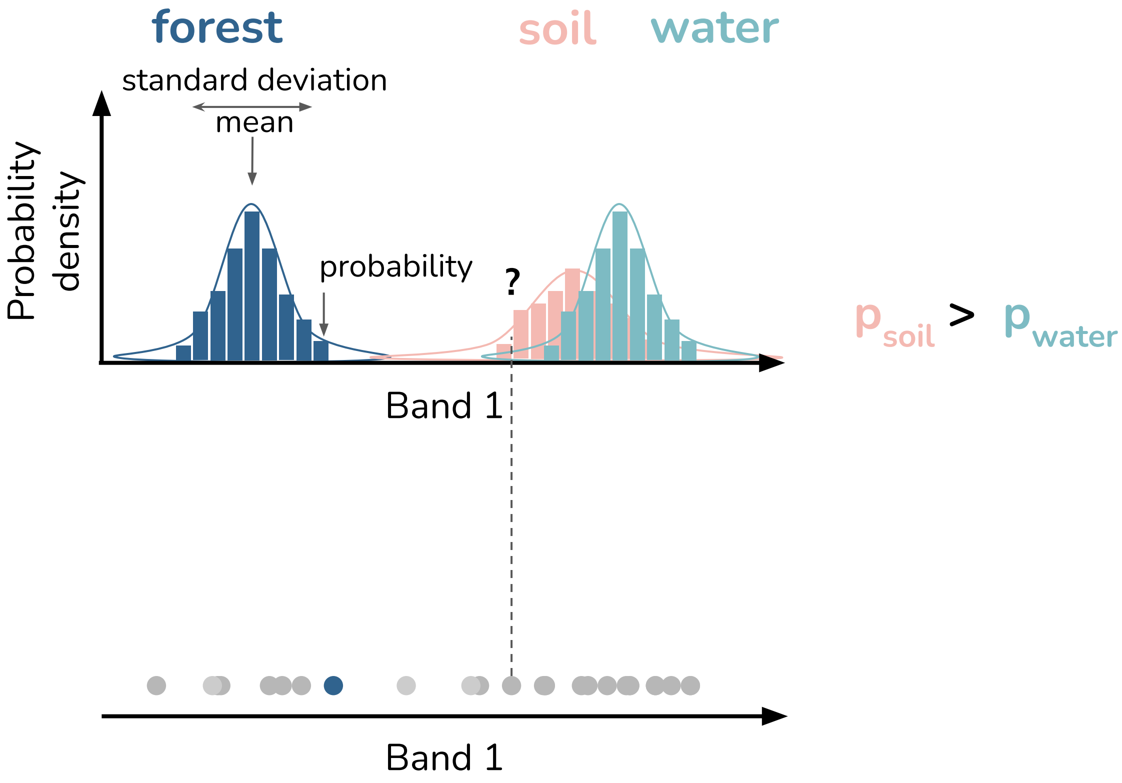
Unknown pixels:



Supervised classification

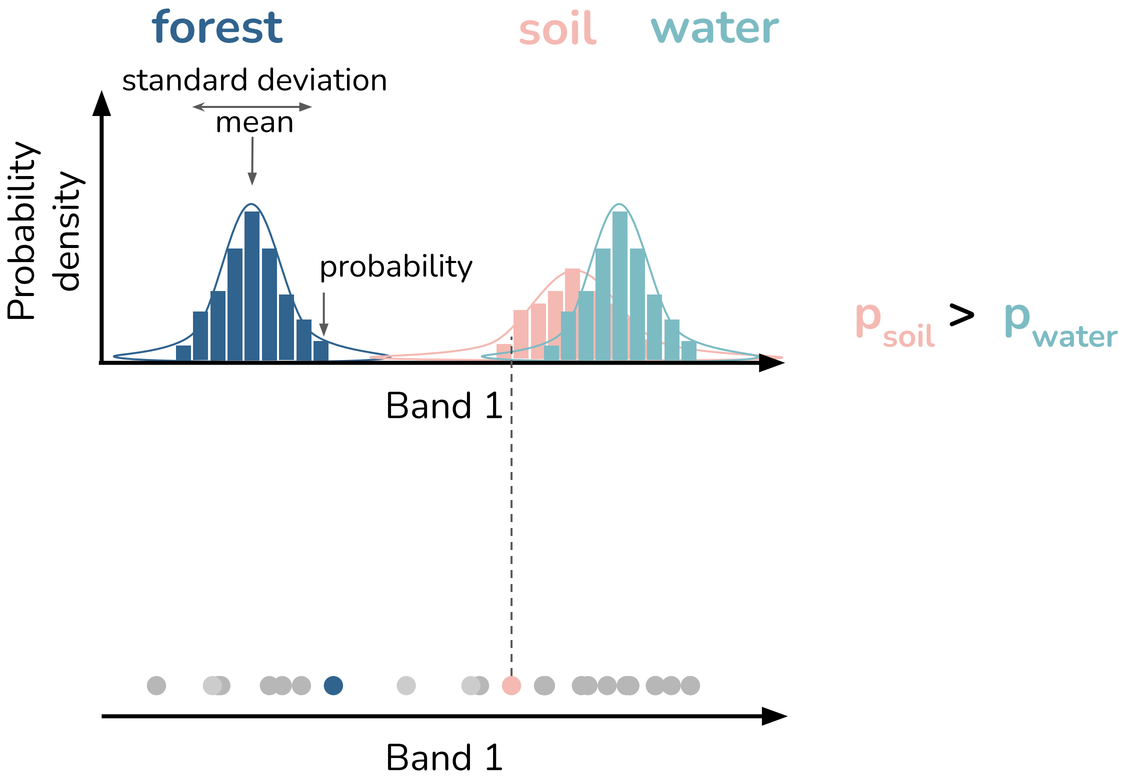
Known pixels:

Unknown pixels:



Supervised classification

Known pixels:

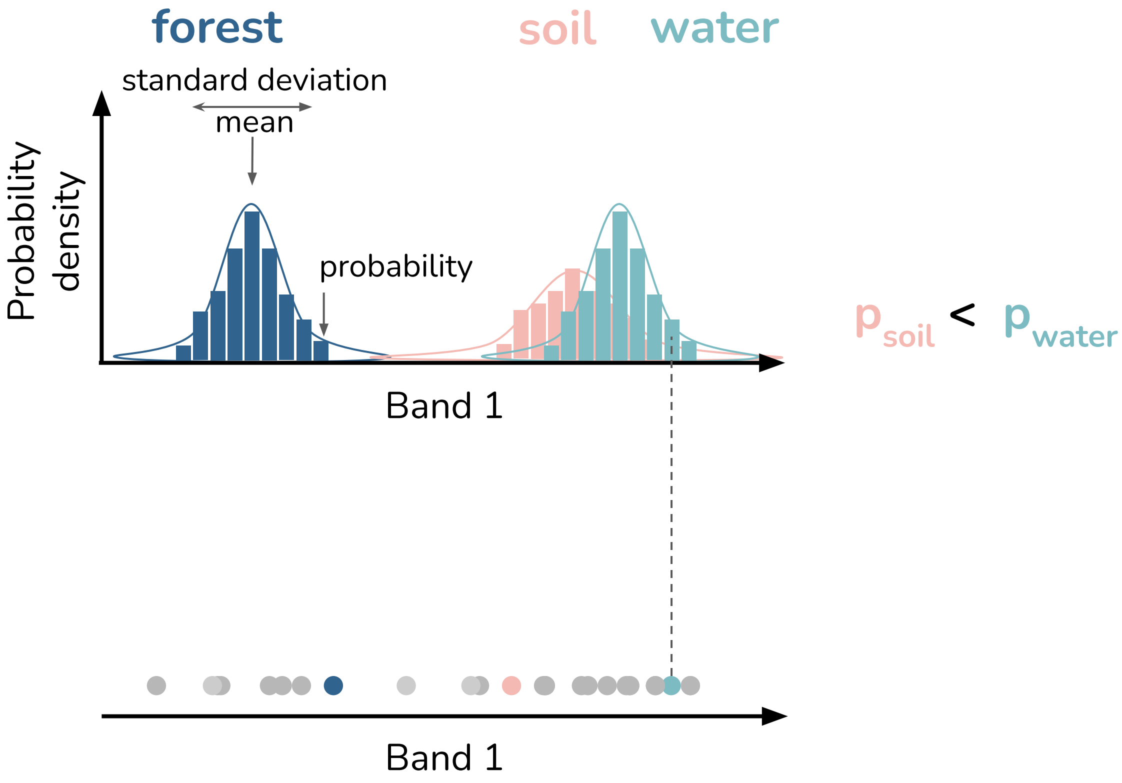


Unknown pixels:

Supervised classification

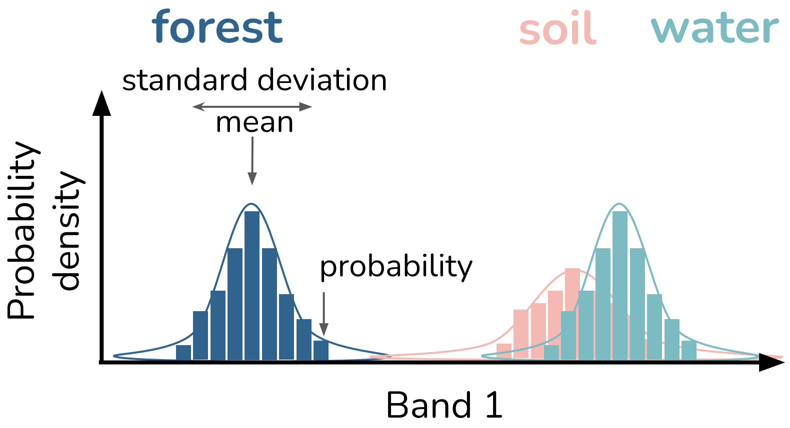
Known pixels:

Unknown pixels:

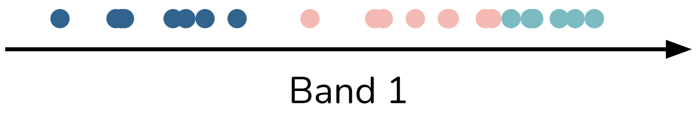


Maximum likelihood

Known pixels:

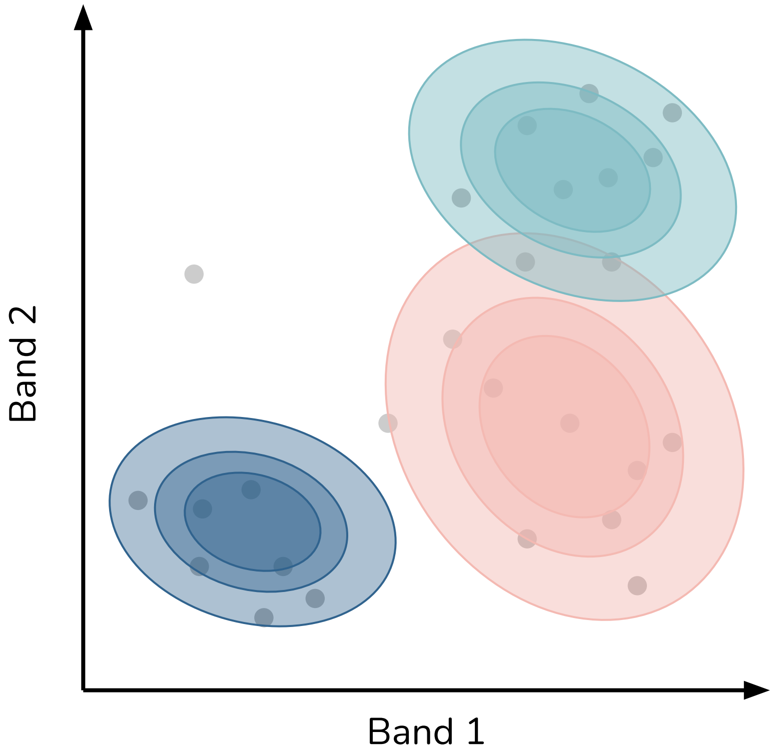


Unknown pixels:

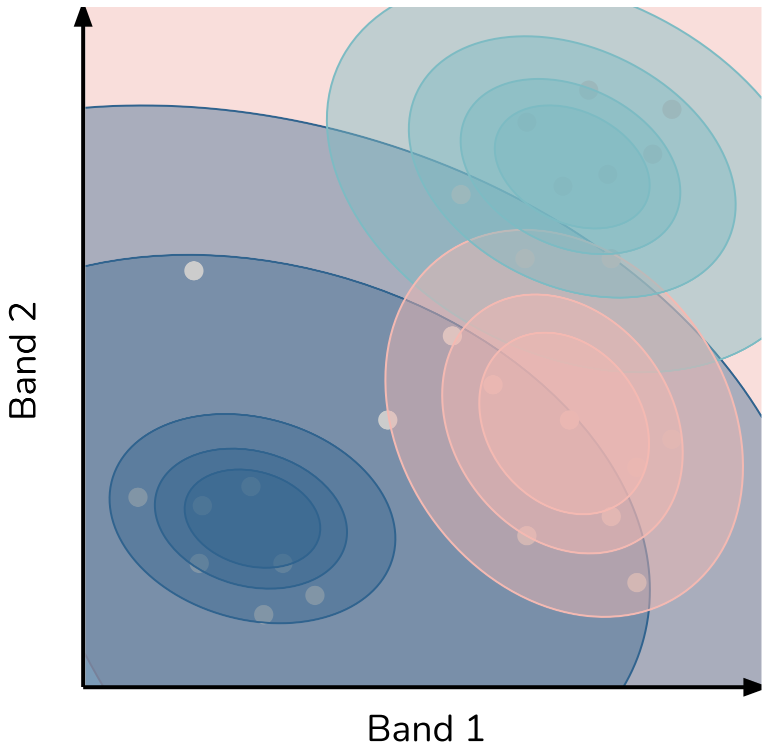


maximum likelihood

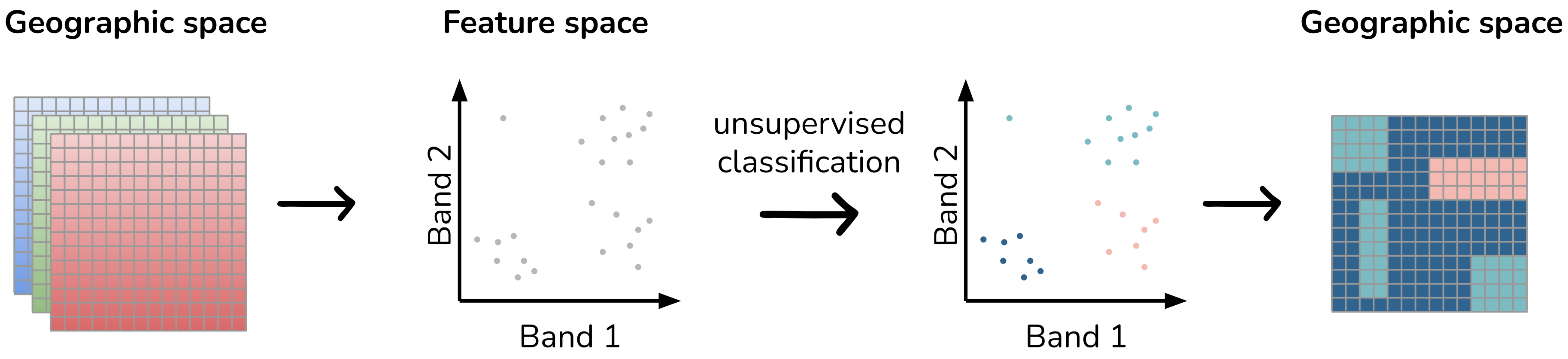
Maximum likelihood



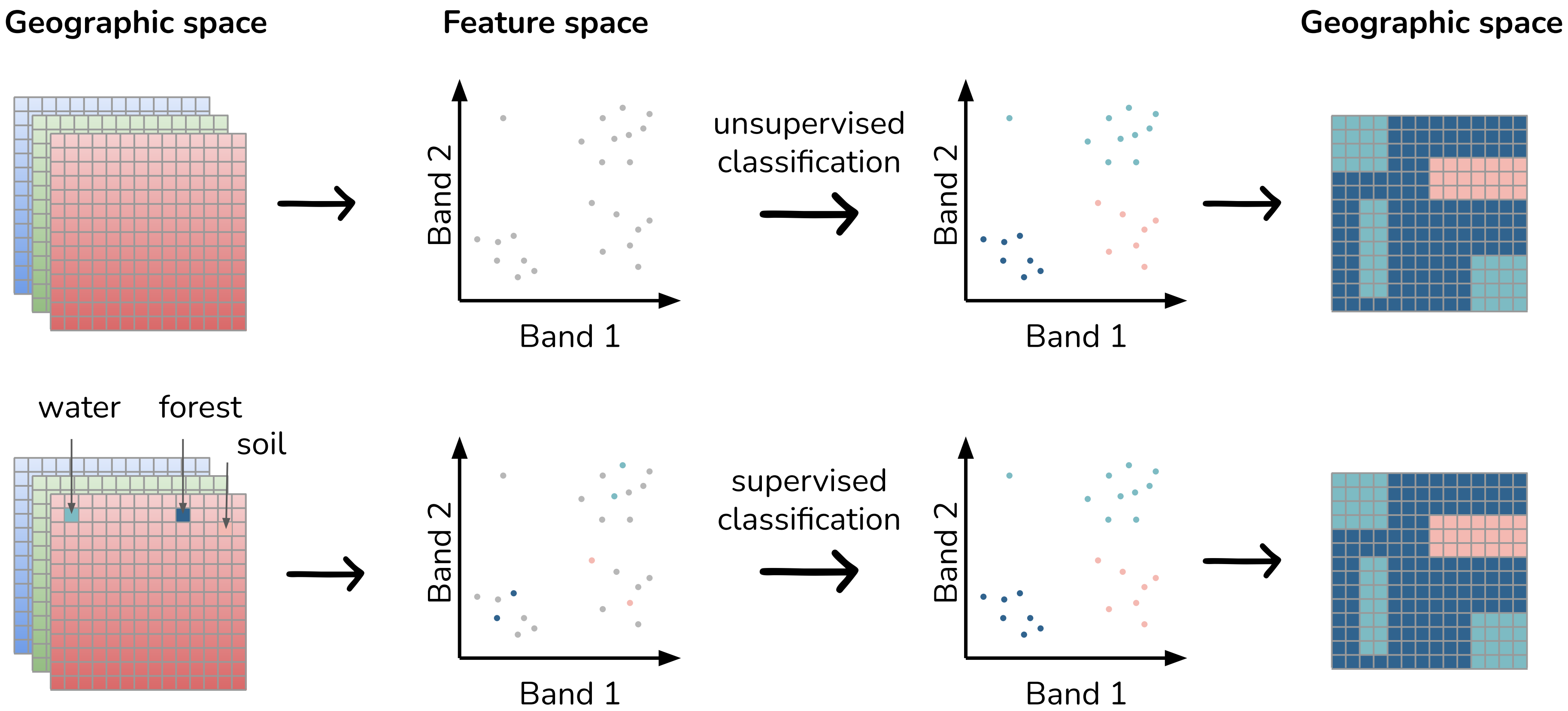
Maximum likelihood



(un)supervised classification



(un)supervised classification



Classification approaches

Unsupervised

- Algorithm identifies groups of pixels with similar spectra

Supervised

- Algorithm identifies groups of pixels with similar spectra

Classification approaches

Unsupervised

- Algorithm identifies groups of pixels with similar spectra
- User assigns meaning to resulting classes

Supervised

- Algorithm identifies groups of pixels with similar spectra
- User provides examples for desired groupings

Classification approaches

Unsupervised

- Algorithm identifies groups of pixels with similar spectra
- User assigns meaning to resulting classes
- Bulk of analyst’s work comes after the classification process

Supervised

- Algorithm identifies groups of pixels with similar spectra
- User provides examples for desired groupings
- **Bulk of analyst’s work comes before the classification process**

Classification approaches

Unsupervised

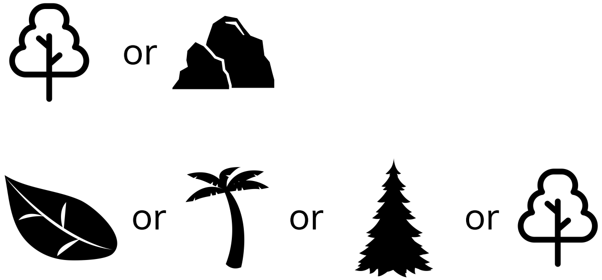
- Algorithm identifies groups of pixels with similar spectra
- User assigns meaning to resulting classes
- Bulk of analyst's work comes after the classification process
- Pros:
    - No prior knowledge of area required
    - Human error is minimized
    - Relatively fast/easy
    - Unique spectral classes are produced
- Cons:
    - Spectral classes may not represent features on the ground
    - Does not consider spatial relationships
    - Can be time-consuming to interpret
    - Spectral properties may vary over time/images

Supervised

- Algorithm identifies groups of pixels with similar spectra
- User provides examples for desired groupings
- **Bulk of analyst's work comes before the classification process**
- Pros:
    - Spectral classes represent features on the ground
    - Training areas are reusable
- Cons:
    - Information classes may not match spectral classes
    - Difficulty and coast of selecting training sites
    - Does not consider spatial relationships

Supervised classification: training data

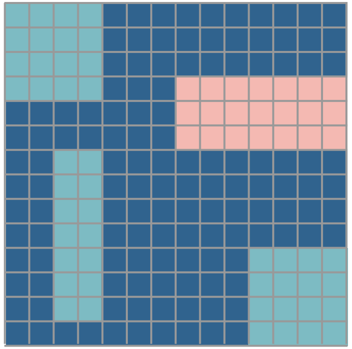
Classification scheme:



Does the resolution match your scheme? (spatial/temporal/spectral/radiometric)

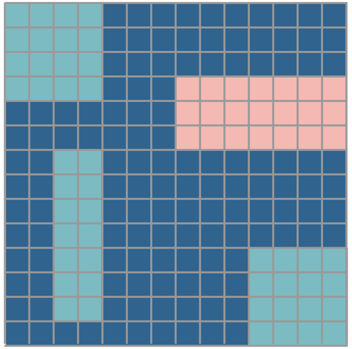
Does your training data capture the heterogeneity of each class?

Testing how we did!



How accurate is this map?

Testing how we did!



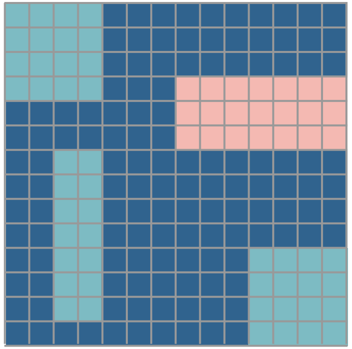
How accurate is this map?

"True answer"

Our guess based on remote sensing data

	forest	soil	water
forest			
soil			
water			

Testing how we did!



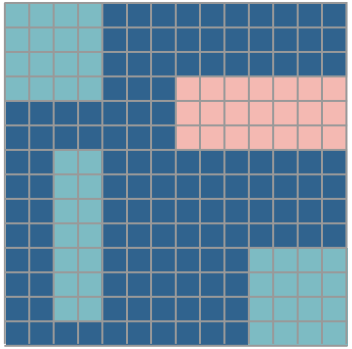
How accurate is this map?

"True answer"

Our guess based on remote sensing data

	forest	soil	water
forest	25	0	0
soil			
water			

Testing how we did!



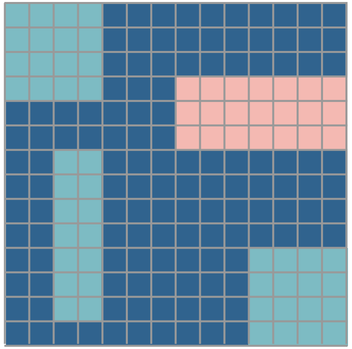
How accurate is this map?

"True answer"

Our guess based on remote sensing data

	forest	soil	water
forest	25	0	0
soil	0	12	0
water	0	0	18

Testing how we did!



How accurate is this map?

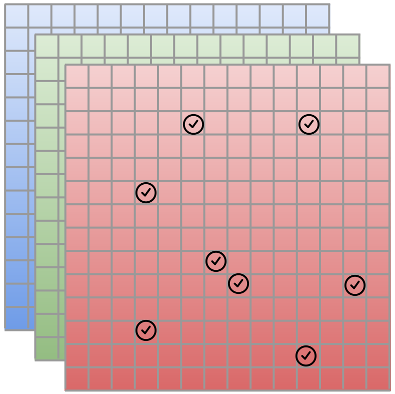
"True answer"

Our guess based on remote sensing data

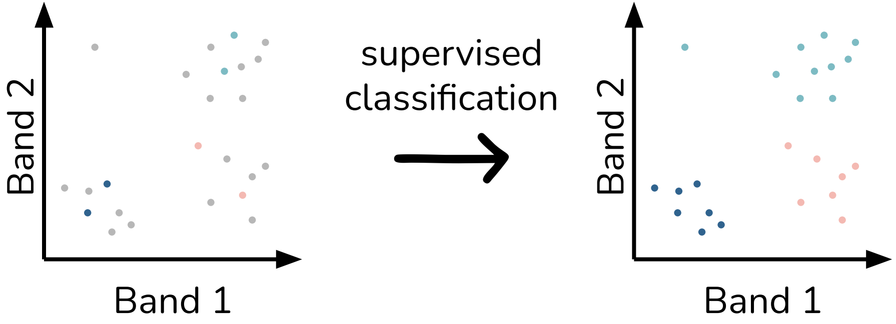
	forest	soil	water
forest	25	0	0
soil	0	12	0
water	0	0	18

Accuracy = sum of correct matches ÷ total number of cells

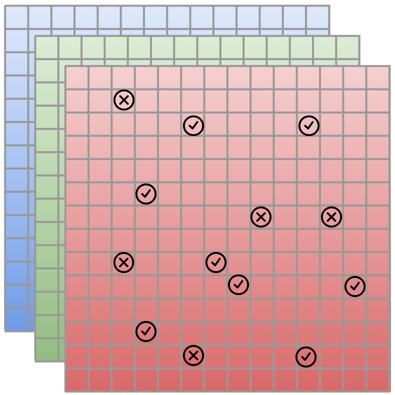
Testing how we did!



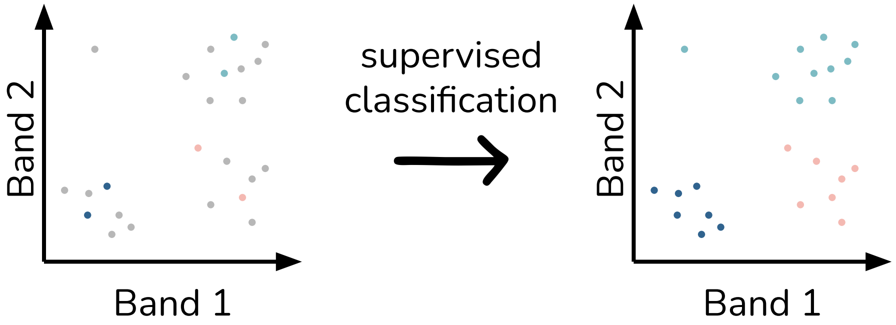
training:



Testing how we did!



training:



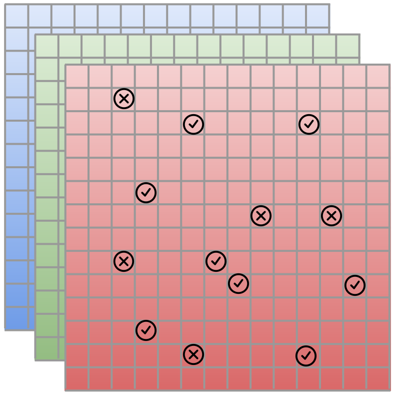
testing:

"True answer"

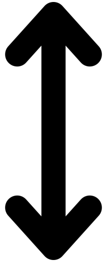
	forest	soil	water
forest			
soil			
water			

Our guess based on remote sensing data

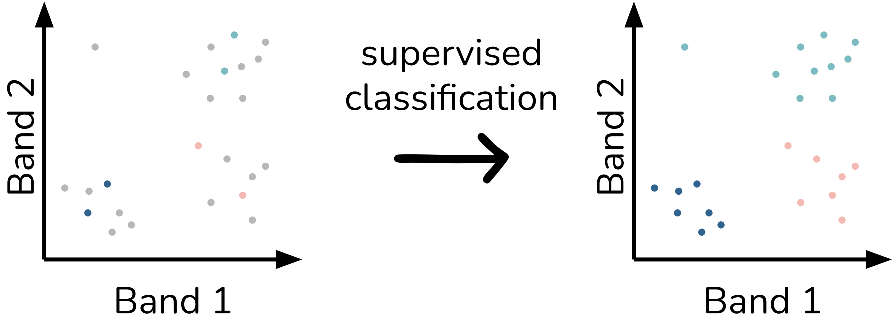
Testing how we did!



✓ training:



⊗ testing:



"True answer"

Our guess based on remote sensing data

	forest	soil	water
forest			
soil			
water			

Summary

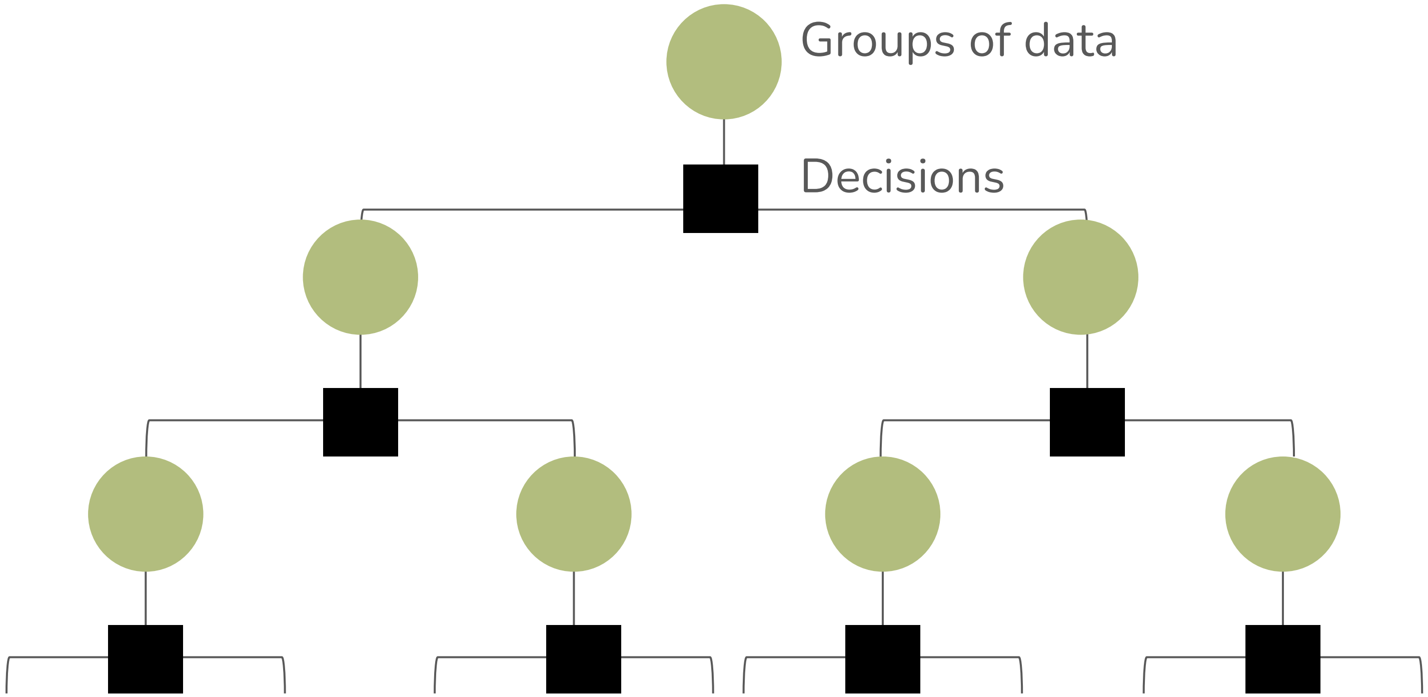
- **To classify objects, we try to separate them based on spectral features**
- **Lots of different ways to do this!**
  - Unsupervised approaches don't require any information upfront, but is hard to interpret
  - Supervised approaches are easier to interpret, but require information upfront

Today's classification task

Classify southern SB county into the following land covers:

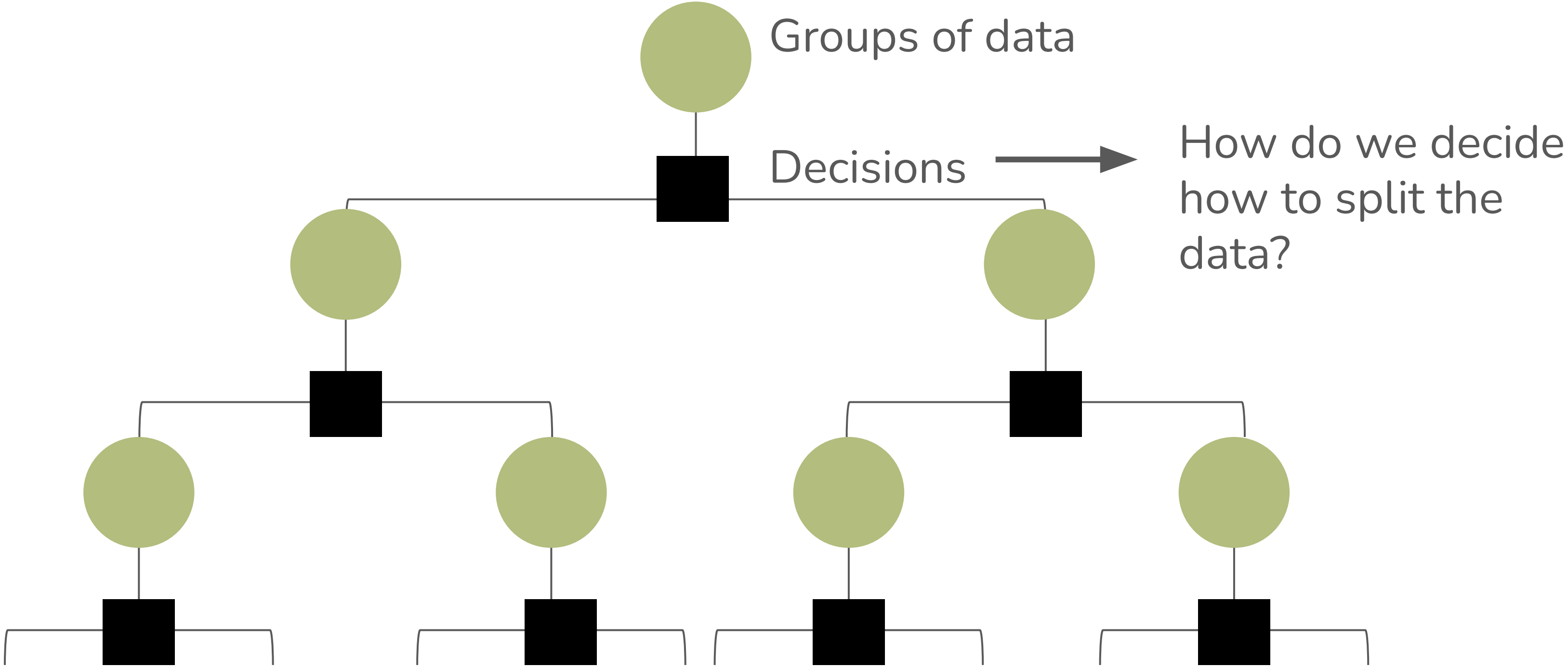
- Green vegetation
- Dry grass or soil
- Urban
- Water

Decision trees



Land cover classes!

Decision trees



Land cover classes!

Classification and Regression Trees (CART) algorithm

By minimizing the Gini Impurity:

$$G = 1 - \sum_{i=1}^{c} p_i^2$$

c is the number of classes
$\boldsymbol{p_i}$ is the probability of a randomly chosen element in the node being labeled as class ***i***

Classification and Regression Trees (CART) algorithm

Consider the example, where we would like to predict who will purchase a product.

Age	Income	Buy?
30	20,000	Yes
40	50,000	Yes
20	30,000	No
50	60,000	No
60	80,000	Yes

We start by splitting our data into two groups based on an arbitrary threshold.

Age	Income	
30	20,000	Yes
20	30,000	No

Age	Income	
40	50,000	Yes
50	60,000	No
60	80,000	Yes

We then compute the Gini Impurity for each subset of data.

Age	Income	
30	20,000	Yes
20	30,000	No

Age	Income	
40	50,000	Yes
50	60,000	No
60	80,000	Yes

$$G_{left} = 1 - \left(\tfrac{1}{2}\right)^2 - \left(\tfrac{1}{2}\right)^2 = 0.5$$

$$G_{right} = 1 - \left(\tfrac{2}{3}\right)^2 - \left(\tfrac{1}{3}\right)^2 \approx 0.444$$

... and take the mean to find the Gini Impurity for the split

$$G_{split} = (\tfrac{2}{5})G_{left} + (\tfrac{3}{5})G_{right} \approx 0.48$$

Age	Income	
30	20,000	Yes
20	30,000	No

Age	Income	
40	50,000	Yes
50	60,000	No
60	80,000	Yes

$$G_{left} = 1 - (\tfrac{1}{2})^2 - (\tfrac{1}{2})^2 = 0.5$$

$$G_{right} = 1 - (\tfrac{2}{3})^2 - (\tfrac{1}{3})^2 \approx 0.444$$

We then try all other splits, compare scores, and pick the split that minimizes the Gini Impurity.