Recibido: 23/12/2022
Aceptado: 23/05/2023

REVISTA DE TELEDETECCIÓN
Asociación Española de Teledetección
(2023) 62, 1-19
ISSN 1133-0953
EISSN 1988-8740
https://doi.org/10.4995/raet.2023.19014

Clasificación de uso y cobertura del suelo a través de algoritmos de aprendizaje automático: revisión bibliográfica

René Tobar-Díaz*[1], Yan Gao[1], Jean Francois Mas[1], Víctor Hugo Cambrón-Sandoval[2]

[1]*Centro de Investigaciones en Geografía Ambiental, Universidad Nacional Autónoma de México, Antigua Carretera a Pátzcuaro No. 8701, Col. Ex-Hacienda de San José de la Huerta. C.P. 58190, Morelia, México.*
[2]*Facultad de Ciencias Naturales, Universidad Autónoma de Querétaro. Av. de las Ciencias s/n, Nuevo Juriquilla, 76230 Juriquilla, Querétaro, México.*

Resumen: Los métodos para la clasificación de uso y cobertura del suelo (UCS) han mostrado avances importantes en los últimos años, como la incorporación de las técnicas de aprendizaje automático (*machine learning*-ML) que han ganado popularidad y aceptación por sus resultados. Sin embargo, la falta de consensos metodológicos ha provocado una aplicación desordenada de los métodos ML en la clasificación de UCS. Por lo que a través de la revisión bibliográfica practicada se identificaron puntos de la forma en que se están implementando los métodos, así como posibles implicaciones en la clasificación de UCS al darse de esta manera. Para dicha revisión se utilizaron únicamente artículos científicos publicados entre el año 2010 al 2020 y que consideraran los siguientes algoritmos para la clasificación de UCS: k vecinos más cercanos *(K-nearest neighbor*-KNN), bosque aleatorio (*random forest*-RF), máquina de soporte de vectores (*support vector machine*-SVM), redes neuronales artificiales (*artificial neural network*-ANN) y árboles de decisión (*decision trees*-DT). A través de los resultados obtenidos en la revisión bibliográfica, se reafirma el potencial de los algoritmos y se identifican puntos de mejora para la aplicación de ML en la clasificación de UCS, especialmente en la integración de los conjuntos de datos, la parametrización de los algoritmos y la evaluación de los resultados, generando a su vez una selección de buenas prácticas a partir de las recomendaciones de diversos autores las cuales consideramos serán de utilidad para usuarios interesados en estos métodos.

Palabras clave: aprendizaje automático, uso del suelo, cobertura del suelo, bosque aleatorio, máquina de soporte de vectores, redes neuronales artificiales, árboles de decisión.

Classification of land use and land cover through machine learning algorithms: a literature review

Abstract: Methodologies for land use and land cover (LULC) classification have demonstrated significant advances in recent years, such as the incorporation of machine learning (ML) classification techniques, which have gained popularity and acceptance due to their good performance. However, the lack of methodological consensus has led to a disorderly application of ML methods in the classification of LULC. Through the literature review, we identified some points in how the methods are being implemented for the classification of LULC. For this review, only scientific articles published between 2010 and 2020 were analyzed that incorporated the following algorithms for LULC classification: K-nearest neighbor (KNN), random forest (RF), support vector machine (SVM), artificial neural network (ANN) and decision trees (DT). Using the results of the literature review, we were able to confirm the potential of the algorithms. We also identified areas for improvement in the application of ML to the classification of LULC. These areas include the integration of data sets, parameterization of algorithms, and evaluation of results. Consequently, we generated a selection of guidelines based on the recommendations of various authors that we consider will be useful for users interested in these methods.

Key words: machine learning, land use, land cover, random forest, support vector machine, artificial neural network, decision trees.

To cite this article: Tobar-Díaz, R., Gao, Y., Mas, J.F., Cambrón-Sandoval, V.H. 2023. Classification of land use and land cover through machine learning algorithms: a literature review. *Revista de Teledetección*, *62*, 1-19. https://doi.org/10.4995/raet.2023.19014

* *Corresponding author: rtobar@pmip.unam.mx*

1. Introducción

El uso y cobertura del suelo (UCS) es considerada información importante para fenómenos ambientales y territoriales (Herold *et al.*, 2006), ya que permite analizar los procesos naturales y antrópicos de la superficie terrestre y los problemas ambientales asociados (Koomen y Stillwell, 2007). Por ello, la mejora de las técnicas para el análisis del UCS ha permitido incrementar la calidad y el manejo de grandes cantidades de información. Dentro de estos avances, métodos como aprendizaje automático (ML) han ganado popularidad desde hace más de una década (Dong *et al.*, 2019) frente a métodos tradicionales como máxima verosimilitud y distancia mínima (Szuster *et al.*, 2011; Demirkan *et al.*, 2020), sobre todo en la clasificación de UCS donde su popularidad se ha afianzado (Abdi, 2020), identificándose así un grupo de clasificadores considerados maduros por haber sido puestos a prueba ampliamente y demostrado su capacidad en la clasificación de UCS, siendo estos k vecinos más cercanos (*K-nearest neighbor*-KNN), bosque aleatorio (*random forest*-RF), máquina de soporte de vectores (*support vector machine*-SVM), redes neuronales artificiales (*artificial neural network*-ANN) y árboles de decisión (*decision trees*-DT) (Maxwell *et al.*, 2018; Ge *et al.*, 2020). A pesar de esto, Yu *et al.* (2014) demostraron en un metaanálisis de 1651 artículos sobre la generación de cartografía de cobertura del suelo que, aunque los clasificadores ML han demostrado mejores resultados, 32,3% de las investigaciones aún prefieren métodos considerados tradicionales como máxima verosimilitud, pudiendo esto ser atribuido a múltiples factores como la incipiente integración de ML a los paquetes de software especializados, la dificultad de ajustar los parámetros de optimización y que el comportamiento y resultados obtenidos puede variar en función de las condiciones particulares del área estudiada (Talukdar *et al.*, 2020). Esta revisión busca documentar el uso de los algoritmos ML, KNN, RF, SVM, ANN y DT, en la clasificación de UCS en un periodo de diez años (2010-2020), enfocándose en las siguientes preguntas: 1) ¿cómo se ha implementado ML en la clasificación de UCS en los aspectos de revisión de conjunto de datos e implementación de algoritmos; 2) ¿Qué factores metodológicos, tales como la compilación de datos, implementación de métodos, y la evaluación de resultados, afectan la implementación de ML en la clasificación de UCS?

2. Aprendizaje automático

ML es un área del conocimiento que se puede encontrar entre la estadística, la inteligencia artificial y la informática (Müller y Guido, 2016), es una subárea de la Inteligencia Artificial que se enfoca en el desarrollo de algoritmos y sistemas que pueden aprender de ejemplos para hacer predicciones o tomar decisiones utilizando teoría estadística para la construcción de modelos matemáticos, los cuales pueden ser de carácter predictivo o descriptivo (Alpaydin, 2014; Marsland, 2014). ML puede valerse de diferentes algoritmos, los cuales pueden ser paramétricos o no paramétricos, siendo estos últimos de particular interés ya que a esa categoría pertenecen los algoritmos analizados, se caracterizan por no asumir ninguna distribución específica de los datos y en su lugar se basan en la estructura subyacente para realizar la predicción. La selección específica del algoritmo puede darse en función de factores como las características de los datos o un objetivo, permitiendo optar por el que mejor se ajuste y se considere pueda generar los mejores resultados (Wilson y Keil, 1999). Se han identificado cuatro pasos para la aplicación de ML en la clasificación de UCS: la recopilación de datos para entrenamiento, evaluación y validación, la selección y optimización del algoritmo, la aplicación de la clasificación y la evaluación de la exactitud (Shih *et al.*, 2019).

2.1. Recopilación de datos de entrenamiento, evaluación y validación

Determinar la cantidad de datos puede depender de factores asociados a la naturaleza del estudio, como el sistema de clasificación de coberturas, la extensión de las clases, el nivel de importancia que puedan llegar a tener unas sobre otras e incluso factores económicos (Congalton y Green, 2019). Si bien se debe procurar que los conjuntos de datos sean amplios con una buena diversidad, calidad y balance, esto no suele ser la regla en la práctica, pero considerarlo permitirá que el algoritmo logre una mejor generalización y por ende la reducción de problemas asociados a la naturaleza de los datos (Müller y Guido, 2016). Por ello, un

diseño de muestreo adecuado no solo permitirá obtener datos suficientemente representativos y equilibrados de las clases estudiadas que faciliten la conformación de conjuntos independientes sino también una buena separabilidad que posibilite a los algoritmos de aprendizaje a identificar patrones y diferencias entre las diferentes coberturas del suelo con mayor efectividad, considerando que la separación entre clases no depende solo de la distancia entre los promedios de las distribuciones de probabilidad de las clases en el espacio, sino también de la forma en que se distribuye esa probabilidad en ese espacio, pudiendo hacerlo con métodos como, divergencia, divergencia transformada, la distancia de Jefferies-Matusita y la distancia de Bhattacharyya (Thomas *et al.*, 1987; Alpaydin, 2014; Olofsson *et al.*, 2014).

Los conjuntos de datos se distribuyen en subconjuntos de entrenamiento, validación y evaluación, pudiendo ser divididos los datos a criterio del usuario o utilizando una distribución en función de la cantidad, si esta es limitada lo recomendable es una distribución en porcentajes de, 60 (entrenamiento); 20 (validación); 20 (evaluación) y si la cantidad es amplia 50;25;25, la definición de un conjunto de datos "grande" en el aprendizaje automático, incluida la teledetección, puede variar, ya que el tamaño óptimo dependerá de las características específicas del problema, del rendimiento deseado y de la capacidad de procesamiento permitida (Alpaydin, 2014; Marsland, 2014).

Los datos de entrenamiento son utilizados para construir el modelo ML que brinda una primera aproximación del comportamiento del algoritmo (Alpaydin, 2014). El subconjunto de validación es utilizado para probar la capacidad de generalización del algoritmo e identificar si la selección de parámetros puede mejorar (Knox, 2018). Los de evaluación son empleados para determinar la bonanza del modelo (Müller y Guido, 2016). Ante la probabilidad de que los datos puedan ser insuficientes, el uso de técnicas como validación cruzada o *bootstrapping* se puede hacer necesario (Alpaydin, 2014; Kelleher *et al.*, 2015). La validación cruzada es considerada un método más estable y exhaustivo que la división tradicional de los datos en subconjuntos (entrenamiento, validación y prueba) ya que los datos se dividen repetidamente, permitiendo así entrenar múltiples modelos.

La versión más utilizada es la validación cruzada de *k*-iteraciones, donde *k* representa el número de particiones que se realizarán. A nivel funcional, los datos se particionan en *k* subconjuntos de aproximadamente el mismo tamaño, de los cuales se tomará el primero para prueba y los restantes para entrenamiento, construyendo así un primer modelo con estos y evaluándose con el primero (de prueba), para luego repetir el proceso sustituyendo el conjunto de datos de prueba por el subsecuente, es decir el segundo y luego tomar los restantes conjuntos para entrenar, haciendo esto consecutivamente hasta que se haya hecho con el total de los grupos de datos (Müller y Guido, 2016).

Bootstrapping es una alternativa a la validación cruzada cuando los conjuntos de datos son muy pequeños, ya que permite un mayor traslape haciendo que sus estimaciones posean mayor dependencia. Funciona con base en un conjunto de entrenamiento *x* de *n* tamaño a partir del cual se extrae una cantidad *n* de muestras al azar sustituidas en el conjunto *x*, por este motivo es posible que algunas sean extraídas más de una vez y que otras no se consideren en absoluto (Alpaydin, 2014).

2.2. Selección y parametrización de algoritmos

La selección de un algoritmo para la clasificación de UCS como tal no responde a un procedimiento en particular. Se recomienda la experimentación con diferentes clasificadores y la evaluación de los resultados de exactitud del usuario y el productor por clases individuales, ya que puede haber clases raras que no tengan impacto en la exactitud general, pero ser claves la utilidad de la clasificación y así lograr escoger el algoritmo que mejor se ajuste a las necesidades (Maxwell *et al.*, 2018).

Los parámetros de ajuste definen el nivel de complejidad de un algoritmo, pudiendo ser simples de ajustar como en el caso de K-NN o mucho más complejos como ANN (Marsland, 2014; Alpaydin, 2014; Müller y Guido, 2016). En la parametrización, la experimentación es una alternativa importante para determinar los ajustes óptimos, sin embargo, también se encuentra el método de diseño factorial, coloquialmente llamado búsqueda en rejilla (*grid search*), el cual es una estrategia de ayuda en la selección de parámetros que permite hallar la combinación que mejor se pueda adaptar,

sustituyendo la manera manual (Alpaydin, 2014). No obstante, la relevancia de los ajustes no se circunscribe únicamente al resultado final, ya que si bien un algoritmo ML puede obtener resultados satisfactorios. Esto no quiere decir que el mismo pueda llegar a ser capaz de clasificar nueva información a partir del aprendizaje dado y obtener así una capacidad de generalización acertada, por lo que una selección de parámetros incorrecta puede ocasionar problemas en la complejidad del modelo al ajustarse muy bien a las particularidades de los datos de entrenamiento, pero no ser capaz de clasificar nueva información, fenómeno conocido como sobre-ajuste (*overfitting*) (Figura 1), o por sub-ajuste (*underfitting*), que es un ajuste demasiado simple y genera predicciones erróneas (Shalev-Shwartz y Ben-David 2014; Kelleher *et al.*, 2015; Géron, 2019).

Figura 1. Fenómenos de aprendizaje en aprendizaje automático.

2.3. Evaluación de exactitud

La estimación de la exactitud brinda información sobre la concordancia entre la clasificación elaborada por el clasificador respecto a la información de referencia obtenida por el usuario, proporcionando datos del rendimiento del clasificador, siendo el método más común para esta estimación la matriz de confusión (Tabla 1) (Tso y Mather, 2009). De esta matriz se obtienen métricas como la exactitud general, de usuario y productor, los cuales se relacionan con los errores de omisión y comisión (Campbell y Wynne, 2011).

La tabla 1 representa un ejemplo de la matriz de confusión, se compone de un número *q* de clases representadas en *i* columnas que son los datos de la clasificación y en las filas *j* qué son los valores de referencia, en el caso de las clases coincidentes $n_{11}, n_{22} \ldots n_{qq}$ representan los datos correctamente clasificados y para las no coincidentes

Tabla 1. Matriz de confusión.

Clases	1	2	…	*j*	…	*q*	Sum n_{i+}	Prop. Área
1	n_{11}	n_{12}	…	n_{1j}	…	n_{1q}	n_{1j}	π_1
2	n_{21}	n_{22}	…	n_{2j}	…	n_{2q}	n_{2+}	π_2
⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮
i	n_{i1}	n_{i2}		n_{ij}		n_{iq}	n_{i+}	π_i
⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮
q	n_{q1}	n_{q2}			…	n_{qq}	n_{q+}	π_q
Sum n_{+j}	n_{+1}	n_{+2}	…	n_{+j}	…	n_{+q}		1

$n_{12}, n_{21} \ldots n_{ij}$ las clases clasificadas incorrectamente. Sin embargo, al aplicar un muestreo aleatorio, especialmente si es de tipo estratificado, los datos de cada clase no necesariamente serán conforme al área cubierta, por lo que es importante aplicar un ajuste de acuerdo a la ecuación (1) propuesta por Card (1982) considerando la proporción de las clases.

$$\hat{p}_{ij} = \frac{\pi_i n_{ij}}{n_{i+},} \tag{1}$$

donde, n_{ij} es el número de muestras en *i* y que pertenecen a los datos de referencia de categoría *j*, π_i la proporción del área de la clase *i* y n_{i+} el número de muestras en *i*. Posterior a este ajuste y agregando los valores ajustados $\hat{p}_{jj}$ a las celdas de clases coincidente es posible implementar el cálculo de exactitud general con la ecuación (2), los errores de usuario y productor a través de las ecuaciones (3) y (4):

$$\hat{O} = \sum_{k=1}^{q} \hat{p}_{kk} \tag{2}$$

$$\hat{U}_i = \frac{\hat{p}_{ii}}{\hat{p}_{i+}} \tag{3}$$

$$\hat{P}_j = \frac{\hat{p}_{jj}}{\hat{p}_{+j}} \tag{4}$$

La exactitud general (ecuación 2) $\hat{O}$ representa el valor de proporción general del área correctamente clasificada donde *q* es el número de categorías y $\hat{p}_{kk}$ la proporción de muestras correctamente clasificadas en la diagonal. $\hat{U}_i$ (ecuación 4), estima la exactitud del usuario a través de la división del total

de datos correctamente clasificados representados $\hat{P}_{ii}$ entre la sumatoria de la fila $\hat{p}_{i+}$ y $\hat{P}_j$ la exactitud del productor donde $\hat{p}_{jj}$ representa el mismo valor que $\hat{P}_{ii}$ dividiéndose entre la suma de la columna $\hat{p}_{+j}$ (ecuación 4). ML en general estima métricas de exactitud, precisión, sensibilidad, especificidad y *f-score* a través de los valores resultantes de la comparación entre los datos predichos y de prueba mediante la matriz de confusión que puede ser binaria (Tabla 2) o multiclase (Tabla 3). La diferencia principal entre estas radica en el cálculo de las métricas. En el caso de la binaria se estima de forma global y en la multiclase se calcula para cada una de las clases como si estas fueran matrices binarias independientes agrupándolas posteriormente en una sola.

Tabla 2. Matriz de confusión binaria.

		Observadas		Total
Clasificadas		*Clase 1*	*Clase 2*	
	Clase 1	VP	FN	A
	Clase 2	FP	VN	N
Total		A^1	N^1	

Tabla 3. Matriz confusión multi clase.

		Observadas		
Clasificadas		*Clase 1*	*Clase 2*	*Clase 3*
	Clase 1	*VP*	*FN*	*FN*
	Clase 2	*FP*	*VN*	*VN*
	Clase3	*FP*	*VN*	*VN*

VP= Verdaderos positivos FN= Falsos negativos FP= Falsos positivos VN= Verdaderos negativos A^1= suma de verdaderos positivos y falsos positivos N^1=Suma de falsos negativos y verdaderos negativos.

Dichas métricas si bien poseen nombres diferentes a los utilizados en las evaluaciones de clasificación de UCS, igualmente son obtenidas a través de una matriz de confusión, siendo equivalentes a través de las ecuaciones 5-12 (Kamusoko, 2019).

$$\textit{Exactitud general}=(VP+VN)/(VP+FN+FP+VN) \quad (5)$$

$$\textit{Precision}=VP/(VP+FP) \quad (6)$$

$$\textit{Exhaustividad}=VP/(VP+FN) \quad (7)$$

$$\textit{f-score}=(2\times\textit{Precisión}\times\textit{Exhaustividad})/(\textit{Precisión}+\textit{Exhaustividad}) \quad (8)$$

$$\textit{Exactitud productor}(\textit{Sensibilidad})=VP/A^1 \quad (9)$$

$$\textit{Exactitud productor}(\textit{Especificidad})=VN/N^1 \quad (10)$$

$$\textit{Exactitud usuario}(\textit{valor de predicción positivo})=VP/A \quad (11)$$

$$\textit{Exactitud usuario}(\textit{valor de predicción negativo})=VN/N \quad (12)$$

En la clasificación de UCS la matriz de confusión es utilizada mayormente para observar los errores de clasificación y calcular índices de exactitud. Es necesario corregir sesgos de representación de las categorías mediante la aplicación de la ecuación (1), siendo usualmente enfocada la atención en los resultados de parámetros como, la exactitud general, de clases, de usuario y de productor. Si bien estos parámetros en ML también son considerados, el análisis se suele ampliar al incluir la estimación de precisión y del *f-score* que brinda información sobre la exactitud predictiva alcanzada a partir de la simplificación en un solo valor de la precisión y la exhaustividad del clasificador, lo que permite detectar el desbalance de datos. El índice Kappa es otro de los índices que se utiliza para evaluar la exactitud de una imagen clasificada. Toma en cuenta tanto la coincidencia observada como la coincidencia esperada por azar. Un valor cercano a 1 indica una consistencia perfecta, mientras que un valor cercano a 0 indica una baja consistencia que podría alcanzar un modelo aleatorio (Lillesand *et al.*, 2015). Ahora bien, su utilidad en teledetección ha sido fuertemente cuestionada por considerar que la información que ofrece es poco útil para la evaluación de exactitud, ya que trata de comparar la exactitud con una línea de referencia aleatoria, lo cual no se considera una alternativa viable en la evaluación de mapas (Pontius & Millones, 2011).

3. Algoritmos de aprendizaje automático

3.1. Árboles de decisión (*Single decision tree* DT)

Introducido por Ross Quinlan en 1986, se basa en una división jerárquica para la identificación de objetos, aplicado en el proceso de clasificación, se implementa a partir de una serie de reglas que inicia en la raíz y finaliza en los extremos, identificando la clase perteneciente del píxel. El diseño del árbol de decisiones se centra en las propiedades espectrales de cada clase y la relación entre estas, definiendo así un nodo raíz a partir del cual se integran nodos interiores, que a su vez cuentan

con nodos terminales, razón por la cual recibe el nombre que posee, ya que gráficamente simula un árbol, raíz, ramas (nodos interiores) y hojas (nodos terminales) (Tso y Mather, 2009).

3.2. Bosque aleatorio (*Random Forest* RF)

El algoritmo clasificador RF fue desarrollado por Leo Breiman en 2001. Es un enfoque de aprendizaje automático que crea múltiples árboles de decisión a partir de submuestras aleatorias de los datos de entrenamiento y luego combina sus predicciones para obtener una predicción más precisa, por lo que se consideran como una herramienta eficaz, ya que por ser un método de ensamble, mejora la precisión y estabilidad del mismo en comparación con los árboles de decisiones, especialmente cuando los conjuntos de datos de entrada presentan ruido. La técnica para la generación de RF generalmente se estructura como una combinación de *Bagging,* técnica que ayuda a evitar el sobre-ajuste y mejorar la exactitud de la clasificación y métodos aleatorios subespaciales lo cual determina aleatoriamente diferentes subconjuntos de datos de entrenamiento (Breiman, 2001; Tso y Mather, 2009).

RF ha sido un algoritmo popular en la clasificación de UCS con aprendizaje automático por sus buenos resultados aun en condiciones poco óptimas tales como datos de entrenamiento insuficientes y desbalanceados. Además posee la capacidad para trabajar con datos multifuente que pueden ser imágenes multiespectrales e hiperespectrales así como información geográfica como elevación y pendientes, lo que le hace ampliamente versátil, sumado a la simpleza en su parametrización para la clasificación de UCS, ya que suele ser suficiente con determinar el número de árboles de decisión a generar y el número de variables a seleccionar, características que le hace menos sensible a problemas de sobre o sub-ajuste (Gislason *et al.*, 2006; Belgiu y Drăguţ, 2016).

3.3. K vecinos más cercanos (*k-Nearest Neighbours* K-NN)

Clasificador que no requiere un modelo para ser ajustado (Hastie *et al.*, 2008), fue introducido por Thomas Cover en 1967. Es un método no generalizador de ML, siendo una de sus principales ventajas, la facilidad de su implementación y su manejo adecuado de clases multimodales (Kamusoko, 2019). KNN funciona bajo el supuesto de que los píxeles cercanos en el espacio espectral sean probablemente de la misma clase, siendo así que toma un píxel desconocido y examina los píxeles de entrenamiento disponibles que posean un dominio espectral parecido para luego definir la clase que mejor le representa a partir de este valor y de un número predeterminado de vecinos cercanos (Richards, 2013).

3.4. Máquina de soporte de vectores (*Support Vector Machines* SVM)

SVM se ha considerado como una técnica potente y flexible, desarrollado por Vladimir N. Vapnik y Corinna Cortes en 1993 para aplicaciones de clasificación, cuyo objetivo es encontrar el límite entre dos clases que maximice el margen de separación, dependiendo del conjunto de datos de entrenamiento, de los cuales se apoya en los más cercanos, conocidos como vectores de soporte (Kuhn y Johnson, 2016; Chang y Bai, 2018; Kamusoco, 2019). Funciona a partir de la división de un conjunto de datos en dos clases, mediante la implementación de un hiperplano con un margen máximo entre dos posibles hiperplanos (positivos y negativos) los cuales son descritos a partir de los vectores de soporte. Se basan en el principio de la minimización del riesgo estructural, aprendiendo de un hiperplano lineal, el cual posee un margen máximo que separa un conjunto de ejemplos positivos de unos negativos.

SVM en la clasificación de imágenes requiere que el usuario decida desde un inicio el tipo de estrategia multiclase y cuál será el *kernel* a usar para tener una mejor separabilidad, de base radial o polinomial por ser de las mejores opciones, situación similar para la estrategia multiclase, en la que se suelen utilizar técnicas conocidas como uno contra todos (OAA por sus siglas en inglés) (Richards, 2013).

3.5. Redes neuronales artificiales (*Artificial Neural Networks* ANN)

Los algoritmos ANN están inspirados en el modelo biológico de las neuronas del cerebro humano (Lillesand *et al.*, 2015), fueron introducidos por Warren McCulloch y Walter Pitts en 1943. Son un método no paramétrico, por lo que el rendimiento de la red neuronal depende de lo bien que se haya entrenado y de la distribución de los datos. En la

clasificación de imágenes de sensores remotos se consideran las arquitecturas perceptrón multicapa con retropropagación de errores, el mapa de características autoorganizado, las redes de contrapropagación, las redes de Hopfield y los sistemas de teoría de resonancia adaptativa como idóneas para estos fines. ANN funciona a partir de elementos de procesamiento llamados neuronas y conexiones entre ellos con coeficientes, denominados pesos, los cuales procesan información de entrada y producen una salida. Cada nodo realiza una operación matemática simple, y la combinación de estas operaciones en la red permite que se produzcan resultados complejos, para lo que el usuario puede definir parámetros como el número y tamaño de capas, ratio de aprendizaje y los valores de regularización que mejor se adapten a sus necesidades, ya que si bien existen un gran número de reglas para la implementación de ANN la parametrización en específico debe ser adaptada a las condiciones particulares (Mas y Flores 2008; Tso y Mather, 2009).

4. Métodos de revisión

La búsqueda de artículos se centró en la clasificación de UCS con algoritmos de aprendizaje automático. Se utilizó una aproximación sistemática en dos bases de datos, EBSCO, la cual ofrece un compendio amplio de publicaciones de editoriales reconocidas y Google académico, con el objeto de ampliar la búsqueda.

Los criterios en la exploración fueron: literatura publicada entre 2010 y 2020, no se tomaron en cuenta publicaciones elaboradas a partir de datos obtenidos mediante un sistema de medición y detección de objetos mediante láser (LIDAR por sus siglas en ingles) y/o Radar de Apertura Sintética (SAR por sus siglas en inglés) ni aquellas que no aportaran información suficiente o relevante. Se generaron combinaciones de palabras clave a través de operadores booleanos para obtener una ecuación de búsqueda aplicada en inglés ((*land use cover classification AND ("artificial neural networks" OR "support vector machines" OR "decision trees" OR "random forest" OR "k nearest neighbors") AND ("ANN" OR "SVM" OR "KNN" OR "RF" OR "DT") NOT (lidar OR radar*)), y español ((*clasificacion de uso y cobertura del suelo AND ("redes neuronales artificiales" OR "máquinas de vectores de soporte" OR "arboles de decisión" OR "bosque aleatorio" OR "k vecinos más cercanos") AND ("ANN" OR "SVM" OR "KNN" OR "RF" OR "DT") NOT (lidar OR radar*)). Es de mencionar que la inclusión de las siglas referentes a los algoritmos en la ecuación de búsqueda fue de gran ayuda, ya que su uso en las publicaciones objeto de nuestro interés es constante.

Como resultado de la búsqueda se obtuvieron un total de 706 artículos científicos, a partir de los cuales se realizó una preselección manual, que consistió en una revisión a partir del título y del resumen, con lo cual se buscó identificar que estos abordan aspectos importantes para la revisión como 1) confirmar que no estuvieran basados en imágenes de tipo RADAR y/o LIDAR, 2) que estuvieran hechos con los métodos de clasificación de nuestro interés y preferiblemente presentaran información referente a los resultados obtenidos, como la exactitud de la clasificación de UCS. Producto de esto resultaron 110, los cuales fueron revisados y seleccionados en detalle, evaluando que aportaran información suficiente que estuvieran enfocados en la temática de clasificación o que no la abordan de forma superficial, dejando de lado los que estuviesen repetidos o que se hubiesen pasado por alto los filtros previos y no cumplieran con criterios de selección, quedando un total de 44 artículos (Figura 2).

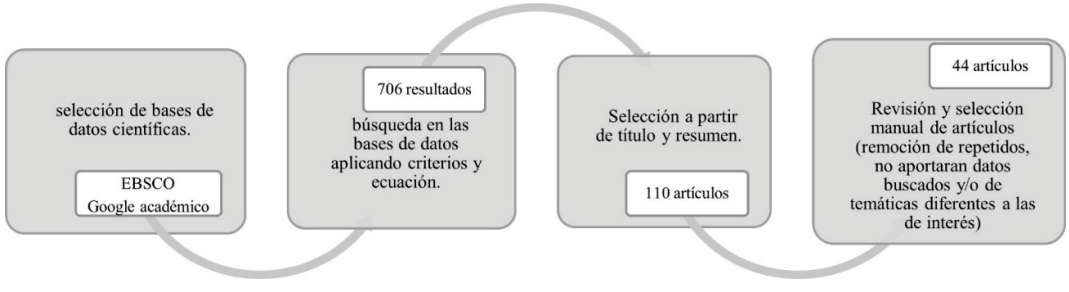

Figura 2. Diagrama general de la revisión de literatura.

5. Resultados

5.1. Características generales de los estudios

Dentro del periodo analizado (2010-2020) (Figura 3) se observa un crecimiento fluctuante. En la primera mitad, de 2011 a 2015, la producción fue limitada concentrando 36% del total con 3,2 artículos por año, estando el 64% restante en la mitad de 2016 a 2020, siendo los años más productivos 2014 (16%), 2017 (18%), 2019 (18%) y 2020 (16%).

La mayoría de los artículos (93%) utilizó datos espaciales provenientes de fuentes como: vehículos aéreos no tripulados (VANT); los programas Landsat, SPOT y Pleiades; los satélites Formosat, Gaofen2 (GF-2) y WorldView; los sensores Hyperion y ASTER; las constelaciones PlanetScope y RapidEye y la misión Sentinel. Del total de trabajos, 48% utilizó imágenes Landsat, incrementando a 55% al considerar aquellos donde se usó de manera conjunta con alguna otra imagen. Los artículos que utilizaron imágenes diferentes a Landsat (45%), se orientaron hacia la experimentación de las imágenes con los algoritmos, aplicándolos en condiciones diversas (Petropoulos *et al.*, 2012; Jia *et al.*, 2014; Elatawneh *et al.*, 2014; Karakacan y Bektas, 2017) brindando información sobre el desempeño de los clasificadores en territorios y bajo características particulares de las imágenes a fin de identificar aquellos con mejores resultados. En las clasificaciones con imágenes de Landsat el número de clases fue entre tres y catorce, y las áreas cubiertas fueron desde los 6,8 km^2 hasta los 23092 km^2.

5.2. Datos de entrenamiento y evaluación

Los conjuntos de datos para entrenamiento y evaluación en el 90% de los casos el número de muestras no fue dado en base a un método en particular a excepción de trabajos como los de Ge *et al.* (2020) donde estimaron que el número de muestras no deberá ser menor de 10-30 veces el número de bandas; Di Shi y Yang (2017) establecieron el tamaño del conjunto de datos a partir del mínimo de muestras necesaria utilizando el método estadístico convencional basado en una distribución multinomial; Puertas *et al.*, (2013) que determinaron el tamaño de la muestra utilizando el enfoque de proporciones multinomiales de Thompson para hacer una selección aleatoria de cada estrato.

La distribución de datos para entrenamiento y evaluación se dio a través de métodos aleatorios, siendo únicamente el 25% de los trabajos que específico el porcentaje de distribución, de los cuales el 73% utilizó una distribución de 70% para entrenamiento y 30% para evaluación (Abdel-Rahman *et al.*, 2014; Adam *et al.*, 2014; Chakraborty *et al.*, 2016; Ge *et al.*, 2020; Saini y Ghosh, 2018; Vélez y Álvarez, 2020; Aguilera, 2020; Abdi, 2020).

La calidad de los conjuntos de datos es trascendental para un buen entrenamiento de los algoritmos y una buena clasificación, por lo que el origen y el tratamiento marcará una diferencia (Marsland, 2014; Alpaydin 2014; Müller y Guido, 2016). Los artículos presentan escasa información sobre la preparación o tratamiento de los conjuntos de datos antes de ser utilizados o después de practicar

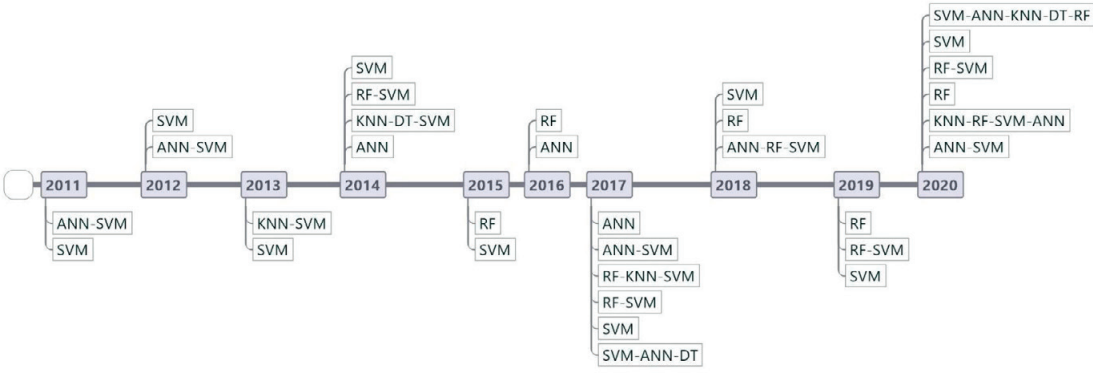

Figura 3. Línea de tiempo de algoritmos utilizados.

pruebas para la parametrización de los algoritmos, es decir, algún tipo de procedimiento que pudiese reducir el ruido, los sesgos, anomalías o desequilibrio de los datos. Solo en 16% de los trabajos se llevaron a cabo procedimientos para equilibrar los datos. Kamusoko (2019) advierte que una práctica usual de los analistas de sensores remotos es, suponer que los conjuntos de datos son precisos por haber implementado comprobaciones en terreno o en imágenes de alta resolución. Sin embargo, esto no exime de la posibilidad de introducir errores durante la adquisición de las muestras o en alguna otra etapa, por lo que sería prudente implementar comprobaciones estadísticas que permitan identificar anomalías.

Las muestras fueron seleccionadas por factores como disponibilidad y objetivo particular del estudio, considerando factores como accesibilidad e inclusive razones económicas. 33% utilizaron imágenes provenientes de *Google Earth*, por su muy alta resolución espacial (Thanh Noi y Kappas 2017; Karakacan y Bektas, 2017; Mfuka *et al.*, 2019).

5.3. Parametrización de algoritmos

El 68% basó el ajuste de sus parámetros en la experimentación propia y en la obtenida mediante bibliografía. No obstante, los detalles de las características de su configuración y el análisis de los resultados del desempeño del algoritmo es superficial. El análisis de los fenómenos de sobre y sub-ajuste no fue un tema de particular interés. Únicamente los trabajos de Puertas *et al.* (2013), Halmy y Gessler (2015), Chen *et al.* (2017) y Jozdani *et al.* (2019) hacen referencia al abordaje del sobre-ajuste, pero sin brindar detalles al respecto. Sin embargo, dichos trabajos forman parte del 25% de los que aplicaron procedimientos de validación cruzada, el cual es un método utilizado tanto para subsanar la limitación de datos como para evaluar el modelo y por ende permite detectar problemas del mismo asociados a los parámetros y datos (Puertas *et al.*, 2013; Adam *et al.*, 2014; Abdel-Rahman *et al.*, 2014; Halmy y Gessler, 2015; Ganbold y Chasia, 2017; Thanh Noi y Kappas, 2017; Jamil y Bayram, 2018; Abdi, 2020; Jozdani *et al.*, 2019; Ge *et al.*, 2020; Vélez y Álvarez, 2020).

5.4. Evaluación de exactitud

La matriz de confusión fue el método principal para la evaluación de la exactitud. Más de la mitad (55%) aplicó simultáneamente el índice Kappa, el 16% incorporó otros métodos estadísticos como la determinación del *f-score* para medir la calidad de la clasificación de cada clase, la cual viene a ser una medida armónica entre la exactitud generada por el usuario y el productor (Christovam *et al.*, 2019) y la prueba de McNemar, que es una prueba para identificar la diferencia significativa en los resultados obtenidos por los clasificadores (Foody, 2004), por lo que su aplicación está orientada hacia las comparaciones de algoritmos. Al ser *f-score* una métrica que facilita la estimación y la interpretación de sensibilidad y precisión, puede ser utilizada para determinar la calidad de la clasificación en cada una de las clases evaluadas (Knox, 2018; Géron, 2019). De los artículos evaluados, solo el 7% hicieron esta estimación (Saini y Ghosh, 2018; Jozdani *et al.*, 2019; Christovam *et al.*, 2019).

Para la presentación de resultados el 68% incorpora la evaluación de la exactitud por cada clase, sin embargo, la relevancia de su análisis aún está supeditado a la exactitud general, medida que como se ha mencionado no representa por completo la complejidad de los resultados obtenidos habiendo aún un porcentaje importante de trabajos que no incorporan la evaluación a nivel de clase (32%).

Los valores máximos de exactitud alcanzados estuvieron entre el 80% (DT) y el 99% (SVM) y los valores mínimos en el rango del 18% (ANN) al 74% (KNN). El promedio total de los algoritmos, ANN, RF, SVM y K-NN se mantiene por arriba del 80%, lo cual muestra que en general los niveles de exactitud alcanzados pueden considerarse altos, a excepción de DT (72%). Por otro lado, K-NN, si bien presenta un valor promedio alto, se encuentra muy por debajo de la media de 16,4 análisis por algoritmo siendo el menos utilizado (Tabla 4).

Tabla 4. Valores máximos, mínimos y promedio en exactitud alcanzados por algoritmos.

Algoritmos	ANN	RF	SVM	K-NN	DT
No. de análisis practicados	13	25	37	4	3
Valor máximo exactitud	97,2	99,8	99,9	94,6	80,2
Valor mínimo exactitud	18,3	64,0	70,0	74,7	67,8
Promedio exactitud	80,8	85,6	86,8	86,0	72,0

SVM fue el más utilizado, siendo con el algoritmo RF con el que más se comparó, generando ambos valores de exactitud en rangos cercanos. Por ejemplo, los obtenidos por Jamali (2019), el cual logró valores en SVM de 99,93% y de 99,83% en RF, o los obtenidos por Abdel-Rahman *et al.* (2014) de los cuales sus resultados fueron en RF 74,50% y SVM 73,50%. La diferencia obtenida entre estos clasificadores en el total de trabajos donde fueron comparados no sobrepasa el 6% en los valores de exactitud general, guardando una buena correlación entre sí (R^2: 0,91) (Tabla 5).

Esta preferencia por SVM podría estar determinada por ser un método bastante aplicado en clasificación de UCS. Según Gualtieri y Cromp (1999) fue introducido a mediados de los años noventa por Boser, Guyon y Vapnik, haciendo que su disponibilidad y familiaridad sea mayor entre los usuarios y los diferentes paquetes de software, condición que posiblemente le ha permitido ser probado y evaluado en mayor medida. Además, a diferencia de otros métodos, la cantidad de muestras que necesita para operar y generar buenos resultados puede ser mucho menor (Foody y Mathur, 2004). Sumado a ello, su configuración no suele ser compleja, lo que permite a los usuarios entender su funcionamiento con mayor facilidad.

El promedio general de exactitud en función de los diferentes sensores se mantiene por encima del 70%. Los que poseen mayor resolución como la imagen obtenida mediante VANT o RapidEye obtuvieron valores superiores al 90%. En el caso de los de mediana resolución como Landsat, es interesante observar que la diferencia entre los algoritmos no es tan amplia. Su exactitud general se mantuvo en un promedio de 87,7% y por algoritmo entre el 80,2% con DT y el 90,6% con ANN, y Sentinel-2, que, si bien los promedios son un tanto menores, se sigue manteniendo superior al 80% (Tabla 6).

Tabla 5. Comparación de exactitud obtenida entre algoritmos RF y SVM.

Autores	Exactitud obtenida
Adam *et al.* (2014)	RF 93,07% - SVM 91,80
Di Shi y Yang (2017)	RF80,05% - SVM-79,38%
Jamali (2019)	SVM 99,93% - RF 99,83%
Christovam *et al.* (2019)	SVM 88% - RF 91%
Abdi (2020)	SVM 75,5% - RF 73,9%
Jozdani *et al.* (2019)	RF 94,47% - SVM 95,43%
Rana y Venkata Suryanarayana (2020)	RF 70% - SVM 76%
Tassi y Vizzari (2020)	RF (Landsat8 = 64%, Sentinel2 = 89,3%, PlanetScope = 77,9) SVM (Landsat8 = 70,4, Sentinel2 = 86,9%, PlanetScope = 73,9)

Tabla 6. Exactitud general y por algoritmo ML promedio por cada sensor.

	Exactitud general promedio en %					
Sensores	DT	K-NN	SVM	RF	ANN	General
Hyperion			82,7		52,0	72,5
AISA Eagle			73,5	74,5		74,0
PlanetScope			73,9	77,9		75,9
FormoSat					77,3	77,3
Pleiades				78,0		78,0
Ortofoto			87,0	80,0	86,0	84,3
Sentinel		94,6	81,9	84,2		84,5
SPOT	68,1		86,9		91,0	86,7
Landsat	80,2	84,3	87,3	86,6	90,6	87,7
HyRANK			88,0	92,0		90,0
VANT			94,1		87,4	90,8
RapidEye			91,8	93,1		92,4
WorldView			92,5	93,0		93,1
GF-2 satellite			94,3			94,3
ASTER			94,2		95,0	94,6
Aerial High Spatial Resolution image -HSR-			95,4	94,5		95,0

Únicamente tres artículos utilizaron imágenes provenientes de sensores no satelitales, Jamil y Bayram (2018) utilizaron ortofotos digitales de muy alta resolución obtenidas a través de una cámara aérea digital, teniendo como objetivo experimentar con algoritmos ANN, SVM y RF en la detección de especies arbóreas y en la clasificación del UCS; Syifa *et al.* (2020) analizaron imágenes de alta resolución obtenidas mediante VANT para diferenciar pinos enfermos de sanos a través de ANN y SVM. Abdel-Rahman *et al.* (2014) trabajaron con imágenes hiperespectrales obtenidas del Sistema de Imágenes Aéreas para Diferentes Aplicaciones (AISA por sus siglas en inglés) a través del sensor de barrido Eagle, con la finalidad de distinguir pinos sanos de los atacados por *Sirex noctilio* y/o dañados por rayos a través de RF y SVM. Coinciden los tres trabajos anteriores en la temática de salud forestal además de utilizar imágenes de fuentes no satelitales.

Si bien hay factores que pueden influir en la exactitud obtenida, como la selección de parámetros por el usuario, los datos del entrenamiento y evaluación, se ha llegado a considerar que el número de clases o la extensión del área estudiada podrían tener un impacto (Maxwell *et al.*, 2018). A este respecto se encontró que la exactitud general como tal no responde a variaciones del área y el número de clases (Figura 4 (a)(c)), y no llega a presentar una relación significativa entre sí. Evidencia de esto es el trabajo de Di Shi y Yang (2017) quienes analizaron la mayor extensión registrada (23092 km^2) obteniendo una exactitud del

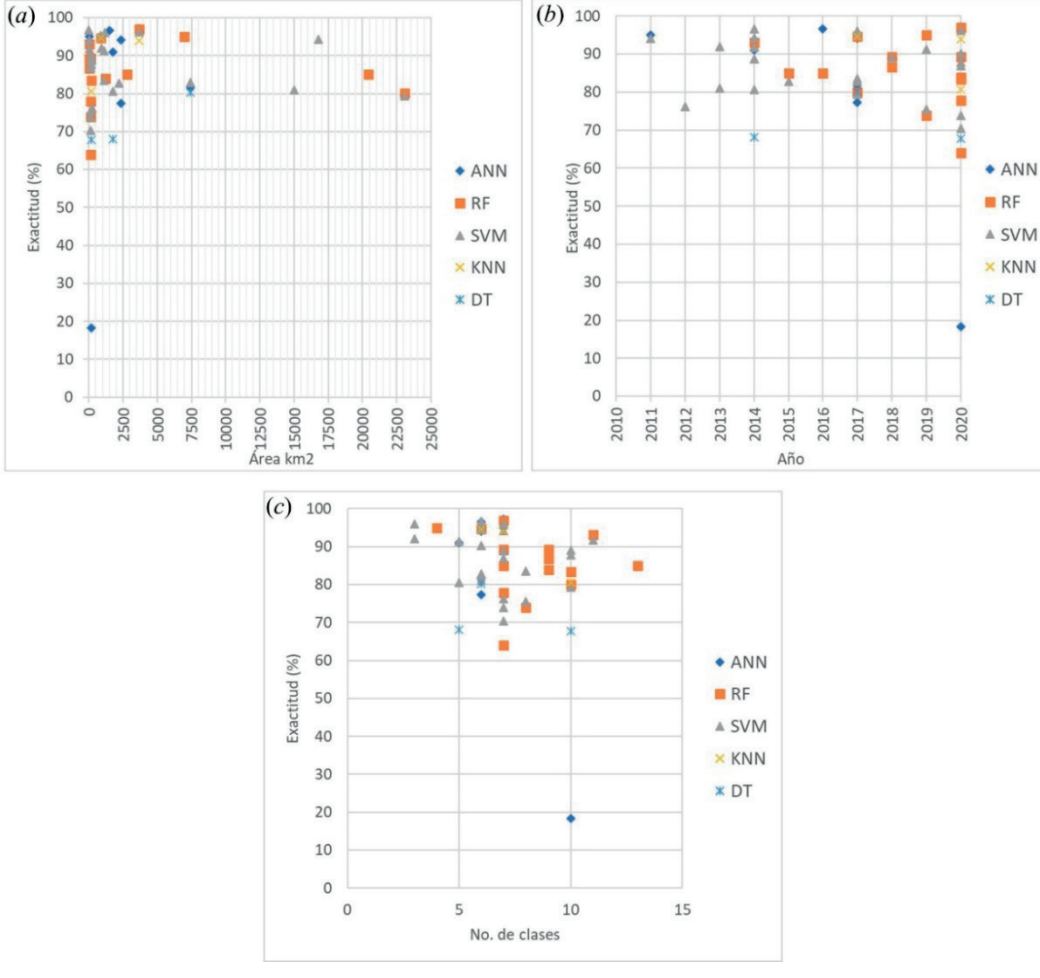

Figura 4. Relación entre área *(a)*, años *(b)* y numero de clases *(c)* con la exactitud alcanzada por algoritmo

79,38% con SVM, en contraste con trabajos como el de Tassi y Vizzari (2020), que al analizar 154 km^2 obtuvo una exactitud del 70,4% con SVM y del 64% en RF.

Respecto a los valores de exactitud alcanzados mayores al 90%, es necesario considerar que pueden estar asociados a problemas como sesgos, sobre-ajuste o conjuntos de evaluación pequeños que provocan que el algoritmo tenga poca oportunidad de equivocarse. En el caso de Jamali (2019), que alcanza más del 99%, es difícil determinar posibles causas de dicho resultado, ya que el trabajo es escueto en la información que provee. No obstante, trabajos con resultados por encima del 90% como el de Ge *et al.* (2020), que proveen información como, el número de muestras utilizadas por clase, el análisis de separabilidad espectral, entre otros, no obstante, hay poca claridad en cuanto al método para la selección de muestras. Únicamente se menciona que para reducir la redundancia y la autocorrelación espacial seleccionaron muestras basadas en píxeles mediante visitas de campo en 468 puntos, conocimiento previo e interpretación visual de imágenes disponibles en *Google Earth*, distribuyendo aleatoriamente en 70% entrenamiento y 30% evaluación, siendo así este último conjunto de datos un subconjunto del de entrenamiento. Olofsson *et al.* (2014) hacen hincapié al indicar que en teledetección las muestras utilizadas para la evaluación deben ser independientes de los datos utilizados para entrenar, por lo que en este caso al ser los conjuntos de datos divididos de uno común, es posible que sean demasiado similares y por ende esté impactando en la estimación de exactitud obtenida.

6. Discusión

Esta revisión del uso de algoritmos ML en la clasificación de UCS otorgó información para responder las preguntas planteadas 1) ¿cómo se ha implementado ML en clasificación de UCS en los aspectos de revisión de conjunto de datos e implementación de algoritmos; 2) ¿Qué factores metodológicos, tales como la compilación de datos, implementación de métodos, y la evaluación de resultados, afectan a la implementación de ML en la clasificación de UCS?

Para responder la primera pregunta se comenzó por la descripción de características generales, fuentes de información y métodos utilizados, ahondando en la construcción de los conjuntos de datos e implementación de los algoritmos.

La producción de artículos muestra un incremento a partir de la mitad del periodo (2010-2015), dándose un enfoque con notable interés hacia los análisis de carácter comparativo del desempeño de los algoritmos. En cuanto a la fuente de información de imágenes, Landsat fue la más utilizada y evaluada, al contrario de imágenes de otras fuentes que en su mayoría únicamente fueron evaluadas con dos algoritmos (RF y SVM) y con las que sería provechoso experimentar más. En cuanto al origen de los datos para conformar los conjuntos de entrenamiento y validación el uso de plataformas como *Google Earth* para obtener muestras de imágenes de alta resolución además de las muestras de campo fue algo usual. No obstante, al ser una plataforma que está diseñada para fines diferentes sería importante implementar o dar a conocer las medidas aplicadas para evitar posibles errores como la discrepancia de fechas.

6.1. Conjuntos de datos y tratamiento

En ML es importante que los conjuntos de datos para entrenamiento, evaluación, y validación se encuentren en las mejores condiciones posibles. Sin embargo, la gran mayoría de trabajos analizados (84%) carecen de evidencia de haber implementado algún tipo de proceso para tratar anomalías o sesgos como ruido o desequilibrio de los datos. En este sentido Kamusoko (2019) considera que probablemente se debe a que es común suponer que los datos poseen suficiente calidad y precisión por haber sido obtenidos mediante comprobaciones en campo o en imágenes de alta resolución. Sumando a esa confianza, se encuentra la popularidad de algoritmos como DT, RF y SVM, por mostrar un buen desempeño y tolerancia para trabajar con datos carentes de calidad y cantidad, lo cual puede que haya venido abonando a esta confianza "ciega" que propicia la construcción de conjuntos de datos de manera limitada y poco fiable (Lu y Weng, 2007). Este aspecto puede dar pie a la incertidumbre en los resultados, por lo que será necesario considerar como buena práctica implementar comprobaciones estadísticas que ayuden a la identificación de desequilibrios, errores o incoherencias que permitan mejorar la calidad tanto de los conjuntos de datos como de los resultados.

También es importante no perder de vista el estudio y análisis de métricas asociadas a las imágenes como el valor de separabilidad de las muestras que se utilizarán, ya que únicamente un 27% de los artículos menciona haber llevado a cabo este tipo de análisis previo a la clasificación de UCS.

6.2. Parametrización y entrenamiento de los algoritmos

El ajuste de los parámetros fue dado de varias maneras, desde la experimentación hasta el uso de "recetas" obtenidas en bibliografía o el uso de valores por defecto. Si bien estas dos últimas no son como tal una práctica prohibida si deja de lado una de las ventajas que otorga ML, que es la optimización de los algoritmos en función de las características del fenómeno y de los datos, por lo que bien vale llevar a cabo el esfuerzo de implementar procesos como la experimentación para la adecuada selección de ajustes. Sin embargo y siendo algo relacionado al tema de los conjuntos de datos, el hecho de llevar a cabo este tipo de procedimiento requiere contar con una buena calidad y cantidad de datos.

La poca implementación de evaluaciones de sub-ajuste, sobre-ajuste o de métodos como validaciones cruzadas, así como la ambigüedad y/o ausencia de información relacionada a los experimentos de optimización y entrenamiento, plantea una serie de retos importantes tanto para usuarios como para los fabricantes de software. Para los usuarios será importante profundizar y aplicar las buenas prácticas que contempla ML para llevar un mejor seguimiento y control de los insumos y procedimientos efectuados, apoyándose en herramientas y técnicas adicionales que no necesariamente se encuentran disponibles en los paquetes de software de teledetección, los cuales a pesar de contemplar cada vez más clasificadores ML aún carecen de una mayor integración de herramientas estadísticas que permitan llevar a cabo estos procesos. Trabajos como, Bishop (2006), Kelleher *et al.* (2015) y Bashir *et al.* (2020) pueden ser una base teórica básica para introducirse al entendimiento de los métodos mencionados.

6.3. Medición de exactitud y desempeño

Los resultados de exactitud fueron obtenidos mediante la elaboración de matrices de confusión. De las métricas obtenidas de dicho método el valor de exactitud general fue el más utilizado para explicar los resultados. No obstante, dicho valor contrario a los valores de exactitud por clase que permiten entender de manera más amplia y completa los resultados, podría ocultar clases con un rendimiento bajo al ser un promedio. Por otro lado, la evaluación del desempeño de los algoritmos ML mediante métodos considerados estándar en ML como *f-score* y la prueba de McNemar es muy baja.

Esta falta de uso de métricas que brinden más detalles sobre los resultados y de la aplicación de métodos que permitan conocer el desempeño de los algoritmos podría estar limitando una mayor comprensión de los resultados. Este aspecto genera a su vez un exceso de confianza que fomente la implementación de ML sin considerar aspectos esenciales previamente discutidos, como la integración de conjuntos de datos en buena calidad y cantidad, una parametrización de algoritmos adecuada y un análisis de resultados más objetivo.

Para analizar los factores metodológicos de ML y su implementación en la clasificación de UCS, han sido tomados como referencia los cuatro pasos para la aplicación de ML en teledetección definidos por Shih *et al.* (2019): la recopilación de datos, la selección y optimización del algoritmo, la aplicación de la clasificación y la evaluación de la exactitud.

6.4. Recopilación de datos

Se puede considerar como la base del proceso ya que de esta información depende que el algoritmo pueda entrenarse correctamente y así lograr obtener resultados satisfactorios, por ello la necesidad de contar con conjuntos de datos amplios y de buena calidad (Huang *et al.*, 2002; Müller y Guido, 2016; Congalton y Green, 2019; Kamusoko, 2019). Sin embargo, definir la cantidad necesaria para el entrenamiento y evaluación de los algoritmos ML no es sencillo, especialmente al implementarse en clasificación de UCS, ya que como tal no hay alguna regla sobre la cantidad mínima de muestras con que se debe contar (Maxwell *et al.*, 2018). En este sentido algunos estudios han investigado el efecto de la variabilidad de las muestras en la exactitud en trabajos de clasificación de UCS (Myburgh y Niekerk, 2013; Li *et al.*, 2014; Qian

et al., 2015; Thanh Noi & Kappas, 2017; Heydari y Mountrakis, 2018; Ramezan *et al.*, 2021). Sin embargo, aún es información incipiente, ya que elementos como, la extensión, características del área, insumos y algoritmos utilizados en estos experimentos es variada, lo que limita poder extrapolarlos a otros estudios, pero constituyen un punto de partida importante a considerar para la estimación del tamaño del conjunto de muestras para un trabajo en particular.

No obstante, lograr integrar un conjunto de datos amplio y con una buena calidad puede ser complicado, por elementos como el tiempo, acceso a la información y capacidad de interpretación. Por lo que evaluar las condiciones de los datos es un aspecto a considerar al seleccionar un algoritmo clasificador, ya que no todos son susceptibles a las condiciones de los datos (Maxwell *et al.*, 2018). Se ha identificado que algoritmos como DT, ANN y KNN son susceptibles a la calidad de los datos en contraste con SVM y RF que toleran mejor este tipo de condiciones y por ende sus resultados se ven menos afectados (Li *et al.*, 2014; Maxwell *et al.*, 2018). Ello no necesariamente significa que se deban descuidar las condiciones de los datos, por lo que sin importar las aptitudes del algoritmo se debe procurar conjuntos de datos con una buena calidad y cantidad para la clasificación de UCS con ML. Para ello conviene llevar a cabo pruebas estadísticas descriptivas multivariantes y univariantes de los datos, así como análisis mediante gráficas que permita distinguir anomalías en la información. Toda esta información será de gran utilidad para identificar problemas como desbalances, valores atípicos y colinealidad (Kamusoko, 2019).

La división del conjunto de datos principal en los subconjuntos de entrenamiento, validación y evaluación dependerá de la cantidad disponible. Si es limitada lo recomendable sería definir una distribución porcentual de 60 (entrenamiento), 20 (validación) y 20 (evaluación) y si es amplia, usualmente se sugiere 50, 25 y 25, para lo cual es posible valerse de una distribución aleatoria (Marsland, 2014). No obstante hacer uso de este último método para la repartición de los datos, puede significar una afectación al desempeño de los algoritmos por provocar un desequilibrio al distribuirles en los diferentes conjuntos donde la probabilidad de selección de la clase es acorde al tamaño que esta representa, por lo que se recomienda revisar las propuestas de solución abordadas en el documento de Maxwell *et al.* (2018).

6.5. Selección del modelo y optimización del clasificador

Seleccionar un algoritmo ML que se ajuste a las condiciones del problema a resolver puede llegar a suponer una inversión alta de tiempo y conocimiento. Apoyarse en la literatura es una opción, sin embargo, es común encontrarse argumentos contradictorios respecto al desempeño de estos, ya que por las diferencias de procedimientos aplicados compararlos no resulta ser sencillo, lo cual puede terminar por confundir más al usuario (Maxwell *et al.*, 2018). Por lo tanto, puede que lo más factible sea hacerlo a partir de la consideración de factores que orienten la decisión, tales como el tamaño y calidad del conjunto de datos disponible, el dominio del usuario respecto al modelo y los ajustes de este, así como el dominio del software desde donde se elabore.

Será necesario hacer la selección inicial de aquel o aquellos clasificadores que ofrezcan un mejor desempeño frente a las condiciones impuestas, ya sea porque sean capaces de sortear de forma más efectiva condiciones desfavorables respecto a los datos o por ajustarse mejor a las necesidades específicas del usuario. No obstante la probabilidad de llegar a contemplar más de una opción es alta, por lo que es recomendable llevar a cabo un proceso de experimentación y comparación para poder hacer una selección en base al que ofrezca mejor desempeño, ya que la selección del algoritmo puede depender mucho de las características particulares de cada caso.

Para la parametrización de algoritmos ML de acuerdo con varios autores (Alpaydin, 2014; Müller y Guido 2016; Maxwell *et al.*, 2018; Shih *et al.*, 2019) la mejor opción puede ser hacer uso de técnicas como validación cruzada, *bootstrapping* o diseño factorial (*grid-search*), ya que permiten entrenar al algoritmo utilizando diferentes configuraciones sin agotar los conjuntos de datos y lograr identificar la que ofrezca mejores resultados. Sin embargo, la desventaja de estos métodos es que no se encuentran integrados en los paquetes de software especializados en teledetección, lo cual implica que deba ser realizado a

través de otros programas como R (R Core Team, 2022) o la librería *scikit-learn* (Pedregosa *et al.*, 2011) implementada en Python.

6.6. Aplicación del clasificador y la evaluación de la exactitud

La aplicación de la clasificación dará origen a nuevos datos que deberán ser evaluados con la finalidad de determinar el grado de exactitud obtenido, para lo cual lo usual es la matriz de confusión y en algunos casos la evaluación del índice Kappa, de los cuales suele ser el dato más representativo la exactitud general. Sin embargo, es necesario revisar con detenimiento los valores obtenidos a nivel de cada clase, ya que en estos es donde se puede identificar clases en las cuales el desempeño no haya sido satisfactorio y por ende aplicar modificaciones, ya sea en los datos o en los parámetros de configuración del algoritmo para mejorar el resultado. Procedimiento que no suele ser tan común al enfocarse mayormente la atención en el resultado general y en la imagen resultante de clasificación de UCS, característica que ha restado interés a la evaluación del desempeño de los algoritmos, ya que pruebas como *f-score* o la prueba de McNemar no suelen ser comunes en este tipo de aplicaciones. Es importante recalcar la importancia de los conjuntos de datos utilizados, ya que deben poseer tanto cantidad como calidad y ser de características relevantes acordes al objetivo de aplicación y de esto dependerá en buena medida los resultados. A este respecto Blum & Langley (1997) y Olofsson *et al.* (2014) han abordado aspectos relevantes que son importantes de considerar desde ML y de la clasificación de imágenes en teledetección.

7. Conclusiones

La revisión practicada sobre ML en la clasificación de UCS ha resaltado la incertidumbre en la aplicación de pasos metodológicos como el tratamiento de los conjuntos de datos, la optimización de parámetros y la evaluación de los resultados a nivel de clase, si bien a nivel de exactitud general no ha reflejado un impacto importante al estar el promedio de este valor en 80% en los trabajos analizados. Sí podría afectar la confianza en los productos obtenidos al no implementarse de la manera esperada los pasos metodológicos necesarios, además que centrar el resultado en un valor global no refleja las afectaciones a nivel de clase derivado de la aplicación de ML de esta forma. En este sentido, se considera importante que los usuarios deben contemplar conocimientos adicionales relacionados a la estadística y ciencia de datos que les permitan llevar a cabo la implementación de los métodos de forma completa, ya que es evidente el exceso de confianza en el uso de parámetros por defecto o "recetas", lo cual, si bien no es prohibido, puede fomentar la omisión de procedimientos con tal de desarrollar trabajos haciendo uso de técnicas novedosas.

La amplia diversidad de escenarios posibles del análisis de UCS con métodos ML plantea condiciones y retos únicos, donde el criterio y habilidades del usuario deberán definir los métodos y técnicas que mejor se acoplen a sus necesidades, por lo cual se insta a considerar los trabajos de Li *et al.* (2014), Maxwell *et al.* (2018), Shih *et al.* (2019) y Kamusoko (2019) para obtener un acercamiento más detallado a técnicas para sortear las dificultades que plantea la clasificación de UCS mediante ML.

Agradecimientos

El autor principal expresa su gratitud al CONAHCYT por brindarle el apoyo para llevar a cabo sus estudios de posgrado. También deseamos agradecer a dos revisores anónimos por sus valiosos comentarios, los cuales contribuyeron significativamente a mejorar el contenido de este artículo.

Referencias

Abdel-Rahman, E.M., Mutanga, O., Adam, E., & Ismail, R. 2014. Detecting Sirex noctilio grey-attacked and lightning-struck pine trees using airborne hyperspectral data, random forest and support vector machines classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, *88*, 48-59. https://doi.org/10.1016/j.isprsjprs.2013.11.013

Abdi, A.M. 2020. Land cover and land use classification performance of machine learning algorithms in a boreal landscape using Sentinel-2 data. *GIScience & Remote Sensing*, *57*(1), 1-20. https://doi.org/10.1080/15481603.2019.1650447

Adam, E., Mutanga, O., Odindi, J., & Abdel-Rahman, E.M. 2014. Land-use/cover classification in a heterogeneous coastal landscape using RapidEye imagery: Evaluating the performance of random forest and support vector machines classifiers. *International Journal of Remote Sensing*, *35*(10), 3440-3458. https://doi.org/10.1080/01431161.2014.903435

Aguilera, M. 2020. Classication Of Land-Cover Through Machine Learning Algorithms For Fusion Of Sentinel-2a And Planetscope Imagery. 2020 IEEE *Latin American GRSS & ISPRS Remote Sensing Conference (LAGIRS)*, 246-253. https://doi.org/10.1109/LAGIRS48042.2020.9165632

Alpaydin, E. 2014. *Introduction to Machine Learning* (3.a ed.). MIT Press.

Bashir, D., Montañez, G.D., Sehra, S., Segura, P.S., & Lauw, J. 2020. An Information-Theoretic Perspective on Overfitting and Underfitting. En M. Gallagher, N. Moustafa, & E. Lakshika (Eds.), *AI 2020: Advances in Artificial Intelligence* (pp. 347-358). Springer International Publishing. https://doi.org/10.1007/978-3-030-64984-5_27

Belgiu, M., & Drăguţ, L. 2016. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing, 114*, 24-31. https://doi.org/10.1016/j.isprsjprs.2016.01.011

Bishop, C.M. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag.

Blum, A.L., & Langley, P. 1997. Selection of relevant features and examples in machine learning. *Artificial Intelligence, 97*(1), 245-271. https://doi.org/10.1016/S0004-3702(97)00063-5

Breiman, L. 2001. Random Forests. Machine Learning, *45*(1), 5-32. https://doi.org/10.1023/A:1010933404324

Campbell, J.B., & Wynne, R.H. 2011. Introduction to Remote Sensing, Fifth Edition. Guilford Publications. https://books.google.com.mx/books?id=NkLmDjSS8TsC

Card, D. 1982. Using map category marginal frequencies to improve estimates of thematic map accuracy. *Photogrammetric Engineering and Remote Sensing*, *48*(3), 431-439.

Chakraborty, A., Sachdeva, K., & Joshi, P.K. 2016. Mapping long-term land use and land cover change in the central Himalayan region using a tree-based ensemble classification approach. *Applied Geography*, *74*, 136-150. https://doi.org/10.1016/j.apgeog.2016.07.008

Chang, N.-B., & Bai, K. 2018. *Multisensor Data Fusion and Machine Learning for Environmental Remote Sensing* (1.a ed.). CRC Press. https://doi.org/10.1201/9781315154602

Chen, Y., Dou, P., & Yang, X. 2017. Improving Land Use/Cover Classification with a Multiple Classifier System Using AdaBoost Integration Technique. *Remote Sensing, 9*(10), 1055. https://doi.org/10.3390/rs9101055

Christovam, L.E., Pessoa, G.G., Shimabukuro, M.H., & Galo, M.L.B.T. 2019. Land use and land cover classification using hyperspectral imagery: evaluating the performance of spectral angle mapper, support vector machine and random forest. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13, 1841-1847. https://doi.org/10.5194/isprs-archives-XLII-2-W13-1841-2019

Congalton, R.G., & Green, K. 2019. *Assessing the Accuracy of Remotely Sensed Data: Principles and Practices, Third Edition* (3.a ed.). CRC Press. https://doi.org/10.1201/9780429052729

Demirkan, D.Ç., Koz, A., & Düzgün, H.Ş. 2020. Hierarchical classification of Sentinel 2-a images for land use and land cover mapping and its use for the CORINE system. *Journal of Applied Remote Sensing, 14*(02), 1. https://doi.org/10.1117/1.JRS.14.026524

Di Shi, & Yang, X. 2017. A Relative Evaluation of Random Forests for Land Cover Mapping in an Urban Area. *Photogrammetric Engineering & Remote Sensing, 83*(8), 541-552. https://doi.org/10.14358/PERS.83.8.541

Dong, J., Metternicht, G., Hostert, P., Fensholt, R., & Chowdhury, R.R. 2019. Remote sensing and geospatial technologies in support of a normative land system science: Status and prospects. *Current Opinion in Environmental Sustainability*, *38*, 44-52. https://doi.org/10.1016/j.cosust.2019.05.003

Elatawneh, A., Kalaitzidis, C., Petropoulos, G.P., & Schneider, T. 2014. Evaluation of diverse classification approaches for land use/cover mapping in a Mediterranean region utilizing Hyperion data. *International Journal of Digital Earth*, *7*(3), 194-216. https://doi.org/10.1080/17538947.2012.671378

Foody, G.M. 2004. Thematic map comparison. *Photogrammetric Engineering & Remote Sensing*, *70*(5), 627-633.

Foody, G.M., & Mathur, A. 2004. Toward intelligent training of supervised image classifications: Directing training data acquisition for SVM classification. *Remote Sensing of Environment, 93*(1-2), 107-117. https://doi.org/10.1016/j.rse.2004.06.017

Ganbold, Ganchimeg, & Chasia, Stanley. 2017. Comparison between Possibilistic c-Means (PCM) and Artificial Neural Network (ANN) Classification Algorithms in Land use/ Land cover Classification. *International Journal of Knowledge Content Development & Technology*, *7*(1), 57-78. https://doi.org/10.5865/IJKCT.2017.7.1.057

Ge, G., Shi, Z., Zhu, Y., Yang, X., & Hao, Y. 2020. Land use/cover classification in an arid desert-oasis mosaic landscape of China using remote sensed imagery: Performance assessment of four machine learning algorithms. *Global Ecology and Conservation*, 22, e00971. https://doi.org/10.1016/j.gecco.2020.e00971

Géron, A. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Gislason, P.O., Benediktsson, J.A., & Sveinsson, J.R. 2006. Random Forests for land cover classification. *Pattern Recognition Letters*, *27*(4), 294-300. https://doi.org/10.1016/j.patrec.2005.08.011

Gualtieri, J.A., & Cromp, R.F. 1999. *Support vector machines for hyperspectral remote sensing classification*. R.J. Mericsko, Ed.; pp. 221-232. https://doi.org/10.1117/12.339824

Halmy, M.W.A., & Gessler, P.E. 2015. The application of ensemble techniques for land-cover classification in arid lands. *International Journal of Remote Sensing*, *36*(22), 5613-5636. https://doi.org/10.1080/01431161.2015.1103915

Hastie, T., Tibshirani, R., Friedman, J.H., & Friedman, J.H. 2009. *The elements of statistical learning: Data mining, inference, and prediction* (Vol. 2). Springer.

Herold, M., Latham, J.S., Di Gregorio, A., & Schmullius, C.C. 2006. Evolving standards in land cover characterization. *Journal of Land Use Science*, *1*(2-4), 157-168. https://doi.org/10.1080/17474230601079316

Heydari, S.S., & Mountrakis, G. 2018. Effect of classifier selection, reference sample size, reference class distribution and scene heterogeneity in per-pixel classification accuracy using 26 Landsat sites. *Remote Sensing of Environment*, *204*, 648-658. https://doi.org/10.1016/j.rse.2017.09.035

Huang, C., Davis, L.S., & Townshend, J.R.G. 2002. An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, *23*(4), 725-749. https://doi.org/10.1080/01431160110040323

Jamali, A. 2019. A fit-for-purpose algorithm for environmental monitoring based on maximum likelihood, support vector machine and random forest. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3/W7, 25-32. https://doi.org/10.5194/isprs-archives-XLII-3-W7-25-2019

Jamil, A., & Bayram, B. 2018. Tree Species Extraction and Land Use/Cover Classification From High-Resolution Digital Orthophoto Maps. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *11*(1), 89-94. https://doi.org/10.1109/JSTARS.2017.2756864

Jia, K., Liang, S., Wei, X., Yao, Y., Su, Y., Jiang, B., & Wang, X. 2014. Land Cover Classification of Landsat Data with Phenological Features Extracted from Time Series MODIS NDVI Data. *Remote Sensing*, *6*(11), 11518-11532. https://doi.org/10.3390/rs61111518

Jozdani, S.E., Johnson, B.A., & Chen, D. 2019. Comparing Deep Neural Networks, Ensemble Classifiers, and Support Vector Machine Algorithms for Object-Based Urban Land Use/Land Cover Classification. *Remote Sensing*, *11*(14), 1713. https://doi.org/10.3390/rs11141713

Kamusoko, C. 2019. *Remote Sensing Image Classification in R*. Springer Singapore. https://doi.org/10.1007/978-981-13-8012-9

Karakacan Kuzucu, A., & Bektas Balcik, F. 2017. Testing the potential of vegetation indices for land use/cover classification using high resolution data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4/W4, 279-283. https://doi.org/10.5194/isprs-annals-IV-4-W4-279-2017

Kelleher, J.D., Namee, B.M., & D'Arcy, A. 2015. *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. MIT Press.

Knox, S.W. 2018. *Machine learning: A concise introduction*. John Wiley & Sons.

Koomen, E., Stillwell, J.(2007) Modelling land-use change en Koomen, E., Stillwell, J., Bakema, A., & Scholten, H.J. *Modelling land-use change: Progress and applications* (Vol. 90)(1-21). Springer Science & Business Media.

Kuhn, M., & Johnson, K. 2013. *Applied Predictive Modeling*. Springer New York. https://doi.org/10.1007/978-1-4614-6849-3

Li, C., Wang, J., Wang, L., Hu, L., & Gong, P. 2014. Comparison of Classification Algorithms and Training Sample Sizes in Urban Land Classification with Landsat Thematic Mapper Imagery. *Remote Sensing*, *6*(2), 964-983. https://doi.org/10.3390/rs6020964

Lillesand, T., Kiefer, R.W., & Chipman, J. 2015. *Remote Sensing and Image Interpretation*. Wiley.

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. https://doi.org/10.1080/01431160600746456

Marsland, S. 2014. *Machine Learning: An Algorithmic Perspective* (2.a ed.). Chapman and Hall/CRC. https://doi.org/10.1201/b17476

Mas, J.F., & Flores, J.J. 2008. The application of artificial neural networks to the analysis of remotely sensed data. *International Journal of Remote Sensing*, *29*(3), 617-663. https://doi.org/10.1080/01431160701352154

Maxwell, A.E., Warner, T.A., & Fang, F. 2018. Implementation of machine-learning classification in remote sensing: An applied review. *International Journal of Remote Sensing*, *39*(9), 2784-2817. https://doi.org/10.1080/01431161.2018.1433343

Mfuka, C., Zhang, X., & Byamukama, E. 2019. Mapping and Quantifying White Mold in Soybean across South Dakota Using Landsat Images. *Journal of Geographic Information System*, *11*(03), 331-346. https://doi.org/10.4236/jgis.2019.113020

Müller, A.C., & Guido, S. 2016. *Introduction to machine learning with Python: A guide for data scientists*. O'Reilly Media, Inc.

Myburgh, G., & Niekerk, A. 2013. Effect of feature dimensionality on object-based land cover classification: A comparison of three classifiers. *South African Journal of Geomatics*. https://www.semanticscholar.org/paper/Effect-of-feature-dimensionality-on-object-based-A-Myburgh-Niekerk/298f8341429248311f9a688741d0ee4344aa404c

Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

Petropoulos, G.P., Kalaitzidis, C., & Prasad Vadrevu, K. 2012. Support vector machines and object-based classification for obtaining land-use/cover cartography from Hyperion hyperspectral imagery. *Computers & Geosciences*, *41*, 99-107. https://doi.org/10.1016/j.cageo.2011.08.019

Pontius, R.G., & Millones, M. 2011. Death to Kappa: Birth of quantity disagreement and allocation disagreement for accuracy assessment. *International Journal of Remote Sensing*, *32*(15), 4407-4429. https://doi.org/10.1080/01431161.2011.552923

Puertas, O.L., Brenning, A., & Meza, F.J. 2013. Balancing misclassification errors of land cover classification maps using support vector machines and Landsat imagery in the Maipo river basin (Central Chile, 1975–2010). *Remote Sensing of Environment*, *137*, 112-123. https://doi.org/10.1016/j.rse.2013.06.003

Qian, Y., Zhou, W., Yan, J., Li, W., & Han, L. 2015. Comparing Machine Learning Classifiers for Object-Based Land Cover Classification Using Very High Resolution Imagery. Remote Sensing, *7*(1), Art. 1. https://doi.org/10.3390/rs70100153

Olofsson, P., Foody, G.M., Herold, M., Stehman, S. V., Woodcock, C.E., & Wulder, M.A. 2014. Good practices for estimating area and assessing accuracy of land change. *Remote Sensing of Environment, 148*, 42-57. https://doi.org/10.1016/j.rse.2014.02.015

R Core Team. 2022. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

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), Art. 3. https://doi.org/10.3390/rs13030368

Rana, V.K., & Venkata Suryanarayana, T.M. 2020. Performance evaluation of MLE, RF and SVM classification algorithms for watershed scale land use/land cover mapping using sentinel 2 bands. *Remote Sensing Applications: Society and Environment, 19*, 100351. https://doi.org/10.1016/j.rsase.2020.100351

Richards, J.A. 2013. *Remote sensing digital image analysis: An introduction* (Fifth edition). Springer.

Saini, R., & Ghosh, S.K. 2018. Exploring capabilities of sentinel-2 for vegetation mapping using random forest. ISPRS - *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* XLII-3, 1499-1502. https://doi.org/10.5194/isprs-archives-XLII-3-1499-2018

Shalev-Shwartz, S., & Ben-David, S. 2014. Understanding Machine Learning: From Theory to Algorithms. *Cambridge University Press*. https://doi.org/10.1017/CBO9781107298019

Shih, H., Stow, D.A., & Tsai, Y.H. 2019. Guidance on and comparison of machine learning classifiers for Landsat-based land cover and land use mapping. *International Journal of Remote Sensing, 40*(4), 1248-1274. https://doi.org/10.1080/01431161.2018.1524179

Syifa, M., Park, S.J., & Lee, C.W. 2020. Detection of the Pine Wilt Disease Tree Candidates for Drone Remote Sensing Using Artificial Intelligence Techniques. *Engineering, 6*(8), 919-926. https://doi.org/10.1016/j.eng.2020.07.001

Szuster, B.W., Chen, Q., & Borger, M. 2011. A comparison of classification techniques to support land cover and land use analysis in tropical coastal zones. *Applied Geography*, *31*(2), 525-532. https://doi.org/10.1016/j.apgeog.2010.11.007

Talukdar, S., Singha, P., Mahato, S., Shahfahad, 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. https://doi.org/10.3390/rs12071135

Tassi, A., & Vizzari, M. 2020. Object-Oriented LULC Classification in Google Earth Engine Combining SNIC, GLCM, and Machine Learning Algorithms. *Remote Sensing, 12*(22), 3776. https://doi.org/10.3390/rs12223776

Thanh Noi, P., & Kappas, M. 2017. Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. *Sensors*, *18*(2), 18. https://doi.org/10.3390/s18010018

Tso, B., & Mather, P.M. 2009. *Classification methods for remotely sensed data* (2nd ed). CRC Press.

Thomas, I.L., Ching, N.P., Benning, V.M., & D'aguanno, J.A. 1987. Review Article A review of multi-channel indices of class separability. *International Journal of Remote Sensing*, *8*(3), 331-350. https://doi.org/10.1080/01431168708948645

Vélez-Alvarado, D.A., & Álvarez-Mozos, J. 2020. Clasificación de usos y cubiertas del suelo y análisis de cambios en los alrededores de la Reserva Ecológica Manglares Churute (Ecuador) mediante una serie de imágenes Sentinel-1. *Revista de Teledetección*, *56*, 131. https://doi.org/10.4995/raet.2020.14099

Wilson, R.A., & Keil, F.C. (Eds.). 1999. *The MIT encyclopedia of the cognitive sciences*. MIT Press.

Yu, L., Liang, L., Wang, J., Zhao, Y., Cheng, Q., Hu, L., Liu, S., Yu, L., Wang, X., Zhu, P., Li, X., Xu, Y., Li, C., Fu, W., Li, X., Li, W., Liu, C., Cong, N., Zhang, H., … Gong, P. 2014. Meta-discoveries from a synthesis of satellite-based land-cover mapping research. *International Journal of Remote Sensing*, *35*(13), 4573-4588. https://doi.org/10.1080/01431161.2014.930206