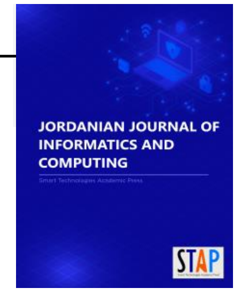
Smart Technologies Academic Press

Jordanian Journal of Informatics and Computing
https://www.jjic.thestap.com/



A Comparative Analysis of Classification Algorithms in Land Cover Mapping: A Study of MLC, Mahalanobis Distance, NNC, and SVM for 2023 and 2018: Case study of Wasit province, central Iraq

Yasir Abdulameer Nayyef Aldabbagh[1, 2], **Ali Ibrahim Zghair Alnasrawi** [3]

[1] *Administrative and Finance Affairs Department, Al-Muthanna University, Muthanna Samawah, Iraq*
[2] *Department of English, College of Education, Sawa University, Al-Muthana, Samawah, Iraq.*
[3] *Ministry of Education, General Directorate for Education, Maysan, Iraq*

ARTICLE INFO

Article History

Received: 30-06-2025
Revised: 10-08-2025
Accepted: 23-08-2025

Vol.2025, No.1

DOI:

Corresponding author.
Email: *yasir.aldabbagh@mu.edu.iq*

Orcid:

ABSTRACT

The research presents in this paper a comparison between different classification procedures for land cover data between the present and four years prior. The techniques include Maximum Likelihood Classification (MLC), Mahalanobis Distance, Nearest Neighbor Classification (NNC), and Support Vector Machines (SVM). Confusion matrices have been employed to evaluate performance, measuring categorical biases across water, vegetation, urban structures, soil, and other land cover types for each method. The results reveal variations in accuracy not only over time but also across algorithms, offering insights into the strengths and limitations of these classification systems under different temporal conditions. The study aims to lay the foundation for future adaptations of these approaches to improve land cover analysis and long-term monitoring.

Keywords: Classification Algorithms, Land Cover mapping, Iraq, Maximum Likelihood Classification (MLC), Mahalanobis Distance, Nearest Neighbour Classification (NNC), Support Vector Machines (SVM).

How to cite the article

1. Introduction

Land cover maps play a pivotal role in natural resource management and environmental planning, especially with the ongoing environmental change resulting from urban expansion, agricultural expansion, and climate fluctuations, the ability to accurately classify and monitor land cover types has become of great importance to environmental researchers and environmental managers. Remote sensing techniques, along with advanced classification algorithms, provide a powerful means to analyze land use patterns and monitor land cover changes over time [1].

In recent years, machine learning techniques have emerged with their great potential to significantly improve the accuracy of land cover classification compared to traditional manual methods, in addition to computational efficiency when conducting analyses and faster analysis time [2]. Among the most prominent algorithms used in this field are maximum likelihood classification (MLC) [3], Mahalanobis distance [4], Nearest Neighbor Classification (NNC) [5], and Support Vector Machines (SVM) [6]. Each of these methods has many advantages and a set of challenges that vary depending on the land cover, datasets used, and different classification scenarios.

This study aims to conduct a comparative analysis of these four classification algorithms by mapping land cover in Wasit Governorate, central Iraq. Wasit Governorate is an agricultural region characterized by diverse landscapes including urban centers, agricultural lands, and natural vegetation. Its strategic location in the central region of Iraq, coupled with the increasing changes in land cover caused by climate change, environmental pollution, and human practices, makes it an ideal area for conducting this study to evaluate the performance of classification algorithms in capturing changes in land cover maps over time.
This study focuses on the analysis of land cover data between 2018 and 2023. In this research, we will evaluate the performance of each algorithm in identifying and classifying land cover types such as Water, Vegetation, Urban, and Soil.

This study contributes to a growing body of research on the application of machine learning techniques in remote sensing, with a particular focus on land cover mapping. By providing a comparative analysis of four major classification algorithms, it aims to guide future research and practical applications in environmental monitoring, land management, and sustainable development in Wasit Governorate and beyond.

2. Research Background

Land cover mapping is one of the most essential tasks for identifying environmental changes and managing natural resources. For many sectors, such as urban development and agriculture to environmental conservation, very accurate land cover type classification is highly needed. [7] introduce RS technology as an effective solution in both time and cost compared to traditional methods for land cover mapping. Satellite imagery, in this respect, has proved highly efficient, as recorded by several studies such as [8-12]. Policy planning, resource management, and environmental monitoring all depend on regional land cover mapping. More precise and thorough mapping at regional sizes has been made possible by developments in remote sensing, machine learning, and data integration; nonetheless, issues with processing speed, categorization consistency, and terrain adaptation still exist. Techniques and Strategies Data Integration: For broad regions, combining data from many sources, including vegetation indices, land surface temperature, topographic characteristics, and multiseasonal and multispectral satellite images, increases mapping accuracy and operating efficiency [13-15].

Methods of Classification: Machine learning classifiers such as Maximum Likelihood (ML), Random Forests (RF), Support Vector Machines (SVM), and Artificial Neural Networks (ANN) are frequently employed. For large-scale mapping, ML offers a compromise between speed and precision, whereas SVM and RF give excellent accuracy but can be computationally demanding [16-18]. By [19], the MLC algorithm attained an impressive 99.63% accuracy in land use/land cover mapping in Awka South LGA, Anambra State, Nigeria. The optimally tuned SVM classifier achieved the best overall accuracy of 94% with Sentinel-2 and Landsat-8 satellite data as presented by [20].

Other methods, such as Decision Trees and Random Forests, are also effective. [21] reported that DTs outperformed MLC and SVM in various land cover change assessments with accuracies above 85%. Similarly, [22,23] have stated that the RF algorithm is the most accurate machine-learning classifier for land use/land cover mapping and has outperformed six other examined algorithms.

3. Methodology

3.1. Study Area

Wasit Province in central Iraq offers a vast variation in landscape, where built-up areas and agricultural fields are interjected with natural vegetation. The changes that this area has witnessed in the past couple of years classify this as a classic case study for land cover change detection in Figure 1.

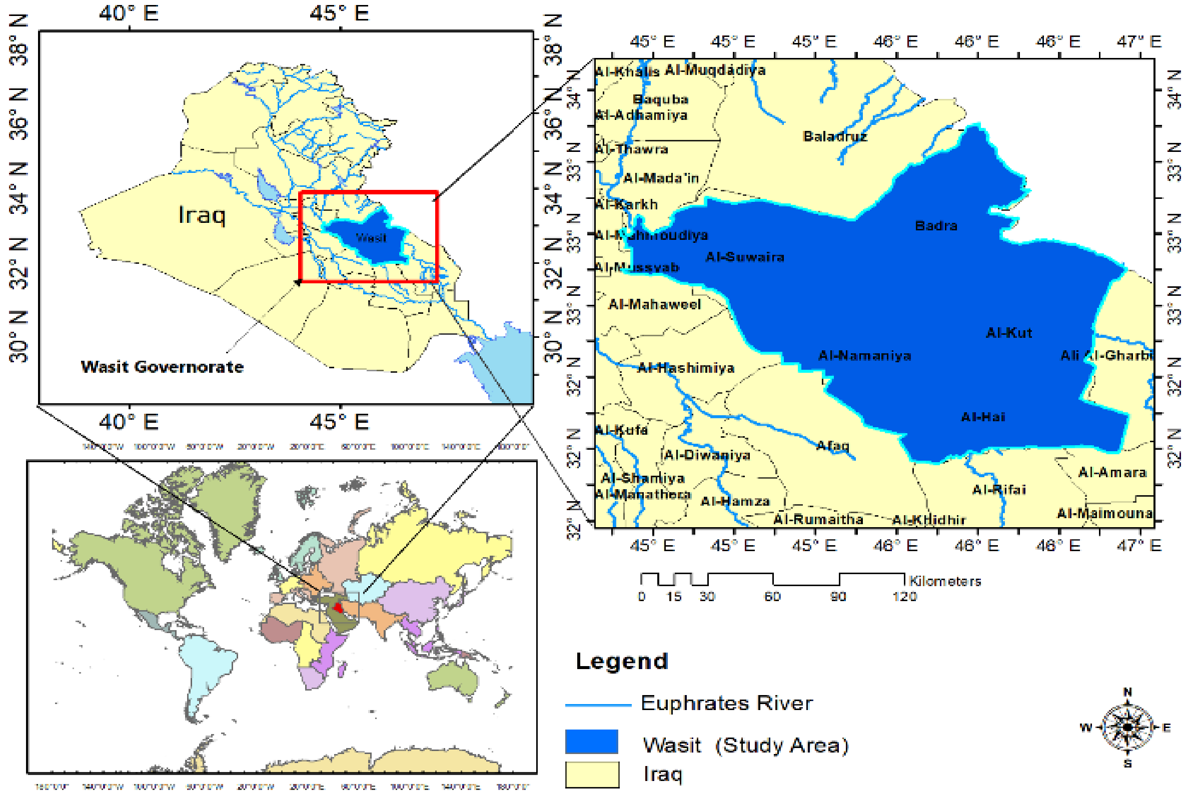


Figure 1. The study area

3.2. Data Collection

Satellite images for the year 2018 and 2023 were downloaded from archiving sources in [Landsat]. Therefore, the imagery pre-processed in advance was atmospherically and geometrically corrected. We prepared the ground truth data by field surveying and making use of existing land cover maps to validate our classification results.

Pre-processing of Landsat images. Landsat images were pre-processed in three steps including (1) radiometric calibration using the gain and offset values retrieved from the metadata files, (2) atmospheric correction by converting the data into the 0-1 range, and (3) geometric correction and co- registration to assure coincident with each other and the ground truth data, which were collected from Google Earth images

Table 1. Landsat images utilized in this research.

#	Landsat Project	Path/Row	Acquisition Date	Spatial Resolution (m)
1	Landsat 8 OLI	167/37	2018/06/15	30
2	Landsat 8 OLI	167/38	2018/06/15	30
3	Landsat 9 OLI	167/37	2023/07/30	30
4	Landsat 9 OLI	167/38	2023/07/30	30

3.3. Classification Algorithms

3.3.1. Maximum Likelihood Classification (MLC):

MLC is one of the most widely used classification algorithms in remote sensing and land cover mapping and is a statistical method that assumes that the data in each class in a feature space follow a multivariate normal (Gaussian) distribution. Maximum Likelihood Classification is based on the principle of probability theory and uses training data to estimate statistical parameters for each class (24).
The probability that a pixel belongs to a certain class i is given by the following equation [24]:

$$p(i|w) = \frac{p(w|i) \times p(i)}{p(w)} \qquad (1)$$

Where:
$p(i|w)$: The likelihood function.
$p(i)$: Probability of occurrence of class i in the study area.
w: a vector represents the pixel features.
$p(w)$: Probability of observing w.

$$p(w) = \sum_{i=1}^{M} p(w|i) \times p(i) \qquad (2)$$

Where:
M: Classes number
Pixel x is assigned to class i by the following rule:
A pixel x is considered to belong to class i if $p(i|w) > p(j|w)$ for all classes j≠i.
Each pixel is assigned to the class with the highest probability or is classified as unclassified if the probability values are below a user-defined threshold (24).
Thus, the training process is done by determining the number of land cover types within the study area (number of classes) and then selecting the pixels that belong to each class (training pixels) using the land cover information of the study area (24).

3.3.2. Mahalanobis Distance Classification

Mahalanobis distance classification is a method used to classify data based on the Mahalanobis distance measure. This method is useful in remote sensing and land cover classification because of its ability to take into account correlations between variables and the varying spreads of data into different classes [25].
The Mahalanobis distance is a measure of the distance between a point and a distribution. The Mahalanobis distance takes into account the correlations between variables and the variance within each class, and provides a standardized measure of distance that is sensitive to the shape of the data distribution. The Mahalanobis distance uses the class covariance matrix to normalize the distance measure, allowing it to efficiently handle correlations between features and differences in scale between features [25].
For a given class i, the Mahalanobis distance d_i between a pixel x and the mean vector μ_i of that class is calculated using the following formula:

$$d_i^2 = (x - \mu_i) \sum_i^{-1} (x - \mu_i) \qquad (3)$$

Where:
x: It is the vector of spectral values of the pixel being classified.
$\mu_{i:}$ It is the mean vector of class i.
$\sum_i$: It is the covariance matrix of class i.
The following steps demonstrate the classification method using Mahalanobis distance.

1) First, representative samples are collected for each class to estimate the mean vector and covariance matrix. This step involves selecting training data that accurately represents the spectral characteristics of each land cover class.
2) Calculating the mean vectors and covariance matrices for each class using the training data.
3) Calculating the Mahalanobis distance for each pixel for each class based on the class mean vector and class covariance matrix.
4) Assigning each pixel to the class with the smallest Mahalanobis distance.
5) Evaluating the classification accuracy using accuracy metrics such as overall accuracy and confusion matrices.

3.3.3. Nearest Neighbor Classification (NNC)

Nearest neighbor classification (NNC) is a classification algorithm used in remote sensing and other fields to classify data based on similarity. The basic principle of classification is that the algorithm classifies a given data point by comparing its attributes with the attributes of known data points (training samples) and assigning it to the class of the closest (most similar) training sample [26].

Nearest neighbor classification relies on a similarity measure, often Euclidean distance, to determine how similar a data point is to the training samples. The idea is that similar data points are more likely to belong to the same class. This algorithm is generalized by looking at the k nearest neighbors (KNN) and then the class of a data point is determined by a majority vote among these k neighbors [27].

The K-Nearest Neighbors (KNN) algorithm is a supervised machine learning method employed to tackle classification and regression problems. Evelyn Fix and Joseph Hodges developed this algorithm in 1951, which was subsequently expanded by Thomas Cover.

(K-NN) algorithm is a versatile and widely used machine learning algorithm that is primarily used for its simplicity and ease of implementation. It does not require any assumptions about the underlying data distribution. It can also handle both numerical and categorical data, making it a flexible choice for various types of datasets in classification and regression tasks. It is a non-parametric method that makes predictions based on the similarity of data points in a given dataset. K-NN is less sensitive to outliers compared to other algorithms [28].

kNN retains every training example and, when presented with a new instance, computes its distance to all stored examples. It then selects the k examples with the smallest distances and assigns the test point the label that appears most frequently among those neighbors. The two critical parameters are the neighborhood size (k) and the distance metric—Euclidean distance is the standard choice for real-valued feature vectors [29].

For two-class problems, one can independently find the k-th nearest neighbor within each class and assign the new point to whichever class's k-th neighbor is closer [30].

Design Decisions and Their Effects

- **Neighborhood Size (k):** The choice of k has a major impact on misclassification rates. Small odd values (e.g., k = 3) are common in binary tasks to avoid ties. Analyses based on Poisson and Binomial models clarify how k influences overall risk, inspiring practical, data-driven methods for selecting k. [29-31].
- **Distance Metric:** The distance function profoundly affects classification accuracy. Locally optimized metrics—designed to perform well with finite samples—can surpass the performance of the standard Euclidean distance. [32].

Extensions and Enhancements

- **High-Dimensional Data:** In many dimensions, assuming constant class probabilities within a neighborhood can introduce bias. Adaptive techniques—such as local linear discriminant analysis or learned flexible metrics—reshape neighborhoods to emphasize the most relevant feature directions, reducing bias and improving accuracy. [33,34].
- **Improved Variants:** The local mean–based pseudo nearest neighbor (LMPNN) replaces individual neighbors with their local class mean vectors, often yielding higher classification accuracy than traditional kNN. [35].

Applications and Empirical Performance

- **Practical Use:** kNN has been applied to problems like avalanche prediction, leveraging historical event data alongside meteorological variables to identify similar past conditions [36].
- **Theory vs. Practice:** Although worst-case theoretical bounds for kNN error can be pessimistic, refined theoretical analyses and empirical studies typically demonstrate much stronger real-world performance [37], Moreover, online adaptations of kNN can achieve low mistake rates in streaming or evolving environments [38,39,40,41,42,43,44].

The K-NN algorithm works by finding the K nearest neighbors to a given data point based on a distance metric, such as Euclidean distance. The class or value of the data point is then determined by the majority vote or average of the K neighbors. This approach allows the algorithm to adapt to different patterns and make predictions based on the local structure of the data [45].

As we mentioned above that the KNN algorithm helps us identify the nearest points or the groups for a query point. But to determine the closest groups or the nearest points for a query point we need some metric. For this purpose, we use below distance metrics:

1) Euclidean Distance: It is the cartesian distance between the two points which are in the plane/hyperplane. Euclidean distance can also be visualized as the length of the straight line that joins the two points which are into

consideration. if we have two points A(x1, y1), B(x2, y2), the Euclidean distance between these points is given by the formula:

$$d(A, B) = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \qquad (4)$$

2) Manhattan Distance: Manhattan Distance metric is generally used when we are interested in the total distance traveled by the object instead of the displacement. This metric is calculated by summing the absolute difference between the coordinates of the points in n-dimensions. if we have two points A(x1,y1), B(x2,y2), the Euclidean distance between these points is given by the formula:

$$d(A, B) = |x2 - x1| + |y2 - y1| \qquad (5)$$

The value of k is very crucial in the KNN algorithm to define the number of neighbors in the algorithm. The value of k in the k-nearest neighbors (k-NN) algorithm should be chosen based on the input data. If the input data has more outliers or noise, a higher value of k would be better. Cross-validation methods can help in selecting the best k value for the given dataset.
The K-Nearest Neighbors (KNN) algorithm operates on the principle of similarity, where it predicts the label or value of a new data point by considering the labels or values of its K nearest neighbors in the training dataset.
KNN works as follows:

1) Selecting the optimal value of K: K represents the number of nearest neighbors that needs to be considered while making prediction.
2) Calculating distance: To measure the similarity between target and training data points, Euclidean distance is used. Distance is calculated between each of the data points in the dataset and target point.
3) Finding Nearest Neighbors: The k data points with the smallest distances to the target point are the nearest neighbors.
4) Voting for Classification or Taking Average for Regression: In the classification problem, the class labels of are determined by performing majority voting. The class with the most occurrences among the neighbors becomes the predicted class for the target data point. In the regression problem, the class label is calculated by taking average of the target values of K nearest neighbors. The calculated average value becomes the predicted output for the target data point.

3.3.4. Support Vector Machine (SVM)

Support vector machines (SVMs) are supervised learning algorithms used in classification and regression tasks that are particularly effective in high-dimensional spaces and in cases where the number of dimensions exceeds the number of samples. They are designed to find the optimal threshold (or hyperplane) that separates different classes in a feature space [46]. The hyperplane represents the decision boundary that separates different classes. In case of two-dimensional space, the hyperplane is a line, while in case of higher dimensions, it becomes a plane or hyperplane. Support vectors represent the data points closest to the hyperplane. They are very important for determining the position and orientation of the hyperplane as the decision boundary is affected only by these support vectors. The margin represents the distance between the hyperplane and the closest data points of each class and the algorithm aims to maximize this margin, providing a clear separation between the classes. Statistical Learning Theory: Support vector machines (SVMs) are based on the structural risk minimization concept, which seeks to maximize the margin between support vectors, the most important data points for classification judgments, in order to determine the best separation hyperplane between classes. Convex Optimization: Convex quadratic programming issues must be solved in order to train SVMs, guaranteeing durable performance and global optimality. Sparsity: SVM solutions are effective for large-scale issues because they are usually sparse and only use a subset of the training data (support vectors). Broad Range of Applications: Among other fields, SVMs are employed in computer vision, bioinformatics, text classification, neuroimaging, and structural reliability analysis. Pattern Recognition: SVMs have demonstrated excellent performance in high-dimensional data settings, including brain imaging and precision psychiatry, and are particularly well-known in the field of pattern recognition.

Classification and Regression: SVMs can be used for both classification and regression applications; extensions and modifications have been created for big, imbalanced, and multi-class datasets. Support vector machines use kernel functions to deal with non-linearly separable data by transforming it into a higher-dimensional space where linear separation is possible. Common kernel functions include:

$$Linear:\ \ K(w,b) = w^T x + b \qquad (6)$$
$$Polynominal:\ \ K(w,b) = (\gamma w^T x + b)^N \qquad (7)$$
$$Gaussian\ RBF:\ \ K(w,b) = \exp(-\gamma \|x_i - x_j\|^n) \qquad (8)$$
$$Sigmoid:\ \ K(x_i, x_j) = \tanh(\alpha x_i^T x_j + b) \qquad (9)$$

Suppose we have a binary classification problem contains two classes (+1, -1). Our training dataset comprises input feature vectors denoted as X, alongside their corresponding class labels Y.
The linear hyperplane equation can be written as follows (61):

$$w^T + b = 0 \qquad (10)$$

The vector W symbolizes the normal vector to the hyperplane, which is the direction perpendicular to the hyperplane. Meanwhile, the parameter b in the equation signifies the offset or the distance of the hyperplane from the origin along the normal vector W. The distance between a data point xi and the decision boundary can be calculated as:

$$d_i = \frac{w^T + b}{\|w\|} \qquad (11)$$

where ||w|| represents the Euclidean norm of the weight vector w.
For Linear SVM classifier (29):

$$\hat{y} = \begin{cases} 1 & w^T x + b \geq 0 \\ 0 & w^T x + b < 0 \end{cases} \qquad (12)$$

Optimization of Hard margin linear SVM classifier (29):

$$Minimize\, \frac{1}{2} w^T w = Minimize\ \frac{1}{2} \|w\|^2 \qquad (13)$$

Where:

$$y_i(w^T x_i + b) \geq 1 \quad For\ i = 1,2, \dots, m$$

Optimization of Soft margin linear SVM classifier (29):

$$Minimize\ \frac{1}{2} w^T w + C \sum_{i=1}^{m} \xi_i \qquad (14)$$

Where:

$$y_i(w^T x_i + b) \geq 1 - \xi_i$$
$$\xi_i \geq 0 \quad For\ i = 1,2, \dots, m$$

3.4. Performance Evaluation

We have employed confusion matrices for the performance evaluation of each classification algorithm output, which gave an all-rounded insight into the accuracy of each methodology. From these, the overall accuracy, producer's accuracy, user's accuracy, and the Kappa coefficient of each land cover class were derived.
Within the realm of machine learning, the confusion matrix (CM) which referred to an error matrix is a table layout designed to represent the performance of a model, often one employed in supervised learning. The architecture of the CM is illustrated in Figure 2.

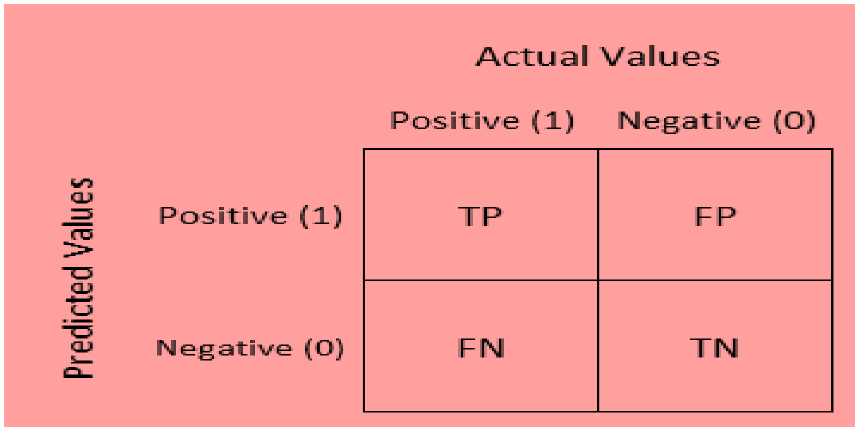


Figure 2. Confusion matrix architecture

Where:

- TP (True Positive): The model classifies the sample as positive and its classification is correct.
- TN (True Negative): The model classifies the sample as negative and its classification is correct.
- FP (False Positive): The model classifies the sample as positive and its classification is wrong.
- FN (False Negative): The model classifies the sample as negative and its classification is wrong.

Accuracy represents the proportion of correct predictions of the model over all predictions (64). Its equation is given as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (15)$$

4. Results and Discussion

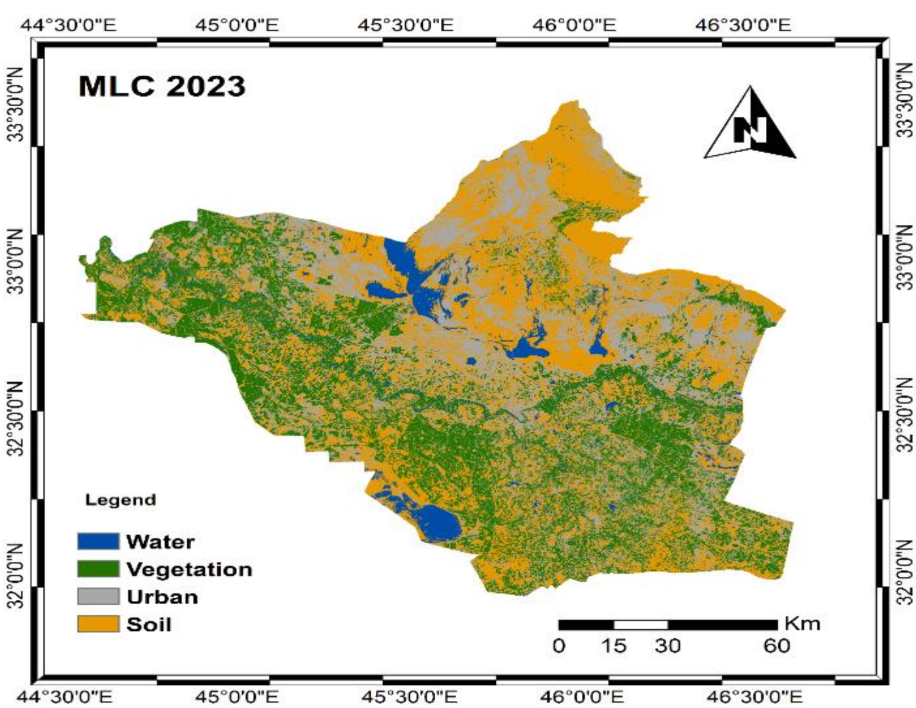


Figure 3. MLC for the year 2023.

Table 2: Confusion Matrix for MLC 2023.

	Water	Vegetation	Urban	Soil	Ground truth
Water	41	0	0	0	41
Vegetation	0	40	1	1	42
Urban	0	0	30	9	39
Soil	0	0	9	30	39
Total	41	40	40	40	161
Overall= 0.875776					

The confusion matrix provides a snapshot of the algorithm's performance.
The diagonal elements represent the correct predictions for each category. For example, the element in the first row and first column (41) indicates 41 correct predictions for water areas.
The off-diagonal elements represent the incorrect classifications. For example, the element in the fourth row and third column (9) indicates that 9 areas were classified as Urban when they were actually Soil.

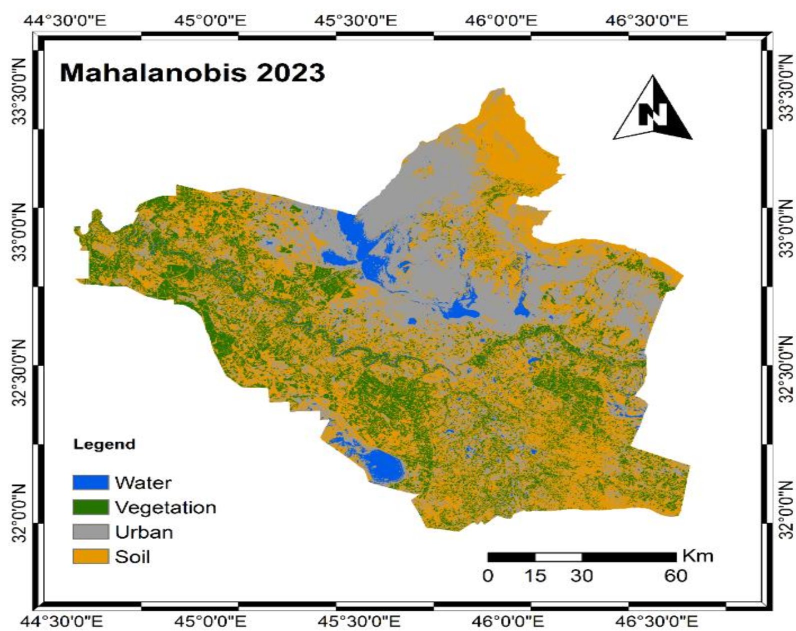


Figure 4. Mahalanobis Distance for the year 2023.

Table 3: Confusion Matrix for Mahala Nobis Distance 2023.

	Water	Vegetation	Urban	Soil	Ground truth
Water	41	0	0	0	41
Vegetation	0	40	0	0	40
Urban	0	0	29	19	48
Soil	0	0	11	21	32
Total	41	40	40	40	161
Overall= 0.813665					

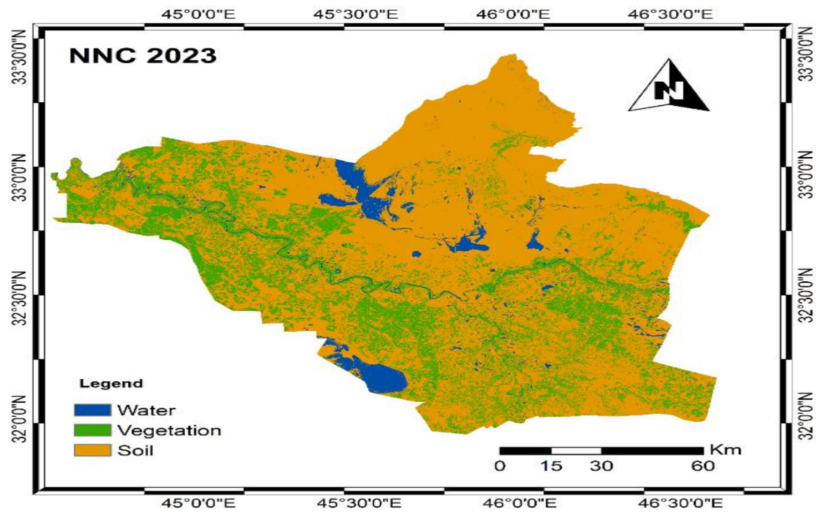


Figure 5: NNC for the year 2023.

Table 4: Confusion Matrix for NNC 2023.

	Water	Vegetation	Urban	Soil	Ground truth
Water	41	0	0	0	41
Vegetation	0	40	0	0	40
Urban	0	0	0	0	40
Soil	0	0	0	40	40
Total	41	40	0	41	160
Overall= 0.751553					

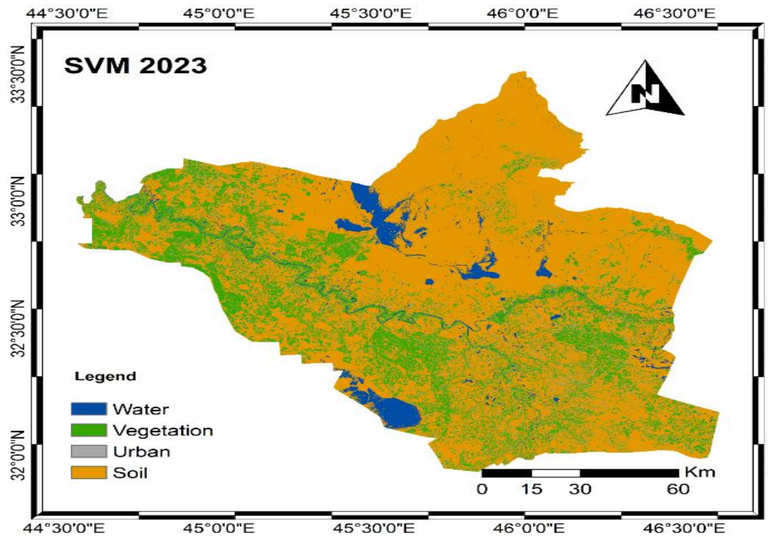


Figure 6: SVM for the year 2023.

Table 5: Confusion matrix for SVM 2023.

	Water	Vegetation	Urban	Soil	Ground truth
Water	41	0	0	0	41
Vegetation	0	40	0	0	40
Urban	0	0	1	0	1
Soil	0	0	39	40	79
Total	41	40	40	40	161
Overall= 0.757764					

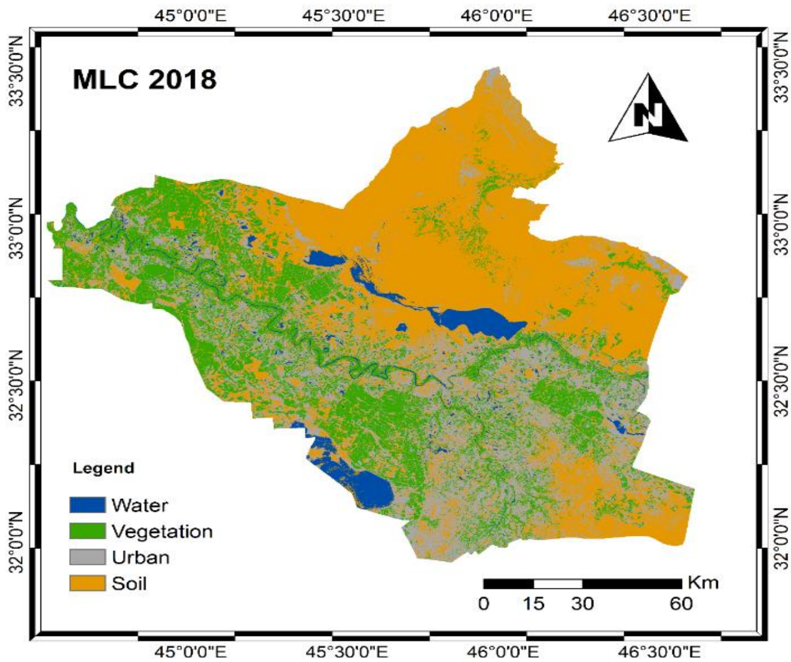


Figure 7: MLC for the year 2018.

Table 6: Confusion matrix for MLC 2018.

	Water	Vegetation	Urban	Soil	Ground truth
Water	40	0	0	0	40
Vegetation	0	40	2	1	43
Urban	0	0	33	11	44
Soil	0	0	5	28	33
Total	40	40	40	40	160
		Overall= 0.88125			

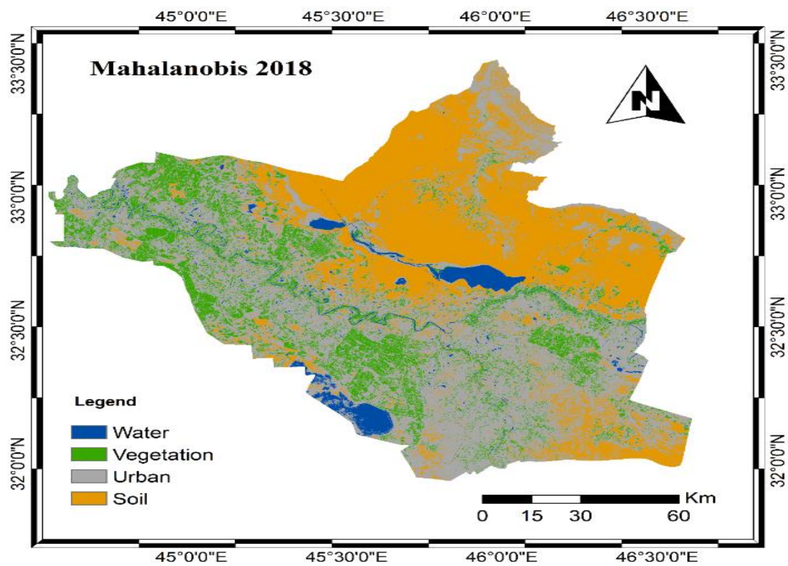


Figure 8: Mahalanobis Distance for the year 2018.

Table 7: Confusion matrix For Mahalanobis Distance 2018.

	Water	**Vegetation**	**Urban**	**Soil**	**Ground truth**
Water	40	0	0	0	40
Vegetation	0	40	0	0	40
Urban	0	0	37	17	54
Soil	0	0	3	23	26
Total	40	40	40	40	160
			Overall=0.875		

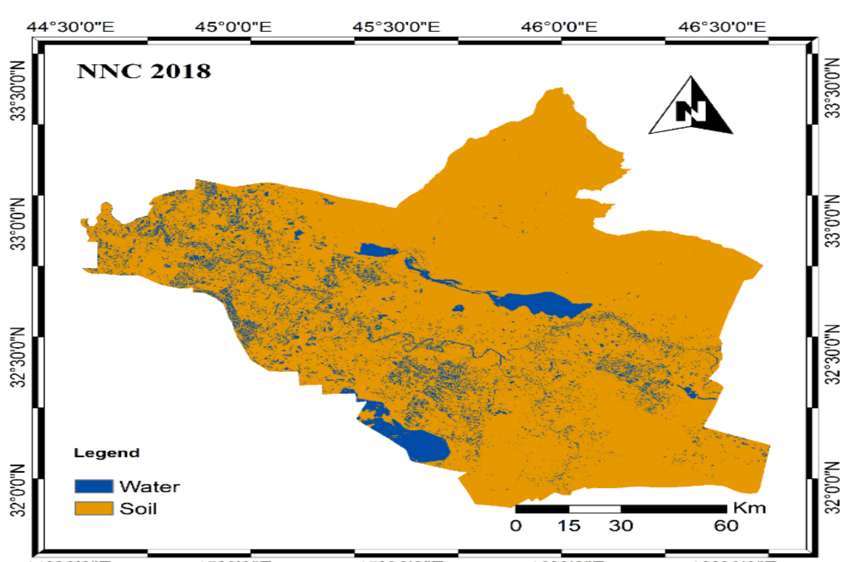


Figure 9: NNC for the year 2018.

Table 8: Confusion matrix for NNC 2018.

	Water	Vegetation	Urban	Soil	Ground truth
Water	40	0	0	0	40
Vegetation	40	0	0	0	40
Urban	22	0	0	18	40
Soil	0	0	0	40	40
Total	102	0	0	58	160
Overall=0.5					

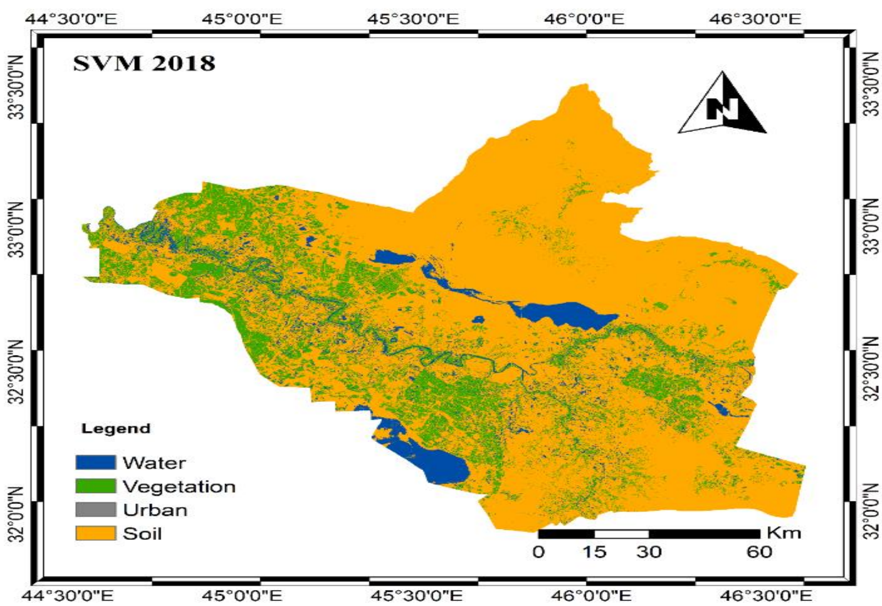


Figure 10: SVM for the year 2018.

Table 9: Confusion Matrix for SVM 2018.

	Water	Vegetation	Urban	Soil	Ground truth
Water	40	1	0	0	41
Vegetation	0	39	0	0	39
Urban	0	0	0	0	0
Soil	0	0	40	40	80
Total	40	40	40	40	160
Overall= 0.74375					

From our analysis, there was a significant change in the land cover in Wasit Province between 2018 and 2023. There was an increase in urban structures, which can be argued to emanate from the increasing urbanization and population growth currently facing the region. Agricultural land took a nosedive, probably because of sprawl growth leading to land conversion into different uses. Vegetation covers therefore had both degrading and reforestation activities in various regions.

Algorithm performance comparison: The four algorithms' performance varied from good to excellent under the different land cover classes and the two periods. Maximum Likelihood Classification (MLC): In both years, MLC provided high overall accuracy, especially for urban and water classes, while it cannot present vegetation in most cases because of its overlapped spectral signature with other classes.
2. Mahalanobis Distance: This method performed well in outlier detection, which proved to be good in distinguishing urban structures from agricultural land. This was less consistent in the case of classes of vegetation where significant spectral overlap occurs.
3. Nearest Neighbor Classification: In the case of NNC good accuracy was observed in the case of both urban and water classes due to their distinct spatial pattern of these land covers. However, it performance was poor in cases where the area consists of mixed vegetation cover, as the neighboring pixel may belong to any different class.
4. Support Vector Machines: Generally, it outperformed all other classes with very high overall accuracy for all land cover classes, especially in discriminating between urban and agriculture. It showed strength in high-dimensional data and was unbothered by the complexity in the landscape in Wasit Province.

5. Conclusion and future works

Land cover maps play a pivotal role in natural resource management and environmental planning, especially with the ongoing environmental change resulting from urban expansion, agricultural expansion, and climate fluctuations, the ability to accurately classify and monitor land cover types has become of great importance to environmental researchers and environmental managers. This work embraces changes in land cover in Wasit Province between 2018 and 2023, from which crucial information on the dynamics of urbanization, decline in agriculture, and vegetation changes will emerge. In this paper, a comparison was made between different classification procedures for land cover data between 2018 and 2023. The techniques studied include maximum likelihood classification (MLC), Mahalanobis distance, neighbor-neighbor classification (NNC), and support vector machines (SVM). Confusion matrices provided a measure of performance in terms of classification biases for water, vegetation, urban structures, soil, etc. for each of the methods. The MLC technique achieved a classification accuracy of 88% for images captured in 2018 and 87% for 2023, outperforming other techniques, as the Mahalanobis distance achieved a classification accuracy of 87% for images captured in 2018 and 81% for 2023, while the nearest neighbor algorithm achieved a classification accuracy of 50% for images captured in 2018 and 75% for 2023, and the SVM algorithm achieved a classification accuracy of 74% for images captured in 2018 and 75% for 2023.
From our analysis, there was a significant change in the land cover in Wasit Province between 2018 and 2023. There was an increase in urban structures, which can be argued to emanate from the increasing urbanization and population growth currently facing the region. Agricultural land took a nosedive, probably because of sprawl growth leading to land conversion into different uses. Vegetation covers therefore had both degrading and reforestation activities in various regions.

Future research could focus on improving classification accuracy by incorporating advanced models such as deep learning-based convolutional neural networks or hybrid models that combine the strengths of multiple classifiers. Additionally, incorporating auxiliary data such as topographic information, soil types, or meteorological data could further improve classification results. Finally, conducting a more in-depth analysis of the reasons for differences in performance across years, especially for the Mahalanobis distance and NNC algorithms, could help improve their use in land cover mapping tasks.

Policy Implications

Results from this study will be very helpful to local policy makers and urban planners. By understanding the trends of change in land cover, they will be able to make relevant decisions with respect to sustainable development, resource management, and environmental conservation. For example, from this increase in urban structures arises a need for proper urban planning, balancing population growth without intrusion on agricultural land and natural habitats. The mixed vegetation cover outcome implies that only focused reforestation and conservation might help enhance ecological resilience against continued development pressures.

In summary, the insights from this study not only help increase academic knowledge in the field of land cover dynamics, but also ensure that at least actionable information is delivered to the local governance to develop sustainable development strategies in Wasit Province.

Corresponding author

Yasir Abdulameer Nayyef Aldabbagh
yasir.aldabbagh@mu.edu.iq

Acknowledgements
We would like to acknowledge the support of local government and institutions in facilitating much of the data and inputs in this study. Collaboration was indeed very crucial in conducting this study; hence, validating it at the community level.

Funding
No funding.

Contributions
Y.A.N.A; A.I.Z.A; Conceptualization, Y.A.N.A; A.I.Z.A; Investigation, Y.A.N.A; A.I.Z.A; Writing (Original Draft), Y.A.N.A; A.I.Z.A; Writing (Review and Editing) Supervision, Y.A.N.A; A.I.Z.A; Project Administration.

Ethics declarations
This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication
Not applicable.

Competing interests
All authors declare no competing interests.

References

[1] Aboelnour, M., & Engel, B. A. (2018). Application of remote sensing techniques and geographic information systems to analyze land surface temperature in response to land use/land cover change in Greater Cairo Region, Egypt. Journal of Geographic Information System, 10(1), 57-88.
[2] Talukdar, S., Singha, P., Mahato, S., Pal, S., Liou, Y. A., & Rahman, A. (2020). Land-use land-cover classification by machine learning classifiers for satellite observations—A review. Remote sensing, 12(7), 1135.
[3] Shivakumar, B. R., & Rajashekararadhya, S. V. (2018). Investigation on land cover mapping capability of maximum likelihood classifier: a case study on North Canara, India. Procedia computer science, 143, 579-586.
[4] Polat, N., & Kaya, Y. (2021). Investigation of the performance of different pixel-based classification methods in land use/land cover (LULC) determination. Türkiye İnsansız Hava Araçları Dergisi, 3(1), 1-6.
[5] Ramezan, C. A., Warner, T. A., Maxwell, A. E., & Price, B. S. (2021). Effects of training set size on supervised machine-learning land-cover classification of large-area high-resolution remotely sensed data. Remote Sensing, 13(3), 368.
[6] Adugna, T., Xu, W., & Fan, J. (2022). Comparison of random forest and support vector machine classifiers for regional land cover mapping using coarse resolution FY-3C images. Remote Sensing, 14(3), 574.
[7] Xie, Z., Chen, Y., Lu, D., Li, G., & Chen, E. (2019). Classification of Land Cover, Forest, and Tree Species Classes with ZiYuan-3 Multispectral and Stereo Data. Remote. Sens., 11, 164.
[8] Mahdianpari, M., Salehi, B., Rezaee, M., Mohammadimanesh, F., & Zhang, Y. (2018). Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. Remote Sensing, 10(7), 1119.
[9] Song, H., Kim, Y., & Kim, Y. (2019). A patch-based light convolutional neural network for land-cover mapping using Landsat-8 images. Remote Sensing, 11(2), 114.
[10] Dong, S., Zhuang, Y., Yang, Z., Pang, L., Chen, H., & Long, T. (2019). Land Cover Classification from VHR Optical Remote Sensing Images by Feature Ensemble Deep Learning Network. IEEE Geoscience and Remote Sensing Letters.
[11] Carrasco, L., O'Neil, A. W., Morton, R. D., & Rowland, C. S. (2019). Evaluating combinations of temporally aggregated Sentinel-1, Sentinel-2 and Landsat 8 for land cover mapping with Google Earth Engine.Remote Sensing, 11(3), 288.
[12] Wu, W., Zucca, C., Karam, F., & Liu, G. (2016). Enhancing the performance of regional land cover mapping. Int. J. Appl. Earth Obs. Geoinformation, 52, 422-432.
[13] Wu, W., Zucca, C., Karam, F., & Liu, G. (2016). Enhancing the performance of regional land cover mapping. Int. J. Appl. Earth Obs. Geoinformation, 52, 422-432. https://doi.org/10.1016/J.JAG.2016.07.014 .
[14] Saah, D., Tenneson, K., Poortinga, A., Nguyen, Q., Chishtie, F., Aung, K., Markert, K., Clinton, N., Anderson, E., Cutter, P., Goldstein, J., Housman, I., Bhandari, B., Potapov, P., Matin, M., Uddin, K., Pham, H., Khanal, N., & Ganz, D. (2020). Primitives as building blocks for constructing land cover maps. Int. J. Appl. Earth Obs. Geoinformation, 85. https://doi.org/10.1016/j.jag.2019.101979 .

[15] Zhang, X., Liu, L., Chen, X., Xie, S., & Gao, Y. (2019). Fine Land-Cover Mapping in China Using Landsat Datacube and an Operational SPECLib-Based Approach. Remote. Sens., 11, 1056. https://doi.org/10.3390/RS11091056 .
[16] Tong, X., Xia, G., & Zhu, X. (2022). Enabling country-scale land cover mapping with meter-resolution satellite imagery. Isprs Journal of Photogrammetry and Remote Sensing, 196, 178 - 196. https://doi.org/10.1016/j.isprsjprs.2022.12.011 .
[17] Preidl, S., Lange, M., & Doktor, D. (2020). Introducing APiC for regionalised land cover mapping on the national scale using Sentinel-2A imagery. Remote Sensing of Environment, 240, 111673. https://doi.org/10.1016/j.rse.2020.111673 .
[18] Elmahdy, S., & Mohamed, M. (2023). Regional mapping and monitoring land use/land cover changes: a modified approach using an ensemble machine learning and multitemporal Landsat data. Geocarto International, 38. https://doi.org/10.1080/10106049.2023.2184500.
[19] Bartsch, A., Höfler, A., Kroisleitner, C., & Trofaier, A. (2016). Land Cover Mapping in Northern High Latitude Permafrost Regions with Satellite Data: Achievements and Remaining Challenges. Remote. Sens., 8, 979. https://doi.org/10.3390/RS8120979 .
[20] Ejikeme, J., Emengini, J., Ugwu, E., & Umenweke, D. (2019). A Comparative Evaluation of Different Techniques of Supervised Classification in Landuse/Landcover Mapping of Awka South L.G.A, Anambra State, Nigeria.. Journal of Environment and Earth Science.
[21] Ghayour, L., Neshat, A., Paryani, S., Shahabi, H., Shirzadi, A., Chen, W., Al-Ansari, N., Geertsema, M., Amiri, M., Gholamnia, M., Dou, J., & Ahmad, A. (2021). Performance Evaluation of Sentinel-2 and Landsat 8 OLI Data for Land Cover/Use Classification Using a Comparison between Machine Learning Algorithms. Remote. Sens., 13, 1349.
[22] Otukei, J., & Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. Int. J. Appl. Earth Obs. Geoinformation, 12, S27-S31. https://doi.org/10.1016/J.JAG.2009.11.002 .
[23] Talukdar, S., Singha, P., Mahato, S., , S., Pal, S., Liou, Y., & Rahman, A. (2020). Land-Use Land-Cover Classification by Machine Learning Classifiers for Satellite Observations - A Review. Remote. Sens., 12, 1135.
[24] Ahmad, A., & Quegan, S. (2012, November). Analysis of maximum likelihood classification technique on Landsat 5 TM satellite data of tropical land covers. In 2012 IEEE international conference on control system, computing and engineering (pp. 280-285). IEEE.
[25] Jain, A. K., Duin, R. P. W., & Mao, J. (2000). Statistical pattern recognition: A review. IEEE Transactions on pattern analysis and machine intelligence, 22(1), 4-37.
[26] Alimjan, G., Sun, T., Jumahun, H., Guan, Y., Zhou, W., & Sun, H. (2017). A hybrid classification approach based on support vector machine and k-nearest neighbor for remote sensing data. International Journal of Pattern Recognition and Artificial Intelligence, 31(10), 1750034.
[27] Saadatfar, H., Khosravi, S., Joloudari, J. H., Mosavi, A., & Shamshirband, S. (2020). A new K-nearest neighbors classifier for big data based on efficient data pruning. Mathematics, 8(2), 286.
[28] Wu, G., Zhao, Z., Fu, G., Wang, H., Wang, Y., Wang, Z., ... & Huang, L. (2019, June). A Fast k NN-Based Approach for Time Sensitive Anomaly Detection over Data Streams. In International conference on computational science (pp. 59-74). Cham: Springer International Publishing.
[29] Elkan, C. (2007). Nearest Neighbor Classification. . https://doi.org/10.1007/springerreference_63665.
[30] Goldstein, M. (1972). Kn -nearest Neighbor Classification. IEEE Trans. Inf. Theory, 18, 627-630. https://doi.org/10.1109/TIT.1972.1054888.
[31] Hall, P., Park, B., & Samworth, R. (2008). Choice of neighbor order in nearest-neighbor classification. Annals of Statistics, 36, 2135-2152. https://doi.org/10.1214/07-AOS537.
[32] Short, R., & Fukunaga, K. (1981). The optimal distance measure for nearest neighbor classification. IEEE Trans. Inf. Theory, 27, 622-626. https://doi.org/10.1109/TIT.1981.1056403.
[33] Hastie, T., & Tibshirani, R. (1995). Discriminant Adaptive Nearest Neighbor Classification. , 409-415. https://doi.org/10.1109/34.506411.
[34] Domeniconi, C., Peng, J., & Gunopulos, D. (2002). Locally Adaptive Metric Nearest-Neighbor Classification. IEEE Trans. Pattern Anal. Mach. Intell., 24, 1281-1285. https://doi.org/10.1109/TPAMI.2002.1033219 .
[35] Gou, J., Zhan, Y., Rao, Y., Shen, X., Wang, X., & He, W. (2014). Improved pseudo nearest neighbor classification. Knowl. Based Syst., 70, 361-375. https://doi.org/10.1016/j.knosys.2014.07.020.
[36] Dasgupta, S., & So, G. (2023). Online nearest neighbor classification. ArXiv, abs/2307.01170. https://doi.org/10.48550/arXiv.2307.01170.
[37] Barber, D. (2011). Nearest neighbour classification. , 322-328. https://doi.org/10.1017/CBO9780511804779.019
[38] Dasgupta, S., & Kpotufe, S. (2020). Nearest Neighbor Classification and Search. , 403-423. https://doi.org/10.1017/9781108637435.024.
[39] Hastie, T., & Tibshirani, R. (1995). Discriminant Adaptive Nearest Neighbor Classification. , 409-415. https://doi.org/10.1109/34.506411.
[40] Gou, J., Zhan, Y., Rao, Y., Shen, X., Wang, X., & He, W. (2014). Improved pseudo nearest neighbor classification. Knowl. Based Syst., 70, 361-375. https://doi.org/10.1016/j.knosys.2014.07.020.
[41] Domeniconi, C., Peng, J., & Gunopulos, D. (2002). Locally Adaptive Metric Nearest-Neighbor Classification. IEEE Trans. Pattern Anal. Mach. Intell., 24, 1281-1285.
[42] Rahmawan, F., , H., Mega, W., & Dhuita, P. (2025). Comparison of The Accuracy of K-Nearest Neighbor and Roberta Algorithm in Analysis of Sentiment on Miawaug Youtube Channel Comments. Teknika. https://doi.org/10.34148/teknika.v14i1.1117 .

[43] Xu, Q. (2025). Application of an Intelligent English Text Classification Model with Improved KNN Algorithm in the Context of Big Data in Libraries. Systems and Soft Computing. https://doi.org/10.1016/j.sasc.2025.200186 .
[44] Faradila, Z., Homaidi, A., & Prasetyo, J. (2025). Classification of Anaemia Status Using The K-Nearest Neighbor Algorithm. G-Tech: Jurnal Teknologi Terapan. https://doi.org/10.70609/gtech.v9i1.6377.
[45] Hu, L. Y., Huang, M. W., Ke, S. W., & Tsai, C. F. (2016). The distance function effect on k-nearest neighbor classification for medical datasets. SpringerPlus, 5, 1-9.
[46] Kok, Z. H., Shariff, A. R. M., Alfatni, M. S. M., & Khairunniza-Bejo, S. (2021). Support vector machine in precision agriculture: a review. Computers and Electronics in Agriculture, 191, 106546.