UNIT 13

UNSUPERVISED CLASSIFICATION

Structure

13.1 INTRODUCTION

In Block-2 of this course you have learnt about various pre-classification techniques. This unit and the next one deal with a very key aspect of digital image processing and thematic information extraction known as image classification. As you have read about image classification in the course MGY-102, you know that it involves conversion of raster data into finite set of classes that represent surface types i.e. a class in the theme of your interest in the imagery. This fundamental task involves either training a model to categorise images into predefined classes (i.e. s*upervised classification*) or it is achieved by using algorithms that autonomously identify patterns and similarities within an image dataset without explicit class labels (i.e. *unsupervised classification*). The goal of this process is to enable machines to recognise and interpret visual content, mimicking

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

human visual perception. In this unit, you will learn more about image classification with focus on unsupervised classification, starting with a brief introduction to image classification, its typology based of various factors, and followed by various unsupervised approaches commonly used for thematic information extraction from remote sensing data. You will study about supervised classification in the next unit.

Expected Learning Outcomes

After studying this unit, you should be able to:

- ❖ define image classification;
- ❖ discuss various types of image classification approaches;
- ❖ differentiate between various types of classifications such as pixel and object based, hard and soft, parametric and non-parametric, image scanning and feature space, single pass and iterative, etc.;
- ❖ identify broad steps in image classification in general, and unsupervised classification in particular;
- ❖ describe the commonly used approaches of unsupervised image classification; and
- ❖ write about some other approaches useful for unsupervised classification.

13.2 IMAGE CLASSIFICATION

You have learnt that image classification is a process through which pixels in the image is grouped into various classes/objects based on their spectral signatures or reflectance properties. It may be noted that image classification is used for several societal applications namely, land use/land cover analysis, agriculture, urban planning, natural resource management, surveillance, updating geographic maps and also for disaster mitigation.

Image classification is defined as a process of assigning land cover classes/themes to pixels in an image (Lillesand and Keifer, 1994). Some of the classes comprise built-up area, urban, forest, grassland, agriculture, water, shadow, rocky areas, bare soil and cloud. Image classification usually represents object of the analysis and generates a map-like image in the form of final product/output. It is an important tool for studying digital images. There are several image classification methods and terminologies used such as, supervised, unsupervised, per-pixel, object-based, hard, soft, parametric, non-parametric, spectral, contextual, etc.

Let us recollect the broad approaches of image classification first.

13.2.1 Approaches

After following image correction, enhancement and transformation stages, image classification begins with both or either of the following two general approaches:

- **Unsupervised Classification**: It is the process of automatic identification of natural groups or structures within a remotely sensed image, and

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

- **Supervised Classification**: It is the process of identification of classes within a remotely sensed image with inputs from and as directed by the user in the form of training data.

Both the classification approaches use spectral signatures of pixels and ground information of the area of the study for assigning each pixel a typical land cover type or providing training samples for that. The unsupervised and supervised image classifications differ from each other in the way the classification is performed. For example, spectral and/or information alone is used in case of unsupervised classification and ground truth is also required for training the supervised classifier and testing the final result.

Some classifiers that combine both labelled and unlabelled data to train the classifier thereby reducing the need for extensive labelled training samples can be categorised into semi-supervised classifiers. Such classifiers are useful when labelled data is scarce but large amounts of unlabelled data are available. This type of classification is useful for the images of remote or inaccessible areas where finding training sites is challenging. Advantage of such classification is that it balances between supervised and unsupervised approaches and reduces dependency on large training datasets. However, it requires a good balance between labelled and unlabelled data and its performance depends on the quality of initial labelled data.

Let us study the stages involved in image classification.

13.2.2 Stages

You have read that the image classification process broadly consists of following three stages: training, signature evaluation and decision making.

Training is the process of generating spectral signature of each class. Training can be carried out either by an image analyst with guidance from his experience or knowledge in case of supervised training or by some statistical clustering techniques requiring little input from image analysts in case of unsupervised training. The training data selection is an important task and it must be ensured that the location of each training sample and its thematic class are correctly recorded, else the classification result can be erroneous.

Signature Evaluation is the checking of spectral signatures for their representativeness of the class they attempt to describe and also to ensure as small spectral overlap between signatures of different classes as possible.

Decision Making is the process of assigning all the image pixels into thematic classes using evaluated signatures which is achieved using algorithms, known as decision rules that set certain criteria. When signature of a candidate pixel passes the criteria set for a particular class, it is assigned to that class.

You may note here that the training and signature evaluation steps are essential steps in supervised classification whereas in the case of unsupervised classification, training and signature evaluation is not involved and the focus is on assigning a thematic class to the classes generated by the computer through minimum input by the human analyst.

Let us now study typologies of image classification.

13.3 TYPES OF CLASSIFICATION

There are several types of image classification such as hard and soft classifiers, per-pixel and object-based classification, parametric and non-parametric classification, as well as statistical, ensemble, and machine learning classification. Some broad typology of the classification is given in the Table 13.1.

Table 13.1: Typology of image classification (modified from Lu and Weng, 2007).

Basis of Classification	Type	Characteristics	Suitability	Advantage	Limitation
Use of training samples	Unsupervised	Image is partitioned into spectral classes based on statistical information using clustering algorithms; analyst merges and labels the spectral classes into information classes	When field information is not available, used for exploratory analysis	No training data required, automatic grouping of pixels	Labelling by analyst required after classification; sensitive to parameters
	SuperviSed	Classifier classifies spectral data into Information classes based on available field/ reference data through the signatures generated from the training samples used	When sufficient field or reference data is available	High accuracy, analyst has control	Labour-intensive; classification dependent on training data
Output type or assignment of class membership to each pixel	Hard / crisp	Assigns each pixel or object to a single, discrete class	Binary decision boundaries; crisp classification	Simple, widely used, clear class boundaries	Does not handle mixed pixels or uncertainty well
	Soft / fuzzy	Provides probabilistic or fuzzy outputs, assigning pixels to multiple classes with varying probabilities	Handles uncertainty; provides membership scores; more realistic outputs	Better handles mixed pixels, captures uncertainty, more information -rich output	Computationally complex, difficult to interpret for some applications
Nature of pixel information used	Per-Pixel	Classifies each pixel individually based on its spectral properties without considering spatial context	Simple, pixel-centric approach, uses spectral values	Easy to implement, fast processing, well-suited for homogeneous areas	Ignores spatial information, prone to salt-and-pepper noise
	Sub-pixel	The spectral value of each pixel is assumed to be a linear or non-linear combination of defined pure materials (or end members), providing proportional membership of each pixel to each end member	Useful for mixed-pixel areas, high-resolution, or fractional mapping of land covers	Handles mixed pixels; provides detailed class proportions	Computationally intensive; requires complex modelling; difficult to interpret.
	Object-oriented	Segments image into meaningful objects and classifies them based on spectral, shape, and contextual data; classification is conducted based on the objects, instead of an individual pixel. No vector data is used	High-resolution data, complex landscapes with clear object boundaries	Uses spatial context; reduces noise; accurate in heterogeneous landscapes	Computationally demanding; sensitive to segmentation quality; requires parameter tuning.

Basis of Classification	Type	Characteristics	Suitability	Advantage	Limitation
Usage of parameters such as mean vector and covariance matrix	Parametric	Assumes that data follows a known statistical distribution (e.g., Gaussian); parameters (e.g. mean vector and covariance matrix) are often generated from training samples	Requires statistical assumptions about data	Computationally efficient, interpretable, well-established methods	Assumptions may not hold true; poor performance with non-normal data; difficult to integrate ancillary data, spatial and contextual attributes, and non-statistical information into a classification procedure; often produces 'noisy' results in complex landscape
	Non-Parametric	Does not assume any specific statistical distribution of data	Flexible, no assumptions about data distribution.	Handles complex and non-linear data, adaptable to various data types	Computationally intensive, requires large training data
Usage of spectral and spatial information	Spectral	Pure spectral information is used in image classification. A 'noisy' classification result is often produced due to the high variation in the spatial distribution of the same class. Maximum likelihood, minimum distance, artificial neural network	Images with high spectral resolution and clear class separations	Fast and easy to implement; well-suited for basic classification tasks	Ignores spatial context; may struggle with spectrally similar classes
	Contextual	The spatially neighbouring pixel information is used in image classification	High spatial resolution imagery, heterogeneous landscapes, urban and agricultural monitoring	Improved accuracy, noise reduction, enhanced object recognition, better boundary delineation, handles mixed pixels	Computationally complex, parameter-sensitive, requires high-quality data, interpretation challenges, struggles with fragmented landscapes
	Spectral contextual (/spectral-spatial)	Spectral and spatial information is used in Classification; parametric or non-parametric classifiers are used to generate initial classification images and then contextual classifiers are implemented in the classified images	High-resolution imagery where spatial features provide valuable information	Improves classification accuracy; reduces misclassification of spectrally similar classes; preserves spatial structure	Computationally intensive; requires careful feature selection
Assignment of label over number of iteration	Single-pass	Assigns labels in one pass through the data	Suitable for quick and simple classification tasks	Fast processing due to single pass	Generally lower accuracy due to lack of refinement
	Iterative	Refines labels over multiple iterations	Suitable for tasks requiring high accuracy and refinement	Improved accuracy through iterative refinement	Requires more processing time and computational resources

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

Basis of Classification	Type	Characteristics	Suitability	Advantage	Limitation
Usage of raw spectral values or transformed values	Image scanning	Scans and classifies the image pixel by pixel	Suitable for detecting local patterns and anomalies	High spatial resolution and detail	Computationally intensive and time-consuming
	Feature space	Classifies based on features extracted from the image	Suitable for capturing global patterns and relationships	Can handle complex and multidimensional data	Requires feature extraction and may lose spatial resolution

So, you have learnt about different types of classification. Let us now discuss some of these types of classifiers in some detail.

Let us first learn about the unsupervised and supervised classification.

13.3.1 Unsupervised and Supervised Classification

You have already learnt about the difference between unsupervised and supervised classification in the course MGY-101 but let us recollect them here in table 13.2

Table 13.2: Comparison of unsupervised and supervised classification.

Aspect	Unsupervised Classification	Supervised Classification
Description	Automatic classification by algorithm by identifying natural groupings based on spectral similarities without prior knowledge or labelled data; classes are determined after classification by interpreting the clusters formed by the algorithm	Classification by algorithm based on user labelled training data, which involves selecting representative samples for each class manually; classes are predefined before classification based on user knowledge and training data
User involvement	Minimal; just to set the number of clusters	High; to define the classes, select training samples, and validate the classification output
Suitability	Suitable for exploratory analysis, areas with unknown land cover types, or when no training data is available	Best for applications where predefined classes are known, such as agricultural mapping, urban planning, and environmental monitoring
Advantage	Requires no prior knowledge or training data; automatic grouping of pixels	High accuracy; analyst has control
Limitation	Labelling by analyst required after classification; sensitive to parameters	Labour-intensive; classification dependent on training data, which involves selecting representative samples for each class manually
Output	Produces clusters that need to be interpreted and labelled by the user	Produces classified maps with clear class labels as defined by the training data

13.3.2 Single-Pass and Iterative Classification

Let us now learn the difference between single-pass and iterative classifications. Table 13.3 compares both types of classification.

Table 13.3: Comparison of single-pass and iterative classification.

Aspect	Single-pass Classification	Iterative Classification
Description	A classification method that assigns labels in one pass through the data	A classification method that refines labels over multiple iterations
Accuracy	Generally lower accuracy due to lack of refinement	Higher accuracy due to iterative refinement
Processing time	Faster processing time as it involves only one pass	Slower processing time due to multiple iterations
Complexity	Simpler and easier to implement	More complex due to iterative process
Resource usage	Lower resource usage	Higher resource usage due to repeated computations
Convergence	May not converge to an optimal solution	Can improve convergence towards an optimal solution
Adaptability	Less adaptable to changes in input data	More adaptable to changes in input data
Suitability	Suitable for smaller datasets	Better suited for larger datasets with complex relationships
Example	K-means clustering	ISODATA (Iterative Self-Organizing Data Analysis Technique)
Advantage	Fast processing due to single pass; easy to implement an understand; requires fewer computational resources; suitable for quick and simple classification tasks	Improved accuracy through refinement; can handle complex relationships better; can adapt to changes in input data; suitable for tasks requiring high accuracy and refinement
Limitation	Lower accuracy due to lack of iterative refinement; less adaptable to changes in input; may not converge to the best solution data	Requires more processing time and computational resources; more complex to implement and understand; higher resource usage

13.3.3 Hard/Crisp and Soft/Fuzzy Classification

Let us now learn the difference between hard and soft classifications. Table 13.4 compares both types of classification.

Table 13.4: Comparison of hard/crisp and soft/fuzzy classification.

Aspect	Hard/Crisp Classification	Soft/Fuzzy Classification
Description	Assigns a single class to each pixel	Assigns pixels to more than one class with probabilities
Class boundaries	Clear, distinct, crisp boundaries	Gradual, overlapping, and fuzzy boundaries
Suitability	Clear-cut land covers, distinct classes e.g. urban-rural mapping, forest types	Complex landscapes, mixed or transitional areas, e.g. agricultural fields, urban areas with mixed pixels
Advantage	Simple, clear results, computationally efficient	Handles mixed pixels, reflects real-world complexity
Limitation	Misclassification of mixed pixels; overly simplified; cannot handle uncertainty	Complex interpretation; computationally demanding; requires additional analysis to convert to hard classes, if needed

13.3.4 Pixel and Object based Classification

Let us now learn the difference between pixel based and object based classifications. Table 13.5 compares both types of classification.

Table 13.5: Comparison of pixel based and object based classification.

Aspect	Pixel-Based Classification	Object-Based Classification (OBIA)
Description	Classifies individual pixels based on their spectral information alone; treats each pixel as an independent unit without considering spatial context	Classifies groups of pixels (objects) that are segmented based on spectral and spatial (shape, texture, size) characteristics; segments the image into meaningful objects or regions before classification
Classification unit	Each pixel is classified independently	Objects (groups of pixels) are classified as a single unit
Suitability	Suitable for coarse to moderate resolution imagery where pixel homogeneity is high	Ideal for high-resolution imagery where spatial detail is significant
Advantage	Simpler and faster to implement; directly uses spectral data, requiring less preprocessing; effective for homogeneous classes	Utilises spatial context, improving classification accuracy; reduces salt-and-pepper effect (noisy classifications); more natural representation of real-world features (e.g., roads, buildings)
Limitation	Prone to salt-and-pepper noise due to pixel-level classification; ignores spatial context, which can lead to misclassification of spectrally similar classes; not suitable for high-resolution imagery where spatial information is crucial	Computationally intensive due to segmentation and feature extraction; requires careful tuning of segmentation parameters; classification quality depends on the segmentation accuracy
Examples of algorithms	Maximum Likelihood, Minimum Distance, K-means clustering, Spectral Angle Mapper., etc.	Multi-resolution segmentation, GEOBIA, Region Growing, Classification and Regression Trees (CART)
Interpretation	Easier to interpret but less realistic for complex scenes	More realistic and easier to interpret in complex environments

While understanding about object based classification, it is also important to learn here about the object detection and understand the difference between object based image classification (OBIA) and object detection. These two are two different types of approaches, each serving specific purposes and utilising different methodologies. As you have read earlier in this unit, object based classification segments an image into meaningful objects (groups of pixels) and classifies them whereas object detection identifies and locates specific objects (e.g., buildings, vehicles) in an image.

The purpose of OBIA is to classify entire regions or segments of an image based on spectral, spatial, and contextual information whereas that of object detection is to detect and precisely locate individual objects of interest within an image. OBIA approach has high spatial context as it considers the relationship between neighbouring pixels and object whereas object detection has limited spatial context as it focuses mainly on the immediate surroundings of the object being detected. Output in OBIA is classified regions or objects with thematic labels (e.g., forest, water, urban) whereas that in object detection is either the bounding boxes, masks, or point locations indicating the presence and position of the objects of interest. OBIA uses both spectral information (colour, reflectance) and spatial features (size, shape, texture) of segments whereas object detection extracts features like edges, shapes, and textures specific to target objects (e.g., cars, buildings). You will learn more about OBIA in the course MGY-009.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

While the OBIA works at a broader scale, classifying objects as regions rather than identifying individual items, object detection approach aims to classify segmented groups of pixels into thematic classes. It is focused on high level of detail such as locating specific, discrete objects within an image. Both the approaches have their unique strengths, making them suitable for different kinds of remote sensing tasks.

13.3.5 Parametric and Non-parametric Classification

Let us now learn the difference between parametric and non-parametric classifications.

Parametric and non-parametric classifications are two major approaches used in remote sensing for classifying image data. The primary difference between them lies in the assumptions they make about the data distribution and the methods they use to classify pixels into different land cover classes. Both the types of classification are compared in Table 13.6.

Table 13.6: Comparison of parametric and non-parametric classifications.

Aspect	Parametric Classification	Non-Parametric Classification
Training Data requirements	Requires less training data; assumptions about data distribution reduce data dependency	Requires more training data to effectively model class boundaries without distribution assumptions
Complexity	Simpler, relies on statistical parameters such as mean and covariance.	More complex; relies on flexible algorithms that can model complex relationships
Flexibility	Less flexible; performance drops if data deviates from assumed distribution	Highly flexible; can adapt to various data types and complex class boundaries
Suitability	Best for simple, well-separated classes with normally distributed data; effective when data fits statistical assumptions; e.g. mapping vegetation types, water bodies, or other classes with distinct spectral signatures and normal distributions	Suitable for complex, high-dimensional data, non-Gaussian distributions, and classes with overlapping features; e.g. complex urban classifications, heterogeneous landscapes, and cases where class boundaries are not clear-cut
Handling of mixed pixels	Limited ability to handle mixed pixels due to rigid boundaries defined by statistical parameters	Better at handling mixed pixels and ambiguous class boundaries; can model complex decision surfaces
Interpretability	Easier to interpret since it uses statistical rules; decisions based on defined parameters	Often considered a "black-box," making interpretation of decision-making processes more difficult
Advantage	Simple implementation; fast computation; effective when assumptions hold	Handles complex, non-linear relationships; no assumptions about data distribution; high accuracy
Limitation	Assumes data follows specific distributions; performance degrades with non-Gaussian data; not suitable for complex or overlapping classes	Computationally demanding; requires a lot of training data; less transparent decision-making process

13.3.6 Image Scanning and Feature Space Classification

Let us now learn the difference between image scanning and feature space classifications.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

Image scanning and feature space classifications are two major approaches used in remote sensing for classifying image data. Image scanning classification directly classifies pixels based on their spectral values in the original image without transforming the data. Each pixel is classified individually based on its spectral signature, often using simple thresholds or clustering algorithms. Feature space classification transforms the raw pixel data into a multi-dimensional feature space, where each dimension represents a different attribute. It classifies pixels based on their feature vectors in the transformed feature space, using more sophisticated algorithms that consider multiple attributes. Both the types of classification are compared in Table 13.7.

Table 13.7: Comparison of parametric and non-parametric classifications.

Aspect	Image scanning Classification	Feature space Classification
Description	Involves directly classifying the raw pixel data without transforming it into a feature space. This method typically operates on the pixel values from the original image; it scans and classifies the image pixel by pixel	Classifies based on features extracted from the image; involves transforming the raw pixel data into a multi-dimensional feature space where each dimension represents a different attribute (e.g., spectral bands, texture, shape). In this space, classifiers can efficiently distinguish between different classes based on their feature vectors
Characteristics	High spatial resolution; direct pixel-level classification	Utilises multidimensional feature space for classification
Accuracy	Can be lower due to mixed pixels	Generally higher due to feature extraction
Processing Time	Faster for smaller images	Slower due to feature extraction
Complexity	Lower complexity	Higher complexity
Resource Usage	Less resource-intensive	More resource-intensive
Convergence	Faster convergence	Slower convergence
Adaptability	Less adaptable to new data	More adaptable to new data
Suitability	Suitable for simple tasks	Suitable for complex tasks
Handling of mixed pixels	Struggles with mixed pixels	Better handling of mixed pixels
Interpretability	Easier to interpret	Harder to interpret
Example	Traditional pixel-based classification	Feature space iterative clustering; classification using Principal Component Analysis (PCA) features; SVM, Random Forests
Advantage	Simple and quick; high spatial detail and resolution;	High accuracy and adaptability; can handle complex and multidimensional data
Limitation	Computationally intensive; lower accuracy with complex data; may miss global patterns	Requires feature extraction; may lose spatial resolution; requires more computational resources

In the following sections, we will discuss the steps and commonly used approaches of unsupervised classification.

13.4 STEPS IN UNSUPERVISED CLASSIFICATION

You have learnt earlier that as its name implies, this form of classification is done without interpretive guidance from an analyst. Unsupervised image classification is a fundamental approach in remote sensing data analysis that involves automatically grouping pixels in an image into clusters or classes without prior knowledge of the classes' characteristics. Unlike supervised classification, where labelled training data is required, unsupervised classification relies purely on the intrinsic spectral properties of the image data. This method is particularly useful when labelled data is scarce, expensive, or unavailable, making it a popular choice for exploratory data analysis in remote sensing. An algorithm automatically organises similar pixel values into groups that become the basis for different classes. This is entirely based on the statistics of the image data distribution, and is often called *clustering*.

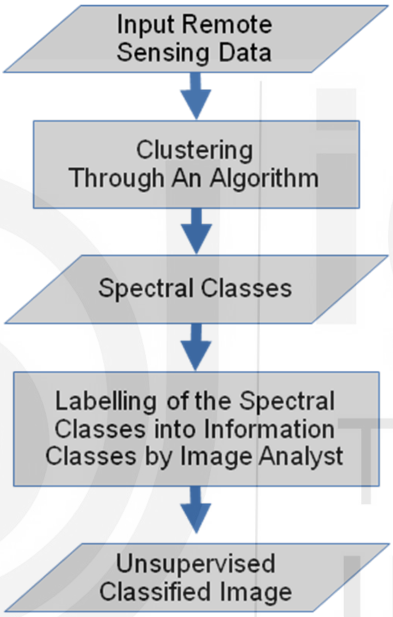

Fig. 13.1: Generalised steps in unsupervised classification.

Unsupervised image classification groups pixels into clusters based on their spectral similarities using algorithms that identify natural groupings within the data. Each cluster represents a distinct land cover type, though the exact nature of each class (e.g., forest, water, urban) is determined through post-classification labelling. The process is automatically optimised according to cluster statistics without the use of any knowledge-based control (i.e. ground referenced data). The method is, therefore, objective and entirely data driven. It is particularly suited to images of targets or areas where there is no ground knowledge. Even for a well-mapped area, unsupervised classification may reveal some spectral features which were not apparent beforehand. The basic generalised steps of unsupervised classification are shown in Fig. 13.1.

The result of an unsupervised classification is an image of statistical clusters, where the classified image still needs interpretation based on knowledge of thematic contents of the clusters. There are hundreds of clustering algorithms available for unsupervised classification and their use varies by the efficiency and purpose. K-means and ISODATA are the two widely used algorithms which are discussed here.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

13.5 COMMONLY USED APPROACHES

In contrast to conventional supervised methods, unsupervised image classification does not require labelled training data, making it an intriguing area in computer vision. This approach does not require explicit class labelling; instead, algorithms recognise patterns and similarities on their own inside an image dataset. One prominent technique in unsupervised image classification is clustering, where the algorithm groups similar pixels or regions together based on inherent features (Fig. 13.2).

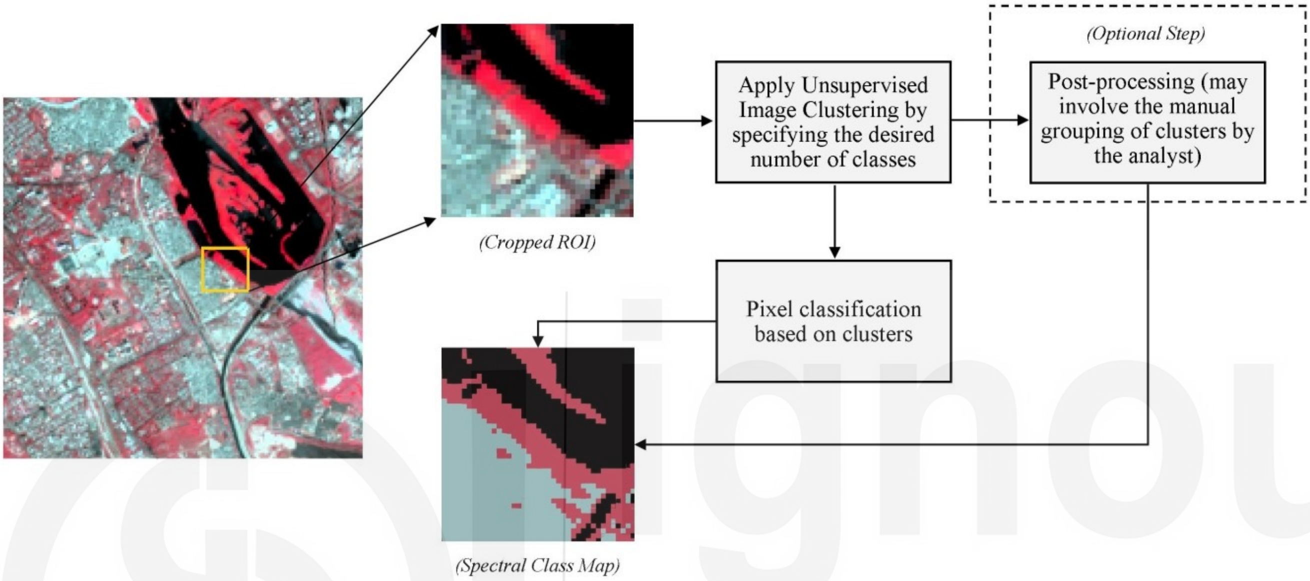

Fig. 13.2: Steps in a typical unsupervised classification. In this diagram a FCC of Delhi-NCR region created using Landsat 8 OLI images (Band 5: Band 3: Band 2 = R:G:B) is undergone classification using a K-means clustering algorithm (*k*=3).

Clustering algorithms, such as K-means or hierarchical clustering, enable machines to uncover hidden structures within the data. By iteratively organising pixels into clusters, the algorithm discerns patterns that may represent distinct objects or textures. This approach is particularly valuable when dealing with vast amounts of unlabelled imagery, as it allows for exploratory analysis and pattern discovery without the manual annotation of training samples. It is particularly suited to images of targets or areas where there is no ground knowledge. Even for a well-mapped area, unsupervised classification may reveal some spectral features which were not apparent beforehand.

Unsupervised image classification finds applications in scenarios where obtaining labelled data is challenging or expensive. It aids in image segmentation, anomaly detection, and uncovering latent patterns in diverse datasets. As technology advances, unsupervised methods contribute to unlocking insights from unannotated visual information, pushing the boundaries of computer vision's capacity to understand and interpret complex visual data.

The result of an unsupervised classification is an image of statistical clusters, where the classified image still needs interpretation based on knowledge of thematic contents of the clusters. There are hundreds of clustering algorithms available for unsupervised classification and their use varies by the efficiency and purpose. k-means and ISODATA are the widely used algorithms which are discussed here basides hierarchical clustering, self organising maps and fuzzy C-means clustering.

13.5.1 k-means Clustering

You have already learnt about the k-means clustering in the course MGY-102. Let us recall it again here. You know that k-means is one of the most widely used unsupervised classification algorithms. It partitions the image into `k` clusters based on the nearest mean value of pixel intensities in a multi-dimensional spectral space. The K-means clustering algorithm is a popular unsupervised machine learning technique used for partitioning a dataset into *k* distinct, non-overlapping subsets or clusters. The goal is to group data points based on their spectral similarity, with *k* representing the predefined number of clusters.

Here is a step-by-step description of how the K-means algorithm works:

1. ***Initialisation***:
   a. Choose the number of clusters (*k*) that you want to identify in the dataset.
   b. Randomly select *k* data points from the dataset as the initial cluster centroids.
2. ***Assignment of data point to a cluster***:
   a. For each data point in the dataset, calculate the distance to each of the *k* centroids. Common distance metrics include Euclidean distance or Manhattan distance.
   b. Assign the data point to the cluster whose centroid is closest, forming *k* clusters.
3. ***Updating Centroids***:
   a. Recalculate the centroid of each cluster by taking the mean of all the data points assigned to that cluster.
   b. The new centroid becomes the representative point for that cluster.
4. ***Iteration***: Repeat the assignment and centroid update steps iteratively until convergence. Convergence occurs when the centroids no longer change significantly or when a predefined number of iterations is reached.
5. ***Result***: The final centroids and the assignments represent the *k* clusters in the dataset.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- The main advantage of this algorithm is its simplicity and speed which allows it to run on large datasets.

Limitation

- It does not yield the same result with each run, since the resulting clusters depend on the initial random assignments (Fig. 13.3).

- It is sensitive to outliers, so, for such datasets k-medians clustering is used.
- One of the main disadvantages to k-means is the fact that one must specify the number of clusters as an input to algorithm.

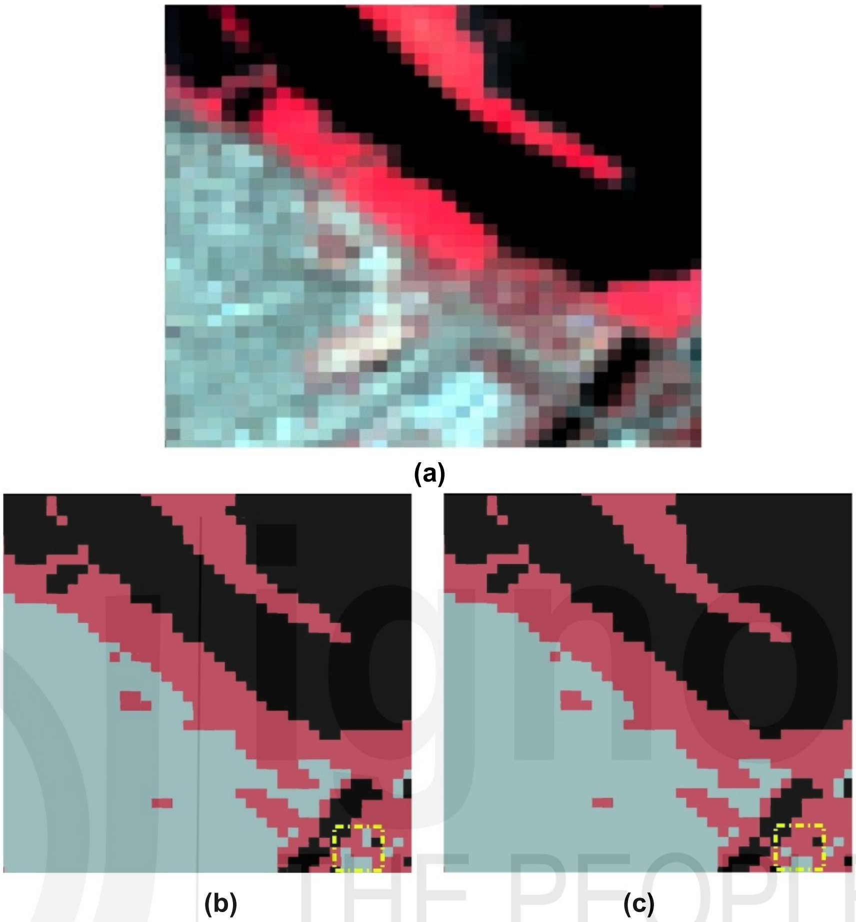

Fig. 13.3: On the left is the cropped ROI from FCC (identical to Fig 13.1) of Landsat 8 OLI image. The next two images are results of K-means performed on two separate tries. Inside the yellow box one can observe how the K-means provide different classification on different runs, despite having the same parameters.

13.5.2 ISODATA Clustering

The ISODATA (Iterative Self-Organising Data Analysis Technique) algorithm is an unsupervised clustering method designed to partition a dataset into a predefined number of clusters based on the data's inherent characteristics. Developed by Stuart Lloyd in 1982, ISODATA draws inspiration from the K-means clustering algorithm but introduces adaptive mechanisms to handle varying cluster shapes and sizes during the clustering process. The key components of ISODATA algorithm are as follows:

1. ***Initialisation**:* ISODATA starts by selecting a set of initial cluster centroids. This can be achieved through random selection or other techniques. The algorithm also requires initial values for parameters such as the minimum and maximum number of clusters, threshold values for cluster splitting and merging, and maximum iteration count.
2. ***Cluster Assignment**:* In the cluster assignment step, each data point is assigned to the cluster with the nearest centroid based on distance metrics

such as Euclidean distance. This process creates an initial clustering of the dataset.

3. ***Update Cluster Centroids***: After assigning data points to clusters, the algorithm computes new centroids for each cluster by taking the mean of the data points within that cluster.

4. ***Merge and Split Clusters***: ISODATA introduces dynamic cluster merging and splitting to adapt to changes in the dataset. If a cluster has too few data points (below a specified threshold), it may be split into two clusters. Conversely, if two clusters are deemed too similar (based on a similarity criterion), they may be merged.

5. ***Update Cluster Statistics***: The algorithm updates cluster statistics, including the mean and variance, after merging or splitting clusters.

6. ***Iterative Process***: Steps 2-5 are repeated iteratively until convergence or until a predetermined maximum number of iterations is reached.

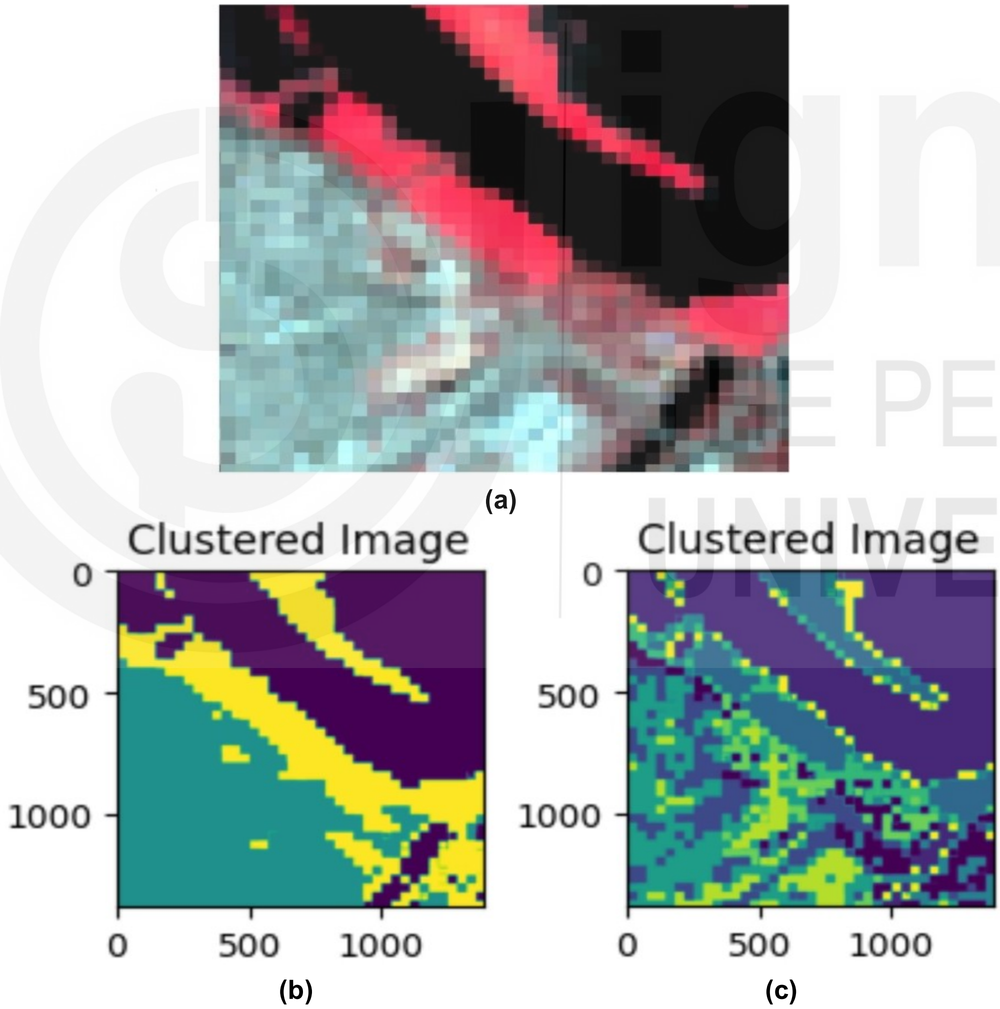

Fig. 13.4: On the top is the cropped ROI from the FCC (identical to Fig 13.2) of Landsat 8 OLI image (a). The two images on the lower panel are the results of ISODATA clustering performed with initial k clusters 3 (b) and 10 (c). It can be noted that despite specifying 10 clusters the final result was produced in 5 clusters as seen in (c), this shows the adaptive nature of ISODATA to cluster variability.

Let us now learn the advantages and limitations of this algorithm.

Advantage

1. ***Adaptive to Cluster Variability:*** ISODATA's ability to dynamically merge and split clusters makes it more adaptive to varying cluster shapes, sizes, and densities in the dataset (Fig. 13.3). This adaptability is particularly useful when dealing with complex and heterogeneous data.
2. ***Automatic Determination of Cluster Number:*** Unlike K-means, ISODATA does not require the user to specify the number of clusters beforehand. The algorithm can automatically adjust the number of clusters based on the characteristics of the data.
3. ***Handles Noisy Data:*** ISODATA is relatively robust to noise and outliers due to its iterative nature and cluster merging/splitting mechanisms. Outliers may be isolated into separate clusters during the process.

Limitation

- ***Sensitive to Initialisation:*** The performance of ISODATA can be sensitive to the initial choice of cluster centroids, and different initializations may lead to different results.
- ***Dependent on Parameters:*** The algorithm's effectiveness depends on appropriately setting parameters such as the minimum and maximum number of clusters, cluster splitting and merging thresholds, and convergence criteria.
- ***Computational Complexity:*** ISODATA can be computationally intensive, especially with large datasets, due to its iterative nature and dynamic cluster operations.

Let us spend 5 minutes to check your progress.

SAQ I

a) What are the types of image classification?

b) Write the generalised steps of k-means clustering.

c) What are the limitations of ISODATA clustering?

13.5.3 Hierarchical Clustering

Hierarchical clustering is a commonly used technique for unsupervised classification of remote sensing images. It groups pixels into clusters based on their attributes (i.e. spectral similarity) without prior knowledge of the class labels. Hierarchical clustering organises data into a tree-like structure called a *dendrogram*, which represents the nested grouping of pixels or objects and their similarities. This method is particularly useful for classifying satellite images where predefined training data is unavailable.

Hierarchical clustering involves two main approaches i.e. agglomerative (bottom-up) clustering and divisive (top-down) clustering. Agglomerative (bottom-up) clustering is the most common in remote sensing. It starts with

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

each pixel or object as its own cluster and iteratively merges the closest cluster based on a distance metric until all pixels are grouped into a single cluster. Divisive (top-down) clustering begins with a single cluster containing all pixels and recursively splits clusters into smaller clusters until each pixel is in its own cluster. However, this approach is less commonly used due to its computational complexity.

Following are the major steps in hierarchical clustering:

1. *Calculation of Distance Matrix:*

After preparing the data to be used for classification on a chosen theme, a distance matrix is created using a distance measure such as Euclidean distance, Manhattan distance, or cosine similarity. This matrix defines the dissimilarities between each pair of pixels or objects. A distance or similarity metric is computed between all pairs of pixels or data points based on their RGB or reflectance values. For example, if there are three bands being used for classification, the distance between pixel 1 (representing class1 having values of 33, 87, 126 in the three bands, respectively) and pixel 2 (representing class2 having values of 121, 187, 56) would be calculated as:

$$\text{Distance} = \text{sqrt}[(100-120)^2+(150-180)^2+(50-60)^2]$$

$$= \text{sqrt}[(-20)^2+(-30)^2+(-10)^2]$$

$$= \text{sqrt}[400+900+100] = \text{sqrt}[1400]$$

$$= 37.42$$

2. *Cluster Initialisation*:

As you have learnt earlier in this subsection, agglomerative approach is the most common hierarchical clustering method. It starts with each pixel as its own cluster (singleton). In each iteration, the closest pair of clusters is merged to form a new cluster. For example, in iteration1, it finds the closest pair of clusters (initially individual pixels) and merges them. In our example, pixels 1 and 2 might be the closest. In iteration 2, it updates distances based on new cluster centroids and finds the next closest pair of clusters to merge. This process continues iteratively until all pixels are grouped into a single cluster or until a stopping criterion is met.

In case of the divisive approach, it begins with all pixels in a single cluster and iteratively splits the cluster into smaller clusters until each pixel is in its own cluster or another stopping criterion is reached.

3. *Merge Clusters:*

Clusters are merged iteratively based on their proximity. The merging criteria depends on the linkage method used, such as Single Linkage (Nearest Neighbour), which merges clusters based on the smallest distance between points; Complete Linkage (Farthest Neighbour) merges clusters based on the maximum distance between points; Average Linkage merges clusters based on the average distance between all pairs of points in the clusters and Ward's

Linkage minimises the variance within clusters during merging, often yielding more compact clusters.

As you merge clusters, you build a dendrogram showing how clusters combine. Initially, the dendrogram will show individual pixels merging into small clusters, which then merge into larger clusters.

4. ***Dendrogram Creation:***

A dendrogram is constructed to visualise the hierarchy of the cluster merges. The vertical axis shows the clusters, and the horizontal axis represents the distance or dissimilarity at which clusters are merged. At the beginning, the dendrogram has each pixel as its own branch. As merging progresses, the branches for pixels 1 and 2 join, forming a branch for a class (assuming both class1 and class2 are subclasses of a broad class) cluster. Similarly, pixels 3 and 4 form a branch for the same class cluster, and Pixels 5 and 6 form a branch for the another cluster. In the end, you will have a single tree structure with three main branches corresponding to the three different land cover types.

5. ***Cluster Selection/Extraction:***

A cut-off point is chosen on the dendrogram to select/extact the desired number of clusters, which are then used to classify the image. For example, if you decide on the number of clusters or the similarity threshold as 3 at which to cut the dendrogram. The cutting the dendrogram at a point where three main branches are clearly separated will give you three clusters: one for class A, one for class B, and one for class C.

The dendrogram graphically represents the hierarchical relationships among clusters. It would have vertical lines representing clusters at different levels of similarity, horizontal lines indicating the merging of clusters, with the position on the distance scale showing the similarity level at which clusters are merged, and the cut-off line which is a horizontal line that can be drawn across the dendrogram to select clusters based on the desired level of similarity.

6. ***Post-Processing:***

At the post-processing stage, you need to analyse the clusters to interpret the land use land cover types they represent. In this case, the clusters align with the known three land cover types. It is also required to validate the clusters with reference data or ground truth, if available.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- It does not require training data or initial class labels, thus making it suitable for exploratory analysis.
- It produces dendrograms that offers a visual representation of data hierarchy and clustering structure, aiding in understanding data relationships.
- It is flexible and intuitive and easily handles different types of distance metrics and linkage criteria, allowing customisation based on data characteristics.

- It can reveal nested or hierarchical structures in data that other clustering methods may not.

Limitation

- The method can be computationally expensive, especially for large datasets, as it requires the calculation of all pairwise distances and repeated merging operations.
- It is sensitive to outliers, which can significantly affect cluster formation.
- The approach is less practical for very large datasets due to memory and processing time constraints.
- The final clustering result depends mainly on the chosen linkage method, which may not always capture the most meaningful groupings.

Hierarchical clustering is useful in land use land cover mapping without predefined classes; change detection to identify changes in land cover over time by clustering multi-temporal images; vegetation mapping to differentiate various vegetation types based on spectral reflectance patterns, etc.

13.5.4 Self-Organising Maps

Developed by Teuvo Kohonen in the 1980s, Self-Organising Maps (SOM) also known as Kohonen maps, are a type of artificial neural networks used largely in unsupervised classification of remotely sensed data. SOMs are useful for clustering remote sensing data based on inherent similarities without prior knowledge of the classes, making them ideal for exploring complex, high-dimensional remote sensing datasets. They are designed to reduce the dimensionality of data while preserving the topological and metric relationships of the input space, making them suitable for feature extraction and clustering in remote sensing data based analysis.

It is useful to perform clustering and visualisation of high-dimensional data by projecting them onto a low-dimensional (usually 2D) grid. It consists of neurons (also called nodes or units), organised in a grid where each neuron represents a prototype vector or weight, corresponding to a particular feature pattern.

Following are the major steps in self-organising maps based clustering:

1. *Initialisation:*

The SOM grid is initialised with random weight vectors, each representing a neuron on the grid. Neurons are the nodes in the grid, each having a weight vector of the same dimension as the input vector. Input data is the vectors representing the features of each pixel or region in the remote sensing image, such as spectral bands.

2. *Training Process:*

Training of SOMs involves adjusting the weight vectors of neurons to map the input data onto the map, preserving the spatial relationships and clustering similar data points together. It is carried out in the following steps:

- **Initialisation**: Weight vectors are initialised randomly or using some heuristic, such as small values close to zero.

- **Input Presentation/Selection**: Each pixel or data point from the image is presented to the SOM. For each training iteration, an input vector is randomly selected from the dataset.
- **Best Matching Unit (BMU)**: For each input vector, the neuron whose weight vector is closest (based on a distance metric like Euclidean distance) to the input vector is identified as the BMU.
- **Weight Update**: The BMU and its neighbouring neurons update their weights to move closer to the input vector. The extent of the update depends on a learning rate and a neighbourhood function, which decreases over time and distance.
- **Iteration and convergence**: The process is repeated for all input data points over multiple epochs, gradually fine-tuning the SOM.

3. *Clustering and Output:*

After training, similar input vectors are mapped to neighbouring neurons, effectively clustering the data in a way that preserves the topological structure of the input space.

The graphical output of a SOM typically consists of a 2D grid (often hexagonal or rectangular) where each cell represents a cluster or class. The grid is coloured or shaded based on the weights of the neurons, depicting patterns and relationships among data points. New input data can be mapped onto the trained SOM by finding the BMU for the new input and using the map to classify or analyse the data based on its location on the map.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- It does not require training data thus making it suitable for exploratory analysis and scenarios where labels are unavailable or costly to obtain.
- It is useful in dimensionality reduction as it can reduce high dimensitonal data into a lower-dimensional thus making it easier to visualise and interpret multiple spectral bands.
- It is flexible and can be applied to a range of data types.
- It is effective at identifying clusters and patterns hence for classification and feature extraction.
- It preserves the topological relationships of data in the input space thus allowing for meaningful clustering and analysis.
- It is relatively robust to noise and outliers because neighbourhood-based updating mechanism smooths the influence of individual noisy data points.

Limitation

- The method can be computationally demanding and time consuming, especially for large datasets, as it involves multiple iterations over the data, which can be slow.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

- It is sensitive to the choice of parameters such as learning rate, neighbourhood function, and map size, which can lead to suboptimal clustering and representation.
- The network structure (e.g., grid size and shape) is fixed before training. If the chosen structure does not match the complexity of the data, the resulting map may not effectively capture the data's relationships.
- Interpreting the resultant map and understanding the specific meaning of clusters can be complex, especially for high-dimensional or abstract data. Visualisation of the map may not always clearly convey the clustering results.
- Outputs of SOMs can vary due to converge to local minima, depending on initialisation and training conditions.

13.5.5 Fuzzy C-Means Clustering

It is a type of clustering algorithm that is widely used for soft classification. In hard classification, each data point belongs to exactly one cluster, whereas FCM allows each data point to belong to multiple clusters with varying degrees of membership. This approach is particularly useful where land cover types often have mixed characteristics, making it difficult to classify them into discrete categories. FCM assigns a membership value to each data point for all clusters. These membership values range between 0 and 1, with the sum of membership values for each data point equal to 1. In a typical graphical output of FCM clustering in remote sensing, each pixel in an image is represented by a colour or shade corresponding to its membership values across different clusters. In FCM clustering, the cluster centers are represented as distinct points in the feature space, typically marked by different colours. Each pixel in the image is coloured based on its highest membership value, with the intensity indicating the degree of membership. Further, unlike the hard clustering approaches you have studied till now, in FCM clustering, boundaries between clusters are not sharp, rather, there are smooth transitions indicating the fuzziness.

The FCM is particularly suitable for remote sensing applications where land cover types are not clearly separable. It is used in case of land use land cover classification, where boundaries between classes such as urban, vegetation, water, and barren land, may be unclear. Also, it is used in case of vegetation and soil mapping when different types of vegetation and soil types have overlapping spectral signatures as there may be gradual transition from one type to another. Further, it is also useful in change detection studies where images from different time periods show gradual changes in the classes of interest.

Following are the major steps in FCM clustering:

1. *Initialisation:*

The first step is to select the number of clusters, the fuzziness parameter (typically >1), and the maximum number of iterations or convergence criterion.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

Then the membership matrix is randomly initialised, where each element of the matrix should satisfy certain criteria.

2. ***Computation of Cluster Centers:***

Then cluster centers are calculated using the current membership matrix. The cluster center for a cluster is computed satisfying criteria for data point and total number of data points.

3. ***Updating of Membership Matrix:***

The next step is updating membership values or degrees based on the distance of each pixel to the centroids.

4. ***Checking Convergence:***

Next step is evaluation of convergence criterion to check if the membership matrix or cluster centers have converged. The convergence can be determined by either change in membership (by assessing if the change in membership degrees between iterations is below a predefined threshold) or change in cluster centres (by assessing if the change in cluster centers between iterations is below a predefined threshold). If convergence criteria are met or the maximum number of iterations is reached then the algorithm is terminated otherwise the process continues until convergence is achieved.

5. ***Output Clusters:***

Once convergence is achieved, the final cluster centers and membership matrix represent the clustering results. Each data point will have membership values for each cluster, indicating the degree to which it belongs to each cluster.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- It does soft classification and captures fuzziness in thematic classes, providing more realistic representation for mixed pixels.
- It is effective in areas where class boundaries are not clear or well-defined.
- It is flexible and allows for degree of class membership thereby accommodating complex scenarios.

Limitation

- The method can be computationally demanding and time consuming, as compared to hard classifiers.
- The algorithm may converge to local minima, depending on the initial cluster centers, which can affect the final outcome.
- It is sensitive to the initial membership value and also choice of fuzziness parameter.
- Although it is robust to within-class variations, it can still be sensitive to noise, particularly in the cases, when the noise levels are high relative to the actual signal.

You have become familiar with several commonly used algorithms for unsupervised classification. Let us now see their comparison in Table 13.8.

Table 13.8 Comparison of various types of unsupervised classification algorithms.

Feature / Method	K-Means Clustering	ISODATA Clustering	Hierarchical Clustering	Self-Organising Maps (SOM)	Fuzzy C-Means (FCM)
Algorithm Type	Partition-based	Partition-based	Hierarchical	Neural Network-based	Partition-based
Cluster Shape	Spherical / Convex	Spherical / Convex	Arbitrary (varies with distance metric)	Arbitrary (depends on map grid)	Spherical / Convex
Number of Clusters	Fixed (predefined)	Dynamic (can merge/split clusters)	Dynamic (depends on dendrogram cut)	Fixed (predefined map size)	Fixed (predefined)
Initialisation	Random or heuristic	Random or heuristic	Not applicable	Random or heuristic	Random or heuristic
Scalability	Good for large datasets	Good for large datasets	Can be slow for large datasets	Can be slow and memory-intensive for large datasets	Can be slow and memory-intensive for large datasets
Convergence Criteria	Minimises within-cluster variance	Minimises within-cluster variance, allows for merging/splitting	Based on dendrogram cut-off	Convergence of weights and membership matrix	Minimises weighted within-cluster variance
Output	Cluster centers and assignment	Cluster centers and assignment, plus merging/splitting information	Dendrogram, cluster assignments at various levels	Cluster assignments and visualization on 2D map	Membership degrees for each data point in each cluster
Flexibility	Less flexible (fixed number of clusters)	More flexible (dynamic cluster number)	Highly flexible (varies with dendrogram cut)	Flexible (visual representation and clustering)	Flexible (soft clustering with membership degrees)
Suitability	Well-separated clusters, fixed number of clusters	Complex data with varying cluster shapes and sizes	Hierarchical relationships, variable number of clusters	Visualising and clustering high-dimensional data	Clustering with overlapping data or ambiguous boundaries

13.6 SOME OTHER APPROACHES

There are some other not so common approaches for unsupervised image classification that you will be getting introduced to here.

13.6.1 Gaussian Mixture Models (GMMs)

Gaussian Mixture Models (GMMs) are probabilistic models used in statistics and machine learning to represent complex data distributions. Comprising a mixture of Gaussian (normal) distributions, GMMs can model diverse patterns within a dataset. Each Gaussian component represents a potential cluster, allowing GMMs to effectively capture intricate structures. Widely applied in clustering, density estimation, and pattern recognition, GMMs excel in scenarios where data exhibits multifaceted characteristics. Their flexibility and ability to express uncertainty make GMMs valuable in various fields, including geospatial data analysis (Sekhar et al., 2002; Okwuashi et al., 2011; Vatsavai et al., 2011), image processing, and speech recognition.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

GMM-based clustering is particularly effective in scenarios where the underlying data distribution is complex and may not adhere to a simple, linear separation (Fig. 13.4). In the context of GMMs, the algorithm identifies clusters by modeling the data as a mixture of multiple Gaussian distributions (Bishop 2006).

Each Gaussian distribution represents a potential cluster, and the algorithm assigns data points to these clusters based on the probability of belonging to each distribution. This unsupervised approach allows for the discovery of hidden patterns and groupings within the data without the need for labelled examples.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- ***Flexibility and Expressiveness**:* GMMs are versatile and can model complex data distributions with multiple components. This flexibility makes them well-suited for capturing intricate patterns and structures in the data, especially when the underlying distribution is not easily characterised by a single Gaussian.
- ***Probabilistic Output**:* GMMs provide probabilistic output, assigning each data point a probability of belonging to different clusters. This allows for a more nuanced representation of uncertainty and provides a richer understanding of the data distribution.
- ***Soft Clustering**:* Unlike some traditional clustering algorithms that assign each data point to a single cluster, GMMs perform soft clustering. This means that data points can belong to multiple clusters simultaneously, reflecting the uncertainty inherent in many real-world datasets.
- ***Effective Handling of Elliptical Clusters**:* GMMs can model clusters with elliptical shapes, making them suitable for datasets where clusters have varying orientations and sizes. This is an advantage over methods like k-means, which assumes spherical clusters.
- ***Handling Mixed Distributions**:* GMMs are capable of capturing mixed distributions within a dataset. This is particularly useful in scenarios where subpopulations with distinct characteristics exist, and a single clustering approach might not be sufficient.
- ***Robustness to Noise**:* GMMs are less sensitive to outliers compared to some other clustering methods. The probabilistic nature of GMMs helps mitigate the impact of noise by considering the overall distribution rather than relying on individual data points.

13.6.2 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that works by grouping data points based on their density in the feature space. DBSCAN is particularly useful for discovering clusters of arbitrary shapes and handling noise in datasets. Its ability to automatically determine the number of clusters without prior knowledge makes it advantageous for various applications, especially in geospatial data analysis (Wang and Wang, 2007). An overview of the working of DBSCAN is given here:

- ***Core Points**:* DBSCAN identifies core points as those with a minimum number of neighbouring points within a specified distance.
- ***Density-Reachability**:* Points within the defined distance of a core point are considered part of the same cluster. This creates a region of connected points that are dense enough to form a cluster.
- ***Border Points**:* Points within the specified distance of a core point but do not meet the minimum density criteria become border points. They are part of the cluster but not considered as influential as core points.
- ***Noise Points**:* Points that are not core points and do not have enough neighbouring points within the distance threshold are considered noise. These points do not belong to any cluster.
- ***Cluster Formation**:* The algorithm iteratively expands clusters by examining the density-reachability relationship. It continues this process until all points are assigned to a cluster or labelled as noise.
- ***Parameter Tuning**:* DBSCAN requires two main parameters - the minimum number of points and the distance threshold. The effectiveness of DBSCAN can be sensitive to the appropriate selection of these parameters, and they need to be chosen based on the characteristics of the data.
- ***Result**:* The final result is a set of clusters, each containing core and border points. Noise points are treated as outliers.

Let us now learn the advantages and limitations of this algorithm.

Advantage

- ***Flexibility in Cluster Shape**:* DBSCAN is capable of identifying clusters with irregular shapes, making it suitable for datasets where clusters may not conform to traditional geometric shapes. Unlike algorithms like k-means, which assume spherical clusters, DBSCAN excels at discovering clusters of varying shapes and sizes, providing more flexibility in capturing complex patterns within the data.
- ***Noise Handling**:* DBSCAN effectively handles noise and outliers in the data. It classifies data points that do not belong to any cluster as noise, allowing for a more robust clustering result. This feature is particularly valuable in real-world datasets, which often contain irregularities or anomalies that can distort the clustering outcomes of other algorithms.
- ***Automatic Determination of Cluster Number**:* DBSCAN does not require the user to specify the number of clusters in advance, as opposed to algorithms like k-means that rely on this input. The algorithm dynamically identifies clusters based on the density of data points, making it well-suited for situations where the optimal number of clusters is unknown or may vary across different parts of the dataset. This adaptability simplifies the clustering process and is especially advantageous in exploratory data analysis.

13.6.3 Feature Space Image Classification

Feature space image classification is also an important technique. As you have read earlier, a **feature space image** is an image where each pixel's value represents its position in the feature space (Fig. 13.5). This type of image is created by transforming the original image data into the feature space, where each pixel is characterised by a set of features rather than just its spectral values. This feature space image is displayed as a raster image, with each pixel coloured to represent cumulative frequency. Bright colors (or grayscale intensity) indicate a high density of points, while dark tones signify a low density. Additionally, pixel values can be coloured based on thematic layers assigned to the same image. Classification of the feature space image could be useful. For example, when using PCA for image classification, the original multispectral image is transformed into a new set of principal components. Each pixel in the feature space image is represented by its coordinates in this new set of components, capturing the most significant variations in the data.

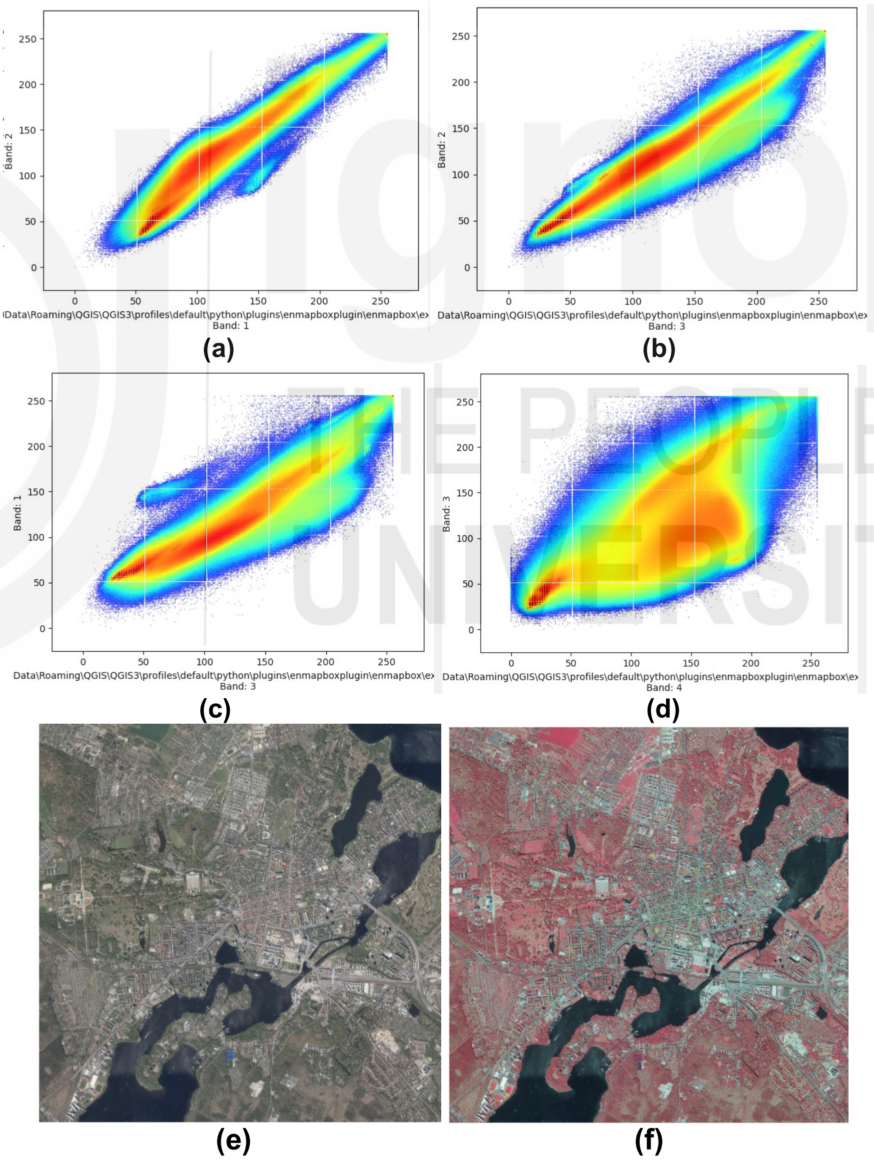

Fig. 13.5: Feature space of a) Band 1 vs Band 2; b) Band 3 vs Band 2; c) Band 3 vs Band 1; d) Band 4 vs Band 3; e) True colour composite (321); and f) False colour composite (432) of an aerial image of Portsdam city, Germany acquired on 01/04/2019. (Source: Aerial image https://github.com/EnMAP-Box/enmap-box/blob/main/enmapbox/exampledata/aerial_potsdam.tif)

Spend 5 minutes to check your progress.

SAQ II

a) What is Gaussian mixture model?

b) What is the usefulness of GMM in thematic information extraction?

c) What are the limitations of density-based spatial clustering?

13.7 CHALLENGES AND RECENT DEVELOPMENTS

You have learnt that unsupervised classification is useful in certain cases when you do not have field data and would like to know natural groupings present in a data. You have also learnt that it is useful in generating initial exploratory analysis particularly in unexplored or rapidly changing environments. It is also useful in change detection analysis as unsupervised classification can highlight changes in land use land cover classes over time. It may also be helpful in geological surveys to identify areas of interest for mineral prospecting.

Despite these advantages, there are certain challenges in unsupervised classification such as in interpreting in assigning informational class or labelling the clusters in a meaningful way. Another challenge is the spectral similarity on which basis the classification has been carried out by an algorithm and the clusters generated, may not correspond well to real-world classes, if those classes have similar spectral properties. Sensitivity to the parameter could be another issue as algorithms require careful tuning of parameters that may influence the outcome. Scalability is another issue as some algorithms may struggle to perform due to them being computationally intensive.

There are certain developments which are driving advancements in the field. The focus is on handling big data, improving robustness, and enhancing interpretability. These trends are enhancing the accuracy and efficiency of clustering and classification processes, making it possible to extract more valuable insights from complex remote sensing datasets. Some of the recent developments in the field include the following:

- **Integration with Machine Learning**: Hybrid approaches combine unsupervised classification with supervised learning to refine results.
- **Deep Learning Adaptations**: Convolutional Neural Networks (CNNs) and autoencoders are being adapted to perform unsupervised feature extraction and clustering. Convolutional Neural Networks (CNNs) analyse both spectral and spatial information in hyperspectral images, improving classification accuracy by leveraging spatial context. Autoencoders are being used for feature extraction and dimensionality reduction, as they learn compact representations of high-dimensional data. These are increasingly used for multispectral and hyperspectral images. Generative Adversarial Networks (GANs) are being used for data augmentation and synthetic data generation, which can improve the quality and quantity of training data for unsupervised models. Further, deep clustering models combining deep learning with

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

clustering algorithms are being used to integrate feature learning and clustering into a unified framework.

- **High-Dimensional Data Handling**: Advanced algorithms are being used alongside clustering to manage high-dimensional hyperspectral data.
- **Multi-Scale and Multi-Resolution Approaches**: Techniques that combine information from multiple scales or resolutions to capture more detailed and accurate features in remote sensing data are being used.
- **Multi-Source Data Integration and Feature Fusion**: Combining data from different sources (e.g., optical, radar, LiDAR) to improve clustering outcomes is another trend. Data fusion methods integrate diverse datasets to provide richer and more accurate information for unsupervised classification. Also, feature fusion i.e. merging features from different types of data (e.g., spectral, spatial, temporal) is being done to enhance the clustering process and extract more meaningful patterns.
- **Hybrid Models**: Hybrid models such as combining traditional unsupervised clustering methods with deep learning features are being used. For example, using SOMs or FCM with features extracted by deep learning models to enhance clustering performance.
- **Handling Big Data**: Development of algorithms that handle large volumes of remote sensing data efficiently is another trend. Techniques such as parallel processing and distributed computing are used to manage and analyse big data in remote sensing applications. Cloud computing platforms is also being leveraged for scalable storage and processing of remote sensing data, thereby allowing for more extensive and complex analyses.
- **Dimensionality Reduction**: Advanced techniques are being used for reducing the dimensionality of high-dimensional remote sensing data while preserving its structure.
- **Adaptive and Dynamic Clustering methods**: Algorithms that can adjust cluster parameters dynamically based on the data characteristics are being used, which improve the adaptability of clustering algorithms to varying data distributions and complexities. Further, techniques that evolve clusters over time or as more data becomes available, useful for monitoring changes in remote sensing data are also being devleloped.
- **Uncertainty and Robustness**: Handling uncertainty is a major issue in the field so methods are being developed that can quantify and manage uncertainty in clustering results. Also, the algorithms are being designed to be less sensitive to outliers and noise, thereby improving the reliability of clustering results in challenging environments.

13.8 SUMMARY

Let us summarise what we have learnt in this unit. We have learnt that:

- Image classification is the process of partitioning image in certain groups of information classes based on their spectral characteristics. There are broadly two approaches of image classification i.e. unsupervised and supervised.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

- Unsupervised image classification is the process of image classification in which user input is minimum and the process is guided by the spectral similarity of the objects present in the image. This unit essentially describes the fundamentals of unsupervised classification approaches
- The commonly used algorithms in unsupervised image classification are k-means and ISODATA clustering.
- There are some other algorithms used in unsupervised image classification.
- However, each algorithm has its own advantages and limitations and choice of the algorithm is guided by the nature of data being used, number and nature of features present in the image and also computational resources.

13.9 TERMINAL QUESTIONS

1. What are the generic steps in unsupervised image classification?
2. Write comparison of unsupervised and supervised classification.
3. What is the difference between hard and soft classification?
4. What is the difference between pixel and object based classification?
5. Write the difference between parametric and non-parametric classification.
6. Compare various algorithms of unsupervised classification.

13.10 REFERENCES

- Banerjee, B., Bovolo, F., Bhattacharya, A., Bruzzone, L., Chaudhuri, S., & Mohan, B. K. (2014). A new self-training-based unsupervised satellite image classification technique using cluster ensemble strategy. IEEE Geoscience and Remote Sensing Letters, 12(4), 741-745.
- Bezdek, J. C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms. Springer.
- Bishop, C.M. (2006). Pattern recognition and machine learning by Christopher M. Bishop. Springer Science+ Business Media, LLC.
- Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining.
- Foody, G. M. (1996). Fuzzy modelling of vegetation from remotely sensed imagery. Ecological Modelling.
- Foody, G. M. (1999). Applications of the self-organizing feature map neural network in community data analysis. Ecological Modelling.
- Jain, A. K. & Dubes, R. C. (1988). Algorithms for clustering data. Prentice Hall.
- Jolliffe, I. T. (1986). Principal Component Analysis. Springer-Verlag.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

- Kaufman, L., & Rousseeuw, P. J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. Wiley-Interscience.
- Kohonen, T. (2001). Self-Organizing maps. Springer Series in Information Sciences.
- Lee, T. W., & Lewicki, M. S. (2002). Unsupervised image classification, segmentation, and enhancement using ICA mixture models. IEEE Transactions on Image Processing, 11(3), 270-279.
- Lillesand, T. M., Kiefer, R. W. & Chipman, J. W. (2015). Remote Sensing and Image Interpretation. John Wiley & Sons.
- Liu, J., & Mason, P. J. (2009). Essential Image Processing and GIS for Remote Sensing. John Wiley & Sons.
- Lloyd, S. (1982). Least squares quantization in PCM. IEEE transactions on information theory, 28(2), 129-137.
- Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. International journal of Remote sensing, 28(5), 823-870.
- Mather, P. M., & Tso, B. (2016). Classification Methods for Remotely Sensed Data. CRC Press.
- Mather, P. M., & Tso, B. (2009). Classification Methods for Remotely Sensed Data. CRC Press.
- Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. Advances in Neural Information Processing Systems.
- Okwuashi, O., Eyo, E., & Eyoh, A. (2011). Supervised Gaussian Mixture Model Based Remote Sensing Image Classification. Global Journal of Environmental Sciences, 10(1&2), 1-10. https://www.ajol.info/index.php/gjes/article/view/79235
- Pal, N. R., & Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. IEEE Transactions on Fuzzy Systems.
- Salah, M. (2017). A survey of modern classification techniques in remote sensing for improved image classification. Journal of Geomatics, 11(1), 21.
- Schmarje, L., Santarossa, M., Schröder, S. M., & Koch, R. (2021). A survey on semi-, self-and unsupervised learning for image classification. IEEE Access, 9, 82146-82168.
- Shekhar, S., Schrater, P. R., Vatsavai, R. R., Wu, W. and Chawla, S. (2002). Spatial contextual classification and prediction models for mining geospatial data. IEEE Transactions on Multimedia, 4(2), 174-188.
- Vatsavai, R.R., Symons, C.T., Chandola, V. and Jun, G. (2011). GX-Means: A model-based divide and merge algorithm for geospatial image clustering. Procedia Computer Science, 4, 186-195.
- Wang, Q., Li, Q., Liu, H., Wang, Y., & Zhu, J. (2014, October). An improved ISODATA algorithm for hyperspectral image classification. In 2014 7th International Congress on Image and Signal Processing (pp. 660-664). IEEE.
- Wang, X. and Wang, J. (2010). Using clustering methods in geospatial information systems. Geomatica, 64(3), 347-361.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

13.11 FURTHER/SUGGESTED READINGS

- Abburu, S. & Golla, S. B. (2015). Satellite image classification methods and techniques: A review. International journal of computer applications, 119(8).
- Ball, G. H. & Hall, D. J. (1967). A clustering technique for summarizing multivariate data. Behavioral Science.
- Bezdek, J. C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms. Springer.
- Dhingra, S. & Kumar, D. (2019). A review of remotely sensed satellite image classification. International Journal of Electrical and Computer Engineering, 9(3), 1720.
- Foody, G. M. (1996). Fuzzy modelling of vegetation from remotely sensed imagery. Ecological Modelling.
- Foody, G. M. (1999). Applications of the self-organizing feature map neural network in community data analysis. Ecological Modelling.
- Jain, A. K. & Dubes, R. C. (1988). Algorithms for clustering data. Prentice Hall.
- Kohonen, T. (2001). Self-Organizing Maps. Springer Series in Information Sciences.
- Lillesand, T. M., Kiefer, R. W. & Chipman, J. W. (2015). Remote Sensing and Image Interpretation. John Wiley & Sons.
- Liu, J., & Mason, P. J. (2009) Essential Image Processing and GIS for Remote Sensing. John Wiley & Sons.
- Lu, D., & Weng, Q. (2007) A survey of image classification methods and techniques for improving classification performance. International journal of Remote sensing, 28(5), 823-870.
- Mather, P. M. & Tso, B. (2016). Classification Methods for Remotely Sensed Data. CRC Press.
- Pal, N. R. & Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. IEEE Transactions on Fuzzy Systems.
- Richards, J. A. & Jia, X. (2006). Remote Sensing Digital Image Analysis: An Introduction. Springer.
- Salah, M. (2017). A survey of modern classification techniques in remote sensing for improved image classification. Journal of Geomatics, 11(1), 21.
- Tou, J. T. & Gonzalez, R. C. (1974). Pattern Recognition Principles. Addison-Wesley.

13.12 ANSWERS

SAQ I

a) You answer should include the points covered in section 13.3.

b) You answer should include the points covered in subsection 13.5.1.

c) You answer should include the points covered in subsection 13.5.2.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh

SAQ II

a) Gaussian Mixture Models (GMMs) are probabilistic models used in statistics and machine learning to represent complex data distributions.

b) Comprising a mixture of Gaussian (normal) distributions, GMMs can model diverse patterns within a dataset. Each Gaussian component represents a potential cluster, allowing GMMs to effectively capture intricate structures.

c) You answer should include the points covered in subsection 13.6.2.

Terminal Questions

1. You answer should include the points covered in section 13.4.
2. You answer should include the difference in terms of mode, requirement of field information, stage of analysis, etc. as covered in subsection 13.3.1.
3. You answer should include the points covered in subsection 13.3.3.
4. You answer should include the points covered in subsection 13.3.4.
5. You answer should include the points covered in subsection 13.3.5.
6. You answer should include the points covered in section 13.5.

Contributors: Dr. Sourish Chatterjee and Prof. Benidhar Deshmukh