



Proyecto Fin de Master en Sistemas Inteligentes

"Master en Investigación en Informática, Facultad
de Informática, Universidad
Complutense de Madrid"

Algoritmo para la localización y estimación de masa
forestal a partir de Imágenes LiDAR

Autor: M^a Pilar Palomino Palomino

Director: Luis Garmendia Salvador

Colaborador externo: Francisco Mauro Gutiérrez

*Grupo de Investigación sobre Tecnologías y Métodos para la
Gestión Sostenible.*

Curso Académico: 2º



El/la abajo firmante, matriculado/a en el Master en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Master: "Algoritmo para la localización y estimación de masa forestal a partir de Imágenes LiDAR", realizado durante el curso académico 2008-2009 bajo la dirección de Luís Garmendia Salvador y con la colaboración externa de dirección de Francisco Mauro Gutiérrez de la E.T.S. Montes, UPM, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.



Agradecimientos

He de expresar mi profundo agradecimiento a todas aquellas personas que me han ayudado a llevar a cabo este trabajo, sin ellas no hubiese sido posible:

Dr. Luis Garmendia, mi director de proyecto, cuya importancia ha sido vital en el desarrollo del mismo. Me ha aportado sus conocimientos y experiencia y gracias a él he conseguido iniciarme en el mundo de la investigación en el que me gustaría seguir trabajando. Su influencia ha sido de vital importancia para la realización de este trabajo.

Francisco Mauro, colaborador experto en el área forestal, gracias por su ayuda incondicional en todo momento y sus horas de dedicación de manera desinteresada. Él me ha contagiado su pasión por la investigación y por el mundo forestal, sin él no hubiese sido posible.

E.T.S. Montes, UPM, por proporcionar todas las imágenes y material necesario para poder llevar a cabo esta investigación.

Por último, en el apartado personal, mi gratitud a mi familia por apoyarme en todo momento para seguir adelante, sobre todo durante este año tan duro. También gracias a Nacho que ha sabido entender mi falta de dedicación durante estos últimos meses y siempre ha estado ahí dándome ánimo.



Índice de contenidos

1	Introducción	9
1.1	Contexto del Trabajo	9
1.2	Motivación y Objetivos	12
1.2.1	Motivación.....	12
1.2.2	Objetivos.....	12
1.3	Estructura del Trabajo	13
2	Estado del arte	15
2.1	Características de la tecnología LiDAR.....	16
2.2	Detección de árboles individuales usando imágenes LiDAR	20
2.2.1	Fundamentos de los distintos métodos de detección de árboles individuales	22
2.2.2	Algoritmo de Wang.....	30
2.3	Problemática que se presenta	32
2.4	Métodos de clasificación de puntos.....	33
2.4.1	Tipos de clustering.....	35
2.5	Lógica Fuzzy	37
2.5.1	Conjuntos borrosos y lógicas borrosas.....	37
2.5.2	Clustering Borroso	39
3	Descripción del problema	43
3.1	Problemática actual	43
3.2	Objetivo, definición y límites del problema.....	43
3.3	Importancia de su resolución	44
3.4	Casos reales	44
4	Algoritmo que se presenta.....	50
4.1	Datos de partida.....	50
4.2	Algoritmo para la localización y estimación de masa forestal a partir de imágenes LiDAR	51
4.3	Descripción de las fases del algoritmo.....	53
4.3.1	Pre-procesamiento de la nube de puntos LiDAR.....	53
4.3.2	Creación Espacio Voxel.....	57
4.3.3	Modelado de copas de árboles individuales	58



4.3.4	Situación de conflicto:	63
4.4	Calcular la localización de cada árbol y la altura	66
4.5	Resolver el problema de la segmentación.....	67
5	Implementación	68
5.1	LocalizedTree	69
5.2	Vox	69
5.3	EspacioVoxel	70
5.4	Árbol	70
6	Resultados Obtenidos	71
7	Evaluación cualitativa de los resultados por expertos	76
8	Evaluación de los resultados obtenidos por métodos automáticos para la detección de árboles.....	84
8.1	<i>Evaluación de la precisión geométrica.</i>	84
8.2	<i>Evaluación del número de aciertos-fallos.</i>	86
8.3	<i>Evaluación de la precisión en la estimación de parámetros forestales.</i>	87
8.3.1	Estimación de Alturas medias	89
8.3.2	Estimación de número de Pies	93
8.3.3	Estimación de biomasa por hectárea	98
8.4	<i>Evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.</i>	102
8.5	<i>Evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.</i>	113
9	Conclusiones	121
9.1	Aportaciones	122
9.2	Futuras líneas de trabajo	123
10	Anexo A: Resultados obtenidos con Statgraphics Comparación de dos muestras- Alturas y ZMax.....	127
10.1	Parcela Volante 1	127
10.2	Parcela Volante 2	131
10.3	Parcela Volante 3	135
10.4	Parcela Volante 4	139
10.5	Parcela Volante 5	143



10.6	Parcela Volante 6	146
10.7	Parcela Volante 7	149
10.8	Parcela Volante 8	152
10.9	Parcela Volante 9	155
10.10	Parcela Volante 10.....	157
10.11	Parcela Volante 11.....	159
10.12	Parcela Volante 12.....	161
10.13	Parcela Volante 13.....	163
10.14	Parcela Volante 14.....	166
10.15	Parcela Volante 15.....	169
10.16	Parcela Volante 16.....	172
10.17	Parcela Volante 17.....	175
10.18	Parcela Volante 18.....	178
10.19	Parcela Volante 19.....	180
10.20	Parcela Volante 20.....	182
10.21	Parcela Volante 21.....	185
10.22	Parcela Volante 22.....	187
10.23	Parcela Volante 23.....	190
10.24	Parcela Volante 24.....	193
10.25	Parcela Volante 25.....	196
11	Anexo B: Modelos de regresión para la calibración de las predicciones	199
12	Bibliografía	222



ABSTRACT

This work describes an algorithm to analyze light detection and ranging (LiDAR) datasets with the aim of estimating forest biomass, positioning trees and measuring their heights and crown dimensions. The algorithm is based on previous studies developed in Germany that showed several problems when working with irregularly shaped trees. This algorithm tries to improve existing methods for this purpose. The detection of trees presents numerous challenges that are described in this work. Clustering methods and fuzzy logic have been used to decide whether or not each point stored in the LiDAR dataset belonged to an existing tree. Finally, the results obtained by the algorithm are satisfactory for the average height, while the number of trees per hectare is overestimated and consequently also too biomass per hectare. The last ones can be improved by adjusting certain parameters. The algorithm has been implemented using gvSIG libraries and the evaluation of results has been done by Statgraphics and R.

KEYWORDS

Laser scanning; segmentation; DTM; DSM; LiDAR; rasterization; Fuzzy clustering; forest measurements, estimation, forest biomass.

RESUMEN

Este trabajo describe un algoritmo para estimar parámetros importantes para la gestión de masas forestales, mediante el posicionamiento de los árboles la medición automática de sus dimensiones partiendo de imágenes LiDAR. Este trabajo se basa en estudios previos realizados en Alemania y muestra los problemas que se presentan en la búsqueda de árboles de diferentes tamaños y formas. Con este algoritmo se pretende mejorar los métodos existentes hasta ahora para este fin. Para llevar a cabo este algoritmo se han usado métodos de clustering y lógica fuzzy. A partir de métodos de clustering y lógica Fuzzy se decide si un punto pertenece o no a un árbol existente. Finalmente los resultados obtenidos por el algoritmo son satisfactorios para la altura media, mientras que el número de pies por hectárea se sobreestima excesivamente y en consecuencia también la biomasa por hectárea. Estos últimos se pueden mejorar ajustando determinados parámetros. El algoritmo ha sido implementando usando librerías de gvSig y la validación de los resultados obtenidos se ha llevado a cabo con Statgraphics y R.



PALABRAS CLAVE

Escaneo láser; segmentación; DTM; DSM; LiDAR; rasterización; agrupación borrosa; parámetros de campo, estimación, biomasa forestal.



1 Introducción

El presente capítulo describe el contexto de la trabajo (sección 1.1) seguido de los objetivos del mismo (sección 1.2) y finaliza con su estructura (sección 1.3).

1.1 Contexto del Trabajo

El cálculo de posiciones de puntos partiendo de las coordenadas conocidas de otros puntos y de un conjunto de mediciones de ángulos y distancias es el pilar fundamental de la topografía y las técnicas de posicionamiento (GPS, levantamientos taquimétricos clásicos, nivelaciones). Los sensores LiDAR (Light Detection and Ranging o Láser Imaging Detection and Ranging) no ofrecen ninguna novedad en cuanto a los principios en los que basan su funcionamiento.

Sin embargo, la aparición de sensores LIDAR ha supuesto una revolución en lo que se refiere a captura de datos espaciales. Los sensores LIDAR son equipos láser que permiten calcular la posición de puntos en el espacio de una forma masiva. Existen variantes de este tipo de aparatos, pero la que está desarrollándose con más fuerza en el campo de la investigación forestal es la variante aerotransportada. Ésta consiste en montar un sensor láser en un avión o helicóptero y sobrevolar la zona a estudiar escaneando la superficie terrestre.

Conceptualmente el funcionamiento de estos sensores es sencillo pues a partir de las coordenadas conocidas del emisor láser y de las mediciones efectuadas de ángulos y distancias se calculan, por trigonometría, las coordenadas de puntos en el espacio cuya posición se desconocía. Sin entrar en detalles complejos referentes a cuestiones logísticas el posicionamiento basado en mediciones de ángulos y distancias mediante técnicas LiDAR se puede resumir en el siguiente esquema:

- El sensor emite pulsos láser y anota la posición en la que se encuentra en ese instante y la dirección en la que ha lanzado los pulsos.
- Algunos pulsos chocan contra la cubierta vegetal y vuelve hacia el sensor, otros pulsos siguen su camino hasta chocar contra el suelo y una vez chocan contra la tierra rebotan y vuelven hacia el avión.



- El sensor recibe los rebotes de los pulsos emitidos y calculando el tiempo que han tardado en ir y volver calcula las distancias a las que han chocado.
- Conocidas las distancias a las que han chocado los pulsos, las direcciones en las que éstos fueron lanzados y la posición desde donde se emitieron se reconstruye la posición original de los puntos donde rebotó cada uno.

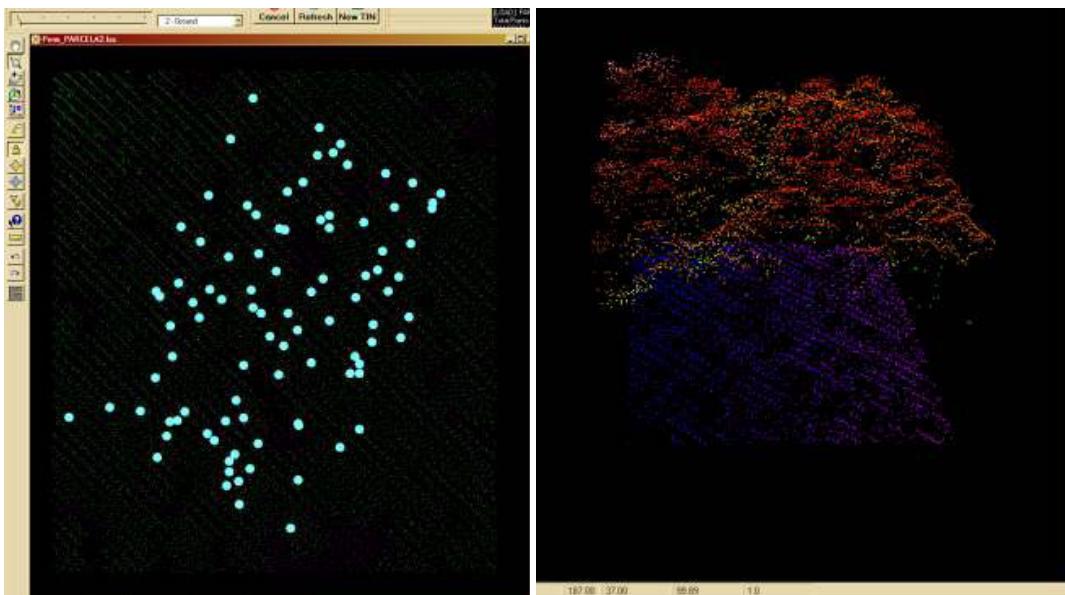


Figura 1. A) Imagen LiDAR vista desde arriba. B) Imagen LiDAR vista desde otro ángulo.

La gran ventaja que ofrecen estos sensores frente a otros de otro tipo es el elevado volumen de puntos que pueden registrar de forma precisa. Sin embargo la gran potencia de estos sensores para captar grandes cantidades de puntos en poco tiempo se paga una cierta incertidumbre. La toma de puntos se hace de una forma indiscriminada, se controla la zona por la que pasa el avión pero no el lugar exacto donde rebotan los puntos láser emitidos por el sensor. Así el resultado de un escaneo LIDAR es una nube de millones de puntos, almacenados normalmente en un fichero tipo *.las, de los cuales parte pertenecen al suelo y parte pertenecen a la superficie de las copas de los árboles o a la superficie de otros objetos que se asienten sobre el suelo. Por ejemplo edificios. El problema de discriminar qué pulsos son suelo y cuales son copas de árboles ha sido resuelto como explica Kraus [12], sin embargo aún hay problemas importantes que resolver.

Las técnicas actuales clasifican a todos los puntos que chocan en las copas de los árboles como puntos sobre vegetación, esta clasificación deja mucho que desear si se desea estimar el número de árboles presentes en el bosque.



Hay numerosos algoritmos para localizar los ápices de los árboles, contarlos y medirlos de forma automática, si bien la mayoría de ellos realizan un proceso previo de “rasterización” de la nube de puntos. En este proceso se pasa de una nube de puntos irregularmente distribuidos a una malla en dos dimensiones de puntos regularmente espaciada pero mucho menos poblada. En esta “rasterización” se pierde o se distorsiona la información pero se gana en operatividad. Las experiencias muestran que pese a que los algoritmos funcionan notablemente en ciertas condiciones hay otros casos en los que la mayor irregularidad de los bosques escaneados hace que estos algoritmos sean totalmente inviables.

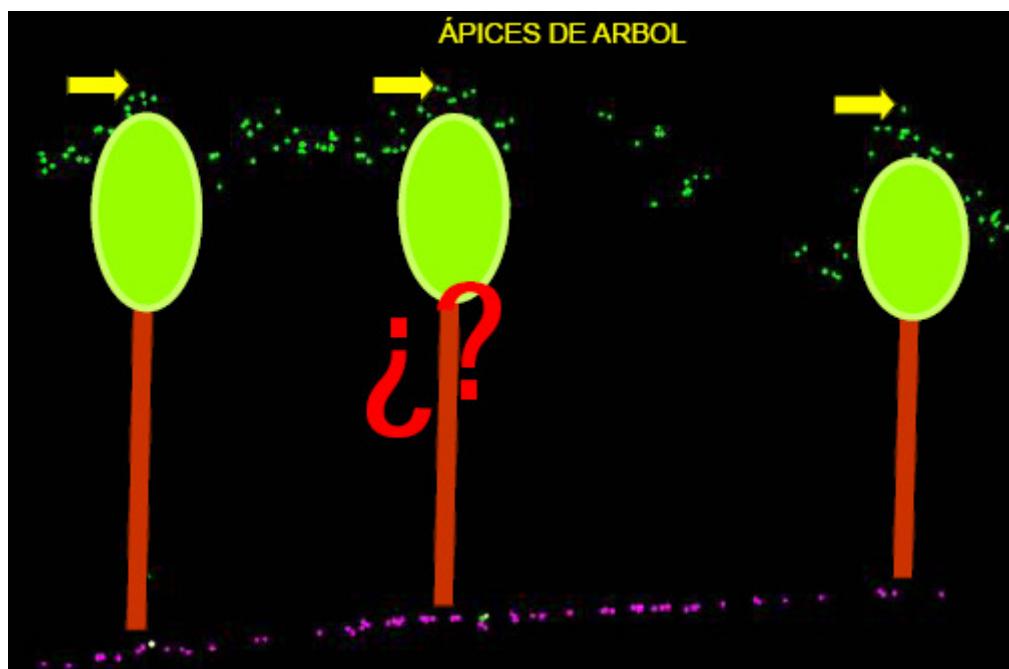


Figura 2. Perfil longitudinal extraído en una parcela de inventario forestal. Los puntos de color rosa se corresponden con puntos clasificados como suelo. Los puntos verdes son puntos clasificados como vegetación. Se puede intuir como los puntos verdes dibujan el contorno de las copas de los árboles.



1.2 Motivación y Objetivos

1.2.1 Motivación

El trabajo de investigación desarrollado tiene su origen en la necesidad planteada por la facultad de Ingenieros de Montes de la Universidad Politécnica de Madrid, con el fin de obtener un método que permita la clasificación de puntos procedentes de imágenes LiDAR para poder estimar la masa forestal y otros parámetros de forma automática sin necesidad de realizar un trabajo de campo. Las imágenes utilizadas para las pruebas son cortesía de esta facultad, al igual que el asesoramiento en todo momento sobre temas forestales, para los que hemos contado con la ayuda de Francisco Mauro.

1.2.2 Objetivos

Los objetivos planteados en este trabajo de investigación son los siguientes:

1. Estudio bibliográfico de la literatura existente sobre estimación de parámetros forestales partiendo de imágenes LiDAR.
2. Estudiar técnicas de inteligencia artificial existentes para detectar posiciones de árboles, masa forestal y otros parámetros partiendo de los datos de la nube de puntos.
3. Diseñar e Implementar un algoritmo que permita detectar árboles individuales, posicionarlos y estimar la masa forestal
4. Analizar los resultados obtenidos con el método implementado.
5. Identificar líneas de investigación futuras.
6. Divulgar y difundir los resultados de la investigación mediante publicaciones científicas u otros mecanismos.
7. Compartir la implementación para que otros puedan usarla y realizar nuevas aportaciones.



1.3 Estructura del Trabajo

Este trabajo está estructurado en 12 capítulos: introducción, estado del arte, descripción del problema, algoritmo que se presenta, implementación del algoritmo, resultados obtenidos, evaluación cualitativa de los resultados por expertos, evaluación de los resultados obtenidos por métodos automáticos para la detección de árboles, conclusiones, dos anexos con análisis estadísticos realizados con statgraphics y la bibliografía.

El capítulo 1 hace una introducción que describe el contexto del trabajo seguido por la motivación y objetivos del mismo y finaliza con su estructura.

El capítulo 2 corresponde al estado del arte donde se describen las características más relevantes de la tecnología LiDAR, métodos de detección de árboles usando LiDAR, aquí se explican las técnicas más generales y el algoritmo de Wang. En este capítulo se describe también la problemática que se presenta y se describen los métodos de clasificación de puntos existentes, que son diferentes tipos de clustering, se hace una breve descripción de en qué consisten y se nombran algunos autores que lo emplean en métodos de detección de árboles. En este capítulo se dedica una sección para hablar de lógica Fuzzy, explicando los conjuntos horrorosos y lógicas borrosas y en otra sección de este capítulo se explica el clustering borroso. Estos son métodos que se usan en nuestro algoritmo de detección de árboles.

En el capítulo 3 se hace una descripción del problema como su nombre indica se describe detalladamente el problema actual de detección de árboles, la importancia de su resolución y se muestran casos reales con los que se pretende acercar al lector a la problemática real que existe en este campo.

En el capítulo 4 se describe de forma detallada el algoritmo que se presenta en este trabajo, para ello se dedica una sección a describir cuáles serán los datos de partida, en otra sección se enumeran las fases del algoritmo, para pasar a describir cada una de ellas de forma detallada en la sección siguiente. En otra sección se explica la forma de localizar cada uno de los árboles y de obtener la altura. Finalizando este capítulo con una sección en la que se explica el problema de segmentación que se da en estos casos.

El capítulo 5 se dedica a la implementación, en él se describen las clases que se usan y las características y funcionalidad de cada una de ellas.



En el capítulo 6 se explican los resultados obtenidos con el algoritmo para dos parcelas con las que se han realizado las pruebas de entrenamiento y ajuste. Se muestran resultados gráficos y alfanuméricos, haciendo un comentario de ellos.

En el capítulo 7 se hace una evaluación cualitativa de los resultados obtenidos para las dos parcelas con las que se hacen las pruebas de ajuste y entrenamiento.

En el capítulo 8 se hace una evaluación de los resultados obtenidos por métodos automáticos para la detección de árboles. Esta evaluación se divide en evaluación de precisión geométrica, evaluación del número de aciertos-fallos, evaluación de la precisión en la estimación de parámetros forestales; en esta se hace una evaluación de alturas medias, número de pies y biomasa por hectárea. En otra sección de este capítulo se hace una evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado y en la última sección del capítulo se hace una evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.

En el capítulo 9 se describen las conclusiones, comentando las aportaciones y las futuras líneas de trabajo.

Los capítulos 10 y 11 son dos anexos con los resultados obtenidos con Statgraphics. En el primer anexo se comparan las alturas obtenidas de campo y las alturas obtenidas con el algoritmo para cada una de las 25 parcelas. En el segundo anexo se realizan modelos de regresión para la calibración de las predicciones. En estos dos anexos se pueden consultar los resultados obtenidos de la validación.

Por último el capítulo 12 presenta la bibliografía seguida para la realización de este trabajo.



2 Estado del arte

Este capítulo describe las características de la tecnología LiDAR, la detección de árboles individuales, la problemática que se presenta para realizar estudios sobre estimación de masa forestal, se explica también en qué consiste el clustering y los distintos tipos de clustering y se explica en qué consiste la lógica fuzzy.



2.1 Características de la tecnología LiDAR

El LiDAR aerotransportado (Light Detection And Ranging) es un sistema activo basado en un sensor láser que se puede instalar en aviones o helicópteros. Esta tecnología está sustituyendo cada vez más a los métodos basados en fotogrametría para obtener MDE (Modelos Digitales de elevaciones) de alta precisión, porque consigue disminuir los costes y plazos de entrega a la vez que se obtienen precisiones mejores de 15 cm. en altura y una mayor densidad de medidas.

El funcionamiento del LiDAR consiste en un sensor de barrido, emite pulsos láser y mide el tiempo que tardan dichos pulsos en llegar a la superficie terrestre y volver hasta el sensor. Además el sensor toma datos precisos acerca de la dirección en que cada pulso fue emitido. Actualmente hay instrumentos LiDAR que miden hasta con una frecuencia de 150 KHz., lo que quiere decir que emiten y reciben 150.000 pulsos láser por segundo.

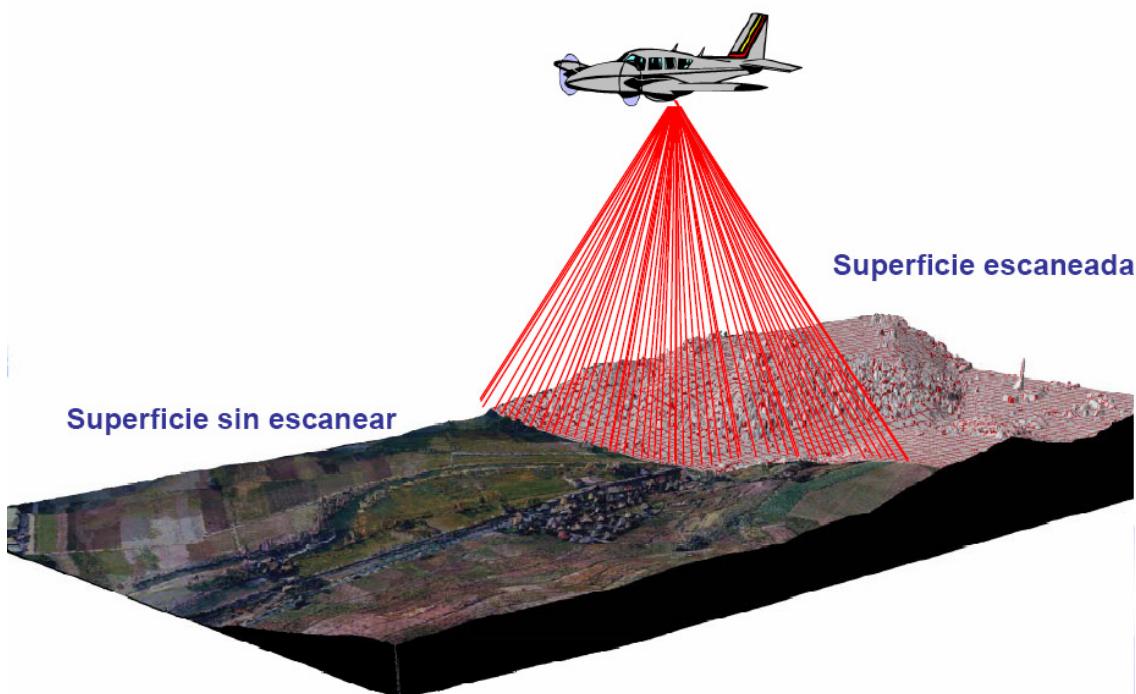


Figura 3. Barrido de un vuelo LiDAR.

En principio todos los rayos láser salen en una única dirección apuntando hacia abajo, pero hay un espejo rotatorio que los desvía a ambos lados del avión con un ángulo de apertura configurable, de forma que se realiza un barrido de una determinada franja



del terreno (la anchura dependerá de la altura del vuelo y del ángulo de apertura de escaneo) conforme avanza el avión.

Por otro lado, en el avión hay instalado un GPS que permite posicionar en tiempo real al instrumento, para saber en cada momento donde se encuentra. Para aumentar la precisión en el posicionamiento del instrumento, se utiliza un GPS diferencial en tierra. La técnica del GPS diferencial consiste en calcular la posición relativa de un GPS respecto a otro. Al medir la posición relativa de un equipo respecto al otro gran parte de los errores de posicionamiento se cancelan, de modo que esta medida gana precisión. Los errores que se cancelan son todos aquellos que son comunes para ambos receptores. Por ejemplo aquellos derivados de la distorsión de la señal GPS cuando esta cruza la Ionosfera. Empleando esta técnica y ubicando el segundo GPS en un punto de coordenadas conocidas se aumentan las precisiones hasta el rango del medio centímetro cuando el GPS móvil se mueve a velocidades pequeñas. Un avión o un helicóptero son objetos que se mueven a una velocidad considerable de modo que tras realizar el vuelo y combinar los datos GPS del avión con los del GPS diferencial en tierra, se pueden obtener precisiones de 5 cm. en la posición del instrumento.

Por último, se dispone del sistema de navegación inercial del avión (INS) que mide con una precisión de 0.001 grados las variaciones en la orientación del avión debidas a turbulencias, pequeños giros, inclinaciones, etc., de forma que se pueden compensar dichos movimientos.

Combinando toda esta información (distancia medida con el láser, ángulo del espejo que realiza el barrido, posición exacta del instrumento con GPS y sistema de navegación inercial), podemos obtener la medición de la altura del terreno con una precisión mejor que 15 cm. en Z.

Cuando el rayo láser llega al terreno se comporta de forma diferente dependiendo de las características de los objetos que se encuentre.

1. En una superficie sólida (edificios, suelo, etc.), el rayo se refleja sin ningún problema y vuelve al avión.
2. En el agua el rayo láser sufre una reflexión especular y no vuelve al avión, por lo que no se obtiene ninguna información, apareciendo un hueco sin datos que habrá que interpolar a partir de las alturas de alrededor o de otros datos batimétricos de los que se pueda disponer.
3. En el caso de la vegetación, el rayo choca en primer lugar con la copa del árbol. En este momento parte del rayo se refleja y vuelve al avión, pero al tratarse de una



superficie no sólida, hay otra parte del rayo que atraviesa la vegetación hasta llegar al suelo, en ese momento rebota y vuelve al avión. El sistema guarda, al menos, las coordenadas y alturas del primer y último pulso. El registro del primer pulso permite describir la superficie (parte más alta) de los objetos, mientras que el registro del último eco se usa para conocer la superficie del suelo. La capacidad multi-retorno se utiliza para detectar vegetación y su altura.

Actualmente es frecuente encontrar sistemas que son capaces de percibir hasta cuatro o cinco retornos parciales debidos a ramas troncos etc.... antes de captar el retorno del suelo. Yendo aun más lejos, los sistemas más modernos conocidos como sistemas de onda completa o fullwaveform, recogen de forma casi continua cual ha sido la historia del pulso al penetrar la cubierta vegetal. Estos sistemas aumentan la cantidad de información captada y añaden nuevas funcionalidades, pero salvo para aplicaciones específicas, los sistemas de primer y último pulso son perfectamente válidos.

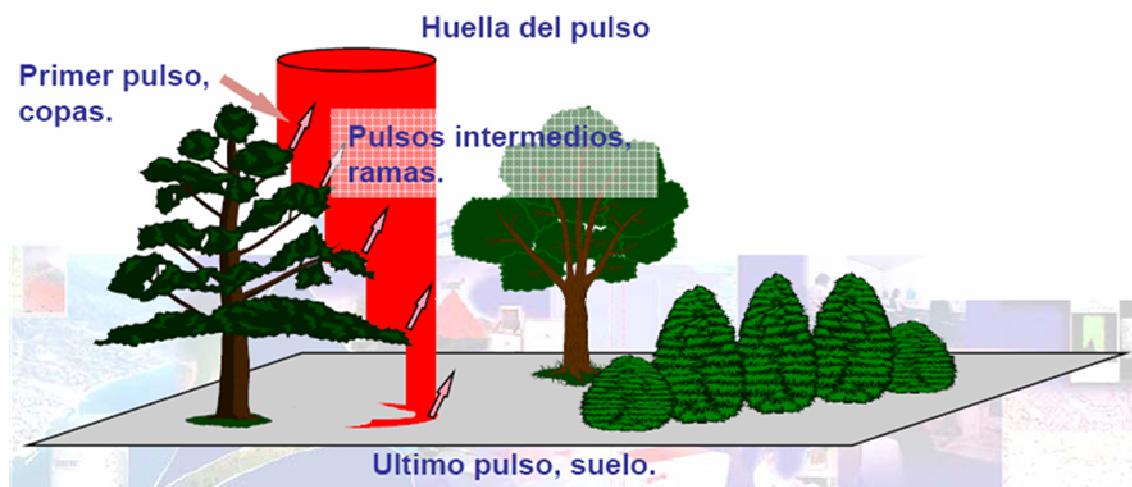


Figura 4. Zona de influencia de un pulso Láser.

Una de las ventajas de la tecnología LiDAR, derivada de su capacidad para atravesar las cubiertas vegetales, consiste en la posibilidad de medir la altura del terreno en zonas completamente cubiertas por arbolado o matorral no denso, lo cual no es posible con técnicas de medición clásicas como la fotogrametría. Esta es una de las principales razones por las que los sistemas LiDAR se están imponiendo a la hora de generar modelos de elevaciones destinados a usos que no requieren más de 10-15cm de precisión en Z.

Un proyecto LiDAR se puede dividir en dos bloques claramente diferenciados: la adquisición de los datos y el procesado de los mismos. Hasta el momento hemos hecho referencia a la parte de adquisición de los datos, pero sin duda la parte más



importante está en el procesado de éstos para obtener los diferentes productos finales. Los productos estándar obtenidos después del procesado de los datos LiDAR se explican a continuación:

- Nube de puntos irregular, en la que se almacena toda la información relevante de cada uno de los ecos recibidos por el sensor.
- MDS (Modelo Digital de Superficie) se obtiene generalmente interpolando los puntos del primer rebote.

Aquí podemos distinguir las alturas de los edificios, vegetación, puentes, coches, farolas, etc.

- Clasificaciones de los puntos pertenecientes a suelo, vegetación, edificios, puentes, líneas eléctricas, etc.
- MDT (Modelo Digital del Terreno) se obtiene interpolando los puntos del último pulso, eliminando los puntos que no pertenecen al terreno.
- Imagen de intensidades. A partir de la amplitud de la señal que vuelve al avión después de rebotar en la superficie terrestre obtenemos una imagen de intensidades que permite realizar distinciones entre superficies, identificando carreteras, etc.

A partir de estos productos estándar, se pueden obtener otros productos derivados que se pueden aplicar a numerosos campos.

En nuestro caso nos vamos a centrar en las aplicaciones en el campo forestal: Dentro del campo forestal la tecnología LiDAR tiene una gran aceptación, gracias a la ventaja de poder medir tanto la altura de la parte alta de la vegetación como la altura del terreno, lo que permite obtener directamente la altura de la vegetación incluso en zonas de vegetación densa donde con el uso de sensores ópticos no es posible tener una visión directa del terreno.

Con datos LiDAR, actualmente, se puede calcular manualmente de forma más o menos precisa la altura de cada árbol individual, localizar la posición del tronco de cada árbol individual, realizar cálculos de volúmenes, obtener la altura media del rodal, realizar cálculos del índice de área foliar (LAI), etc. Todos estos parámetros son de especial importancia a la hora de realizar un inventario forestal y actualmente se obtienen mediante estimaciones realizadas con trabajo de campo. En este trabajo nos vamos a centrar en la detección de árboles individuales, dado que es uno de los pasos previos para estimar distintas variables masa forestal y en la localización de las posiciones de los árboles.



2.2 Detección de árboles individuales usando imágenes LiDAR

Desde que se realizaron los primeros ensayos con sensores láser en entornos forestales comenzaron a desarrollarse modelos y técnicas para la estimación de variables de interés forestal. Durante los primeros pasos de esta tecnología todos los modelos estaban basados en estimaciones generales de las variables a medir. Estos modelos llamados modelos a nivel de rodal, parcela o superficie, son modelos de base estadística y relacionan variables obtenidas directamente del sensor con variables forestales. Las relaciones generalmente se realizan con modelos de regresión o decisión más o menos complejos que proporcionan estimaciones para una superficie.

El constante desarrollo de los sensores, provocó un incremento en las densidades de puntos por metro cuadrado que se podían obtener. A mediados-finales de los años noventa este aumento en la cantidad de puntos por unidad de superficie permitió pensar en detectar árboles individuales, con lo que comenzaron a desarrollarse diversas técnicas y algoritmos orientados a este propósito. El mayor o menor éxito de estos métodos ha estado constantemente relacionado con dos factores, uno las condiciones del bosque y otro la mayor o menor densidad de retornos recibidos por el sensor. A esta nueva forma de proceder se le llamó “Métodos basados en la detección de árboles individuales”.

Actualmente, los métodos estadísticos basados en estimaciones a nivel de superficie, tienen un nivel de desarrollo superior. En gran parte esto se debe a que su aplicación es generalmente más sencilla y no requieren grandes desarrollos, ni técnicas complejas de inteligencia artificial. Por su parte los métodos de detección automática de árboles individuales son uno de los retos que aún quedan por superar en este campo.

Actualmente, y como norma general en la que caben muchas excepciones, se podría decir que los métodos de detección automática de árboles fallan más cuanto más irregular es la masa y cuanto mayor es la presencia de arbolado joven bajo las copas de los árboles maduros. Además, dado que es la probabilidad de que un pulso rebote justo en el ápice de un árbol las estimaciones de alturas generalmente infra-estiman las alturas reales. Así, a día de hoy se han logrado resultados satisfactorios al localizar árboles de forma automática en plantaciones o masas similares de gran homogeneidad en cuanto a edades y tamaños. Sin embargo en masas forestales con una mayor diversidad estructural los resultados deben ser mejorados.



Figura 5. Masa regular. Todos los árboles tienen unas dimensiones similares y no hay un estrato de regeneracion con árboles pequeños.

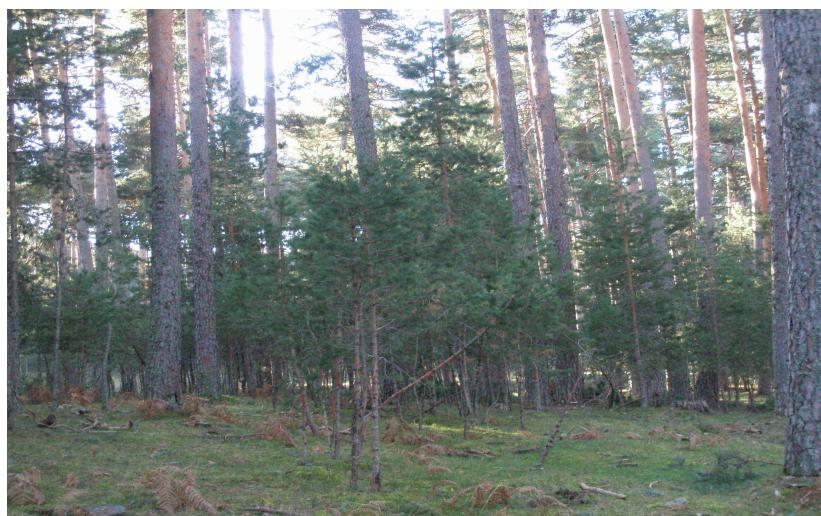


Figura 6. Masa irregular. Entre los pies de mayor tamaño aparecen dispersos árboles jóvenes de dimensiones menores.

A continuación se va ha hacer un breve resumen de los métodos que se han usado hasta ahora para la detección de árboles partiendo de imágenes LiDAR.

Hyyppä e Inkinen (1999) [11] fueron los primeros en obtener los modelos de copas individuales basados en inventarios forestales mediante escáner láser utilizando máximos del modelo de alturas de copas. El método fue probado junto con otros dos algoritmos de segmentación en bosques de coníferas finlandés, austriaco y alemán. Los resultados que se obtuvieron fueron del 40 a 50% de los árboles obtenidos



correctamente. Persson et al. (2002) [18] mejora la delimitación de la copa y llega a vincular el 71% de los árboles resultantes con alturas de árboles de referencia. Otros intentos de utilizar de modelo digital del suelo (DSM) o el modelo de altura de las copas (CHM) para detectar copas de árboles individuales o estimar diámetros de copas han sido realizados por Brandtberg et al. (2003) [2], Leckie et al. (2003) [13], Straub (2003) [20] y Popescu et al. (2003) [3].

Andersen et al. (2002) [3] propone que los modelos de copas se ajusten a un elipsoide en un marco Bayesiano de la nube de puntos. Morsdorf et al. (2003) [14] presentó un práctico procedimiento en dos etapas, donde la localización de los árboles fue definida usando los modelos digitales del suelo y los máximos locales, mientras que la delimitación de la copa fue definida usando clustering k-means en la nube de puntos tridimensional. Wack et al. (2003) [21] primero calculó la altura de todos los puntos láser y los ordenó y definió un cono como parte superior de los árboles. Si el punto se localizaba muy cerca del árbol, este se borraba, el proceso era recursivo. El 93% de los eucaliptos plantados pudieron ser correctamente delimitados.

2.2.1 Fundamentos de los distintos métodos de detección de árboles individuales

La mayoría de los métodos de detección de árboles individuales mencionados en el epígrafe anterior tienen en común un proceso de pasterización sobre los datos de partida. Gran parte de los problemas de los métodos de detección de árboles individuales están relacionados con la rasterización. La forma en que afecta el proceso de rasterización a los resultados obtenidos puede apreciarse en cierta medida en la siguiente explicación. Esta explicación simplifica en exceso pero ilustra de una forma muy gráfica los procesos de rasterización y los problemas que acarrea.

Si se dejase caer una fina tela sobre la nube de puntos. Esta tela dibujaría de forma continua la forma de las copas de los árboles. Si posteriormente se mide la altura de esta fina tela, en los nodos de una malla equiespaciada, se obtiene una representación organizada de una forma mucho más sencilla de la información captada por la nube de puntos. El problema es que cuando aparecen generaciones jóvenes de arbolado bajo el dosel principal de las copas, la tela imaginaria no llega a contactar con ellas. De modo que la malla que se obtiene ha perdido la información de esas capas de regenerado.

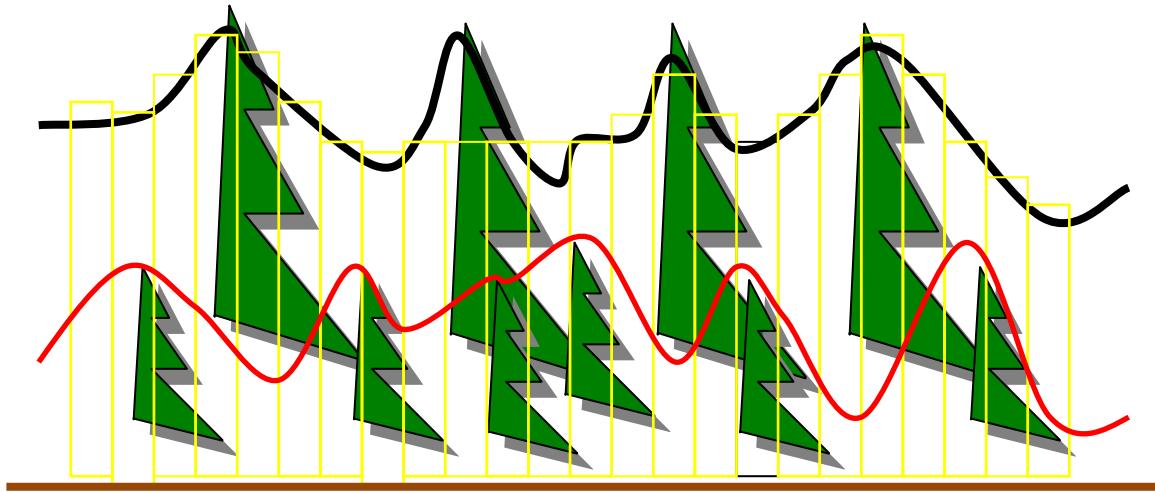


Figura 7. Pérdida de información en el proceso de rasterización.

En la figura anterior se muestra una línea roja que representa la información que se pierde, en el proceso de rasterización. Toda esa información pertenece a árboles bajos, de ahí que la mayor parte, por no decir todos los algoritmos que buscan árboles rasterizando la información de partida, pierdan los árboles de alturas más bajas.

Algunos de estos métodos se habían desarrollado parcialmente para su aplicación sobre imágenes convencionales. La idea subyacente a estos métodos es que para ciertas longitudes de onda o para ciertas combinaciones de “color” la respuesta observada cerca de los ápices de los árboles eran, netamente distintas de las respuestas observadas en zona alejadas del centro del árbol. Al menos visualmente se puede percibir un gradiente en la respuesta observada de forma que las zonas centrales del árbol presentan una intensidad de color máximo que va decreciendo gradualmente al acercarnos al borde de la copa del árbol. Intentando explotar este hecho, se comenzaron a aplicar algoritmos para detectar de forma automática máximos locales. Estos máximos locales se consideraban como árboles y en base a la cantidad de árboles detectados se hacían diferentes predicciones.

La aparición de sensores LiDAR, ha venido a proporcionar conjuntos de datos relativos a las alturas del arbolado, en los cuales, es claro el gradiente de alturas que se establece en las copas de los árboles. Aunque los datos procedentes de los sensores LiDAR, originalmente no poseen un formato matricial, es posible convertirlos a un formato ráster, pero hay que tener en cuenta que en este proceso siempre hay una pérdida de información. Una vez realizado este proceso de rasterización se pueden aplicar algunos de los algoritmos ya existentes para el tratamiento de imágenes.



Las primeras experiencias se llevaron a cabo en el mundo forestal con sensores LiDAR y algoritmos de detección de árboles fueron realizadas por Hyppä [11]. Estas experiencias se realizaron aplicando algoritmos sencillos de detección de máximos locales en ventanas de tamaño fijo, y no se introdujo ninguna modificación, que hiciese que el algoritmo adaptase a las características locales del arbolado.

Con el fin de mejorar los resultados obtenidos en la detección de árboles individuales, se empezaron a aplicar algoritmos más flexibles y complejos, algunos de estos algoritmos son adaptaciones de métodos desarrollados con anterioridad para otros fines. Así aparecen algoritmos de búsqueda de máximos locales empleando ventanas de tamaño variable llevados a cabo por Popescu [19]. Estos algoritmos intentan proporcionar estimaciones más ajustadas de la altura del arbolado o algoritmos capaces de proporcionar estimaciones del tamaño de la copa del árbol (Gougeon [7]).

Actualmente muchos trabajos de investigación se centran en el desarrollo de algoritmos complejos, con múltiples variables, para la segmentación de imágenes y delineación de árboles individuales, trabajando sobre datos ráster.

Finalmente, en paralelo a los métodos para imágenes ráster están empezando a desarrollarse algoritmos para la detección de árboles individuales aplicables a los datos brutos procedentes de los sensores LiDAR. Esta variante pretende dejar a un lado los procesos de rasterización, en los que siempre hay una pérdida de información, y tratan de explotar al máximo todos los datos almacenados durante el vuelo. El método de Wang [22] es uno de los primeros métodos desarrollados para datos brutos.

2.2.1.1 Método 1. Método de máximo local con ventana fija de tamaño 3x3.

El método consiste en buscar árboles localizando los máximos locales. Las primeras experiencias llevadas a cabo por Hyppä [11] lo que proponía era realizar una búsqueda sobre ventanas de tamaño fijo (tamaño 3x3 sobretodo). Al realizar esta búsqueda anotaban como árboles aquellos píxeles que estuviesen rodeados por píxeles con alturas inferiores o iguales al píxel central. Estos métodos tienen la ventaja de su sencillez pero tienen una serie de desventajas como son, que proporcionan estimaciones generalmente menores de la altura del arbolado y que son sensibles al tamaño del píxel.

A continuación se muestra un sencillo gráfico de cómo funcionaría este método.



10	9	7
9	9	6
7	6	5

10	12	11
13	15	13
11	13	12

Figura 8. Funcionamiento del método de máximos locales. El píxel central de la ventana de la derecha se anotaría como un árbol, pues su altura es superior a la de todos los árboles que le rodean.

Hemos aplicado este método sobre las parcelas que hemos usado para realizar las pruebas de entrenamiento de nuestro algoritmo. Los resultados obtenidos encuentran menor cantidad de árboles de los que hay. A continuación se muestra un ejemplo con una de las parcelas (Parcela 1) en las que se puede ver el contraste entre la cantidad de árboles del inventario de campo (puntos en verde) con los árboles obtenidos por este método (puntos en amarillo). En el inventario de campo hay un total de 317 árboles y con este método se localizan 39, que es una cantidad considerablemente menor.

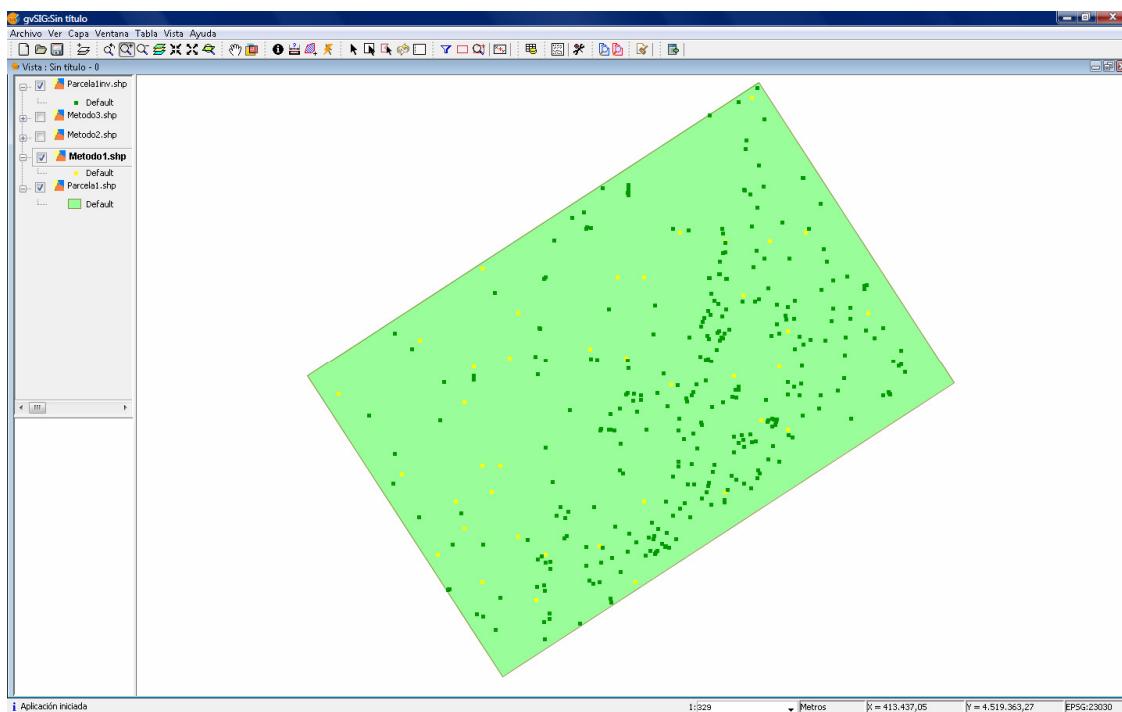


Figura 9. Método de máximos locales aplicado a la parcela 1.

Con la imagen anterior se observa el gran inconveniente que muestra este método. En principio la idea de tomar máximos locales es buena, pero con eso no basta hace falta algo más.



2.2.1.2 Método 2. Método de máximo local con ventana fija de tamaño 3x3 e interpolación de alturas.

El segundo método es un método de máximo local en el que además hay un proceso de interpolación. Partiendo de los datos de la ventana 3x3 se ajusta una superficie cuya expresión en $z=a_0+a_1x+a_2y+a_3xy+a_4x^2+a_5y^2$, al realizar la interpolación es posible obtener valores de Z superiores a los valores asociados a los 9 píxeles que forman la ventana. De esta forma se pretende superar el problema de las estimaciones a la baja de las alturas del arbolado. Mediante la interpolación se obtiene para cada píxel una superficie continua. Los coeficientes se ajustan empleando los nueve datos de la ventana. Con los valores de los coeficientes se pueden determinar los máximos y mínimos de la función ajustada. El criterio para etiquetar un píxel como árbol es el siguiente:

Obtenidos los seis coeficientes que definen la superficie, un píxel se etiqueta como árbol si en la superficie interpolada se detecta un máximo local el cual está incluido en la superficie correspondiente al píxel central. La altura que se considera representativa del árbol no es la del píxel central si no la de la superficie interpolada en el punto donde se ha detectado el máximo local. Este método permite, en determinados puntos, obtener estimaciones de alturas ligeramente superiores a las almacenadas la capa ráster sobre la que se esté aplicando el algoritmo.

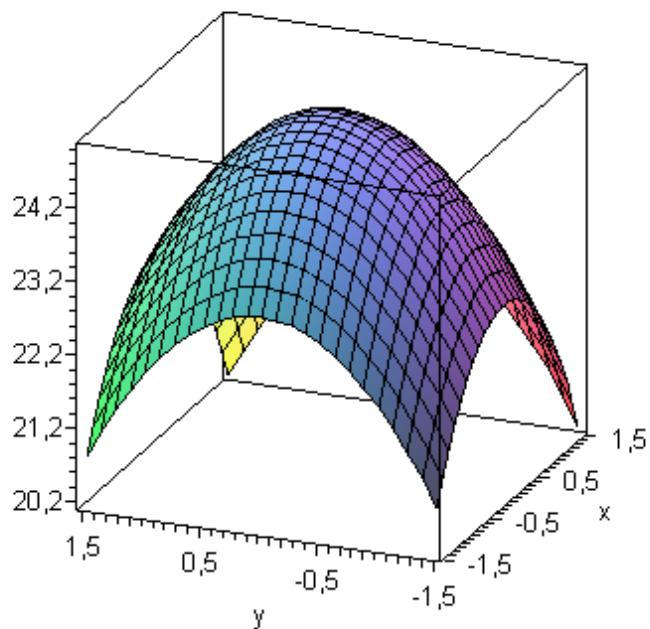


Figura 10. Malla interpolada con un máximo local.

A continuación se muestra el resultado de ejecutar este método en la parcela 1 de nuestras parcelas de entrenamiento:

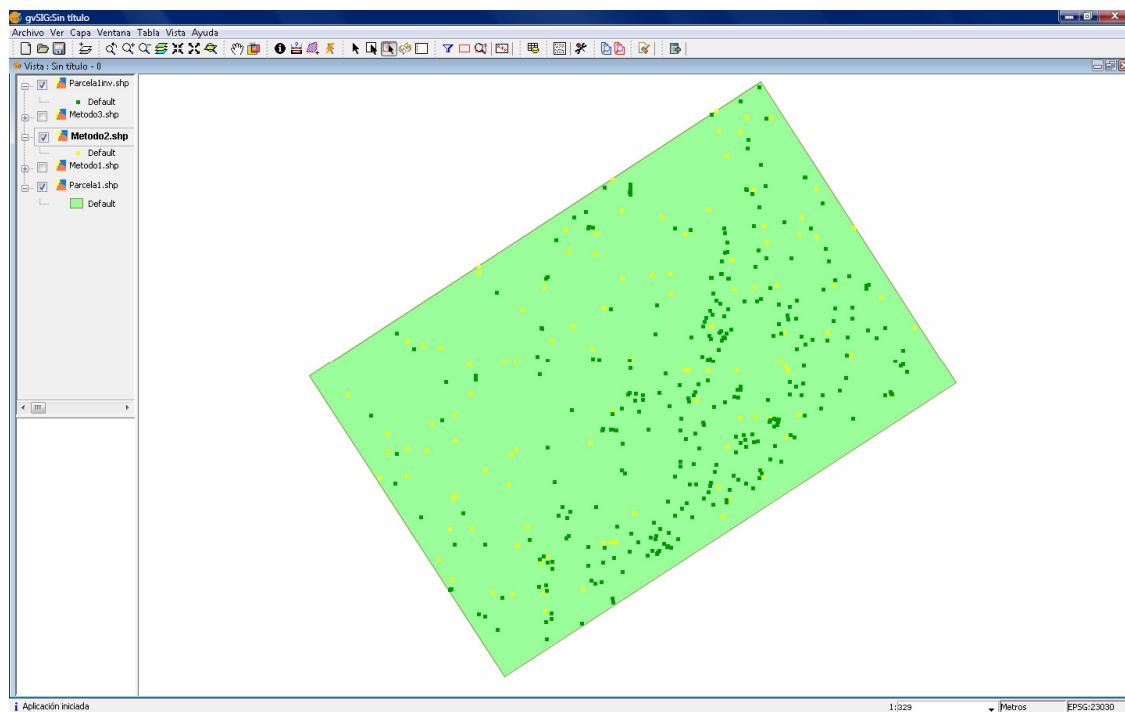




Figura 11. Método de máximos locales e interpolación de alturas, aplicado a la parcela 1.

Se observa que los resultados de este método mejoran bastante los resultados del método anterior, pero aún así, los resultados no son buenos, a grosso modo se puede ver que hay menor cantidad de árboles que los que hay en el inventario de campo. Con este método se obtienen 108 árboles frente a los 317 árboles del inventario. Además se observa que las localizaciones tampoco son muy acertadas.

2.2.1.3 Método 3. Método cuencas vertientes.

Este último método es uno de los ejemplos más notables de reutilización de algoritmos ya existentes. El método se basa en una idea sencilla pero muy ingeniosa, se gira el modelo digital de copas o DCHM, es decir se le da la vuelta y de esa forma obtenemos una superficie en la que las copas de los árboles forman ligeras depresiones en las que se depositaría y almacenaría el agua si se vertiese sobre esta superficie.

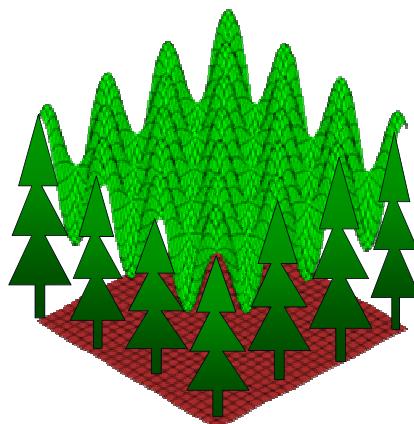


Figura 12. Muestra los huecos entre los árboles.

Los métodos de delineación automática de cuencas vertientes tradicionalmente empleados en hidrología requieren un procesado previo de los modelos digitales de elevaciones que determine los puntos de acumulación del flujo de agua. Es decir requieren un método que detecte y rellene las depresiones del terreno donde se acumularía el agua. Volviendo al ejemplo de la detección de árboles, si aplicásemos alguno de estos algoritmos para detectar y llenar las depresiones del terreno, el resultado sería que las depresiones formadas por las copas de los árboles serían encontradas. Cada una de estas depresiones correspondería a un árbol. Estos métodos están implementados en distintos programas para análisis de información geográfica



(GIS), especialmente en aquellos que se emplean para el planeamiento hidrológico. Lo único que hay que hacer para aplicar este método a datos LiDAR es obtener la imagen en negativo del DCHM. No hacen falta adaptaciones específicas para este problema.

Para el presente trabajo, este método se ha aplicado empleando ArcGIS 9.2. El algoritmo parte de un modelo digital de elevaciones, a partir del cual calcula un mapa del “Flujo del agua”, Se etiquetan como sinks o huecos, aquellas celdas en las que se produciría una acumulación del flujo del agua. Estas celdas son las que finalmente se consideran como árboles.

Los resultados obtenidos por este método aplicados a la parcela 1 son los siguientes:

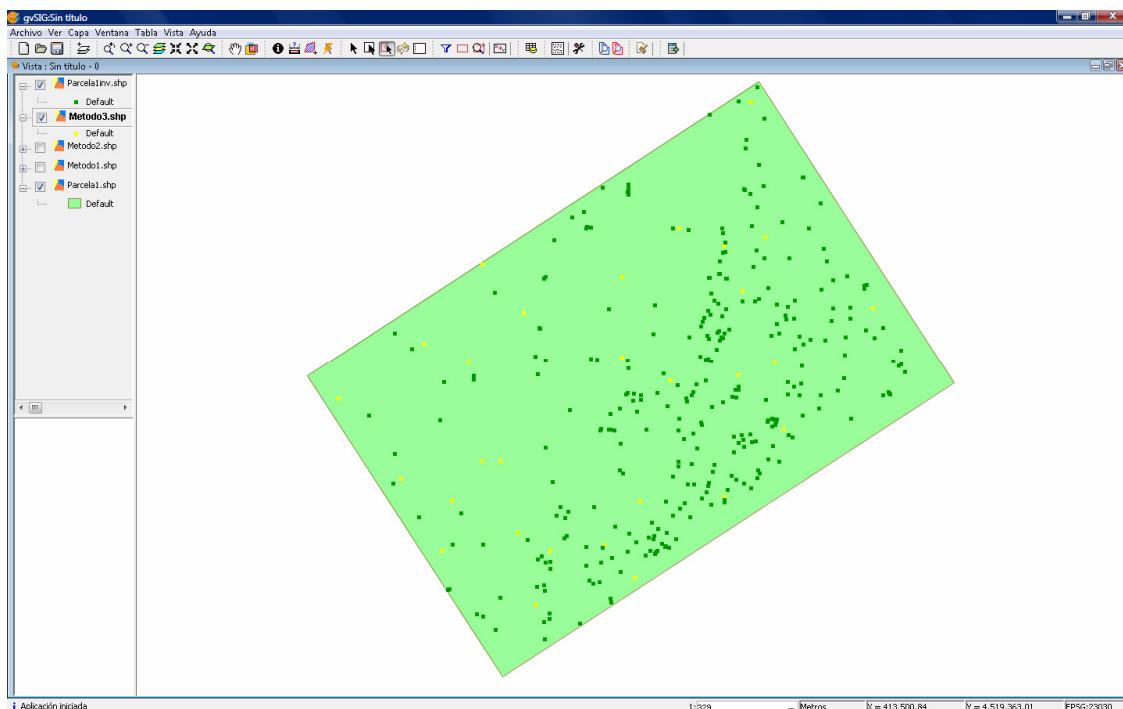


Figura 13. Método cuencas vertientes aplicado a la parcela 1.

Se observa que la cantidad de árboles encontrados es también excesivamente inferior, en este caso se obtienen 29 frente a los 317 del inventario de campo.

Se debe de elaborar un algoritmo más complejo que permita localizar todos los árboles y posicionarlos de una manera más exacta que las técnicas actuales.



2.2.2 Algoritmo de Wang

Wang [22] desarrolla un método para detectar árboles individuales usando los datos brutos de las imágenes LiDAR. Al leer su artículo publicado en la revista Sensor 2008 [23], nos pareció interesante y decidimos estudiarlo más a fondo. De este algoritmo hemos tomado ideas generales para poder desarrollar el nuestro. A continuación se explica en detalle el algoritmo de Wang.

Los pasos que sigue Wang para realizar su algoritmo son los siguientes:

1. Pre-procesado de la nube de puntos LiDAR.

El pre-procesado lo divide en dos partes, por un lado la subdivisión del área de estudio en grids más pequeños y por otro la normalización de los puntos brutos.

Estos dos puntos son comunes a nuestro algoritmo, pero lo realmente importante es destacar que en la división de la parcela en trozos nosotros tendremos en cuenta la ampliación de las zonas de borde, como se explica de forma detallada en la sección 4.1.1.

2. Proyecciones horizontales de la imagen, o lo que nosotros llamamos definición del espacio voxel. Este es uno de los puntos clave y por el que nos pareció interesante el algoritmo de Wang, ya que al dividir todos los puntos por capas y hacer un estudio de las capas por alturas nos permitirá detectar árboles de las capas más bajas lo cual ha sido un problema en la mayoría de los algoritmos de detección de árboles.

3. Modelado de copas de árboles individuales. Este punto lo divide en:

- 3.1. Delimitación de los contornos de copas de árboles individuales

Lo que Wang [22] hace es buscar las siguientes morfologías:



Figura 14. Elementos estructurales usados en el proceso morfológico jerárquico.

En nuestro caso las morfologías buscadas son bastante diferentes. Nosotros damos prioridad a encontrar voxel cuyo número de puntos sea mayor que cero y que sean



adyacentes unos a otros, pero no limitamos la forma, puesto que las formas de los árboles no son figuras geométricas perfectas, sino que es muy variada. Aunque las parcelas a estudiar contienen especies de pino silvestre que son coníferas, el crecimiento de estos es muy irregular. En la sección 4.3 de nuestro algoritmo se explica de forma detallada todo el proceso que hemos realizado para determinar los contornos de copa. Wang coge como semilla los que tienen un número de puntos mayor y después se va fijando en los vecinos, de forma que cuando haya menor número de puntos cerrará la morfología.

Wang asume que donde hay un mayor número de puntos brutos debería de ser donde se presenta la copa. De modo que considera los píxel con un nivel de gris mayor y guarda sus vecinos, considerando las morfologías presentadas en la figura anterior. Lo que hace es ir uniendo regiones de diferentes niveles de gris que son vecinas. El problema que encuentra es realizar esas uniones cuando se van juntando árboles, sobre todo en las capas más bajas, para ello realiza el siguiente paso 3.2.

3.2. Mejora de la delimitación de los contornos de copa.

Uno de los problemas con los que se encuentra Wang y también nos hemos encontrado nosotros es ¿Cuándo paro de crecer mientras busco un árbol? En su artículo comenta que se ha de parar cuando los árboles se juntan unos con otros, pero llegados a este punto no nos quedaba clara la forma de hacer esto. Y por otro lado esto también tiene inconvenientes, puede que si te vas expandiendo a lo ancho según vas buscando hacia abajo un árbol se incluyan puntos de árboles bajos (nuevos árboles) a un árbol mayor. De forma que nuestro algoritmo a partir de este punto difiere en el de Wang. Para entender mejor como hacemos este proceso ver el apartado correspondiente de nuestro algoritmo.

3.3. Pre-orden transversal.

Para realizar este paso Wang sigue una serie de pasos que describe en su artículo [22], lo que pretende fundamentalmente es ir agrupando los vecinos verticalmente.

Como ya hemos dicho en el punto 3.2 de este algoritmo, este punto también difiere del nuestro.



3.4. Modelado en 3D de las copas de los árboles.

Wang va más allá y como cuenta con los datos, una vez detectadas las copas de los árboles decide construir el modelo en 3D de la copa. Esto es algo que no se ha hecho en nuestro algoritmo pero se podría hacer con algún programa apropiado ya que se cuenta con los datos necesarios una vez que se ha detectado el árbol.

Los puntos débiles de este algoritmo son la detección de árboles jóvenes (árboles de altura menor) y detección de árboles con dos copas (árboles muy anchos). En nuestro algoritmo hemos tratado de mejorar estos dos puntos el primero con una estimación de alturas de copas. Y el segundo con la fusión de árboles cuando se cumplen determinadas condiciones. Esto se explica de forma detallada en el capítulo de nuestro algoritmo.

Antes de definir nuestro algoritmo contactamos con Wang comentándole que estábamos interesados en su algoritmo y le preguntamos a cerca de determinadas dudas que nos quedaban al leer su artículo [23] y también evidentemente sobre los resultados obtenidos. Él nos respondió amablemente proporcionándonos otro artículo más detallado [22] y con una evaluación de los resultados obtenidos. Le pedimos también la implementación de su método para poder mejorar los puntos débiles y hacer nuevas aportaciones, pero no accedió a ello puesto que las fuentes eran propiedad de la Universidad en la que ella estudiaba, de manera que optamos por coger sólo las ideas y desarrollar nuestro propio algoritmo. Wang hizo la implementación de su algoritmo en C, nosotros hemos optado por realizar la implementación en Java integrándolo en un GVSIG (Software libre para Sistemas de Información geográfica). En el de Wang había puntos que no se explican detalladamente y no sabíamos cómo abordarlos. Una prueba que hubiese sido muy interesante y pretendemos realizar en el futuro es poder ejecutar el algoritmo de Wang con las parcelas de nuestras pruebas y de esa forma ver resultados reales.

2.3 Problemática que se presenta

Estimar la masa forestal es uno de los grandes problemas con los que se encuentran los expertos en esta materia. No existen programas que lo hagan con total exactitud. Los árboles son muy variados y el crecimiento y morfología de los mismos no sigue unas reglas fijas que puedas predecir, por ello no existen programas ni algoritmos que realicen la estimación de forma exacta. Los principales problemas a la hora de realizar un algoritmo de este tipo son los árboles pequeños. Las imágenes LiDAR con las que estamos trabajando se toman desde el aire



de modo que los puntos correspondientes a los árboles bajos en muchas ocasiones no aparecen o se pueden confundir con el tronco o partes de otro árbol más grande. Otro de los grandes problemas es que muchos bosques y en la zona de estudio en concreto (Valsain) se pueden encontrar muchos árboles con dos copas y un solo tronco que es algo difícil de estimar.

Otro de los grandes problemas es que no siempre los árboles crecen hacia arriba. Podemos encontrar árboles que crezcan inclinados y una pequeña inclinación en árboles de más de 20 metros puede llegar a tener una extensión muy grande que puede hacernos confundir un único árbol con varios árboles.

Estos problemas unidos a que el tiempo de procesado de una imagen LiDAR es muy alto dada la gran cantidad de puntos, hacen que sea un problema de difícil resolución en el que queda mucho por hacer.

2.4 Métodos de clasificación de puntos.

El problema que se plantea es un problema de clasificación de puntos. Se necesita clasificar todos los puntos agrupándolos en n clases. Otra forma de nombrarlo es "Clustering".

Clustering o agrupación puede ser considerado uno de los más importantes problemas de aprendizaje no supervisado, por lo tanto, como cualquier otro problema de este tipo, se trata de encontrar una estructura en una colección de datos no etiquetadas. Una definición aproximada de clustering podría ser "el proceso de organización de objetos en grupos cuyos miembros son similares, de alguna manera" [8].

Un clúster es una colección de objetos que son "similares" entre ellos y son "distintos" a los objetos pertenecientes a otros grupos. El objetivo de cualquier proceso de clustering debe orientarse a conseguir que las muestras se distribuyan en clases, de tal forma que las pertenecientes a una misma clase sean tan similares entre sí como sea posible y a la vez diferenciarse de las muestras de otras clases, también tanto como sea posible. Por este motivo es necesario definir alguna medida de similitud, que en el caso que nos ocupa serán funciones de distancia, de forma que la distancia entre dos muestras x_k y x_l determine el grado de similitud entre ellas. En nuestro caso cada clase se corresponderá con un árbol.

Se puede mostrar gráficamente:

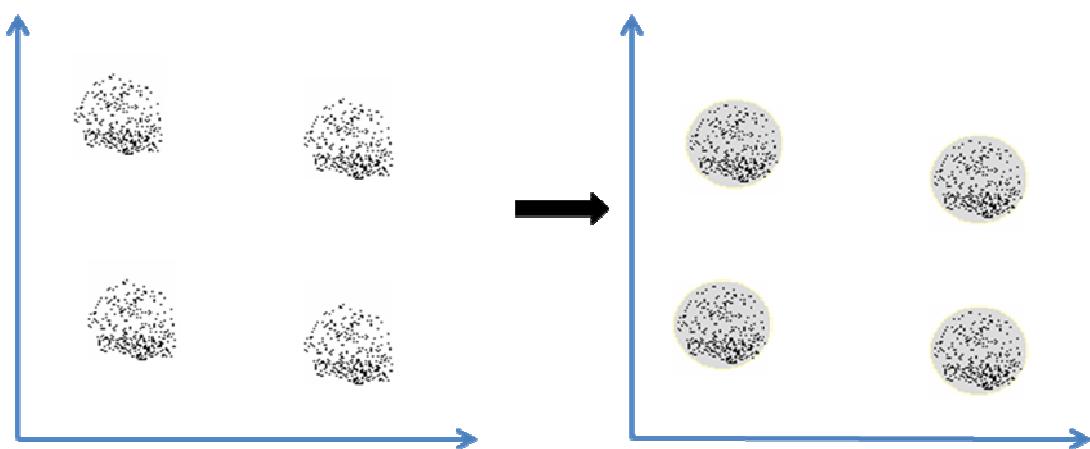


Figura 15. A) conjunto de puntos. B) conjunto de puntos agrupados en clusters.

En este caso, identificar fácilmente los 4 grupos en los que los datos se pueden dividir, el criterio de similitud es la distancia: dos o más objetos pertenecen al mismo grupo si están "cerca" de acuerdo a una determinada distancia (en este caso la distancia geométrica). A esto se le llama clustering basado en la distancia. Otro tipo de clustering es clustering conceptual: dos o más objetos pertenecen al mismo grupo si éste define un concepto común a todos esos objetos. En otras palabras, los objetos se agrupan de acuerdo a sus condiciones de conceptos descriptivos, y no por simples medidas de similitud [8].

Así, el objetivo del clustering es determinar las características intrínsecas en un conjunto de datos sin etiqueta, para poder agruparlos. En el caso de este trabajo lo que se pretende es agrupar puntos de tal forma que los puntos de un mismo grupo formen un árbol. De forma que cada árbol será un clúster diferente. Será un clustering basado en la distancia, ya que las agrupaciones se harán en función de las distancias de unos puntos con otros. Se iterará por todos los puntos de forma que todos los puntos (x,y,z) pertenezca a un supuesto árbol estimado por el algoritmo.

Los pasos a realizar en el clustering son:

- Seleccionar una medida de similaridad.
- Elegir técnica a usar (tipo de clustering).
- Elegir método o algoritmo dentro de la técnica.
- Decidir cuántos clusters hacer.



- Interpretar los clusters formados (deducir las propiedades que diferencian a los individuos dentro de cada clúster del resto de clusters).

Hay una serie de problemas con el clustering. Entre ellos: las actuales técnicas de agrupamiento no abordan adecuadamente todas las necesidades y simultáneamente hacer frente a gran número de dimensiones y gran número de datos puede ser un problema debido a la complejidad de tiempo; la eficacia del método depende de la definición de "distancia" (para clustering basado en distancia, como será nuestro caso); si no existe una obvia medida de distancia, esta debe de ser definida, lo cual no es siempre fácil, especialmente en espacios multidimensionales; el resultado de los algoritmos de clustering puede ser interpretado de diferentes maneras [8].

2.4.1 Tipos de clustering

Los algoritmos de clustering pueden ser clasificados como se enumeran a continuación [8]:

- Exclusive Clustering
- Overlapping Clustering
- Hierarchical Clustering
- Probabilistic Clustering

En el primer caso exclusive clustering o también llamado K-means clustering, los datos se agrupan de manera exclusiva, de modo que si un determinado dato pertenece a un grupo definido entonces no puede ser incluido en otro grupo. Un simple ejemplo se muestra en la figura siguiente, donde la separación de los puntos es alcanzado por una línea recta en un plano Vd.-dimensional.

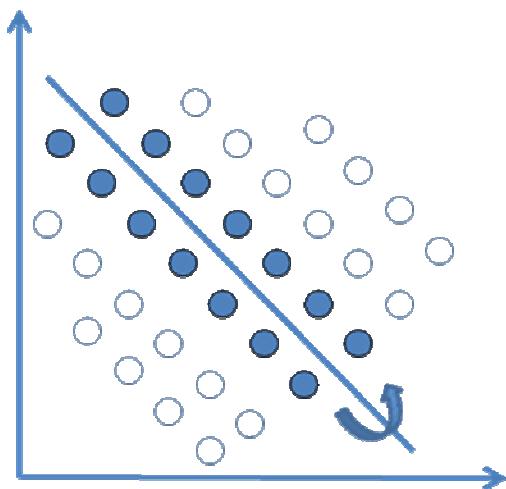


Figura 16. Gráfico de clustering exclusivo.

La idea principal es definir k centroides o puntos semilla, uno para cada grupo. Estos centroides deberían ser colocados de una astuta manera, ya que la diferente localización causa distintos resultados.

Autores como Morsdorf [14], Alexander [1], Glauton [6] y García [5] usaban k -clustering en sus artículos. Todos ellos elegían como punto semilla los máximos locales, aunque variaban la forma de realizar el clustering. Por ejemplo Morsdorf [14] tomaba como semilla para el k -clustering los máximos locales. La extracción de los máximos locales la hacía igual que Hyppä [10], aplicando un filtro suavizado en DSM, para encontrar píxeles que tengan los 8 vecinos más pequeños que el píxel central, el resultado de esto es una gran segmentación. Posteriormente va reasignando puntos a las clases más cercanas y recalculando los centroides.

Por el contrario, el segundo tipo, overlapping clustering o también llamado Fuzzy clustering, utiliza conjuntos borrosos para agrupar datos, de modo que cada punto puede pertenecer a dos o más grupos con diferentes grados de adhesión [8]. En nuestro caso usaremos Fuzzy clustering para conjuntos de puntos que puedan pertenecer a más de un árbol. Se definirá un grado de pertenencia para cada punto del conjunto a cada uno de los árboles. La asignación de cada punto a uno u otro árbol se hará en función del grado de pertenencia de los puntos a cada árbol. Elegiremos siempre para cada punto el árbol que mayor grado de pertenencia tenga. Este punto es fundamental en nuestro algoritmo, ya que es el que nos va a permitir asignar



puntos individuales a un árbol y no asignar un vox completo (que lleva un grupo de puntos que están próximos). En la sección 2.5 se explica más detalladamente este tipo de agrupamiento.

Un algoritmo hierarchical clustering se basa en la unión más cercana entre los dos grupos. El inicio se realiza siempre mediante el establecimiento de cada dato como un grupo. Después de unas iteraciones se alcanzan finalmente los clusterings deseados [8]. En el caso del algoritmo descrito en este trabajo se usará clustering jerárquico cuando se fusionen dos árboles en uno. Esto se hará cuando dos árboles tengan muchos puntos coincidentes. Dando lugar a árboles con dos copas o árboles más grandes. En el capítulo 4 se describe este proceso con mayor detalle.

Finalmente, el último tipo de clustering utiliza un completo enfoque probabilístico, también conocido como mezcla Gaussiana. Brandtberg [2] usaba este tipo de clustering en su artículo.

2.5 Lógica Fuzzy

Uno de los puntos fuertes de este algoritmo es la resolución de conflictos a la hora de asignar determinados puntos a uno u otro árbol. Para resolver estas situaciones se hace uso de lógica Fuzzy. En la sección 2.5.1 se explican las ideas principales de los conjuntos borrosos y lógicas borrosas y en la sección 2.5.2 se explica en qué consiste el clustering Fuzzy o borroso.

2.5.1 Conjuntos borrosos y lógicas borrosas

La teoría de los conjuntos borrosos forma parte de la “Inteligencia Artificial”. La Inteligencia Artificial intenta investigar cómo hacer que los ordenadores hagan lo que las mentes humanas hacen. Las palabras “inteligencia” y “artificial” despistan un poco y no parecen llevarse muy bien, lo que requiere que se deba explicar. El ser humano es enormemente antropocéntrico, se creyó el centro del universo, y ahora le molesta que pueda llamarse “inteligencia” a algo que no esté en un ser humano.



La lógica clásica, la teoría de conjuntos clásica o la teoría de probabilidad no siempre resultan adecuadas para tratar la imprecisión, la incertidumbre, la no especificidad, la vaguedad, la inconsistencia y la complejidad del mundo real. Esto motiva la evolución de los conjuntos borrosos y las lógicas borrosas y explica su papel en la reestructuración de los fundamentos de las teorías científicas y sus aplicaciones, por lo que se están produciendo grandes avances tanto en áreas teóricas como en gran variedad de aplicaciones.

La teoría de la probabilidad sólo es capaz de representar uno de los tipos de incertidumbre. *Lotfi A. Zadeh*, en 1965, escribió su artículo¹ en el que introduce una teoría sobre unos objetos, los conjuntos borrosos, que son conjuntos de frontera no precisa y cuya función de pertenencia indica un grado. El primer texto que apareció sobre subconjuntos borrosos data de 1965, cuando *L. Zadeh*, escribió el artículo “*Fuzzy Sets*” y posteriormente el libro “*Fuzzy Sets and Systems*” (1965) donde creó la base teórica sobre subconjuntos borrosos y de la que parten todas las investigaciones posteriores sobre el tema.

La teoría clásica de conjuntos de *Cantor* no recoge aquellos fenómenos reales cuyas características son "imprecisas" o "borrosas". En la esfera de los predicados subjetivos, y por tanto imprecisos, la teoría de conjuntos clásica se enfrenta con obstáculos difíciles de superar.

El redefinir el concepto clásico de conjunto nos lleva a construir desde la base otras nuevas matemáticas. Es preciso que redefinir las operaciones entre conjuntos, estudiar la estructura del conjunto de partes, las relaciones entre conjuntos, las estructuras algebraicas, los números, y muy especialmente, la lógica.

La teoría de subconjuntos borrosos tiene su origen en querer trabajar con la imprecisión, la incertidumbre y la vaguedad, por lo que pretende eliminar las limitaciones habituales que se encuentran en la teoría clásica de conjuntos, haciendo de ésta un caso particular de la anterior. La lógica queda modificada, ya no es una lógica bivalente en la que si algo es cierto su contrario es necesariamente falso, pues no verifica el principio de contradicción ni la ley del “*tercio excluso*”.

El desarrollo de la tecnología computacional ha abierto diversos campos de investigación. Se pretende que una máquina, o sistema de ellas, pueda producir razonamientos o acciones que si fuesen realizados por una persona serían



considerados inteligentes. El intento de automatizar el razonamiento y el aprendizaje requiere trabajar sobre las lógicas borrosas.

Las lógicas borrosas necesitan generalizar las conectivas entre conjuntos borrosos. Los conectivos lógicos AND, OR y NOT, y las operaciones entre conjuntos intersección, unión y negación se generalizan respectivamente mediante normas triangulares, conormas triangulares y negaciones.

Cuando se pretende utilizar las relaciones borrosas para efectuar inferencias de razonamiento aproximado de la forma

Si x es P entonces y es Q

x es P'

y es Q'

Se puede utilizar la regla composicional de inferencia de Zadeh y, en el caso de un universo en discurso, se obtendrán consecuencias lógicas

2.5.2 Clustering Borroso

El Algoritmo de clustering borroso o Fuzzy clustering (FCM) es un método que permite que un dato pertenezca a dos o más clusters. Este método (desarrollado por Dunn en 1973 y mejorado por Bezdek en 1981) se utiliza con frecuencia en el reconocimiento de patrones. Se basa en la reducción al mínimo de la siguiente función objetivo [8]:

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2 \quad , \quad 1 \leq m < \infty$$

Donde m es cualquier número real mayor que 1, u_{ij} es el grado de pertenencia a x_i en el grupo j, x_i es la posición i de la d-dimensional de los datos medidos, c_j es el centro de la d-dimensión del cluster, y || * || es cualquier norma que expresa la similitud entre los datos medidos y el centro.

El particionado Fuzzy se realiza a través de una optimización iterativa de la función objetivo mostrada anteriormente, con la actualización de los miembros del grupo u_{ij} y centros c_j de los clusters por:



$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad ; \quad c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m}$$

Esta iteración se detiene cuando, $\max_{ij} \left\{ |u_{ij}^{(k+1)} - u_{ij}^{(k)}| \right\} < \varepsilon$, donde δ es un criterio de terminación entre 0 y 1, mientras que k son los pasos de la iteración. Este procedimiento converge a un mínimo local.

El algoritmo se compone de los siguientes pasos [8]:

1. Inicializar $U = [u_{ij}]$ matriz, $U (0)$
2. En el k -paso: cálculo de los centros de vectores $C (k) = [c_j]$ con $U (k)$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m}$$

3. Actualización de $U (k)$, $U (k+1)$

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

4. Si $\|U (k+1) - U (k)\| < \delta$ entonces para; en caso contrario vuelve al paso 2.

Los datos están sujetos a cada grupo por medio de una función miembro, lo que representa el comportamiento de este algoritmo fuzzy. Para ello, simplemente tenemos que construir una matriz llamada U cuyos factores son los números entre 0 y 1, y representan el grado de adhesión entre los datos y los centros de los clusters [8].

Para una mejor comprensión, podemos considerar este ejemplo gráfico:

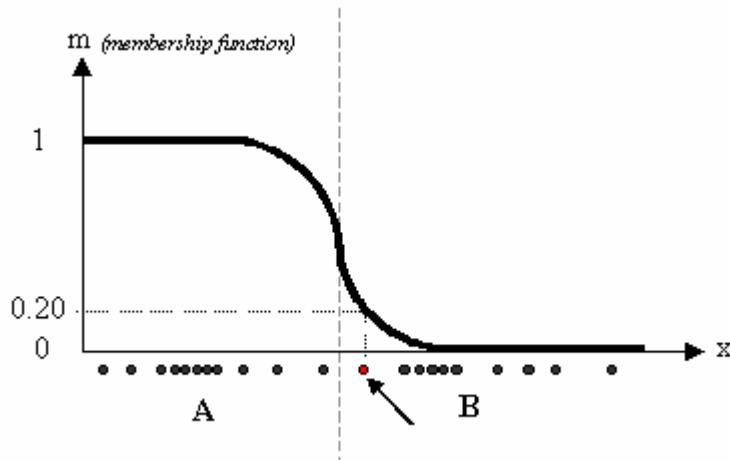


Figura 17. Representación gráfica de la función miembro de Fuzzy clustering.

Se agrupan los puntos en dos clúster «A» y «B», en base a la proximidad de las dos concentraciones de datos. El dato que se muestra como una mancha de color rojo marcado pertenece más al clúster B que al clúster A. El valor 0.2 de «m» indica el grado de pertenencia al clúster A. Ahora, en lugar de utilizar una representación gráfica, se introduce una matriz U cuyos factores son tomados de las funciones miembros:

$$U_{MC} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ \dots & \dots \\ 0 & 1 \end{bmatrix} \quad U_{MC} = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ \dots & \dots \\ 0.9 & 0.1 \end{bmatrix}$$

Figura 18. A) Matriz de K-means clustering. B) Matriz de Fuzzy clustering.

El número de filas y columnas depende del número de datos y clúster que consideremos. Más exactamente tenemos $C = 2$ columnas ($C = 2$ clúster) y N filas, donde C es el número total de grupos y N es el número total de datos.



En las matrices anteriores hemos examinado los casos: k-means (a) y FCM (b). Se puede notar que en el primer caso (a) los coeficientes son siempre unitarios. Esto indica el factor de pertenencia sólo a un clúster (1 pertenece a ese clúster y 0 no pertenece a ese clúster). Otras propiedades se muestran a continuación:

Las dos primeras propiedades pertenecen al caso de K-means Clustering y la última pertenece al caso Fuzzy Clustering:

- $u_{ij} \in [0,1] \quad \forall i, j$
- $\sum_{j=1}^c u_{ik} = 1 \quad \forall i$
- $0 < \sum_{i=1}^N u_{ij} < N \quad \forall N$



3 Descripción del problema

Este capítulo describe el problema abordado en este trabajo comenzando con una reseña de la problemática actual (apartado 3.1) y luego definiendo el objetivo del trabajo y el problema cuyo objetivo se plantea resolver encuadrándolo bajo ciertas limitaciones y alcances que tendrá este trabajo (apartado 3.2). Se explica por qué es importante resolver este problema (apartado 3.3) y se dan ejemplos, de casos concretos para dar al lector una mejor idea de cómo se ven estos (apartado 3.4).

3.1 Problemática actual

Actualmente los mecanismos que se usan para conocer la masa forestal son procesamiento de imágenes ráster de alta resolución, aunque estos métodos no son del todo fiables. La otra opción es trabajo de campo que es caro y supone mucho tiempo. De ahí, que gran parte de los bosques no estén inventariados. Los inconvenientes de realizar un trabajo de campo son muchos. Se ha de invertir mucho tiempo y supervisión humana. Además los bosques crecen y cambian, con lo cuál el inventario ha de estar en evolución constante. Si este trabajo es manual es muy difícil por no decir imposible que todos los bosques estén inventariados siguiendo este método. Por ello son necesarias técnicas que mejoren esta situación. Ya que el inventario de campo es muy importante para prevenir incendios, para las industrias de madera, para estudios sobre impacto medioambiental y para otros fines de interés forestal.

3.2 Objetivo, definición y límites del problema

El objeto de este trabajo es mejorar, mediante un marco teórico y práctico, la tasa de éxitos obtenida en el trabajo de Wang [22]. Se pretende complementar el proceso descrito en su artículo, añadiéndole variantes beneficiosas.

El problema tratado consiste en describir e implementar un algoritmo para estimar la masa forestal partiendo de imágenes LIDAR. Además el algoritmo calculará la posición de los árboles encontrados, y tamaños como la altura y los tamaños máximos en X e Y.



Se pretende minimizar el índice de error en la estimación de la masa y en la localización de la posición de los árboles respecto a los resultados obtenidos por los otros métodos comentados en este trabajo.

3.3 Importancia de su resolución

La solución de este problema es de suma importancia debido a que, gracias a la tecnología LiDAR y algoritmos de extracción de variables en bosques se podría tener un mejor control de todo el inventario de campo, siendo más rápido y eficiente el control de parámetros forestales tales como el número de árboles, la altura de estos, las dimensiones de la copa, la separación entre ellos. Estos valores son de suma importancia para controles forestales que tienen como objetivo evitar incendios, control de industrias madereras, etc.

El estudio de variable forestales mediante herramientas de detección asistida por computadora permitiría al experto una mayor velocidad de procesamiento para todas las imágenes láser, consiguiendo información que hoy en día se desconoce en muchos bosques y parcelas forestales, puesto que la forma actual de conseguir esa información es hacer un trabajo de campo, es decir, desplazándose a la zona y realizar mediciones, esto es un proceso caro y tedioso. No hay que olvidar que hay zonas donde la accesibilidad a pie es limitada, de modo que con algoritmos como este se evitarían horas de trabajo y se aumentaría la información conocida de nuestros bosques.

3.4 Casos reales

Se desea encontrar el mayor número de aciertos en cuanto número de árboles, dimensiones y posicionamiento de los mismos. A continuación se muestran imágenes LiDAR de ejemplo, en la que se pueden observar la distribución de puntos desde distintas vistas. Las imágenes son visualizadas con LASEdit.

La imagen que se muestra a continuación es la imagen LAS de una parcela, en el que se encuentra el modelo de elevaciones de los puntos, es decir los puntos se muestran en diferentes colores según la altura a la que se encuentren.

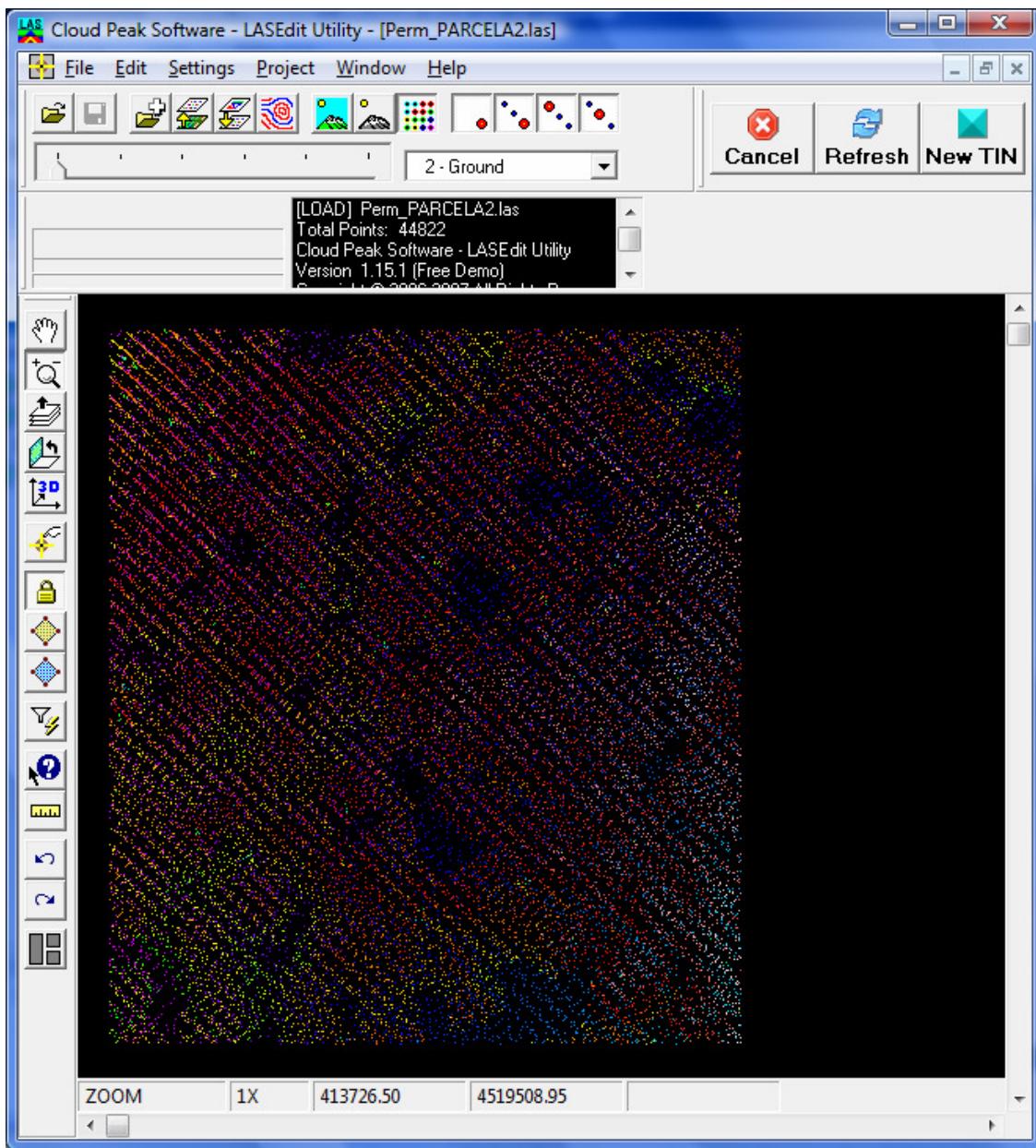


Figura 19. Modelo de elevaciones de parcela LAS en 2D.

La imagen que se muestra a continuación es también un modelo de elevaciones pero en 3D, donde se puede apreciar mejor la clasificación de puntos por alturas.

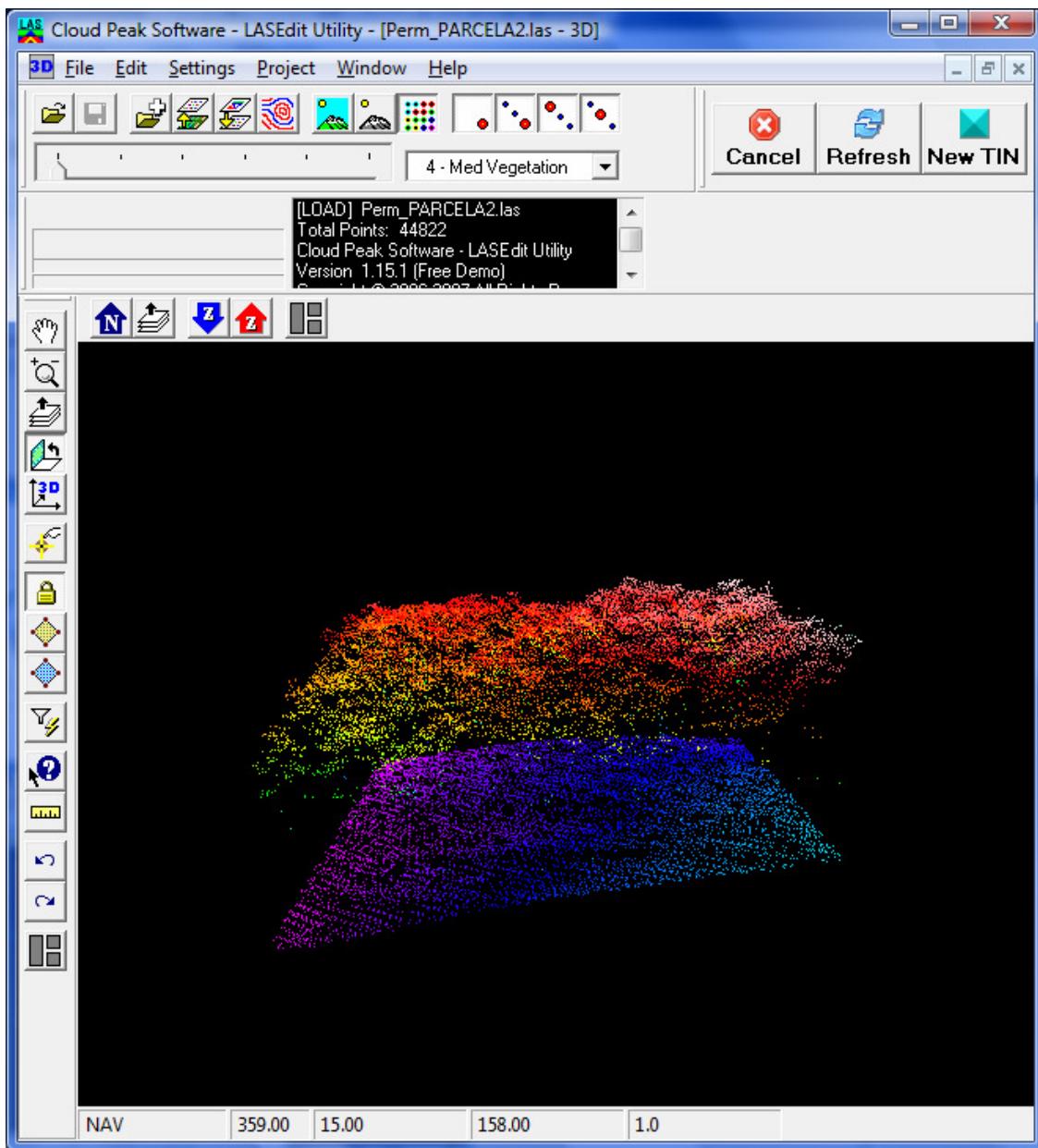


Figura 20. Modelo de elevaciones de parcela LAS en 3D.

A continuación se muestra la leyenda que nos indica los colores que se asignan a cada punto dependiendo de la altura del mismo.

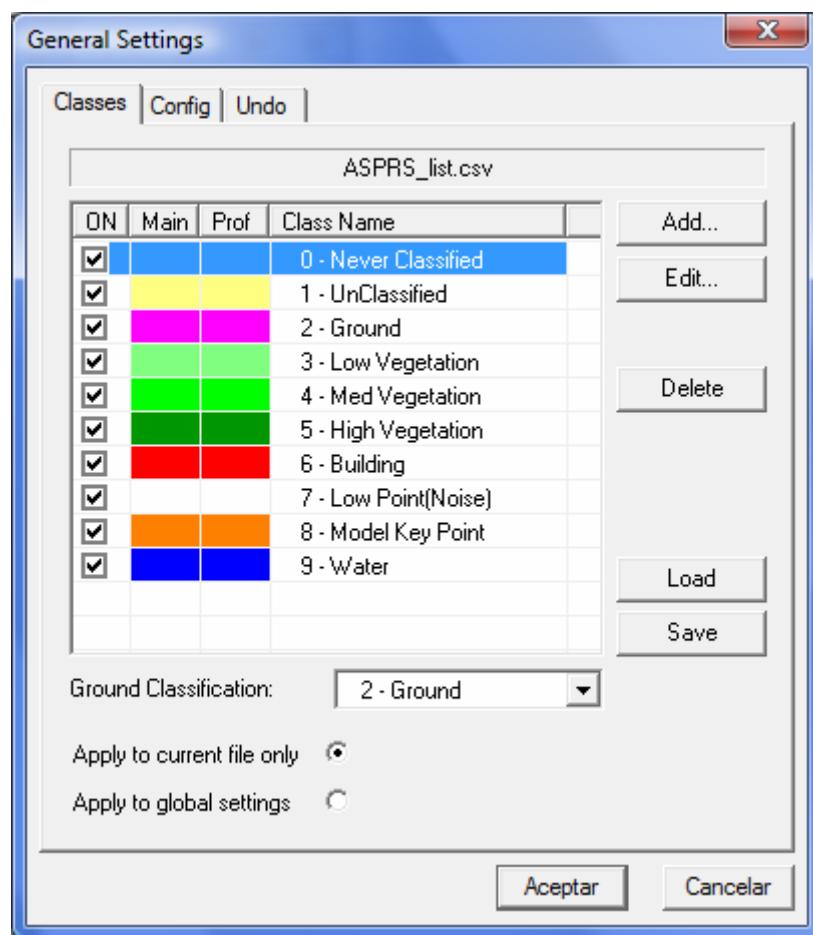


Figura 21. Clasificación de los puntos por colores

A continuación se muestra la misma imagen LAS pero se muestra la vista de clasificación en la que vamos a ver los puntos que son suelo, que se muestran en rosa y los puntos que son vegetación que se muestran en verde. La imagen se muestra en 2D en la parte izquierda y en 3D en la parte derecha.

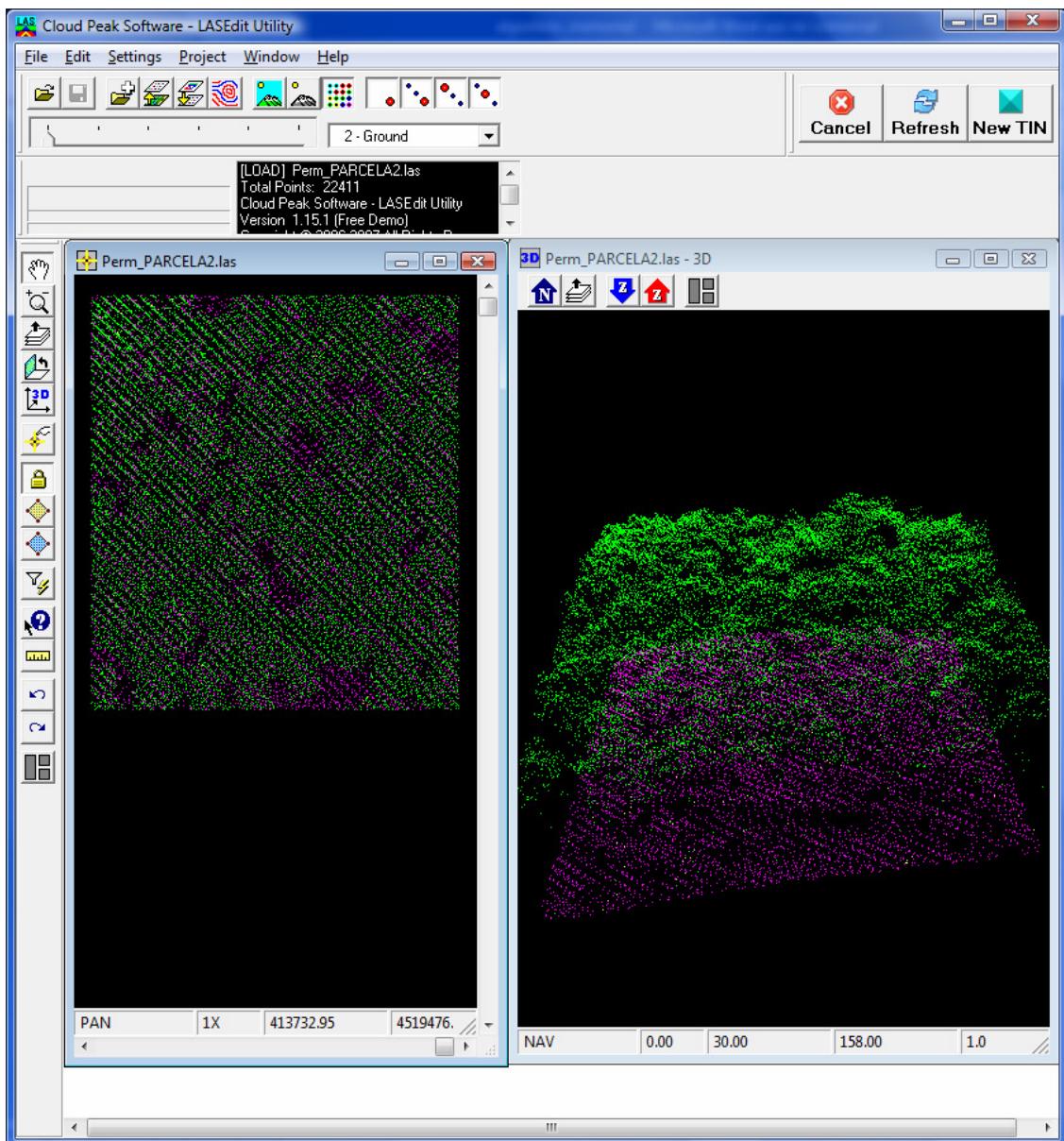


Figura 22. Clasificación de puntos de parcela LAS. Parte izquierda en 2D y parte derecha en 3D.

Si se hace un corte transversal en la parcela en 2D, podremos distinguir muy fácilmente los puntos que pertenecen al suelo de los que pertenecen a vegetación. Se observa que todos los rosas quedan en la parte inferior de la figura, ya que se corresponden con el suelo y los puntos verdes que son los que se corresponden con la vegetación se muestran más arriba. Aunque es importante destacar que estos se muestran a



diferentes alturas. De esa forma el lector se puede hacer una idea clara de cuál va a ser un caso real y ponerse en situación de lo que se pretende conseguir.

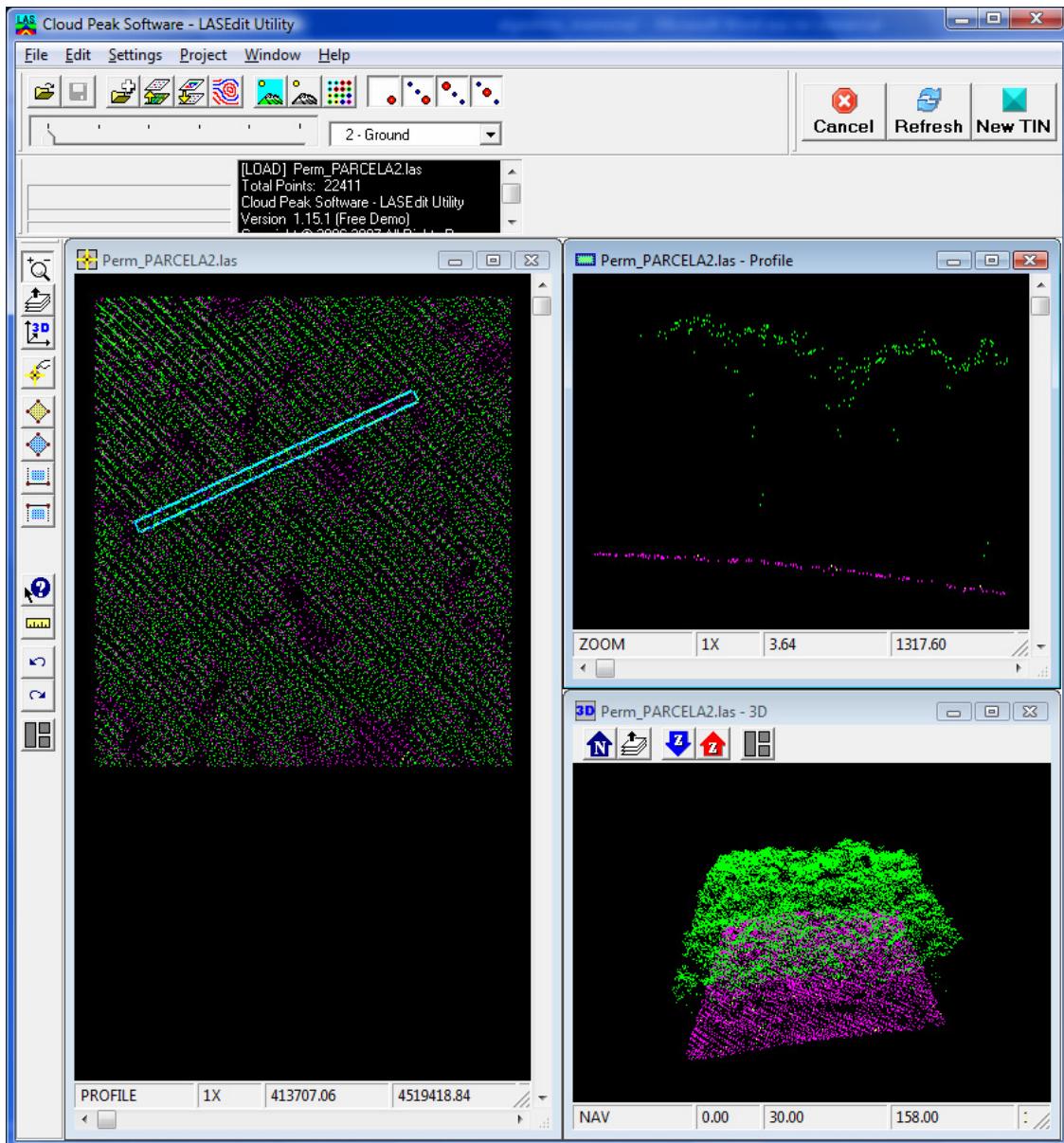


Figura 23. Perfil longitudinal de parcela LAS.



4 Algoritmo que se presenta

4.1 Datos de partida

El punto de partida de este algoritmo es uno de los productos Standard que se entregan tras un vuelo LiDAR y que consiste en una nube de puntos clasificados en un fichero o en un conjunto de ficheros con extensión .las. Generalmente la empresa o la entidad encargada de realizar el vuelo LiDAR suele realizar una serie de tareas preliminares sobre las nubes de puntos de tal forma que el cliente suele recibir un fichero .las ligeramente corregido y clasificado. Esta clasificación, suele realizarse empleando una serie de programas cuyas licencias suelen ser muy caras y no suelen ser rentables para los usuarios finales de los datos LiDAR. Generalmente las empresas que se dedican a realizar vuelos LiDAR tienen un poder adquisitivo suficiente como para adquirir estas licencias y amortizar la inversión ofreciendo los archivos .las clasificados como producto final a los clientes. Dado que este tipo de producto es el más estándar centraremos nuestros desarrollos en este tipo de archivo.

Las clasificaciones de puntos se orientan fundamentalmente a discriminar las partes fundamentales del medio físico. Esto es, clasifican los puntos en grandes categorías como suelo, vegetación, edificios etc. Estas categorías son realmente útiles a la hora de generar modelos de elevaciones, bien del suelo, bien del suelo más los objetos que se asientan sobre él, pero deben realizarse con los programas mencionados. Finalmente, es oportuno mencionar los desarrollos realizados por Robert J. McGaughey y su equipo en Pacific Northwest Research Station, Sattle Wassington, perteneciente al servicio forestal de estados unidos. McGaughey y sus colaboradores cuentan con una amplia experiencia en desarrollo de software para el tratamiento de datos LiDAR. Su principal aportación en este campo es FUSION [4]. FUSION es un programa de uso libre para la visualización y el análisis de datos LiDAR. Gran parte de los procesos que automatiza han sido seleccionados por su especial relevancia para estudios relacionados con el mundo forestal. En su versión de Marzo de 2009, FUSION incluye un algoritmo de filtrado de la nube de puntos LiDAR que permite separar los puntos del suelo del resto. Esta clasificación extremadamente sencilla en puntos de suelo y puntos que son cualquier otra cosa permitiría independizar completamente nuestro algoritmo de terceros programas de alto coste.

Estos algoritmos están en una fase de desarrollo bastante madura, y si bien se están realizando mejoras en este campo, actualmente, al menos en el mundo forestal los



esfuerzos se centran en la discriminación o extracción de información relativa a unidades estructurales del territorio de menor escala, por ejemplo árboles, parcelas forestales etc.

4.2 Algoritmo para la localización y estimación de masa forestal a partir de imágenes LiDAR

A continuación se enumeran las fases del algoritmo, que se explicarán detalladamente en la sección 4.3:

1. Pre-procesamiento de la nube de puntos:

- 1.1. Sub-división de datos (parcelas) en áreas de interés.
- 1.2. Normalización de la nube de puntos.

2. Creación del Espacio Voxel:

Entrada: Puntos (x,y,z)

Salida: Lista de Voxel

Cada Voxel tiene la siguiente información: el número de puntos que contiene, la lista de puntos (x,y,z) que contiene, las dimensiones máximas y mínimas en las coordenadas X, Y, Z, información de si ha sido añadido, y la lista de árboles a los que pertenece este vox.

3. Modelado de copas de árboles individuales:

3.1. Recorrido del Espacio Voxel desde la capa más alta a la capa más baja. Para cada voxel comprobar:

3.1.1. Si el número de puntos del voxel es mayor que 0 y el vox no pertenece a ningún árbol pasar al punto 3.1.2.

3.1.2. Crear Árbol nuevo con este Vox

3.1.3. Marcar vox como perteneciente a un árbol

3.1.4. Buscar vecinos a lo ancho:

3.1.4.1. Recorrido de los 8 vecinos para este voxel. Para cada uno de los 8 vecinos:



3.1.4.1.1. Comprobar si el número de puntos del voxel es mayor que 0 y el vox no pertenece a ningún árbol hacer:

3.1.4.1.1.1. Añadir Vox al árbol creado previamente y pasar al punto siguiente (3.1.4.1.1.2)

3.1.4.1.1.2. Marcar vox como perteneciente a un árbol.

3.1.5. Buscar vecinos a lo alto. (Z es la altura del árbol creado)

3.1.5.1. Recorrido Espacio Voxel desde la altura Z-1 hasta Altura_ máxima

3.1.5.2. Recorrido Espacio Voxel desde Ymin-1 hasta Ymax+1

3.1.5.3. Recorrido Espacio Voxel desde Xmin-1 hasta Xmax+1

3.1.5.4. Para cada Vox comprobar: Si número de puntos pertenecientes al Vox > 0 y no pertenece a ningún árbol:

3.1.5.4.1. Añadir Voxel al árbol creado previamente. Y pasar al punto siguiente (3.1.5.4.2)

3.1.5.4.2. Marcar Vox como perteneciente a un árbol

3.1.5.5. Sino, si el número de puntos > 0 pero ya pertenece a un árbol:

3.1.5.5.1. Si el número de puntos del vox es mayor que 5 entonces considero que es un tronco y fusiono dos árboles

3.1.5.5.2. Sino, actualizar los vox coincidentes con otro árbol.

3.1.5.5.3. Si el numero de vox coincidentes es mayor que min(altura Arbol1, altura Arbol2) / 2 entonces fusiono. Sino paso al punto 3.1.5.5.4

3.1.5.5.4. Aplico lógica Fuzzy y asigno por separado los puntos de este vox en función del grado de pertenencia de estos a cada árbol.

3.1.5.6. Sino, es el caso en el que el número de puntos es 0 entonces añado este vox a la lista de vox sin puntos del árbol.



4. Crear una capa con los árboles encontrados, eliminando los que tengan menos de tres puntos (x,y,z) o sólo tengan puntos en un vox y midan más de 10 metros.
5. Resolver el problema de la segmentación.

Una primera versión de este algoritmo se describió en el artículo presentado en “Jornadas Internacionales de Didáctica de las Matemáticas en Ingeniería” organizado por el Grupo de Investigación de la Universidad Politécnica de Madrid de Matemática aplicada a la ingeniería civil (MAIC) [17].

4.3 Descripción de las fases del algoritmo.

A continuación se explica cada uno de las fases del algoritmo propuesto en este trabajo.

4.3.1 Pre-procesamiento de la nube de puntos LiDAR

El gran tamaño de los archivos con extensión .las asociados a superficies de terreno de mediano tamaño unido a otras restricciones específicas de este algoritmo hacen necesario realizar una serie de tareas preliminares para preparar los datos de una forma adecuada. Estas tareas, son fundamentalmente dos. Por un lado la subdivisión de los datos en áreas de interés. La otra tarea es la normalización de la nube de puntos. Estos puntos son comunes al algoritmo de Wang [22], en su artículo los enumera, aunque no describe en detalle la forma de hacerlo. Nosotros describiremos de forma más detallada estos puntos. Como se verá en el epígrafe correspondiente, en este paso se hará uso de la clasificación previa realizada contenida en el fichero .las.

4.3.1.1 Subdivisión de datos en áreas de interés

Debido al gran coste computacional de memoria y almacenamiento, no es posible analizar todos los puntos de un área grande en un sólo paso. De modo que el área a estudiar se va a dividir en varios grids (celdas) pequeños como se muestra en la siguiente figura:

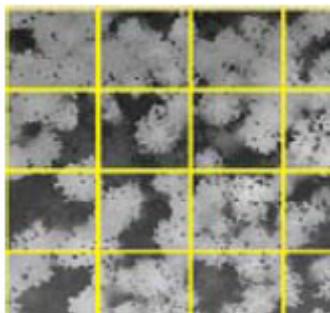


Figura 24. Área de estudio dividida en grids.

Actualmente, en la implementación que se ha hecho del algoritmo no se está usando la subdivisión de datos en grids. Es un punto que habría que desarrollar para cubrir parcelas de gran tamaño, de forma que la aplicación del algoritmo se realizaría en cada grid por separado. Realizar esa subdivisión no es algo prioritario ya que con herramientas GIS convencionales se pueden dividir las parcelas en trozos más pequeños y de ese modo realizar la ejecución en varias parcelas más pequeñas.

Las parcelas con las que contamos para realizar las pruebas tienen unas dimensiones de 90m * 95m y de 95m * 75m, según aconseja Wang [22] en su artículo los tamaños correctos de los grids pueden ir desde 10m * 10m hasta 200m * 200m. En nuestro caso todas las parcelas están dentro de los límites que recomienda Wang, por lo que hemos obviado el paso de subdivisión. Lo que es muy importante es que siempre que se realice esta subdivisión, se ha de ampliar x metros a cada lado el grid o parcela, para evitar que los puntos de cualquier árbol queden dispersos en varios grids. Los metros que se amplíen a cada lado han de seguir un criterio que es el siguiente. Estos metros tienen que ser como mínimo el diámetro de copa del árbol más grande, para que de ese modo nos aseguremos que los puntos de todos los árboles estarán como mínimo en un único grid. Para poder fijar correctamente los metros que se han de ampliar nuestras parcelas a cada lado, se ha hecho un estudio de los diámetros de copa de una muestra de 1075 árboles, en la que se observa que el árbol más grande tiene un diámetro de copa de 18 metros, con lo cual en nuestras parcelas se han ampliado aproximadamente este valor a cada uno de los lados.

A continuación se incluye un histograma en el que se muestra la distribución de los radios de copa en la muestra de 1075 árboles. Como se puede observar en el mismo la mayoría de los árboles tienen un radio de copa de 1 metro, pero hay una minoría un radio de copa de 9 metros o lo que es lo mismo 18 metro de diámetro, que se debe de considerar para que en el caso de que esos árboles estén próximos al borde no queden sin contabilizar como podría pasar si esto no se tuviese en cuenta.

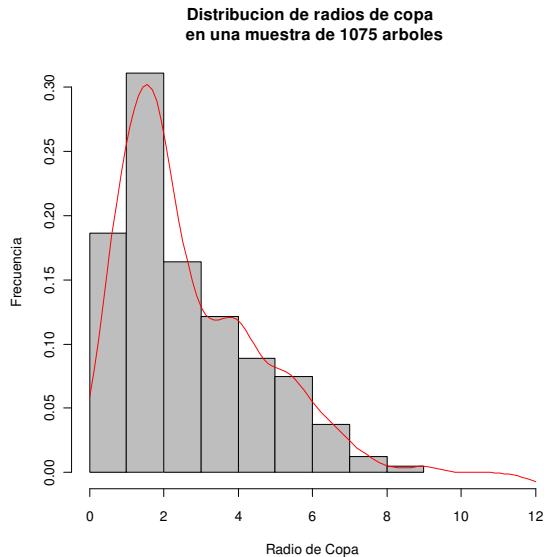


Figura 25. Distribución de radios de copa.

Hay que tener en mente, que al hacer esta división si quedan puntos de un árbol en un borde, los podremos ignorar, ya que esos puntos se contabilizarán en el grid vecino.

4.3.1.2 Normalización de puntos

El proceso de normalización es un procedimiento casi estándar ampliamente aceptado. Autores como Wang [22] y Morsdorf [14], también hacen uso de él en sus artículos. En sí, no aporta ninguna novedad técnica si bien procederemos a explicarlo con cierto detalle por ser uno de los pasos previos a realizar para aplicar el algoritmo.

Las nubes de puntos de los ficheros .las contienen la altitud de los puntos bien sobre el nivel del mar, bien sobre el elipsoide del sistema de referencia empleado. Para la aplicación de este algoritmo, no interesa la altura de los puntos sobre cualquiera de estas superficies. La información que realmente será relevante a la hora de decidir qué es o no un árbol será la altura de los puntos sobre el suelo, por ello la influencia del terreno debe de ser eliminada.



Para realizar la normalización hay que restar la altura del suelo a la altura de cada punto. Pero en los archivos .las no se tiene la altura del suelo para cada punto. En la información de origen se incluye la clasificación de todos los puntos brutos indicando si son suelo o vegetación. De modo que la altura del suelo, sólo se tiene para los puntos clasificados como tal. Por lo menos para aquellos puntos clasificados como vegetación habrá que calcular cual es la altura del suelo. Los puntos clasificados como vegetación, no tienen porqué estar en la misma vertical que ningún punto clasificado como suelo luego en principio se desconoce que altura hay que restar. Para solucionar este problema se genera un modelo digital del terreno (DTM) que permite estimar la altura del suelo en los huecos que quedan entre puntos clasificados como tal.

Tomando solo los puntos clasificados como suelo y aplicando un determinado método de interpolación seremos capaces de generar el modelo digital de terreno (DTM) y con él, podremos estimar la altura del suelo correspondiente a todos los puntos del fichero .las Una vez obtenido el DTM, solo hay que restar la altura del terreno a la altura de cada punto. El resultado es una nube de puntos normalizada, o lo que es lo mismo una nube de puntos en la que la influencia del terreno se ha eliminado.

El método de interpolación no es un factor crítico en la mayoría de los casos. Salvo que la intensidad del escaneo (Número de puntos por cuadrado) y la densidad de la masa forestal impidan tener un número suficiente de puntos por metro cuadrado clasificados como suelo la relevancia del método de interpolación es mínima. Los métodos de interpolación más extendidos son, Triangulaciones de Delaunay, Métodos de distancia inversa IDW, Métodos geoestadísticos (Krigging). La información de partida de este trabajo usa como método de interpolación la distancia inversa.

En las figuras siguientes se muestra el resultado de aplicar normalización a la nube de puntos:

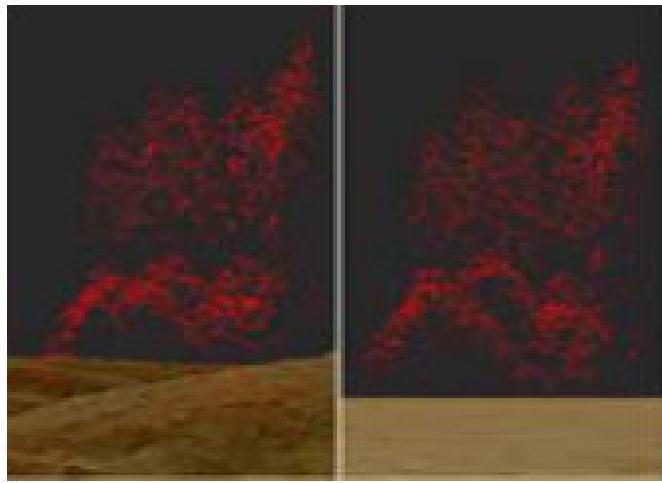


Figura 26. A) Imagen LiDAR con influencia del terreno. B) Imagen LiDAR normalizada.

4.3.2 Creación Espacio Voxel

Para llevar a cabo este paso nos centraremos exclusivamente en una celda si se ha subdividido el área de estudio en grids o en nuestro caso en una parcela. Se define un espacio local de voxel.

Vóxel, voxel o vox [19], a partir de ahora se usará indistintamente cualquiera de los tres sinónimos (la palabra proviene de la contracción del término en inglés "volumetric pixel") es la unidad cúbica que compone un objeto tridimensional. Constituye la unidad mínima procesable de una matriz tridimensional y es, por tanto, el equivalente del píxel (o píxel) en un objeto 2D. Al igual que los píxeles, los voxels no contienen su posición (x , y , z) en el espacio 3D, sino que esta se deduce por la posición del voxel dentro del archivo de datos por la fila columna y capa o layer que ocupan.

A continuación se muestra un gráfico de cómo sería el espacio Voxel [22].

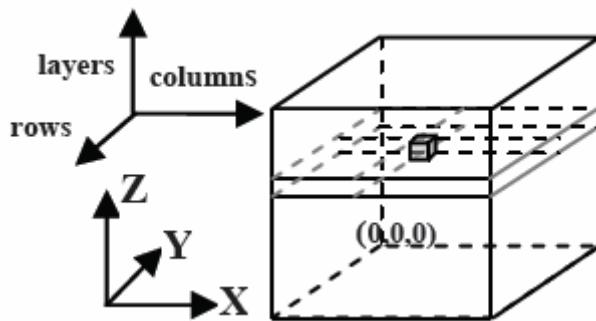


Figura 27. Espacio Voxel.

Una vez definido el espacio voxel, se hará un barrido de todos los puntos normalizados de cada celda (x,y,z), los puntos obtenidos del paso anterior, para identificar a qué voxel pertenece cada uno de ellos, de manera que a cada voxel se le va a añadir información del número de puntos (x,y,z) que contiene y cuáles son esos puntos. Cada voxel puede contener de 0 a n puntos (x,y,z), dependiendo de la densidad de la nube de puntos brutos y del tamaño de cada voxel.

La forma de realizar este paso es filtrar los puntos obtenidos del paso anterior por altura. Este filtro se hará en intervalos de 1 metro, el filtrado por x e y también será de 1 metro. Al hacer el filtro de los puntos por alturas, obtenemos los puntos que se encuentran a una altura determinada. Se iterará por todas las alturas para realizar este paso por los puntos de cada altura de forma individual.

El resultado de este paso será una serie de imágenes en 2D (ó capas ráster), donde la tercera dimensión vendrá marcada por la posición del voxel dentro del espacio, para cada voxel se guardará el número de puntos (x,y,z) que contiene o lo que llamaremos nivel de gris. El voxel también guardará todos los puntos (x,y,z) que contiene y las dimensiones máximas y mínimas de x , y , z . Es importante destacar, que a pesar de que estamos “rasterizando” la imagen no perdemos la información de los puntos brutos del LiDAR, (x,y,z), ya que para cada voxel guardo siempre toda la información de los puntos brutos (x,y,z) que contiene, de forma que puedo trabajar tanto con datos ráster, como con información LiDAR en cualquier momento, de manera que la rasterización que se hace no implica pérdida de información.

4.3.3 Modelado de copas de árboles individuales

Los puntos brutos pertenecientes a cada una de las copas de cada árbol individual pertenecerán a un grupo en función de los puntos pertenecientes a cada voxel del espacio voxel definido en el punto anterior. La forma de los clústers (árboles) está muy



relacionada con la resolución de cada píxel y el espesor de cada capa, en nuestro caso hemos optado por una resolución de 1 metro x 1 metro y un espesor de 1 metro también.

Los clusters de las proyecciones horizontales de cada capa representan la distribución de las copas de los árboles en el correspondiente nivel de altura. De esa forma cada árbol individual debería de ser visible en la misma localización de varias capas verticales vecinas. El concepto básico de extracción de un árbol es realizar un seguimiento de los clusters de arriba abajo a través de la proyección de imágenes en diferentes niveles de altura.

En este punto se realiza un K-means clustering en el que se toma como punto semilla los puntos más altos. A partir de ellos busco puntos en la misma capa que estén cerca y hacia abajo (en otras capas). Se hace esto iterativamente hasta que todos los puntos pertenecen a algún clúster.

Los pasos para obtener los árboles individuales empezando por la capa de más arriba y acabando por la capa de más abajo son los siguientes:

Desde la capa más alta a la capa más baja hacer:

1. Delimitar el contorno de la copa: Para ello se hará un recorrido por todos los vox de cada capa, en el momento que encontramos un vox con un número de puntos mayor que 0 y que no pertenezca a ningún árbol se pasa al punto 2.
2. Crear un árbol nuevo con ese vox.
3. Marcar ese Vox como perteneciente a un árbol. Pasar al paso 4.
4. Buscar en anchura. Se busca en los ocho vox vecinos de este vox en esta misma capa. El requisito de búsqueda será que el número de puntos es mayor que cero y este vox no pertenece a ningún árbol, es ese caso pasará a formar parte de este árbol. De esa forma las morfologías que indican el comienzo de un árbol serán un cuadrado y todas las combinaciones posibles entre los 8 vox vecinos.

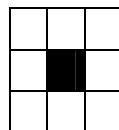


Figura 28. Espacio de búsqueda localizado un Vox.

A continuación se muestra un ejemplo se posibles morfologías. Éstas dependerán del tamaño del voxel y evidentemente del tipo de árboles. En nuestro caso el tipo de árbol es pino silvestre que es una conífera, aunque su copa puede llegar a ser bastante irregular. Pero generalmente tendrá una forma de cono, con lo que cuando se realice la búsqueda hacia la capa de abajo



se ampliará el rango de búsqueda en un voxel a cada lado de los encontrados en la primera capa.

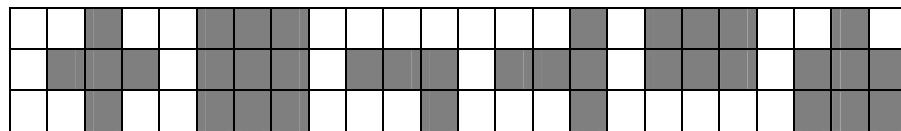
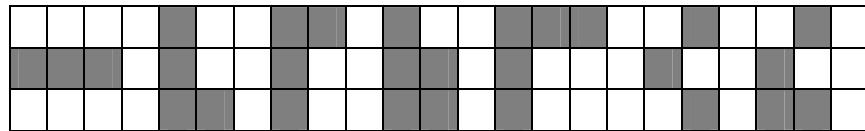
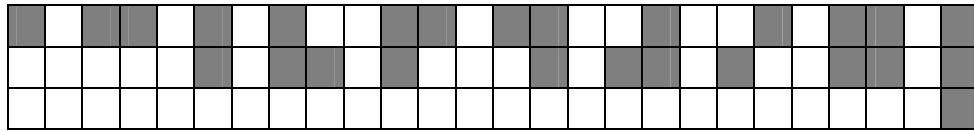


Figura 29. Posibles morfologías de la parte más alta de las copas de los árboles.

Cuando se encuentre alguna de las posibles formas guardo el posible árbol con las coordenadas de cada píxel que lo forma en una lista de posibles árboles. Para cada uno de los posibles árboles se guardan siempre los valores Xmin, Xmax, Ymin, Ymax, altura del árbol y altura de copa y un identificador.

5. Buscar en altura: Buscar desde la capa en la que localizo su voxel más alto de cada árbol hacia abajo. Para realizar esta búsqueda hago un recorrido hacia abajo desde el valor Xmin-1 hasta el Xmax+1 y desde Ymin-1 hasta Ymax+1. La búsqueda hacia abajo se hace ampliando el rango en 1 a la izquierda, derecha, arriba y abajo, ya que la forma de los árboles es cónica aunque con irregularidades. A continuación se muestran unos ejemplos gráficos de cómo sería la búsqueda hacia abajo de un árbol, en gris se muestran los voxels que se encontraron en la capa superior, en azul los voxels que se van a incluir en la búsqueda de puntos en esta capa y en las inferiores junto con los grises y en blanco se muestran aquellos voxels que no se incluyen en la búsqueda:

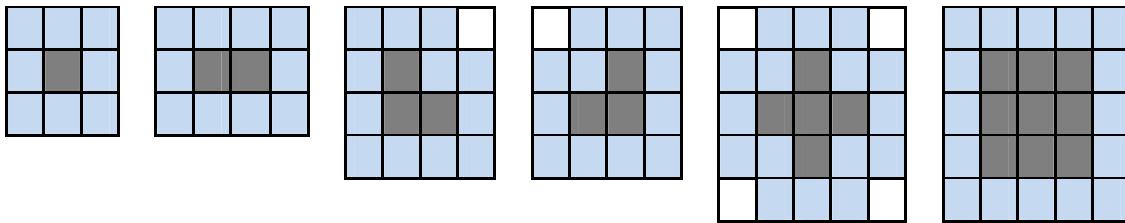


Figura 30. Zona de búsqueda de vox vecinos en altura.

Los vox en blanco más los vox en gris de la figura anterior serían los vox en los que se buscarían en las capas más bajas. Los vox en gris nos indican que son vox en los que he encontrado puntos en la capa inmediatamente superior. Mientras que los vox en blanco son vox vecinos en los que podría encontrar puntos pertenecientes a este mismo árbol en las capas más bajas.

La búsqueda hacia abajo es acotada, esto es, no se va a buscar desde que se encuentra el punto más alto de la copa hasta el suelo, sino que se establece un criterio de parada para el crecimiento del árbol hacia abajo, basado en las mediciones realizadas en los 1076 árboles de las parcelas permanentes para los cuales se tienen mediciones del Tamaño de la Copas. A continuación se incluye un gráfico con la relación altura del árbol – Tamaño de la copa.

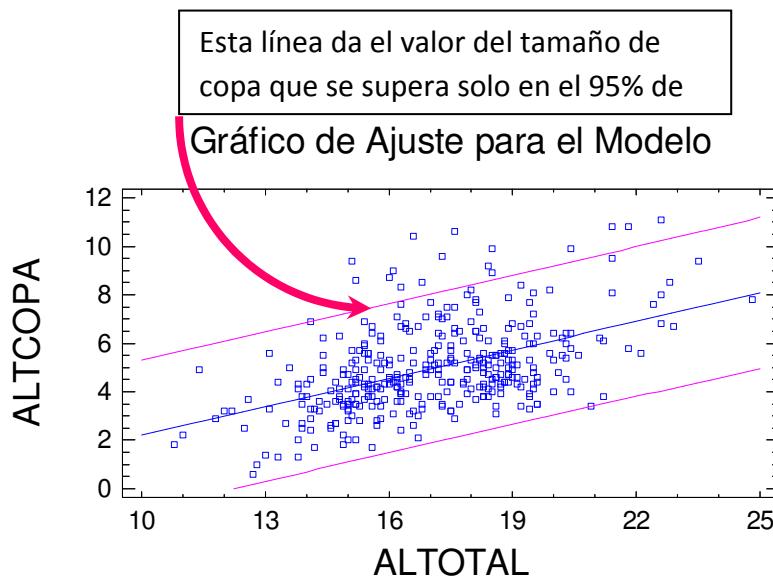


Figura 31. Relación gráfica entre la altura total de un árbol y la altura total del mismo.

Si unimos los datos de los clúster con los datos de campo, se puede obtener una regresión de la altura de los árboles LiDAR y la altura de los árboles del inventario de



campo. La altura de los árboles es derivada como la máxima altura de los puntos LiDAR pertenecientes a un clúster específico.

Si se cumplen todas las hipótesis necesarias para los “modelo lineales generales” se puede asociar una función de distribución para la variable a predecir en cada valor de la variable predictora. De modo que se puede construir intervalos de confianza para una predicción hecha en base a un valor. Si nos fijamos en la figura 31, la línea azul será la predicción del tamaño de la copa para un árbol de una determinada altura. Las líneas rosas son los límites de confianza del 95% para la predicción, (estos límites tienen una expresión analítica sencilla (llamémoslos D(H)). Los modelos de regresión generalmente incluyen otras dos líneas que son los intervalos de predicción para un promedio de observaciones. De modo que se tienen tres cosas. La línea azul que es el valor promedio que se estima para cada valor de alttotal. El intervalo que queda entre las líneas rosas y la línea azul, nos indican los valores entre los que fluctúa la predicción que se hace. Es decir que esto indicaría que los árboles de X metros de altura tendrá una copa entre m y n metros. Las líneas rosas son límites de confianza para un árbol de tamaño x, no para la media de los árboles de tamaño x. Se usa una regresión normal:

$$\text{Altura de copa} = D(H) = 1.24687 + 0.268302 * \text{ALTOTAL} + (1.96 * 2.355 * \text{ALTOTAL})$$

La idea fundamental es que cuando se encuentre el punto más alto de la copa de un árbol a una cierta altura H se la haga crecer hacia abajo una distancia que como mucho será igual al límite del 95% de la predicción D(H). En este caso D(H) sería la línea rosa de arriba. Con esto se puede perder algún árbol pero se pueden detectar, otros que estén en las capas más bajas, y esta es otra importante ventaja de este algoritmo frente a otros. La función D(H) que se incluye es la que ese muestra más arriba.

Con esta función se obtiene un valor de la altura de copa tal que el 95% de los árboles con una determinada altura tienen una copa menor.

- 5.1. Si en esos rangos de búsqueda se encuentran voxel con un número de puntos mayor que cero y que no pertenece a ningún árbol se añaden a la lista de voxel del posible árbol que añadí previamente y se marca el voxel como escrutado, para saber que ya pertenece a un árbol.
- 5.2. Si el número de puntos del voxel es mayor que cero pero ya pertenece a otro árbol, nos encontramos ante un conflicto. Esto puede suceder con más frecuencia en las capas más bajas. La situación de conflicto es que existe un voxel que pertenece a más de un árbol. Para solucionar esta situación se aplica el criterio que se explica más adelante en la sección 4.2.4.
- 5.3. En el caso de que el número de puntos del voxel sea cero lo añado a la lista de voxel sin puntos de cada árbol, que se usará en un futuro para ver los solapamientos entre árboles y poder agrupar dos árboles en uno, ya que es habitual encontrar dos árboles con dos copas y un tronco. De esta forma



seremos capaces de detectar este tipo de árboles, que era una de los puntos débiles del algoritmo de Wang [22].

6. Si no se ha superado la altura de copa, volver al punto 5.1, restando 1 a la altura de copa. Si por el contrario ya se ha alcanzado, volver al punto 1, con una altura de $Z = Z-1$ hasta que lleguemos a la capa más baja.

La tecnología que se emplea en este apartado es un k-means clustering en el que se toma como semilla los puntos más altos que cumplen unas determinadas condiciones. Lo que se hace es iterar hasta que se llega al suelo y ya no queda ningún vox con puntos que no pertenezca a ningún grupo. En este caso cada grupo sería un árbol.

4.3.4 Situación de conflicto:

A continuación vamos a explicar detalladamente cómo se resolvería la situación de conflicto que se expuso anteriormente.

Se podría encontrar conflicto en cualquier capa, pero con mayor probabilidad en las capas más bajas. La situación de conflicto es la siguiente: Un vox con un número de puntos mayor que cero, que ya pertenezca a un árbol e intentemos asignarlo a otro árbol. Ante esa situación se van a seguir los siguientes pasos:

1. Comprobar si el número de puntos del vox conflictivo o nivel de gris, es mayor o igual que cinco (consideramos que indica una alta densidad de puntos por metro cuadrado) en cuyo caso se deduce que se trata de un vox que guarda los puntos de un tronco ya que hay gran densidad de puntos por metro cuadrado. Al deducir que se trata de un tronco lo que se hace es fusionar el árbol al que pertenecía este vox y este nuevo árbol en uno sólo, puesto que parece tratarse de un árbol con dos copas y un tronco. Lo que se hace es fusionar los dos clusters más cercanos que tienen en común un voxel. Es lo que se podría llamar clustering jerárquico que lo que hace es unir clusters cercanos. De forma que si se detecta un vox con un nivel de gris alto en un punto conflictivo se fusiona, si el nivel de gris no es alto pasamos al punto 2.

2. Si no se cumple lo anterior, se actualizará una tabla hash en la que se guarda como clave los identificadores de los árboles coincidentes con el árbol que se está construyendo en este momento y como valor se guardan los vox coincidentes. Este paso es fundamental para conocer posteriormente el número de vox coincidentes entre dos árboles. En el punto siguiente se determina si fusionar o no dos árboles en función del número de vox coincidentes.

3. Una vez que ya hemos analizado todos los vox ya tendremos actualizada la tabla hash de coincidencias de vox con puntos y además para cada árbol no sólo se guardan



los vox que pertenecen a él y que tienen puntos, sino que se guardan también los vox que no tienen puntos. De esa forma lo que se hace si no se ha cumplido el punto 1, es mirar si el número de vox coincidentes entre el árbol que se está construyendo y los árboles con los que haya coincidencias, si la suma de coincidencias es mayor que el mínimo de la suma de las alturas dividido entre 2, considero que coinciden muchos puntos, de modo que fusiono los dos árboles en uno. Llamamos vox coincidentes a la suma del los vox que tienen puntos (x,y,z) más los vox que no tienen puntos y que coinciden entre dos árboles. Si el número de vox coincidentes cumple dicha condición lo que se hace es fusionar dos árboles en uno puesto que se deduce que se trata de un mismo árbol. La condición que han de cumplir es superar: $\min(\text{altura Arbol1}, \text{altura Arbol2}) / 2$. Si se cumple esta condición considere que para el árbol más pequeño coinciden como mínimo la mitad de sus vox en altura, con lo cual supongo que son coincidencias suficientes para determinar que los dos árboles Árbol 1 y Árbol 2 son el mismo. En este punto también se está usando clustering jerárquico, ya que unimos los dos clúster más cercanos, los que consideramos que tienen suficiente número de coincidencias. La condición se ha establecido en función de los resultados obtenidos con las pruebas para las parcelas permanentes 1 y 6. Aunque esta condición debería de ajustarse más para poder mejorar los resultados.

4. Si no se cumple el punto 3 (que el número de vox coincidentes supere dicho valor) lo que se hace es repartir los puntos pertenecientes al vox conflictivo entre los árboles que tenga más próximos ya que se descarta la fusión entre árboles. Para realizar el reparto de puntos se emplea **Lógica Fuzzy**.

Desde el punto de vista fuzzy, para cada clase ω_i , se puede definir un conjunto borroso μ_i sobre el universo X , $\mu_i : X \rightarrow [0,1]$ que asigna a cada muestra x_k lo que se conoce como *grado de pertenencia* de dicha muestra al subconjunto ω_i , denominado *clase* a partir de este momento. El grado de pertenencia de x_k a la clase ω_i se expresa como $\mu_i(x_k) = \mu_{ik}$. [24]

En el caso de conjuntos borrosos se dice que una muestra puede pertenecer a diferentes clases y así se habla por ejemplo de que x_k pertenece a una clase ω_i con grado de pertenencia μ_{ik} y a la vez a la clase ω_j con grado de pertenencia μ_{jk} . Como ejemplo, supongamos que se tienen dos clases ω_i y ω_j , en este caso podríamos decir que la muestra x_k pertenece a dichas clases con los siguientes grados de pertenencia $\mu_{ik} = 0.7$ $\mu_{jk} = 0.3$, donde estos valores pertenecen al intervalo continuo $[0,1]$.[8]

Dada una nueva muestra x_s , se ha de decidir a qué clase pertenece ω_j , de entre las disponibles. En nuestro caso la muestra sería un punto (x,y,z) y la clase sería un árbol.

De esta forma se obtiene la siguiente secuencia de grados de pertenencia para x_s ,

$$\mu_{s1} \equiv \mu(x_{s1}, \omega_1), \mu_{s2} \equiv \mu(x_{s2}, \omega_2), \dots, \mu_{sc} \equiv \mu(x_{sc}, \omega_c)$$



Finalmente, x_s se clasifica como perteneciente a la clase ω_j que proporcione el máximo grado de pertenencia según la siguiente regla de decisión,

$$\mu_{sj} > \mu_{sk} \forall k \neq j \text{ con } k, j = 1, 2, \dots, c$$

Aplicando lo explicado en esta sección para nuestro algoritmo, definimos unos fuzzy set [24] que indican el grado de pertenencia de cada punto del vox a cada uno de los árboles por la siguiente fórmula:

$$\mu_{ji} \equiv \mu(x_{sj}, \omega_i) = \max(0, 1 - d(x_{sj}, \omega_i)).$$

Donde:

μ_{ji} es el grado de pertenencia del punto j al árbol i.

$d(x_{sj}, \omega_i)$ es la distancia del punto j al árbol i. Al tratarse de distancias, es necesario realizar la transformación siguiente $[1 + d(x_{sj}, \omega_i)]^{-1}$, de forma que si la distancia es cero, el grado de apoyo resulta ser la unidad y si la distancia es la máxima posible el grado de apoyo se aproxima a cero [24].

La decisión sobre qué punto pertenece a qué árbol se toma en función del grado de pertenencia y el objetivo es asignar el punto al árbol que mayor grado de pertenencia tenga.

Se puede usar cualquier tipo de distancia, pero en este caso hemos optado por la distancia **Euclídea**.

Se denomina **distancia euclídea** entre dos puntos A(x1,y1) y B(x2,y2) del plano a la longitud del segmento de recta que tiene por extremos A y B. Puede calcularse así:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Para asignar cada punto a un árbol, se hará en función de este grado de pertenencia definido y lo que se hará es que cada punto se asignará al árbol al que tenga mayor grado de pertenencia. Al asignar puntos individuales a cada árbol también se ha de modificar los valores máximos y mínimos de x, y, z de cada árbol.



Para calcular la distancia Euclídea se tomará como puntos de posicionamiento del árbol el punto más alto del mismo, que será el que se use para posicionarlo.

De forma que para una situación de conflicto hay dos posibles acciones Fusión de árboles o asignar puntos de forma individual a un árbol u otro en función del grado de pertenencia. Se tomará una u otra acción en función de las condiciones que se cumplan.

Los puntos anteriores se pueden resumir diciendo que para resolver la situación de conflicto se van a emplear dos métodos fundamentales: la fusión o reparto de puntos en función del grado de pertenencia a un árbol u otro.

4.4 Calcular la localización de cada árbol y la altura

Una vez que se hayan realizado todos los pasos no nos quedará ningún vox ni ningún punto por asignar. De modo que ya tendremos una lista con todos los árboles existentes. Llegados a este punto debo de calcular la localización de cada árbol. Para determinar la localización de cada árbol se va a seguir el criterio de posicionar cada árbol en el punto más alto de su copa, que sería lo más sencillo. Se podría aplicar cualquier otro criterio, como calcular el epicentro de todos los puntos que contiene el árbol.

En nuestro caso para posicionar el árbol se ha hecho lo siguiente: coger los puntos brutos ($P(x,y,z)$) que pertenecen al voxel y quedarnos con el que tenga el valor de z más alto y esa será la altura total del árbol, la altura de copa se guardará al ir buscando cada árbol y esta altura se obtiene gracias a la expresión analítica descrita en el punto anterior.

Cuando se realizaron las pruebas de entrenamiento se observó que se localizaban árboles altos que en realidad no existían, eso puede ser debido a que los árboles no crecen hacia arriba totalmente rectos, de modo que una pequeña desviación de un árbol muy alto puede tener un impacto realmente alto. Para reducir este problema se tomó la siguiente medida: eliminar aquellos árboles que tuvieran pocos puntos (menos de 3) o aquellos que sólo tuviesen puntos en un vox y midiesen más de 10 metros, esto último para no eliminar a los árboles más pequeños. De esa forma conseguimos mejorar un poco la situación pero no lo suficiente. Habría que hacer más pruebas de entrenamiento e ir afinando los resultados.



4.5 Resolver el problema de la segmentación

Como se explico en el primer punto se divide toda la zona en grid. Esa división hay que hacerla de tal forma que se coja siempre x metros más a cada uno de los lados. Ese x debería de ser un valor mayor que el diámetro de cualquier árbol, como se muestra en la figura 26 del apartado 4.1.1 en nuestro caso podría tener un valor mínimo de 18 metros. De esa forma nos aseguramos que ningún árbol tenga la parte de los puntos en un grid otra parte en otro grid. Llegados a este punto ha llegado el momento de juntar todos los grid y resolver los problemas de los bordes para que no contabilicemos ningún árbol varias veces y que tampoco dejemos de contabilizar ningún árbol.

Para resolver el problema de la segmentación, superponemos los grid y los árboles que haya en zonas superpuestas se contabilizan sólo una vez. Cuando se esté aplicando el algoritmo si se encuentran puntos en zonas de borde muy al límite los ignoraremos, ya que pertenecerán a un árbol que contabilizaremos en el grid vecino. En nuestro algoritmo no se ha implementado la parte de la segmentación, ya que las parcelas de ejemplo con las que contábamos para la ejecución del algoritmo son procesables todas unidas.



5 Implementación

La implementación del algoritmo se ha realizado como una extensión de gvSIG. GvSIG es una herramienta gratuita orientada al manejo de información geográfica. Se caracteriza por una interfaz amigable, siendo capaz de acceder a los formatos más usuales de forma ágil tanto ráster como vectoriales.

Dada su naturaleza de software libre (open source) es de gran interés para la comunidad internacional de desarrolladores y, en concreto, para los ambientes universitarios por su componente I+D+I. Desde gvSIG se permite ampliar las funcionalidades de la aplicación, así como desarrollar aplicaciones totalmente nuevas a partir de las librerías utilizadas en gvSIG.

Para poder dar resultados exactos y tener la posibilidad de contrastar estos resultados con otros estudios era necesario realizar el desarrollo de este algoritmo. Después de un estudio de otra herramienta GIS de mercado ArcGIS, la cual conozco de debido a varios años trabajando en el mundo GIS con dicha herramienta, opté por usar gvSIG ya que es una herramienta gratuita. Puesto que este algoritmo es fruto de un trabajo de investigación, me parece interesante que cualquier desarrollador, estudiante o profesional en aspectos forestales pueda usarlo sin ningún problema de licencia.

A continuación se va describir de una forma breve el desarrollo realizado. No entraremos en detalle en cuestiones particulares de la implementación en gvSIG ya que no es el objetivo de este trabajo.

El lenguaje de programación para el desarrollo es java.

Hay cuatro clases:

- LocalizedTree es la clase principal desde la que se hace la integración en GVSIG para que este desarrollo sea una extensión de la herramienta GVSIG.
- Vox es la clase desde la que se van a gestionar todas las operaciones relacionadas con los vox.
- EspacioVoxel es un singleton y gestiona todas las operaciones relacionadas con el espacio vox.
- Árbol es la clase encargada de la gestión de las operaciones relacionadas con cada árbol.



5.1 LocalizedTree

Esta clase se encarga de proporcionar la estructura necesaria para añadir esta extensión a GVSIG y de ir realizando todo el proceso para la ejecución correcta, es decir, primero itera por todas las alturas filtrando los puntos correspondientes a cada altura y rellenando EspacioVoxel.

Una vez que se ha completado EspacioVoxel realiza la búsqueda de los árboles según el proceso descrito en el apartado 4. Una vez que se han buscado todos los árboles genera una capa gráfica con ellos y lo añade al mapa. En esta clase se va a guardar una tabla hash con los árboles resultantes del algoritmo. La clave de la tabla hash será un identificador secuencial y el valor será el árbol con todos los atributos correspondientes. Es importante destacar que el identificador del árbol no va a cambiar en ningún momento, en caso de que se fusionen dos árboles en uno, el árbol resultante se quedará con el identificador del primer árbol y el otro se eliminará de la lista, pero su identificador no se le asignará a ningún otro árbol nuevo.

5.2 Vox

Para cada Vox se guarda la siguiente información:

double dX: Coordenada X.

double dY: Coordenada Y.

double dZ: Coordenada Z.

int nPuntos: Número de puntos que contiene el vox.

boolean bEscrutado: nos indica si el vox ya ha sido escrutado.

Vector IArboles: lista de árboles a la que pertenece el vox.

Vector<Value[]> vPuntos: lista de puntos que contiene el vox.

double dXMax=0.0: Máximo valor de x del vox.

double dXMin=0.0: Mínimo valor de x del vox.

double dYMax=0.0: Máximo valor de y del vox.

double dYMin=0.0: Mínimo valor de y del vox.

double dZMax=0.0: Máximo valor de z del vox.

double dZMin=0.0: Mínimo valor de z del vox.

double dSueloMax=0.0: Máximo valor del suelo en el vox.

double dSueloMin=0.0: Mínimo valor del suelo en el vox.

boolean bVoxParcial=false: Nos indica si este vox pertenece a más de un árbol.



Es importante destacar que el vox en todo momento contiene información de todos los puntos (x,y,z) que lo forman, con lo cuál en cualquier momento se puede pasar de información vox a información de puntos, sin perder la ventajas de trabajar con la información bruta y las ventajas de compactarla en vox (píxel).

5.3 EspacioVoxel

Destacar que esta clase es un singleton, puesto que el proceso de realizar la búsqueda por capas se hace con distintas hebras, y de esta forma se evita que varias hebras actualicen el espacio vox simultáneamente, pudiendo producirse algún tipo de incoherencia. Teniendo una instancia única no habrá esa posibilidad.

5.4 Árbol

Para cada árbol se guarda la siguiente información:

int nIdArbol: Identificador del árbol.

Los tres atributos siguientes guardan el posicionamiento del árbol, que es el punto más alto del mismo. Es decir, es el primer punto del árbol que se detecta.

double dX: Coordenada X del punto más alto del árbol.

double dY: Coordenada X del punto más alto del árbol.

double dZ: Coordenada Z del punto más alto del árbol.

Vector<Vox> lVox: Vector con los véxel que contiene este árbol.

Vector<Vox> lVoxSinPuntos: Vector con los véxel sin puntos que contiene el árbol.

Vector <Value[]> vPuntos: Puntos sueltos que se le añaden al árbol, cuando se fusiona. Coordenadas mínimas y máximas de las x, y y z del árbol para poder determinara las dimensiones máximas del árbol tanto el altura como en anchura.

double dXMax=0.0: Coordenada máxima de X del árbol.

double dXMin=0.0: Coordenada mínima de X del árbol.

double dYMax=0.0: Coordenada máxima de Y del árbol.

double dYMin=0.0: Coordenada máxima de Y del árbol.

double dZMax=0.0: Coordenada máxima de Z del árbol.

double dZMin=0.0: Coordenada máxima de Z del árbol.

double dZSuelo=0.0: Coordenada del suelo correspondiente al árbol, se calcula como la media entre la mínima y máxima del vox.



6 Resultados Obtenidos

Las pruebas se han realizado con dos parcelas (llamadas parcelas permanentes) de las que teníamos información de campo para poder contrastar los resultados y que son han servido para depurar y afinar el algoritmo. Este algoritmo se ha ejecutado también en otras 25 parcelas más pequeñas (llamadas parcelas volantes) de las que no se tiene la información de posicionamiento de los árboles, pero se conoce información suficiente para poder hacer un análisis de los resultados. Los resultados obtenidos de estas parcelas no se mostrarán aquí gráficamente, se mostrarán en el capítulo 7 en el que se realizará un análisis de los mismos.

A continuación se muestran gráficamente los resultados obtenidos con la ejecución del algoritmo para las dos parcelas permanentes con las que hemos realizado las pruebas de entrenamiento y de las que se posee información gráfica, estas parcelas las hemos llamado parcela 1 y parcela 6. La numeración no sigue un orden pues estas parcelas pertenecen a un grupo más numeroso de parcelas empleadas para desarrollar modelos a nivel de rodal. Se eligieron estas dos parcelas pues suponían dos estados tipo del nivel de desarrollo de una masa forestal suficientemente diferentes. Con estas dos parcelas se cubre la gran mayoría de las estructuras forestales presentes en el bosque. Así si se consigue que un algoritmo proporcione resultados satisfactorios en ambas parcelas, es previsible que al usar el algoritmo en el resto del monte los resultados sean igualmente buenos.

A continuación se muestran los resultados obtenidos para la parcela 1:

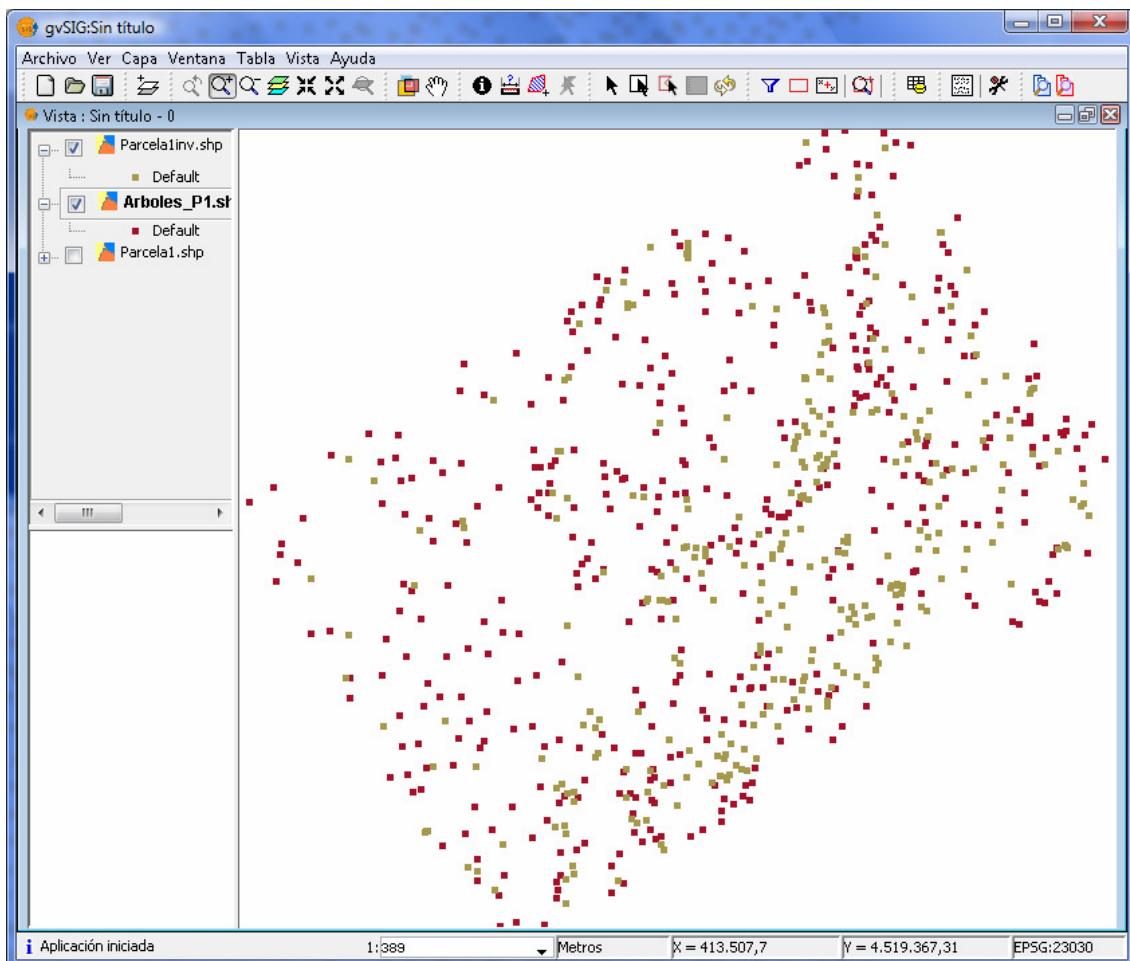


Figura 32. Resultado gráfico del algoritmo para la parcela 1.

En la imagen se muestran en color marrón los árboles procedentes del inventario de campo y en color granate los árboles procedentes de algoritmo. Los árboles procedentes de campo son 317, mientras que los árboles obtenidos con el algoritmo son 442. En nuestro algoritmo se obtienen árboles en exceso.



gvSIG:Sin título

Archivo Ver Capa Ventana Tabla Campo Vista Ayuda

Tabla: Tabla de atributos: Arboles_P1.shp

ID	X	Y	Z	XMin	XMax	YMin	YMax	ZMin	ZMax	ZSuelo
1688.0	413509.3	4519360.0	9.4676	413509.3	413509.3	4519360.0	4519360.0	9.4676	9.4676	1318.712
86.0	413487.4	4519340.0	29.53182	413485.7	413488.9	4519339.0	4519348.0	25.02993	29.53182	1318.017
159.0	413492.2	4519338.0	28.88134	413491.8	413492.2	4519338.0	4519340.0	27.33937	28.88134	1318.9
672.0	413498.9	4519367.0	25.29675	413498.8	413498.9	4519367.0	4519368.0	21.17513	25.29675	1316.864
1397.0	413507.0	4519349.0	15.07174	413504.0	413514.6	4519345.0	4519350.0	2.90063	22.48345	1319.565
285.0	413489.1	4519327.0	27.82941	413489.1	413490.9	4519326.0	4519327.0	23.91705	27.82941	1319.1
1106.0	413500.9	4519365.0	22.63966	413499.0	413501.2	4519364.0	4519365.0	20.83618	22.85583	1317.187
1883.0	413512.5	4519355.0	7.53083	413512.5	413512.5	4519355.0	4519355.0	7.53083	7.53083	1319.529
798.0	413484.1	4519353.0	24.09449	413483.0	413485.8	4519351.0	4519354.0	21.65222	24.09449	1316.895
29.0	413493.6	4519337.0	30.24062	413491.5	413494.5	4519337.0	4519341.0	20.79966	30.48752	1318.882
771.0	413458.1	4519334.0	24.49219	413457.0	413458.1	4519334.0	4519334.0	21.06476	24.49219	1318.995
1098.0	413489.1	4519360.0	22.38385	413488.3	413490.4	4519360.0	4519362.0	19.8727	22.71495	1316.953
1952.0	413497.5	4519340.0	6.44414	413497.2	413498.5	4519339.0	4519340.0	0.4117	30.74978	1319.385
33.0	413488.8	4519341.0	30.31818	413487.7	413491.1	4519340.0	4519348.0	23.21801	30.63113	1318.086
281.0	413499.7	4519326.0	27.97314	413498.1	413499.7	4519325.0	4519326.0	24.78105	27.97314	1320.276
420.0	413468.2	4519320.0	26.12009	413467.4	413469.1	4519320.0	4519321.0	21.22701	26.12009	1319.071
456.0	413466.3	4519340.0	26.03163	413465.1	413467.9	4519339.0	4519340.0	22.8773	26.03163	1318.625
1768.0	413502.9	4519348.0	8.59264	413502.9	413506.3	4519347.0	4519349.0	3.74927	8.95248	1319.297
1818.0	413478.9	4519316.0	7.26114	413478.3	413484.2	4519315.0	4519317.0	2.86936	7.40229	1319.157
496.0	413495.9	4519357.0	26.94236	413495.5	413496.0	4519357.0	4519358.0	25.05734	26.94236	1317.722
1480.0	413494.4	4519318.0	11.4686	413494.4	413498.9	4519318.0	4519319.0	2.2526	11.4686	1320.181
143.0	413499.6	4519327.0	28.02777	413498.1	413504.8	4519326.0	4519329.0	15.45961	28.02777	1320.164
653.0	413491.1	4519357.0	25.61513	413489.0	413491.5	4519357.0	4519359.0	19.97018	25.87617	1317.057
41.0	413500.4	4519342.0	30.45606	413500.4	413501.8	4519342.0	4519349.0	27.32003	30.45606	1319.297
779.0	413472.3	4519343.0	24.62652	413471.7	413473.2	4519343.0	4519344.0	20.66708	24.62652	1318.101
273.0	413474.1	4519322.0	27.08872	413474.1	413475.4	4519321.0	4519323.0	22.20844	27.08872	1318.921
1442.0	413520.7	4519346.0	12.09113	413518.7	413528.2	4519343.0	4519348.0	3.22618	12.09113	1320.161
786.0	413520.1	4519346.0	24.40933	413520.1	413520.4	4519345.0	4519346.0	22.05124	24.40933	1320.19
2110.0	413502.4	4519348.0	4.04349	413500.2	413502.9	4519347.0	4519349.0	0.0	28.28662	1319.026
2044.0	413516.2	4519353.0	5.58616	413514.0	413516.6	4519351.0	4519354.0	0.00434	9.21906	1319.596
17.0	413486.1	4519321.0	30.87394	413486.1	413491.0	4519321.0	4519347.0	26.52775	30.91501	1319.304
2032.0	413496.7	4519339.0	5.72615	413494.2	413498.7	4519336.0	4519344.0	0.0	30.74978	1319.049
1518.0	413504.6	4519350.0	11.43806	413504.6	413511.2	4519350.0	4519352.0	7.48277	11.96182	1319.375
37.0	413484.7	4519342.0	20.11297	413482.0	413488.0	4519340.0	4519348.0	0.03065	20.11297	1317.682

0 / 442 Total registros seleccionados.

Aplicación iniciada Metros X = 413.521,9 Y = 4.519.326,67 EPSG:23030

Figura 33. Resultado alfa-numérico para la parcela 1.

A continuación se muestran los resultados obtenidos para la parcela 6:

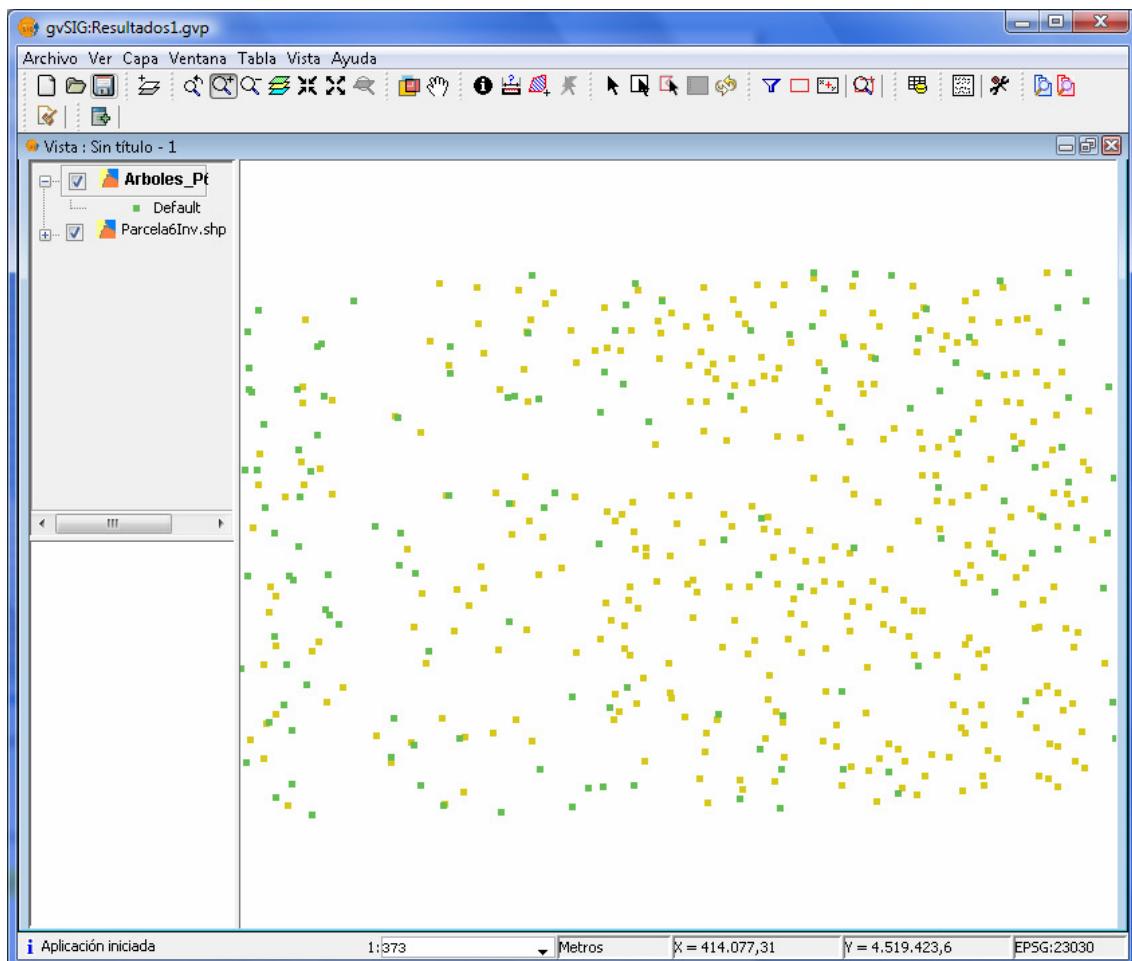


Figura 34. Resultado gráfico del algoritmo para la parcela 6.

En la imagen se muestran en color marrón los árboles procedentes del inventario y en color verde los árboles obtenidos con el algoritmo.



ID	X	Y	Z	XMin	XMax	YMin	YMax	ZMin	ZMax	ZSuelo
1240.0	414118.5	4519402.0	17.30198	414113.0	414121.8	4519399.0	4519408.0	0.0	17.30198	1353.144
159.0	414083.7	4519403.0	21.09045	414082.0	414084.5	4519403.0	4519404.0	12.61026	21.09045	1357.204
1397.0	414114.1	4519387.0	14.48945	414113.9	414117.6	4519386.0	4519388.0	0.0	14.65617	1358.968
798.0	414106.4	4519425.0	18.13667	414104.1	414108.4	4519425.0	4519429.0	0.0189	18.3664	1352.464
905.0	414125.1	4519389.0	17.13886	414124.0	414130.9	4519388.0	4519392.0	0.0	17.13886	1357.12
29.0	414077.9	4519419.0	23.03528	414075.1	414085.5	4519416.0	4519424.0	13.97839	23.05321	1354.9
157.0	414083.5	4519401.0	21.86612	414082.1	414085.3	4519401.0	4519402.0	14.51507	21.86612	1358.49
375.0	414120.3	4519424.0	20.14079	414117.5	414124.9	4519421.0	4519427.0	14.25374	20.14079	1352.41
536.0	414114.6	4519410.0	19.34586	414104.5	414142.8	4519410.0	4519411.0	16.20453	19.34586	1353.485
350.0	414083.4	4519416.0	20.05007	414083.3	414084.8	4519416.0	4519417.0	17.45994	20.05007	1355.762
1488.0	414105.5	4519417.0	6.77335	414105.5	414105.5	4519417.0	4519417.0	6.77335	6.77335	1355.066
545.0	414121.6	4519414.0	19.80822	414116.0	414124.9	4519411.0	4519417.0	0.0	19.80822	1352.387
718.0	414091.2	4519398.0	18.07952	414087.0	414098.4	4519396.0	4519406.0	0.0	18.78444	1353.668
371.0	414137.5	4519423.0	20.171	414132.2	414141.8	4519422.0	4519429.0	14.08834	20.82165	1349.67
1155.0	414099.4	4519389.0	14.49955	414096.0	414104.2	4519386.0	4519392.0	0.0016	16.57059	1354.836
167.0	414081.7	4519409.0	21.28046	414080.1	414088.8	4519407.0	4519415.0	12.53649	21.28046	1354.36
783.0	414092.8	4519420.0	18.39425	414089.7	414098.9	4519415.0	4519423.0	0.0	18.5357	1352.085
955.0	414115.7	4519408.0	17.20865	414112.0	414120.5	4519405.0	4519413.0	0.0	17.33739	1354.51
781.0	414130.0	4519419.0	18.2912	414128.7	414130.9	4519417.0	4519420.0	14.90764	18.2912	1353.192
80.0	414078.1	4519417.0	22.11583	414078.1	414079.4	4519417.0	4519417.0	19.92787	22.11583	1355.658
777.0	414120.3	4519418.0	18.23125	414118.1	414124.9	4519414.0	4519419.0	0.03553	18.2923	1352.387
1444.0	414104.0	4519388.0	12.30573	414104.0	414104.9	4519388.0	4519389.0	11.25648	12.80544	1360.447
786.0	414127.3	4519421.0	18.29003	414126.1	414129.9	4519418.0	4519421.0	0.35243	18.29003	1353.257
754.0	414128.7	4519410.0	18.24627	414124.0	414131.9	4519407.0	4519415.0	0.01194	18.74736	1353.223
141.0	414077.7	4519390.0	21.44437	414075.0	414078.5	4519388.0	4519391.0	17.93455	21.44437	1360.804
934.0	414097.0	4519400.0	17.15225	414093.0	414102.8	4519396.0	4519409.0	0.0	19.79762	1354.811
505.0	414088.6	4519393.0	19.01601	414086.7	414095.8	4519391.0	4519396.0	0.0	19.01601	1354.501
78.0	414082.9	4519414.0	22.20773	414079.5	414090.2	4519412.0	4519424.0	0.0	22.20773	1354.213
183.0	414141.2	4519417.0	21.44525	414134.0	414144.9	4519413.0	4519423.0	0.56789	21.44525	1350.301
1309.0	414103.8	4519415.0	16.09076	414103.1	414107.7	4519413.0	4519416.0	0.01569	16.53198	1355.221
769.0	414088.9	4519415.0	18.51322	414085.0	414096.9	4519412.0	4519427.0	0.0	18.98171	1353.839
1246.0	414115.5	4519403.0	14.65623	414113.0	414120.6	4519401.0	4519406.0	0.0	16.59014	1353.144
551.0	414138.3	4519416.0	19.09571	414137.1	414139.2	4519415.0	4519417.0	15.36232	19.57611	1352.879
571.0	414127.8	4519422.0	10.16067	414126.0	414128.7	4519421.0	4519426.0	12.49127	10.16067	1352.412

Figura 35. Resultado alfanumérico del algoritmo para la parcela 6.

Con el algoritmo se obtiene 142 árboles, frente a los 365 del inventario de campo, este caso el inventarios es bastante más pequeño.



7 Evaluación cualitativa de los resultados por expertos

En este apartado se va a realizar una evaluación a grossó modo de los resultados obtenidos y mostrados en el capítulo anterior.

Además de la cantidad de árboles obtenidos con el algoritmo es importante la localización de los mismos, para ver en que coincide y en que difiere. Esta es quizás la primera evaluación del método. Estos datos han sido analizados con por el colaborador externo, experto en temas forestales y que ha opinado y guiado en todo el proceso de búsqueda de la solución.

Para posicionar los árboles en este algoritmos se ha seguido el criterio de localizarlo en el punto (x,y,z) más alto del árbol, es decir en el primer punto que se encuentra al ir buscando árboles.

Este análisis sólo se va a realizar para las parcelas 1 y 6, que son para las que poseemos información de campo sobre la localización de cada árbol.

PARCELA 1:

Los datos obtenidos para esta parcela en cuanto a precisión geométrica son bastante buenos, en general el algoritmo es capaz de posicionar los árboles con bastante exactitud. A continuación se muestran unos ejemplos sacados del resultado gráfico de esta parcela donde se puede apreciar este hecho. La pequeña variación de posición en determinados árboles puede ser debida a que un error provocado por otro criterio de localización del árbol en los datos de campo, posicionar el árbol donde empieza el tronco por ejemplo, o también puede ser debido a pequeños errores de medición.

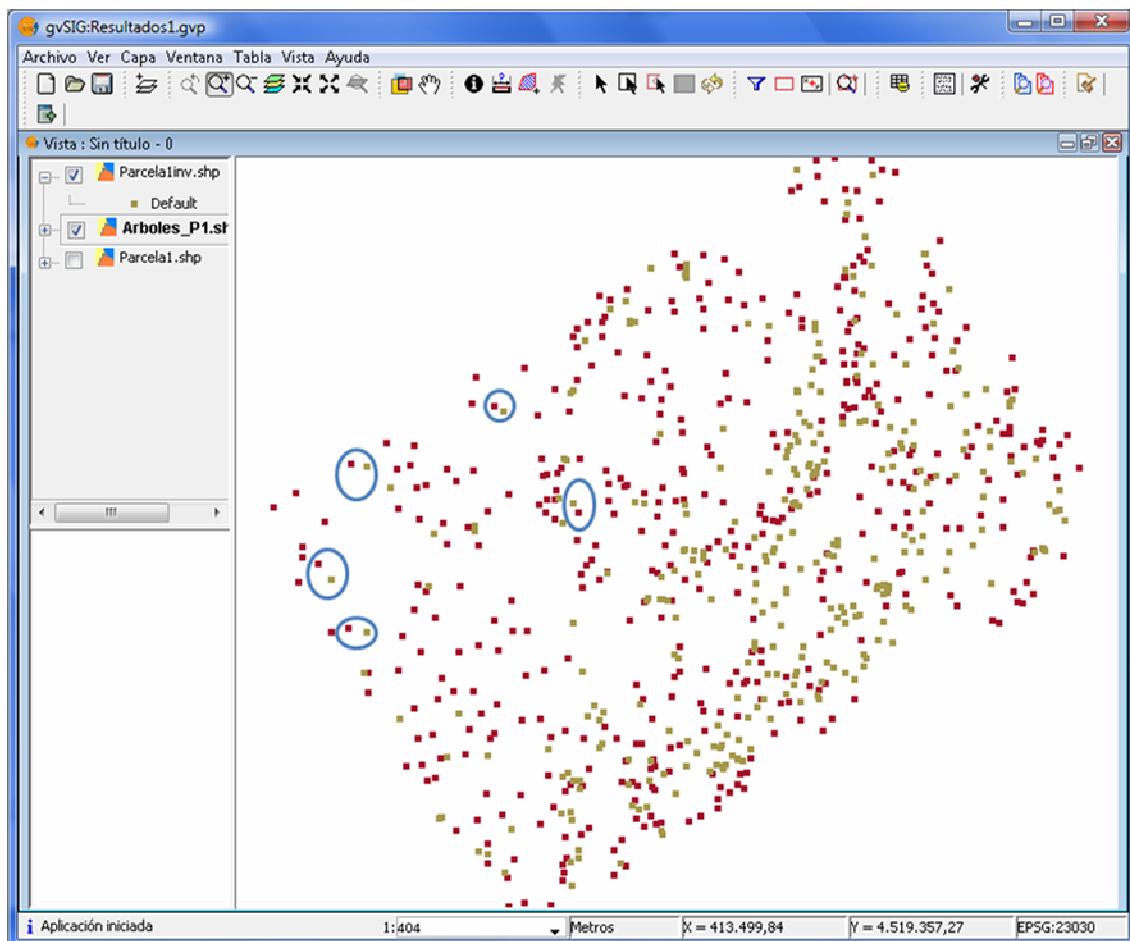


Figura 36. Árboles del inventario y del algoritmo que coinciden en localización.

También se aprecian fallos, ya que hay árboles que nuestro algoritmo no detecta, posiblemente porque son muy pequeños o hayan sido fusionados con otro por su cercanía. A continuación se muestran unas imágenes donde se puede apreciar esta observación:

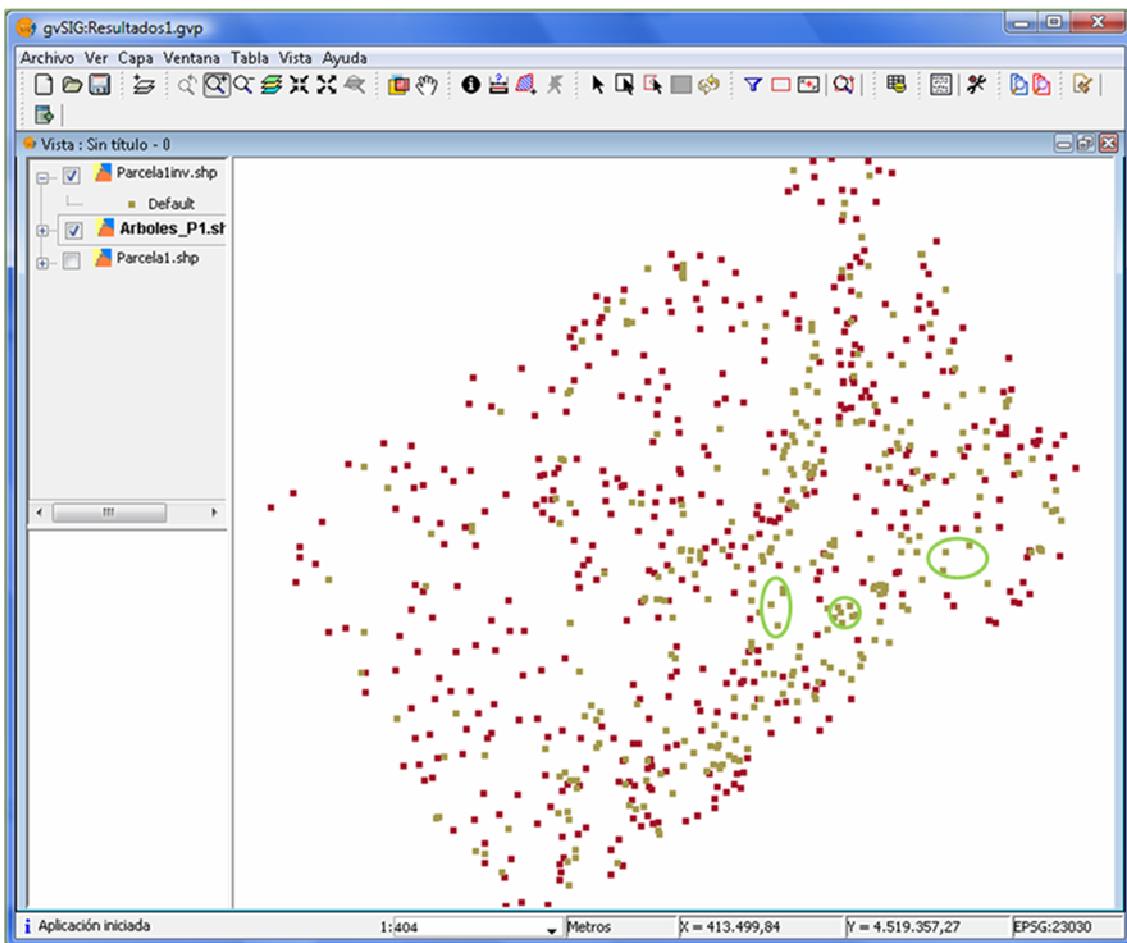


Figura 37. Árboles que aparecen en el inventario de campo pero no aparecen en los resultados de nuestro algoritmo.

Por otro lado nuestro algoritmo estima árboles donde no los hay. Esto puede ser por diversas causas entre ellas puede ser debido a árboles que no crecen hacia arriba, es decir a árboles que crecen inclinados y pequeñas inclinaciones en árboles de 30 metros de longitud pueden dar una desviación muy grande. Por ello en el algoritmo se optó por tomar una medida para intentar corregir este fallo. La medida fue no contar como árboles aquellos que tuvieran pocos puntos menos de 3 o aquellos que sólo tuviesen puntos en un vox y midiesen más de 10 metros, esto último para no eliminar a los árboles más pequeños. De esa forma conseguimos mejorar un poco la situación pero no lo suficiente. Habría que hacer más pruebas de entrenamiento e ir afinando. A continuación se muestra un ejemplo de esta situación en la que los puntos de color granate son los resultantes de nuestro algoritmo y los de color marrón son los resultantes de las mediciones de campo.

Se muestra con círculos naranja las situaciones en las que ocurre esta situación.

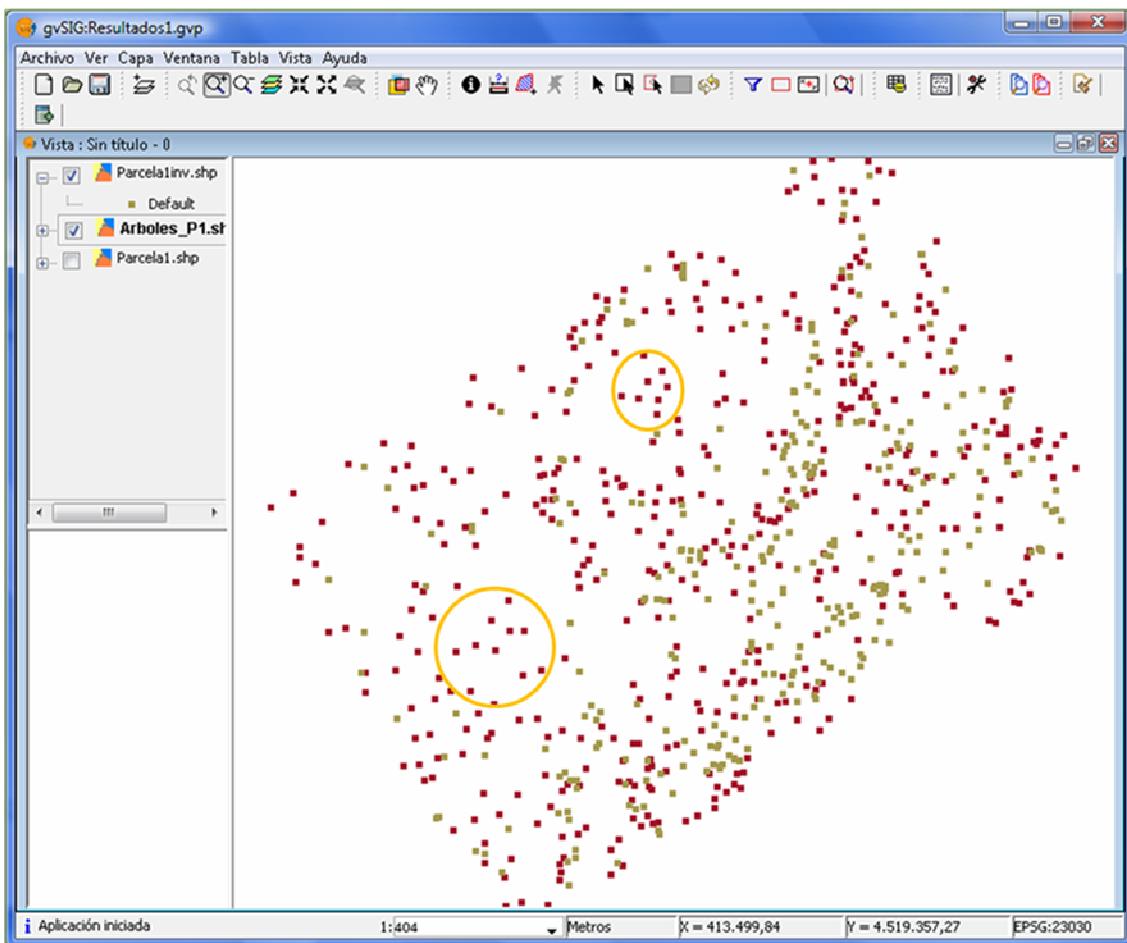


Figura 38. Estimación de árboles que no existen en el inventario de campo.

PARCELA 6:

En esta parcela se observa que a simple vista hay mucha menor cantidad de árboles prácticamente la mitad, esto puede ser debido a que en ella la mayoría de los árboles se localizan muy próximos unos a otros. Con lo cual en el proceso de fusión se han debido de fusionar. A continuación se muestra una imagen en la que se puede apreciar esta situación.

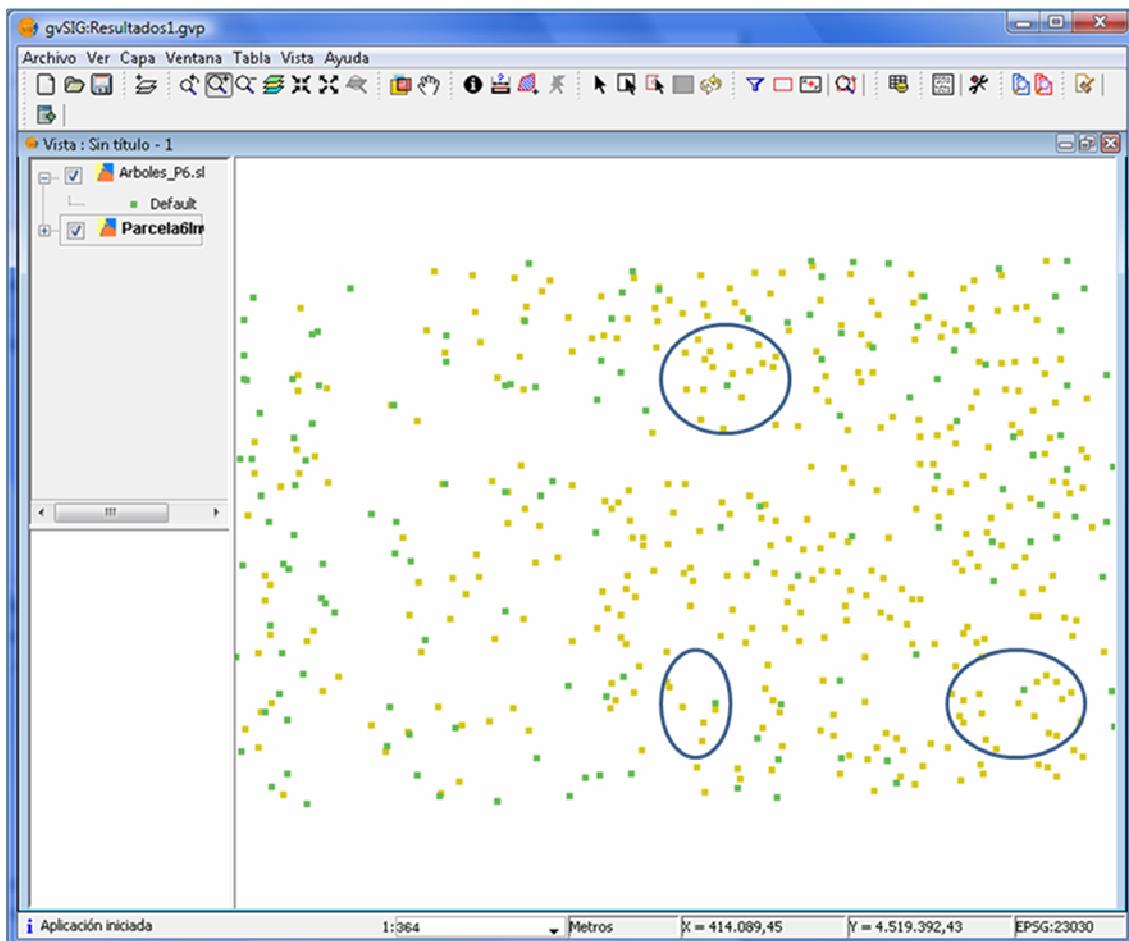


Figura 39. Imagen en la que el algoritmo fusiona varios árboles en 1.

En la figura anterior se muestran con elipses azules lo que podría corresponder a la extensión del árbol de nuestro algoritmo (el verde) mientras que en el inventario de árbol en esas zonas se localizan muchísimos árboles.

Al igual que en la parcela 1 hay árboles que se localizan correctamente. A continuación se muestra una imagen de ello:

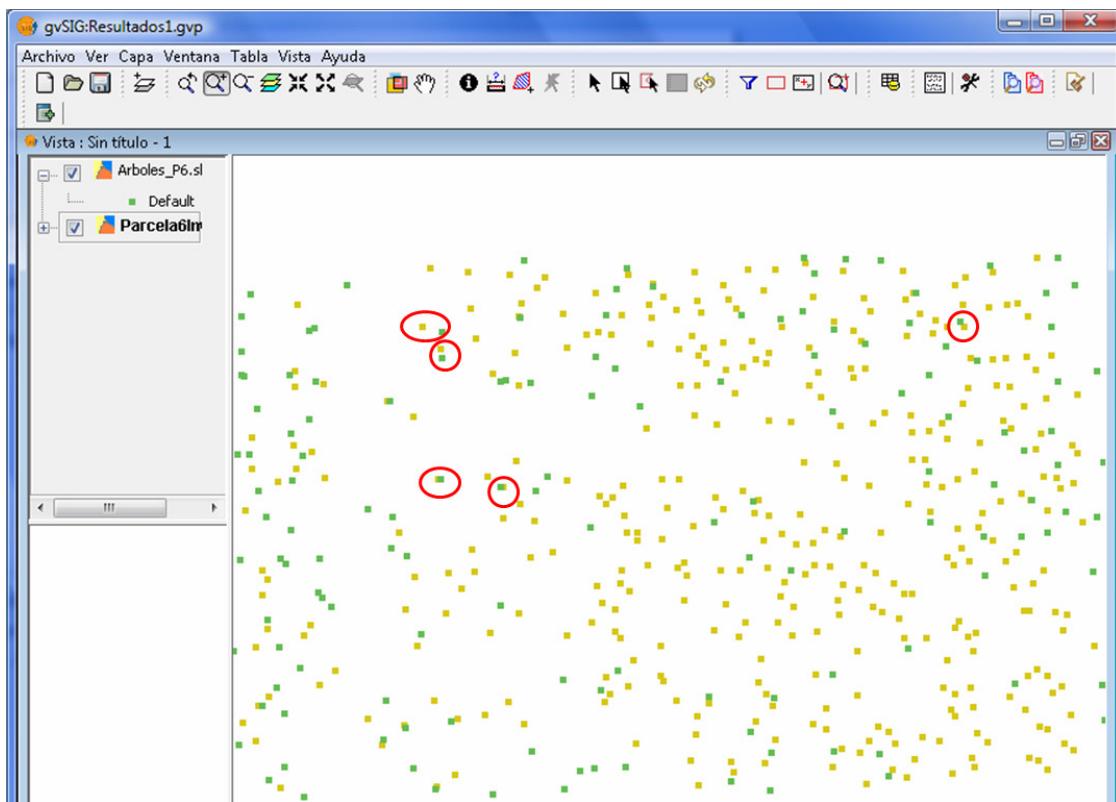


Figura 40. Imagen en la que el algoritmo localiza árboles correctamente.

Se muestran con un círculo rojo las coincidencias entre árboles del inventario y árboles obtenidos con el algoritmo.

Y por el contrario hay árboles que no se encuentran:

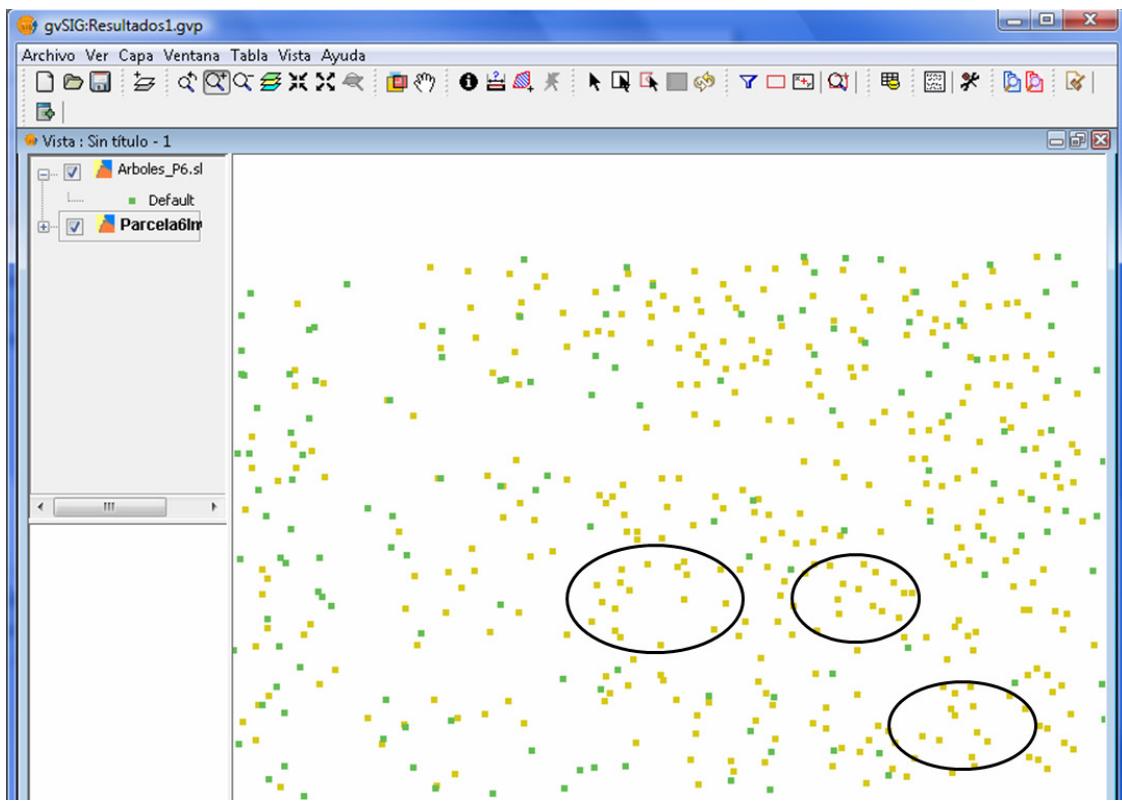


Figura 41. Árboles que existen en el inventario de campo pero no en el resultado del algoritmo.

O se encuentran árboles que no existen, aunque en esta parcela esto ocurre con menos frecuencia:

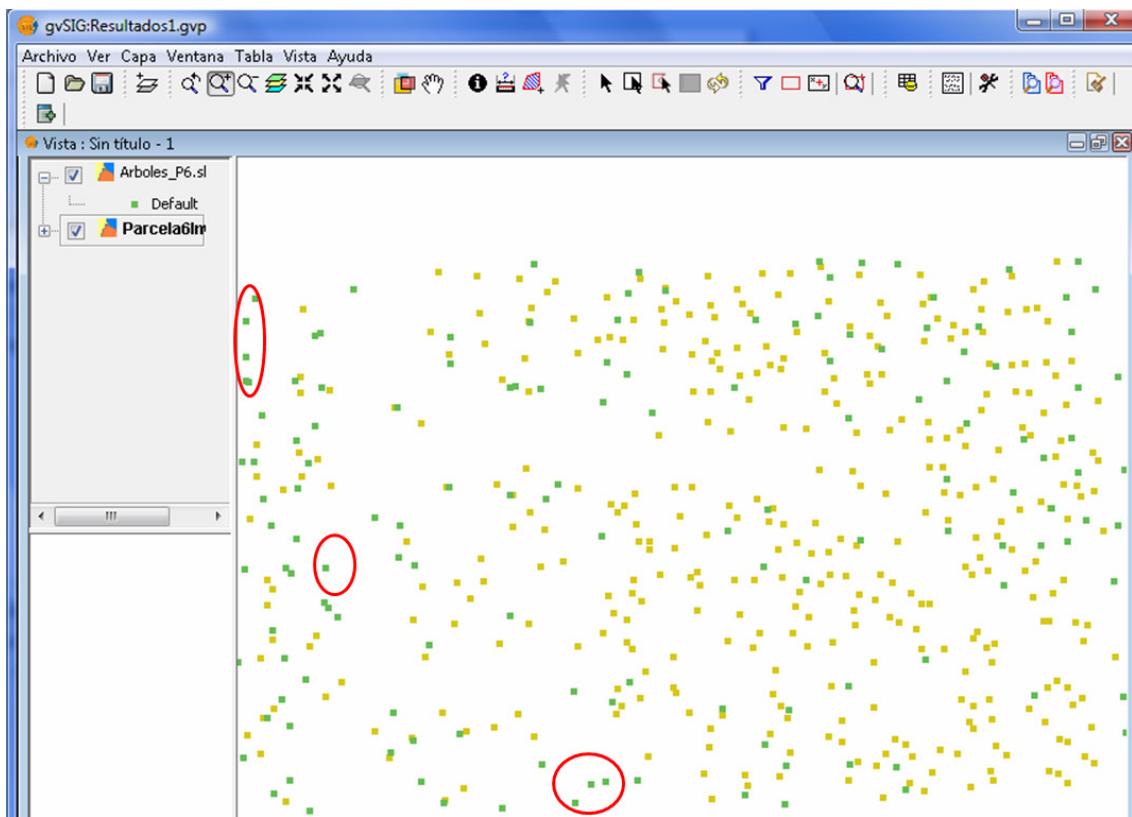


Figura 42. Árboles localizados con el algoritmo que en realidad no existen.



8 Evaluación de los resultados obtenidos por métodos automáticos para la detección de árboles

Tras la fase de diseño e implementación de un determinado método automático para la búsqueda de árboles individuales, hay que evaluar los resultados que dicho método ofrece. El output de un método de detección de árboles individuales es un listado de "árboles ficticios" con al menos tres campos (X,Y,H) donde (X,Y) son las coordenadas en el plano y H la altura estimada para cada árbol detectado. Existen diversas formas de evaluar los resultados que el método proporciona. Haremos una pequeña revisión de las distintas modalidades de evaluación de resultados. Cada una de estas formas de evaluación permitirá ver como son de precisos los resultados obtenidos, dependiendo del posterior uso que se vaya a hacer con la salida de un determinado método. Vamos a ver cuatro tipos distintos de evaluación de resultados que informan sobre aspectos distintos por lo que ninguno excluye a otro. Para la validación de nuestro algoritmo sólo se ha realizado dos tipos de evaluación: la evaluación de la precisión en la estimación de parámetros forestales y la evaluación de la precisión estimando las funciones de tamaños de arbolados. Ambas evaluaciones se han realizado para las 25 parcelas llamadas "parcelas volantes".

Tipos de evaluación:

1. Evaluación de la precisión geométrica.
2. Evaluación del número de aciertos-fallos.
3. Evaluación de la precisión en la estimación de parámetros forestales.
4. Evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.

8.1 Evaluación de la precisión geométrica.

Se analiza cuando queremos obtener estimaciones de lo cerca que están en el espacio los "árboles ficticios" de los árboles reales. Este tipo de análisis es quizás el menos interesante pues en general los métodos de detección de árboles individuales proporcionan estimaciones poco precisas de parámetros más importantes. Tener un listado de coordenadas precisas de los árboles del bosque sería por supuesto muy



interesante para la gestión forestal, sin embargo no es una información esencial. Por otro lado generalmente los métodos de detección de árboles individuales nos son capaces de detectar todos los árboles e inventan otros que no existen y generalmente producen estimaciones incorrectas de otros parámetros más importantes para la gestión del monte.

La evaluación de la precisión geométrica es compleja entre otras cosas porque no hay una regla general para asociar un árbol real (cuya posición ha de conocerse) a otro "ficticio". Este es el primer problema que hay que solventar. Una vez resuelta la forma en que se van a ligar ambos listados de árboles hay que ver como son de importantes las diferencias de coordenadas reales y ficticias.

Dado que hay que conocer las posiciones de una serie de árboles reales los inventarios previos deben recopilar esta información de forma precisa. Esto implica grandes cantidades de trabajo de campo pues hay que emplear métodos más o menos laboriosos para calcular las posiciones de todos los árboles que se vayan a utilizar en este tipo de análisis.

A continuación se muestra un ejemplo de este tipo de evaluación en el que se puede ver la variación entre la localización de un mismo árbol con distintos métodos. Y en algunos casos incluso se encuentran árboles que no existen o por el contrario no se encuentran árboles que sí existen.

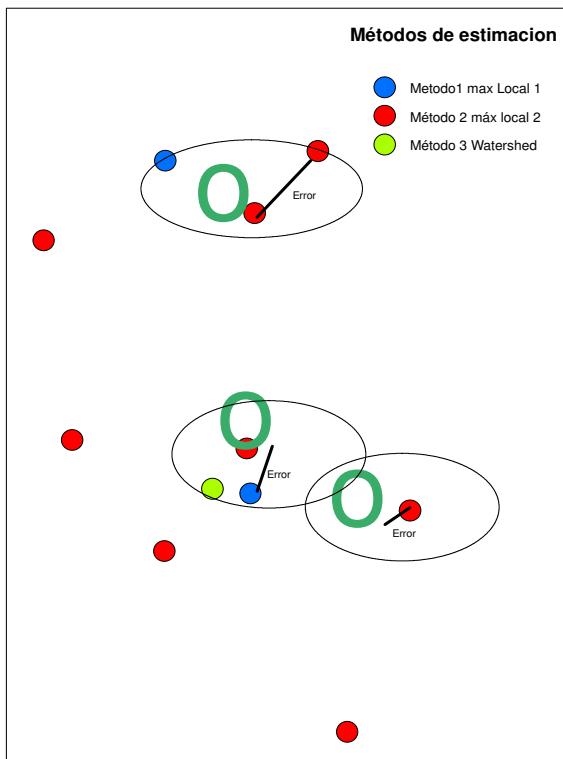


Figura 43. Precisión geométrica por distintos métodos.

8.2 Evaluación del número de aciertos-fallos.

Este tipo de evaluación es en cierta medida similar al primero de los métodos descritos. Uno de los principales inconvenientes es, como se decide si hay acierto o no. En casos favorables pueden emplearse métodos visuales, fotointerpretación sobre ortofotos, sin embargo hay que tener en cuenta que en muchas ocasiones las ortofotos son complejas, poco nítidas o difusas y no es fácil diferenciar de visu los árboles presentes en la realidad. Además la metodología general de elaboración de ortofotos puede provocar distorsiones geométricas, que no se presentan en los datos LiDAR, y que pueden invalidar el estudio si este se realiza mediante foto interpretación.

Se podrían emplear datos de campo (Coordenadas XY de los árboles reales) y establecer criterios cuantitativos para clasificar como acierto o error.



La bondad del método se evalúa estudiando el número de árboles detectados, el número de árboles no detectados y el número de puntos etiquetados como árbol pero que en la realidad no lo son.

8.3 Evaluación de la precisión en la estimación de parámetros forestales.

Este tipo de evaluación es sin duda el más empleado. La gestión forestal no suele realizarse, árbol a árbol, generalmente interesa que el monte presente una serie de características generales relacionadas con el conjunto de la masa forestal y éstas son las que se tratan de estimar a partir de los datos LiDAR. Algunas de estas características son, la altura media del arbolado, la densidad o número de árboles por hectárea, la altura dominante o altura de los cien árboles más altos de una hectárea, la biomasa por unidad de superficie, etc. La forma de evaluar la precisión de las estimaciones de éstas características, obtenidas al aplicar un determinado método es la siguiente:

Empleando los árboles del listado de "árboles ficticios" detectados dentro de una serie de parcelas en las que se ha hecho un inventario previo se calculan las variables de masa (Altura media, altura dominante, densidad....) tal y como se haría si los árboles detectados fueran reales. Estas estimaciones se comparan con las mediciones realizadas in situ para cada parcela. Así para cada parcela se obtienen dos valores para cada variable a estudiar. Uno la medición de campo de dicha variable y otro la estimación de dicha variable a partir del algoritmo. Posteriormente se evalúan las diferencias entre estas estimaciones y los datos obtenidos en campo.

Desde el punto de vista práctico es muy importante que para validar este tipo de predicciones no es necesario conocer las coordenadas de cada árbol. Tan solo se necesita conocer las coordenadas del centro de las parcelas donde se han recogido los datos de campo. Esto hace mucho más sencillas las labores de inventario. Así el esfuerzo necesario para cada parcela se reduce en torno a una quinta o una décima parte con lo que se pueden medir muchas más parcelas de campo que permitirán realizar un análisis estadístico más fiable. Para esta parte del trabajo se han empleado en total 25 parcelas de 20 metros de radio.

Aunque las coordenadas de los árboles no intervienen en el proceso de cálculo de ninguna de las variables consideradas, sí que es necesario conocer la ubicación del centro de cada parcela. Para ubicar los centros de las parcelas en los trabajos de



campo se empleo un GPS diferencial modelo Topcon HiperPro. Este receptor permite realizar correcciones diferenciales de fase con lo que la precisión de las coordenadas adquiridas puede alcanzar los 2 cm. Aun así los valores de precisión son netamente peores en entornos con arbolado denso pues la cubierta forestal impide la correcta recepción de la señal GPS. Considerando el efecto del dosel de copas la precisión de los equipos GPS diferencial se sitúa en torno a los 50 cm, (Nasset 2002)

Es importante tener en cuenta que para realizar ciertas estimaciones es necesario contar con modelos de regresión que permitan relacionar alturas con otras variables. Con los datos láser es posible estimar de forma directa alturas. Otras variables asociadas al árbol como el diámetro del tronco o la cantidad de biomasa deben predecirse en base a la altura estimada. En la validación del método que se presenta se evalúan las predicciones de altura media, número de pies y biomasa por unidad de superficie. Para predecir la biomasa por unidad de superficie se emplean la altura de cada uno de los árboles detectados y se estiman las biomassas de cada árbol individual. Posteriormente se calcula la biomasa por unidad de superficie como suma de las biomassas asociadas a cada árbol. Los modelos empleados para predecir la biomasa de cada árbol son los modelos desarrollados por el instituto nacional de investigación agraria (Montero 2005). Estos modelos relacionan la biomasa del árbol con su diámetro de modo que para predecir biomassas fue necesario estimar el diámetro de los árboles a partir de su altura. La predicción del diámetro en función de la altura se ha realizado en base a un modelo de regresión linealizable realizado con una muestra de 1076 árboles en el que el R^2 fue igual a 0.94.

Gráfico del Modelo Ajustado

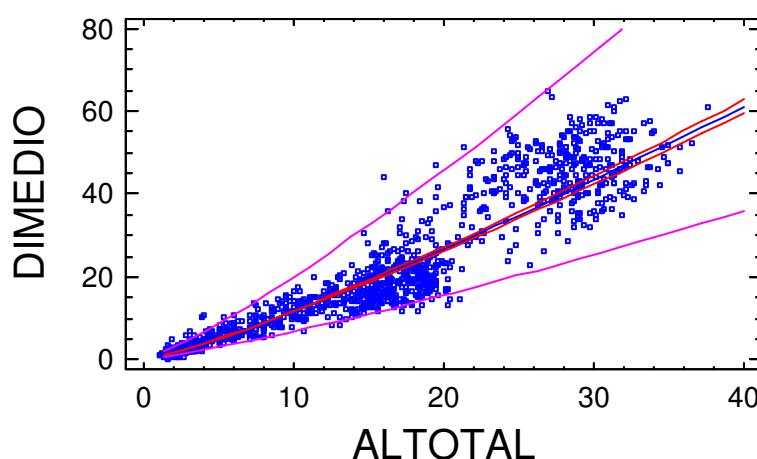




Figura 44. Predicción del diámetro en función de la altura.

8.3.1 Estimación de Alturas medias

La primera variable que se va a analizar en este apartado es la altura media. La altura media de la masa es una variable con un relevante interés para el estudio de ecosistemas forestales. Numerosos aspectos relativos al nivel de madurez de la masa forestal pueden inferirse una vez se conoce la altura media de la misma. El indudable interés de esta variable ha conducido a distintos investigadores a desarrollar modelos para estimar la altura media a partir de datos procedentes de sensores remotos. Así, hace ya 13 años, Nilsson (1996) [16], estudió la capacidad de los sensores LiDAR para estimar esta variable. El estudio de Nilsson se realizó con el fin de obtener estimaciones de altura media para extensiones de terreno relativamente grandes empleando para ello modelos a nivel de rodal.

Una vez obtenidas las predicciones de alturas medias obtenidas a partir del algoritmo. Se calcularon los errores de la estimación como diferencias entre la altura media estimada y la altura media obtenida con datos de campo. Estos errores se representan en la figura 45 a. Esos mismos errores expresados en términos relativos se muestran en 45 b. Como se puede ver en esta figura. Los errores en la estimación de la altura media oscilan la mayoría entre $\pm 5m$. Si bien pueden llegar a ser incluso el doble. En general se observa una ligera mayoría de estimaciones a la alta pues la cantidad de observaciones a la derecha de cero (Altura media real < Altura media estimada) es mayor que la cantidad de observaciones a la izquierda de cero (Altura media real > Altura media estimada). Si observamos los errores en términos relativos. Se puede decir que la mayoría de las estimaciones tienen errores relativos de $\pm 20\%$, si bien estos errores pueden alcanzar el 60%.

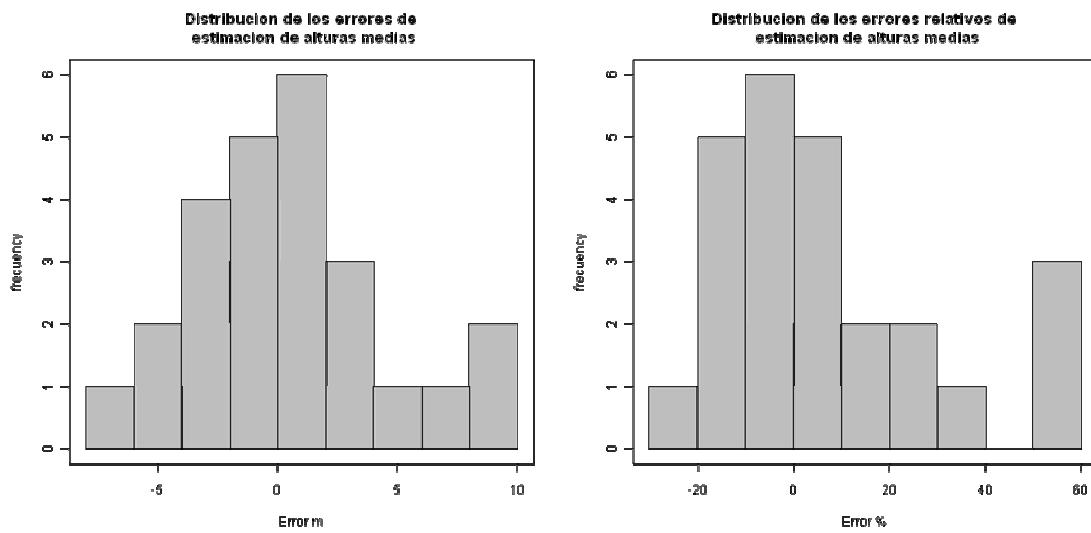


Figura 45. A) Distribución de errores absoluta de estimación de alturas medias. B) Distribución de errores relativos de estimación de alturas medias.

Una vez conocida la magnitud de los errores en términos generales se procedió a estudiar si éstos estaban relacionados con la propia estimación de la altura media. La figura 46 muestra esta relación fuerte entre los errores y la propia estimación. Aparentemente hay una relación entre errores y predicción que hace pensar en una posible calibración del modelo que mejore las estimaciones de alturas medias. La calibración del modelo se realizó mediante un modelo de regresión lineal relacionando predicciones con datos reales. Este modelo además de ser el más sencillo de todos los distintos modelos considerados, fue el que presentó un mayor porcentaje de varianza $R^2=80.7\%$, explicada en la tabla 1. La homocesticidad de los residuos es uno de los principales supuestos en los que se basan los modelos de lineales de regresión fueron contrastadas. Esta propiedad de los residuos fue evaluada empleando para ello la figura 48. Como se puede observar en esta gráfica los residuos se distribuyen de forma similar a lo largo del rango de variación de la variable predictora. Para más detalles consultar el anexo B.

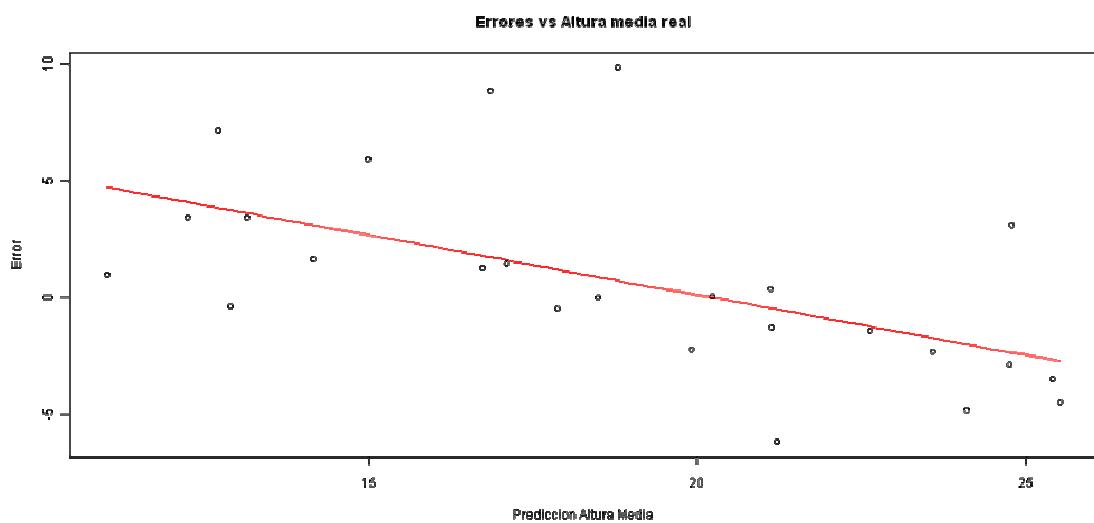


Figura 46. Errores por altura media real

Gráfico del Modelo Ajustado

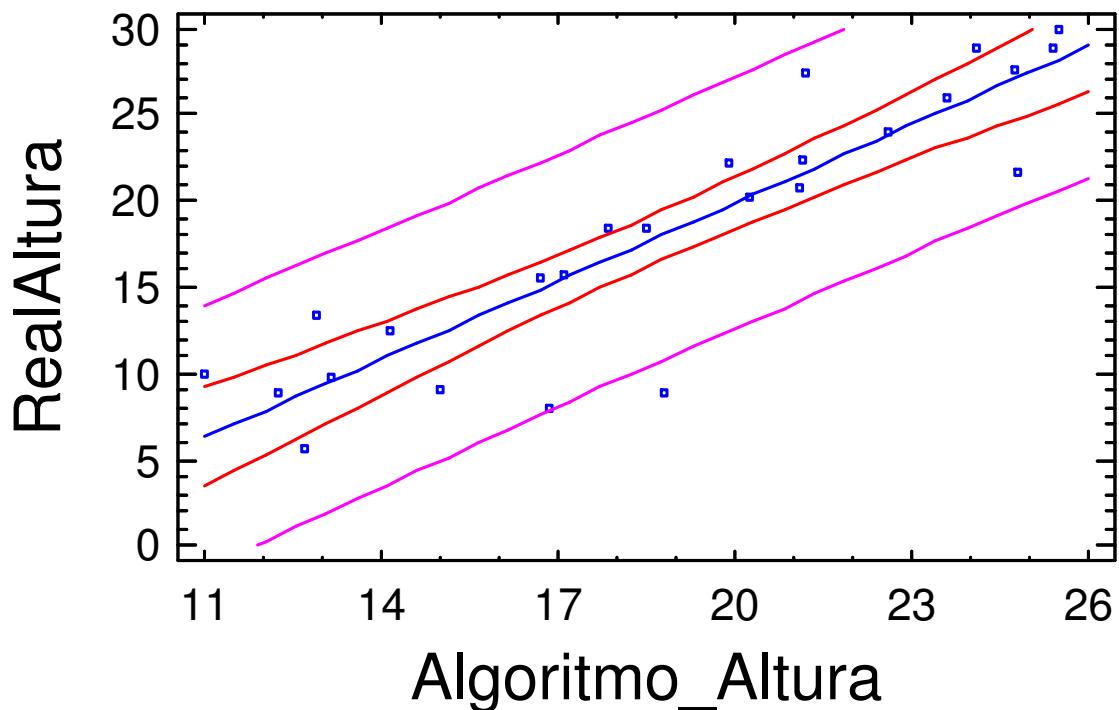


Figura 47. Gráfico de Modelo Ajustado (Altura Real –Altura Algoritmo)

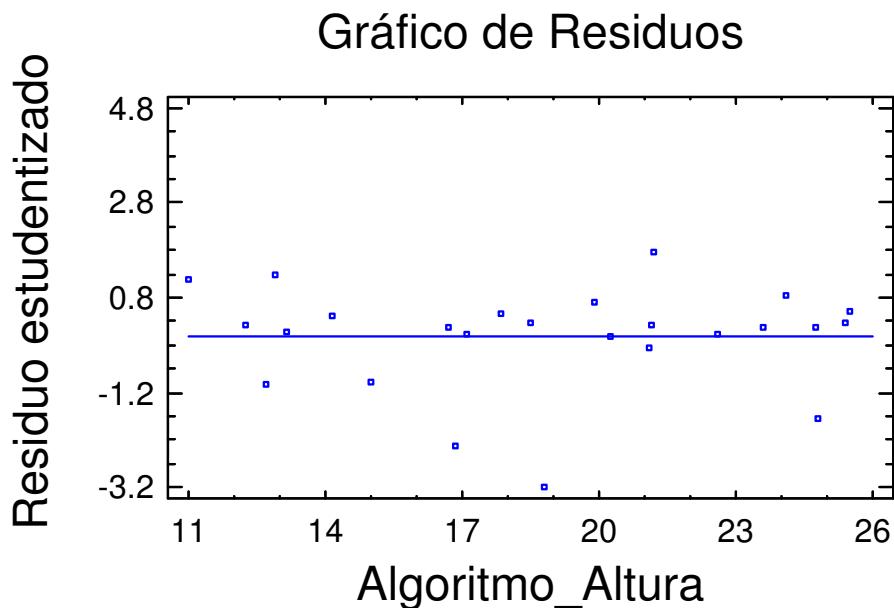


Figura 48. Residuo estudiantizado.

Modelo	Correlación	R-cuadrado
Lineal	0.8984	80.70%
Raiz cuadrada-X	0.8910	79.40%
Raiz cuadrada-Y	0.8849	78.31%
Logarítmico-X	0.8804	77.51%
Exponencial	0.8602	74.00%
Multiplicativo	0.8526	72.70%
Inverso-X	-0.8497	72.20%
curva-S	-0.8319	69.21%
Inverso-Y	-0.7762	60.25%
Doble inverso	0.7667	58.78%
Logístico	<sin ajuste>	

Tabla 1. Comparación Modelos Alternativos

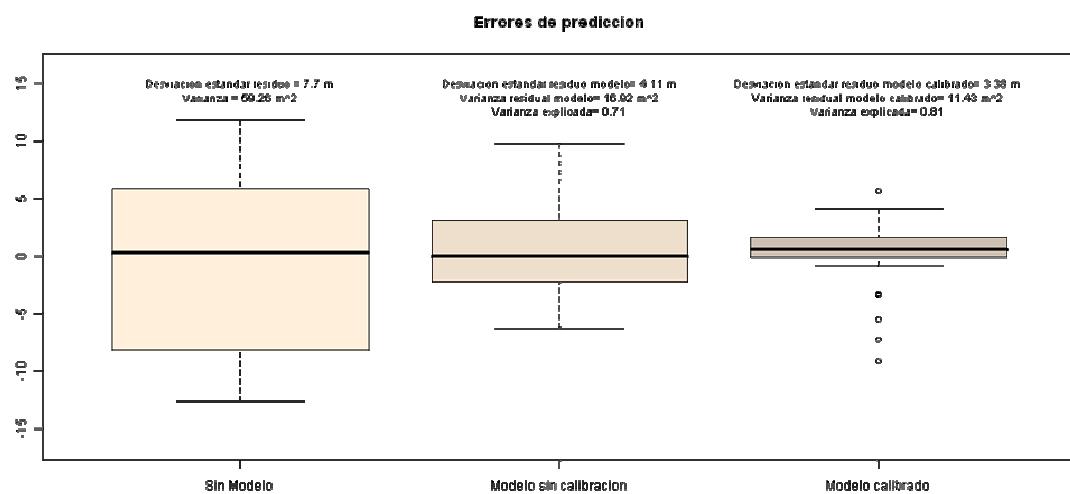


Figura 49. Errores de predicción.

El aumento de la varianza explicada una vez aplicado el modelo (Estimación directa empleando la salida del algoritmo) y una vez aplicado el modelo calibrado (resultados del algoritmo corregidos según el modelo de regresión ajustado), se puede observar en la figura 49. Si bien los resultados obtenidos son francamente buenos, Reducción de un 81% de la varianza residual, estos resultados eran esperables. La altura media no es muy sensible al problema general de la sobre-estimación del número de pies mostrada por este algoritmo, siempre y cuando las proporciones de árboles detectados por encima y debajo de la media medida en campo aparezcan en las mismas proporciones con las que estos aparecen en la realidad. El algoritmo que aquí se presenta ha mostrado resultados satisfactorios estimando los parámetros más importantes de las funciones de distribución de los árboles y esto implica estimar de forma aproximada las proporciones de pies por encima y debajo de la media real Apartado 8.3.

8.3.2 Estimación de número de Pies

La segunda variable que se va a analizar en este apartado es densidad del arbolado o número de árboles por hectárea. Igual que la altura media esta es una variable de indudable interés para la gestión forestal. Si bien esta variable necesita ser complementada con información acerca del tamaño de los árboles. Conocer de forma precisa el número de pies por hectárea es esencial para planificar tareas como, claras,



cortas para favorecer la regeneración o tratamientos para prevenir incendios forestales.

Siguiendo el mismo procedimiento que con la altura media primero se estudiaron los errores al aplicar el algoritmo sin ningún tipo de corrección figura 50 a. Esos mismos errores expresados en términos relativos se muestran en la figura 50 b. Como se puede ver en esta figura. Los errores en la estimación del número de pies presentan errores relativos muy grandes. Esto se debe a que la fusión de vértices no está funcionando como es debido y en general se produce una sobre estimación muy importante del número de pies. Además toda predicción de variables cuyo valor se derive de la suma de la aportación de cada uno de los distintos árboles (La biomasa por hectárea es una de estas variables) se verán seriamente afectadas por este error. Producido estimaciones completamente erróneas debido al ruido introducido al sobre estimar el número de pies.

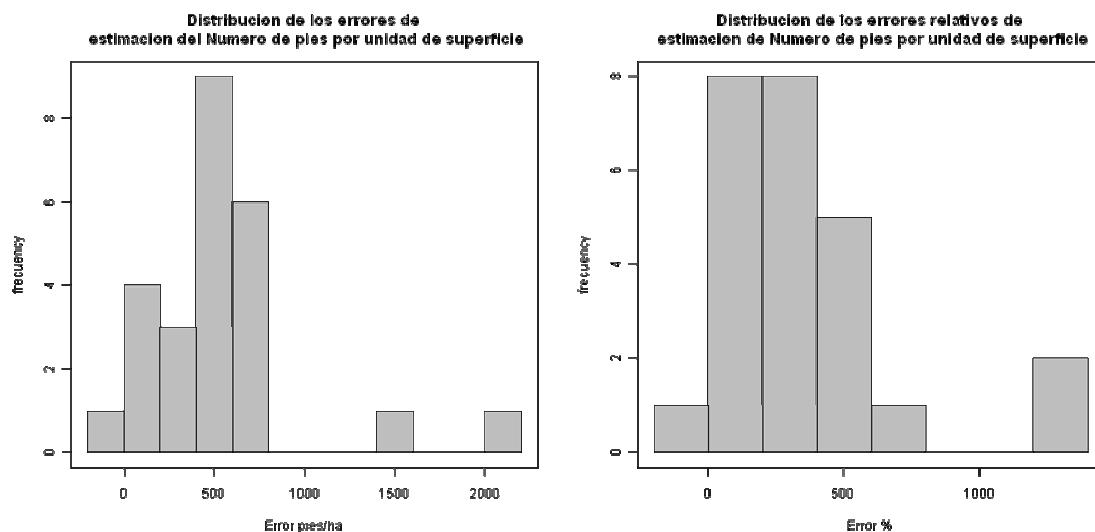


Figura 50. A) Distribución de los errores absoluta de estimación del número de pies por unidad de superficie. B) Distribución de los errores relativos de número de pies por unidad de superficie.

Si se observa la relación de los errores con la estimación se puede apreciar una ligera correlación entre errores y predicción Figura 51. Igual que con las predicciones de altura media se ajustó un modelo de regresión lineal para corregir la sobreestimación del número de pies. En este caso, tanto el porcentaje de varianza explicada por el



modelo lineal apenas explica un 0.1% de la varianza del error. El mejor modelo linealizable tampoco mejora sustancialmente y tan solo llega explicar un 4% de la varianza del número de pies por hectárea. Dado que no hay grandes diferencias entre ambos modelos se optó por emplear el modelo lineal, por su mayor sencillez, en la calibración de los resultados obtenidos por el algoritmo. El modelo ajustado, los residuos y la bondad de ajuste de modelos alternativos puede observarse en Figura 52, Figura 53 y tabla 2.

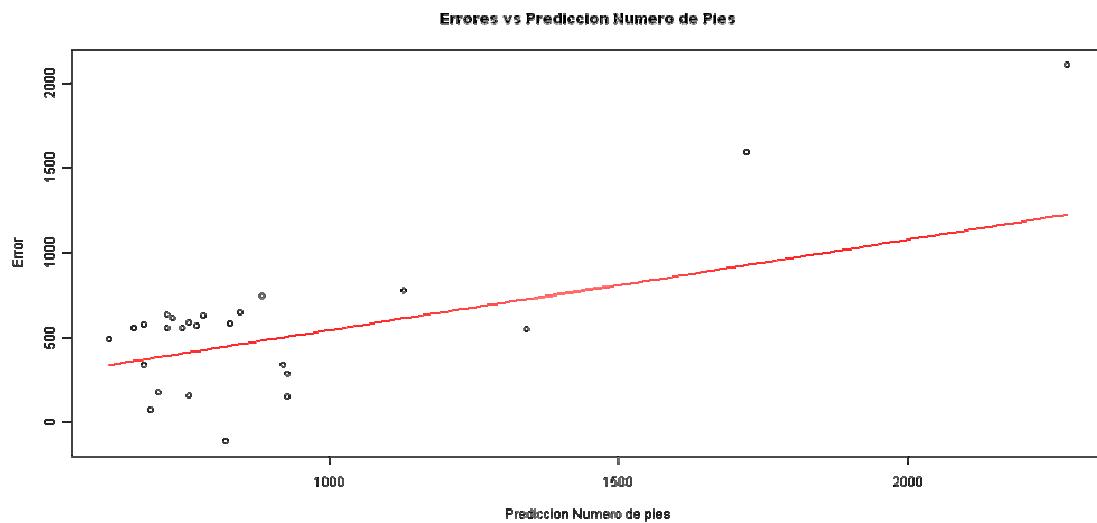


Figura 51. Errores por predicción número de Pies.

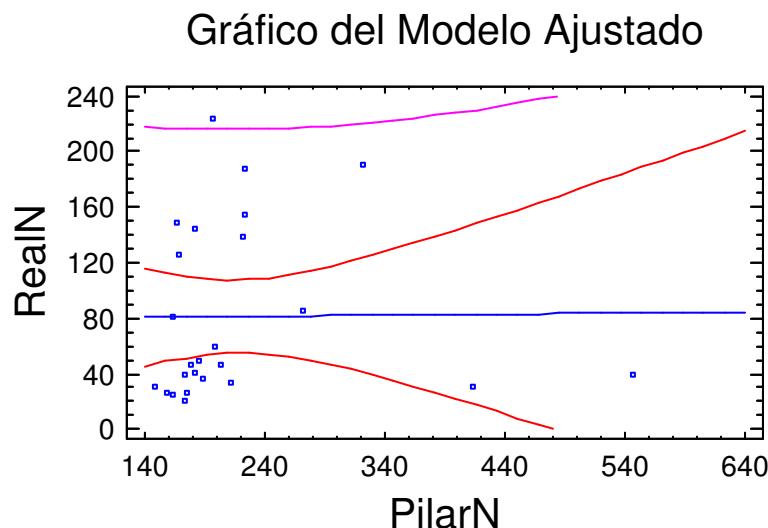


Figura 52. Gráfico del modelo ajustado. (Número de Pies Real- Número de Pies del algoritmo)

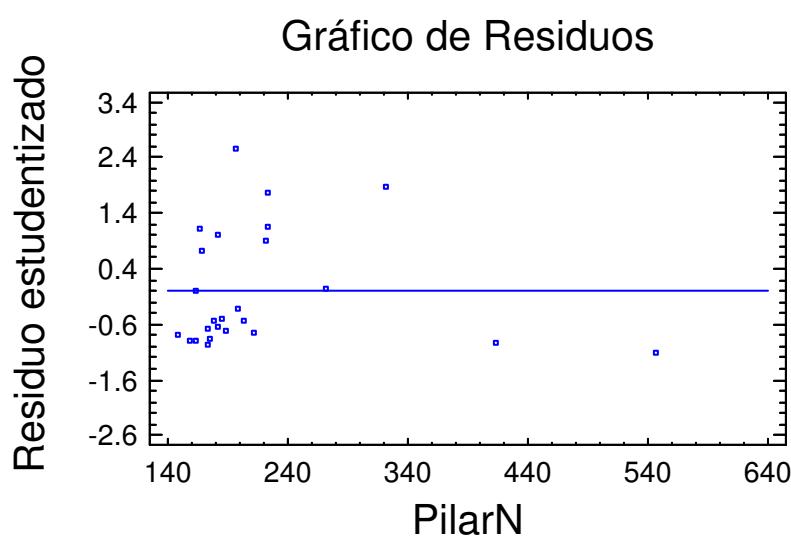


Figura 53. Gráfico de Residuos estudiantizado.



Modelo	Correlación	R-cuadrado
Doble inverso	0.2111	4.46%
curva-S	-0.1835	3.37%
Inverso-X	-0.1720	2.96%
Multiplicativo	0.1011	1.02%
Logarítmico-X	0.0933	0.87%
Raiz cuadrada-X	0.0515	0.26%
Inverso-Y	-0.0486	0.24%
Exponencial	0.0167	0.03%
Lineal	0.0110	0.01%
Raiz cuadrada-Y	0.0108	0.01%
Logístico	<sin ajuste>	
Log Probit	<sin ajuste>	

Tabla 2. Comparación de modelos alternativos.

En la figura 54 se pudo observar claramente como los errores que se obtienen al aplicar el algoritmo sin ninguna corrección están completamente distorsionados, presentando un sesgo importante. Las estimaciones directas del número de pies introducen incertidumbre y su uso está completamente desaconsejado. Por otro lado la calibración con los datos de campo tampoco mejora los resultados.

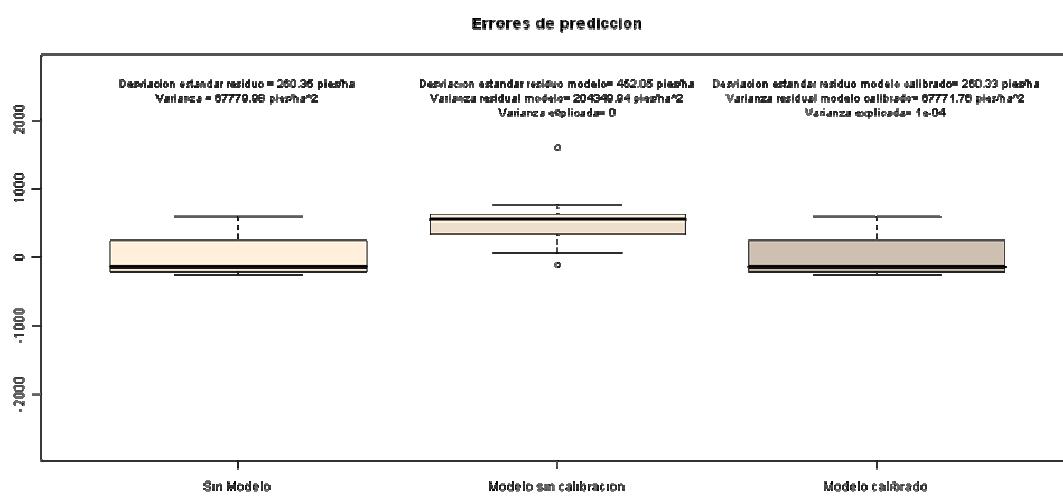




Figura 54. Errores de predicción

8.3.3 Estimación de biomasa por hectárea

La última variable que se va a analizar en este apartado es la biomasa por hectárea. La biomasa o cantidad de materia viva por unidad de superficie está directamente relacionada con la cantidad de CO₂ almacenada por el bosque. Esto la hace especialmente importante dada la influencia de esta variable en procesos ecológicos de carácter medioambiental como el calentamiento global.

La forma de estudiar la precisión de las predicciones se realizó de la misma forma que el estudio de las otras variables. La biomasa por hectárea se calcula como la suma de las biomasas de todos los árboles incluidos en la superficie a estudiar de modo que ya podemos intuir que las estimaciones directas de biomasa estarán notablemente sesgadas debido al efecto de sobre-estimación del número de pies. La biomasa de cada árbol está relacionada con el tamaño de este. El algoritmo que se presenta, produce estimaciones correctas del tamaño del arbolado. Así tenemos que los dos principales factores que influyen en el cálculo final de la biomasa por hectárea, son calculados de forma desigual por el algoritmo. El tamaño medio del arbolado es estimado con precisión por el algoritmo, sin embargo, para la cantidad de pies por hectárea el algoritmo no proporciona ninguna mejora. Los errores obtenidos al aplicar de forma directa el algoritmo se pueden observar en la figura 55.

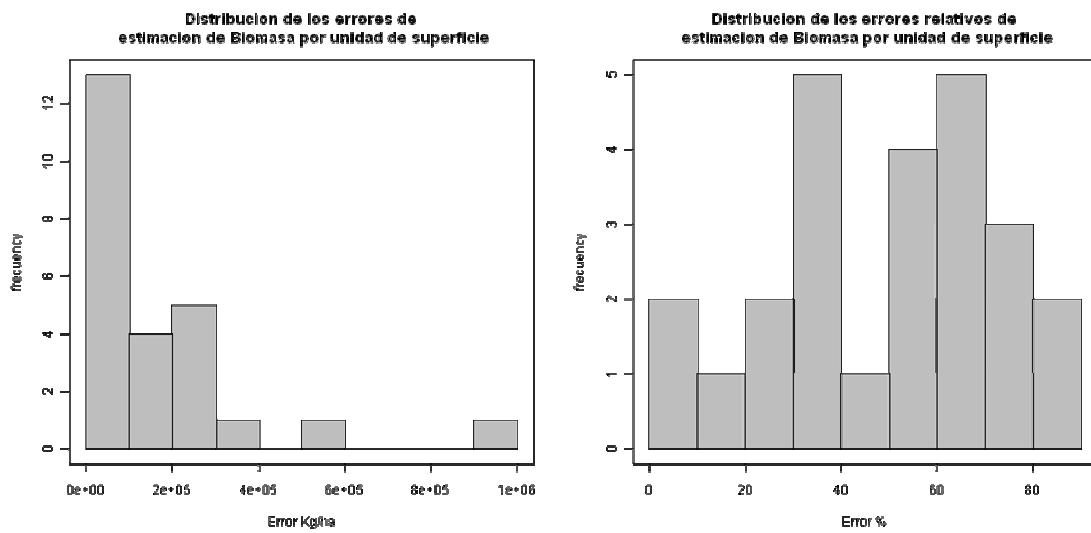


Figura 55. a) Distribución de los errores absolutos de estimación de biomasa por unidad de superficie. B) Distribución de los errores relativos de estimación de biomasa por unidad de superficie.

Igual que se hizo con la altura media y el número de pies se observó la relación entre los errores y las predicciones. En este caso se observa una importante relación entre ambas variables que sugiere emplear un modelo de regresión para corregir los errores. El mejor modelo de regresión linealizable (modelo logarítmico) explicó un 52% de la varianza de esta variable. El modelo lineal explicó un 43% de esta varianza. Dada la mayor simplicidad del modelo lineal y la poca importancia de la mejora aportada por el modelo linealizable se optó por emplear el modelo lineal. El modelo ajustado, los residuos y las varianzas explicadas por modelos alternativos se pueden observar en la tabla 3 y en las figuras 57, figura 58.

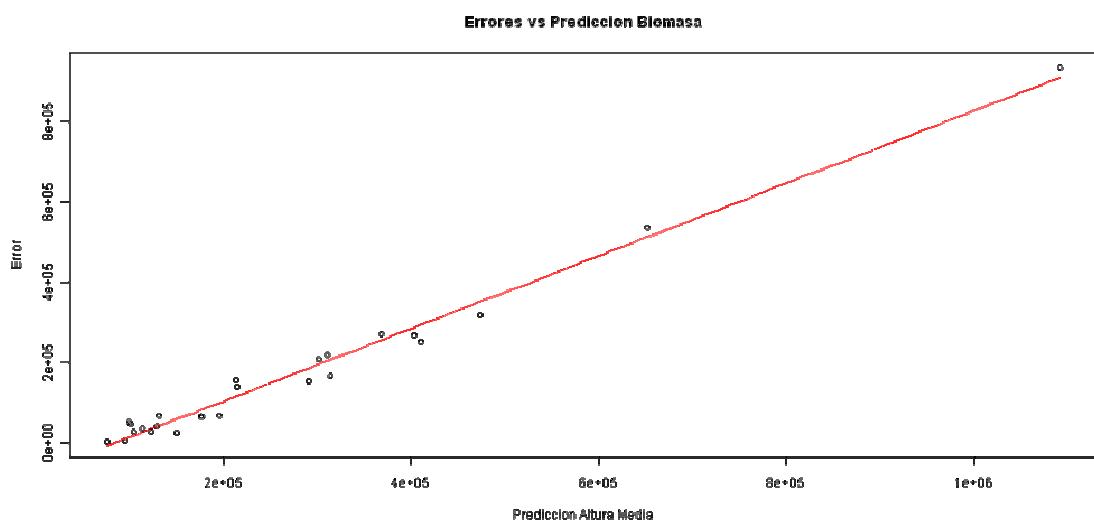


Figura 56. Errores por predicción de biomasa.

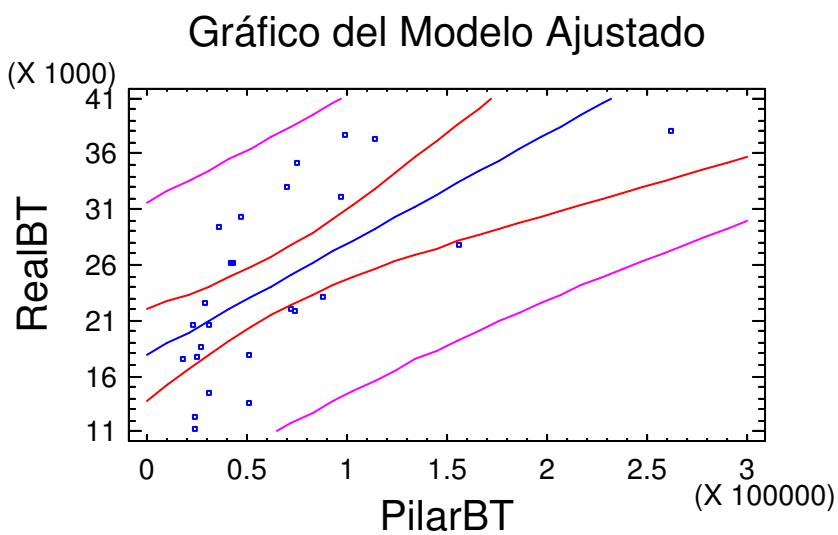


Figura 57. Grafico de modelo ajustado. Biomasa real frente a Biomasa estimada por el algoritmo.

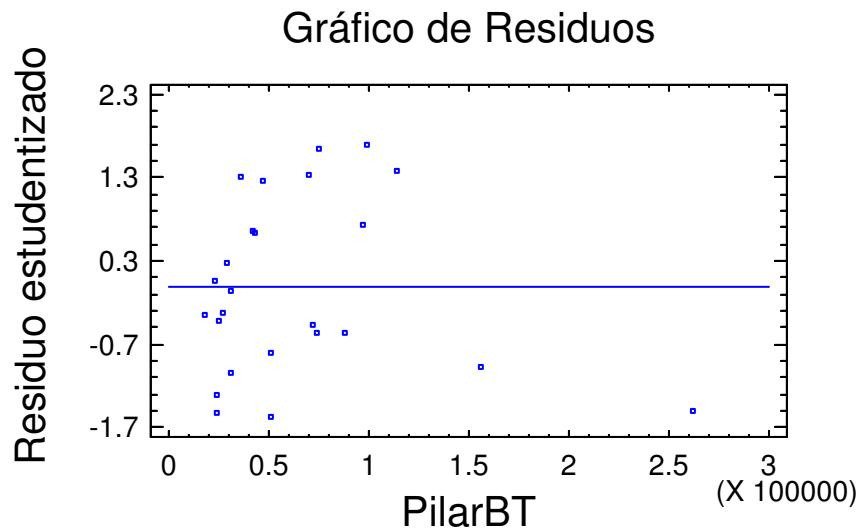


Figura 58. Residuos Estudentizado.

Modelo	Correlación	R-cuadrado
Logarítmico-X	0.7313	53.48%
Raiz cuadrada-X	0.7063	49.88%
Inverso-X	-0.7056	49.79%
Multiplicativo	0.7006	49.09%
curva-S	-0.6921	47.90%
Doble inverso	0.6541	42.79%
Lineal	0.6517	42.47%
Raiz cuadrada-Y	0.6305	39.76%
Exponencial	0.6048	36.58%
Inverso-Y	-0.5418	29.36%
Logístico		<sin ajuste>
Log Probit		<sin ajuste>

Tabla 3. Comparación de modelos alternativos



8.4 *Evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.*

Conocer como se distribuyen los árboles por tamaños es muy relevante para los gestores forestales. Conociendo ésta distribución y el número de pies por hectárea se puede estimar prácticamente cualquier otra variable descriptora del estado del monte. Pero la distribución de tamaños presenta una novedad importante respecto a las variables consideradas con anterioridad. Una función de distribución no es una variable puntual como pueda ser una coordenada o una altura media, una función de distribución es una curva con una forma determinada, formada por "infinitos" puntos.

La forma de proceder es similar a la de los casos anteriores. Primero se obtiene el histograma o la curva de distribución que mejor se ajusta a los datos obtenidos mediante trabajos de campo. Después se realiza lo mismo con los datos de los "árboles ficticios" obtenidos al aplicar el algoritmo que se esté estudiando y posteriormente se comparan las diferencias entre la realidad y la estimación realizada con el algoritmo.

La bondad del algoritmo para predecir los principales descriptores de las funciones de distribución de alturas, se evaluó visualmente empleando diagramas de cajas y bigotes. Estos diagramas agrupan los de forma gráfica 5 parámetros descriptores de la función de distribución de una variable. Empleando estos diagramas se estudiaron las principales diferencias entre las estimaciones y la realidad. En el apéndice A. Se puede encontrar un comentario particularizado para cada una de las parcelas. En este apartado describiremos los resultados generales más relevantes.

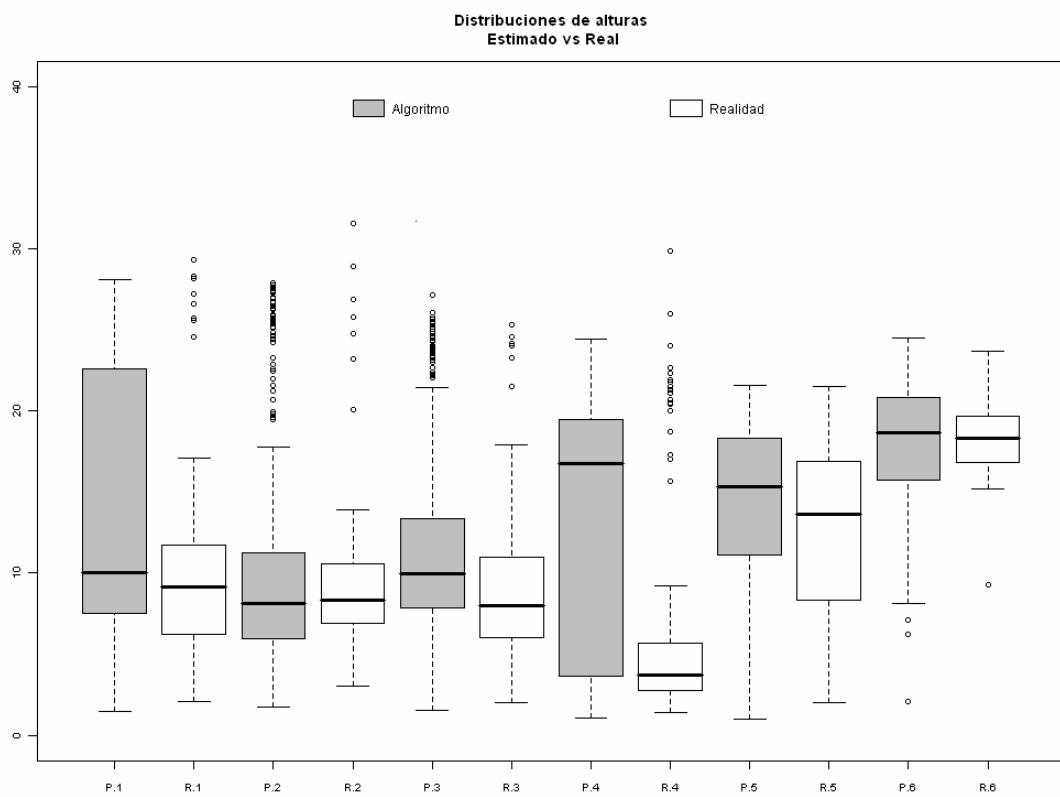


Figura 59 A. Gráficos de Cajas y Bigotes. Parcelas 1-6.

Analizando la figura anterior se observa que para las 6 primeras parcelas nuestro algoritmo sobreestima la altura, mientras que los árboles de alturas bajas no los detecta en su totalidad. En las parcelas 1, 3, 4 y 5 se produce ese mismo comportamiento. Centrándonos en la parcela 1 por ejemplo se observa que el rango de alturas estimado por el algoritmo abarca desde los 7 metros hasta los 23 metros aproximadamente, mientras que en la realidad el rango de altura oscila entre 6 y 12 metros. De modo que nos pasamos por arriba y no llegamos por abajo. Pero en la parcela 1 junto con la parcela 4 son en las que más difieren los rangos de alturas de los árboles estimados con los de la realidad. En la parcela 4 este resultado puede deberse a que las alturas de los árboles son muy bajas entre 3 y 6 metros. Para las parcelas 2 y 6 la estimación es más exacta. Por ejemplo en la parcela 2 y en la parcela 6 se puede observar que nuestro algoritmo estima árboles con entre 1 y 2 metros, de diferencia por arriba y 1 o 2 metros de diferencia por abajo. A pesar de que nuestro algoritmo se comporta de forma muy similar para ambas parcelas los rangos de alturas de los árboles de estas parcelas son muy dispares, la parcela 2 presenta árboles bastante bajos que la parcela 6.



Si analizamos la altura media estimada de los árboles de cada parcela, coincide con bastante exactitud para las parcelas 1, 2 y 6. Difiere en exceso para la parcela 4, por lo que se comentó anteriormente, porque la altura de los árboles es muy pequeña, en las otras dos parcelas la altura media difiere en 2 metros aproximadamente en exceso. Esto es habitual en nuestro algoritmo puesto que se sobreestiman los árboles altos.

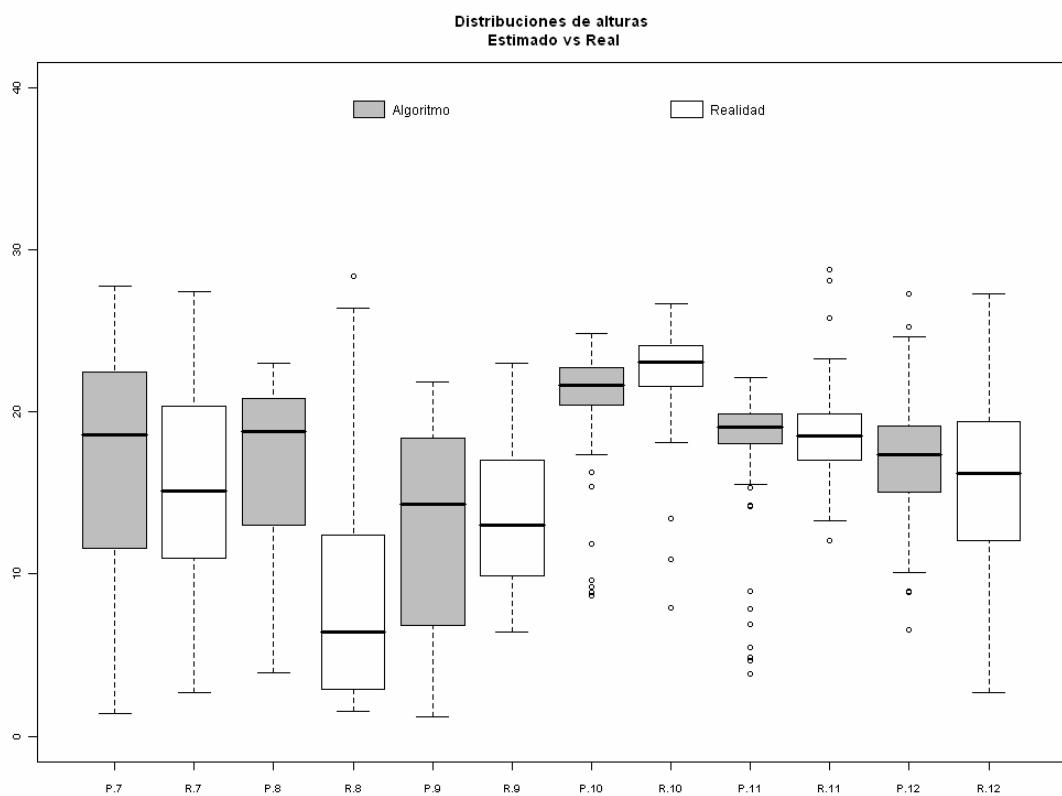


Figura 59 B. Gráficos de Cajas y Bigotes. Parcelas 7-12.

Analizando estas 6 parcelas se observa un comportamiento bastante similar a las 6 anteriores y se puede destacar nuevamente un mal comportamiento para parcelas en las que predominan los árboles bajos si nos centramos en la parcela 8 en la que la altura media de los árboles es aproximadamente de 6 metros y el árbol más alto es de unos 12 metros.

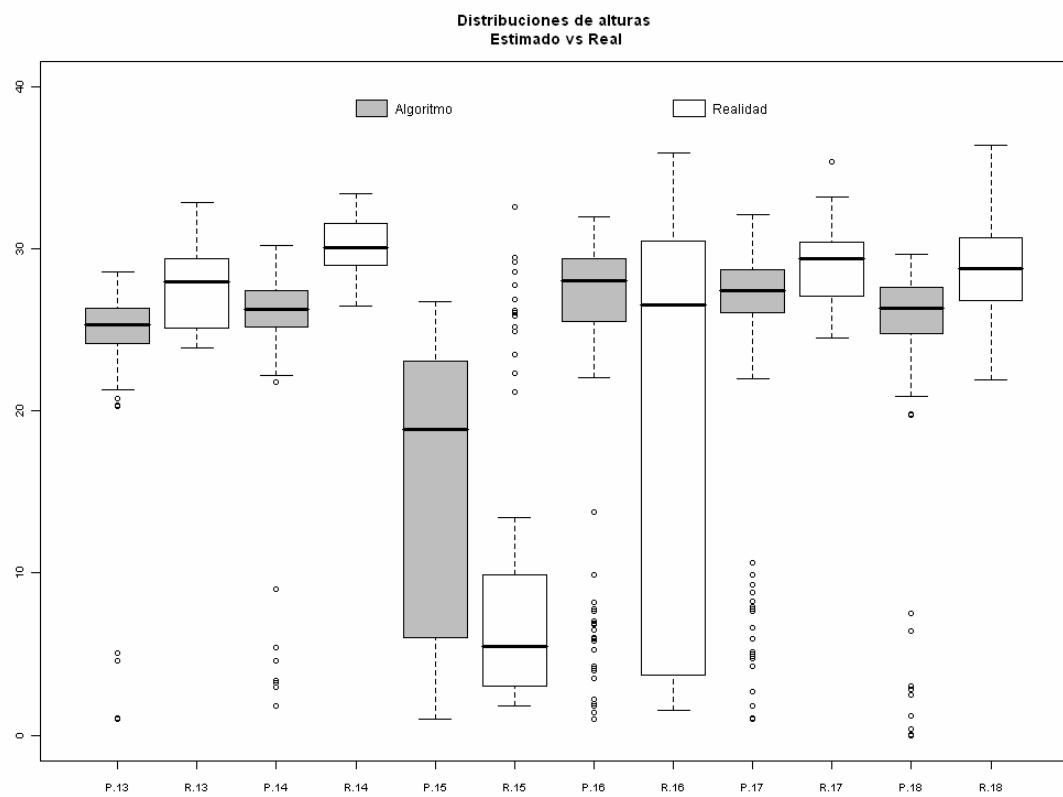


Figura 59 C. Gráficos de Cajas y Bigotes. Parcelas 13-18.

Se aprecia que las alturas de los árboles de estas 6 parcelas son ligeramente mayores que las alturas de los árboles de las parcelas anteriores, a excepción de la parcela 6. En esos casos se observa que no detectamos la totalidad de los árboles altos, esto puede ser por el filtrado que se hizo para evitar encontrar árboles altos en zonas donde no existía realmente ningún árbol, de modo que se tendría que ajustar es filtrado para no eliminar árboles altos que realmente existen. En la parcela 15 predominan los árboles bajos, siendo la altura media de 5 metros aproximadamente, lo que hace que nuestro algoritmo no se comporte muy bien y encuentre árboles en exceso por arriba y no localice los árboles bajos. El rango de alturas de los árboles de la parcela 16 es muy amplio, se pueden encontrar árboles desde 4 metros hasta los 30 metros, en este caso nuestro algoritmo encuentra los árboles altos pero no los árboles más bajos, la altura media estimada en esta parcela difiere en 2 metros en exceso para nuestro algoritmo.

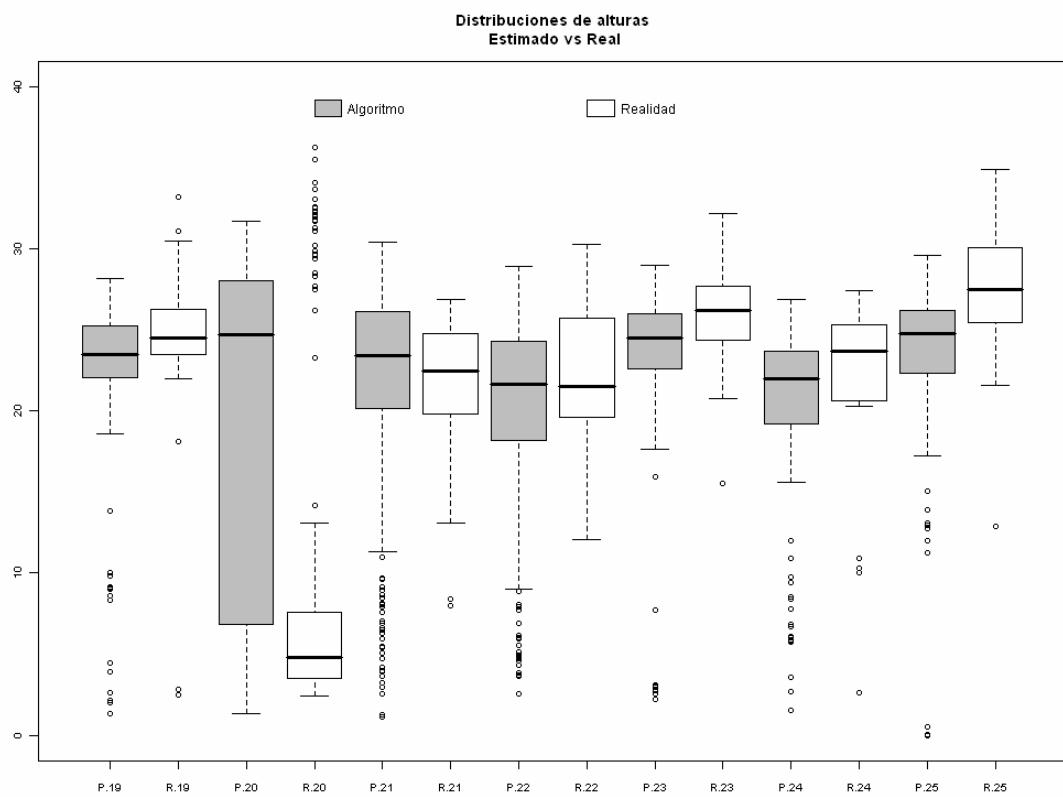


Figura 59 D. Gráficos de Cajas y Bigotes. Parcelas 19-25.

Para estas últimas parcelas el comportamiento es similar a los ya descritos. De modo que como resumen al observar todos los gráficos de cajas y bigotes podemos concluir que en general el algoritmo se comporta mal para las alturas bajas, ya que no localiza la totalidad de los árboles de alturas bajas. Para alturas excesivamente altas superiores a 25 metros, el algoritmo estima menos árboles, probablemente por el filtro que se añadió para eliminar árboles evitando de ese modo contabilizar árboles altos que no existían realmente sino que se confundía con los árboles que no crecen rectos. En el resto de casos el algoritmo generalmente sobreestima árboles altos y no contabiliza los árboles de alturas menores.

Para contrastar estadísticamente cómo de diferente es el histograma o la función de densidad obtenida con los datos de campo del histograma o la función de densidad obtenida con el output del algoritmo, se emplean contrastes de homogeneidad. Estos contrastes se basan en analizar cómo se distribuyen las diferencias entre histogramas o como están de "separadas" las funciones de distribución. Contrastos de este tipo son el contraste de la χ^2 o el test de Kolmogorov-Smirnov. En las siguientes gráficas se puede observar las funciones de densidad ajustadas así como el p-value obtenido en el



test de Kolmogorov-Smirnov. Es importante darse cuenta que las funciones de distribución para este paso se calcularon empleando un kernel gaussiano y el ancho de banda fue seleccionado empleando la regla de Silverman (1986). En el anexo A se pueden observar las mismas funciones de densidad pero el parámetro de suavizado por defecto del programa empleado para el análisis estadístico sobresuavizaba las estimaciones de las funciones de densidad, haciendo prácticamente imposible detectar máximos en la función de densidad debidos a la presencias de diferentes estratos de edad en el arbolado. Considerando este hecho se decidió calcular estas mismas funciones de densidad calculando el parámetro de suavizado empleando el criterio mencionado.

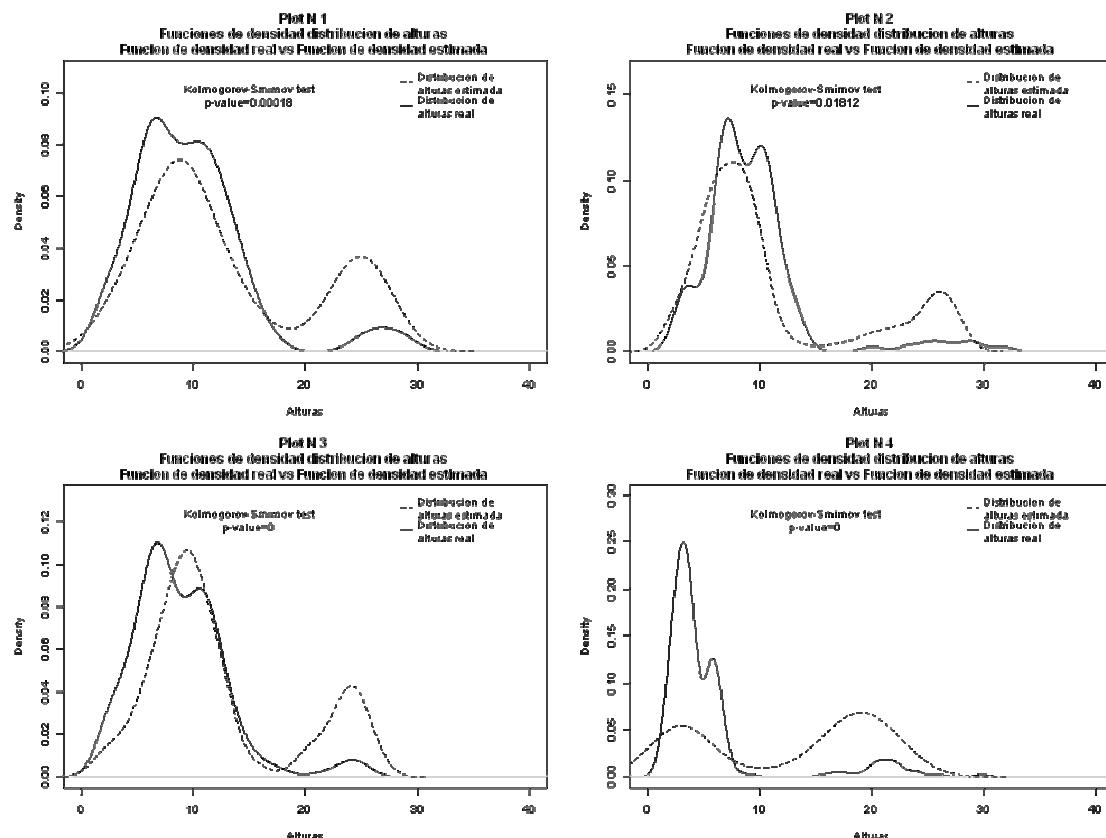


Figura 60 A. Funciones de densidad suavizada. Parcelas 1-4.

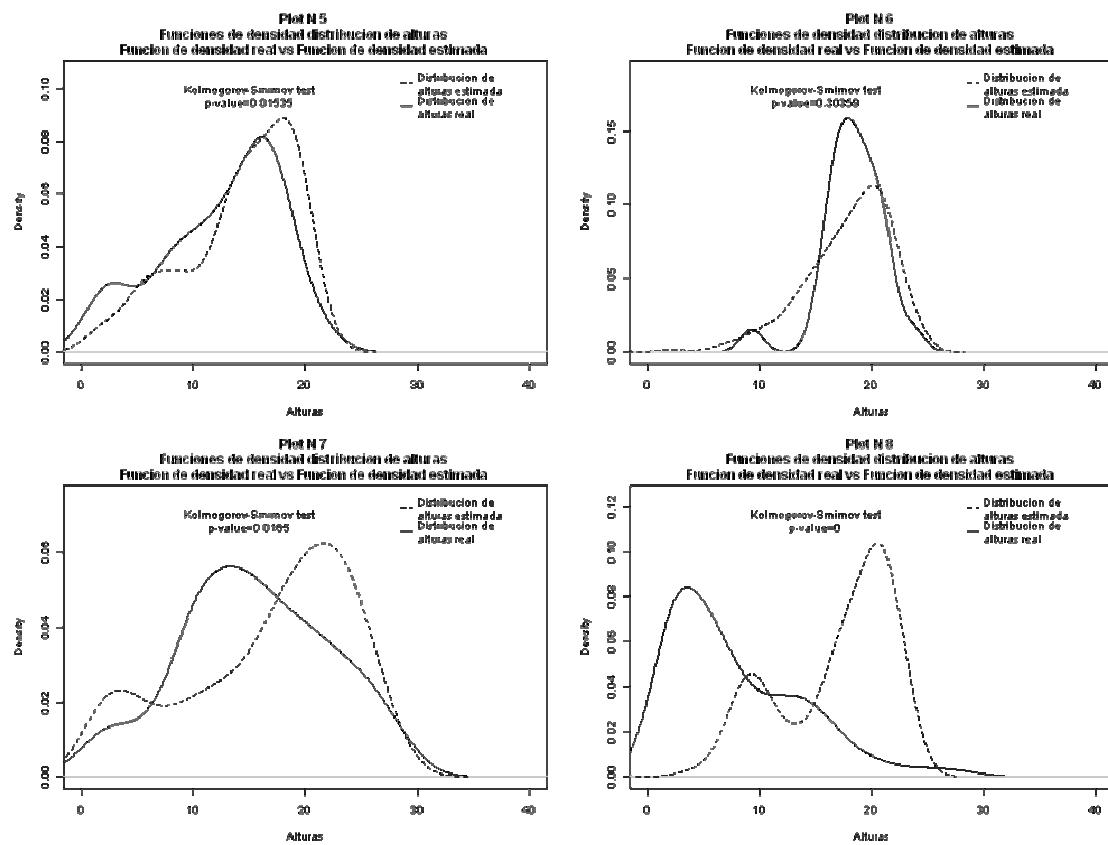


Figura 60 B. Funciones de densidad suavizada. Parcelas 5-8.

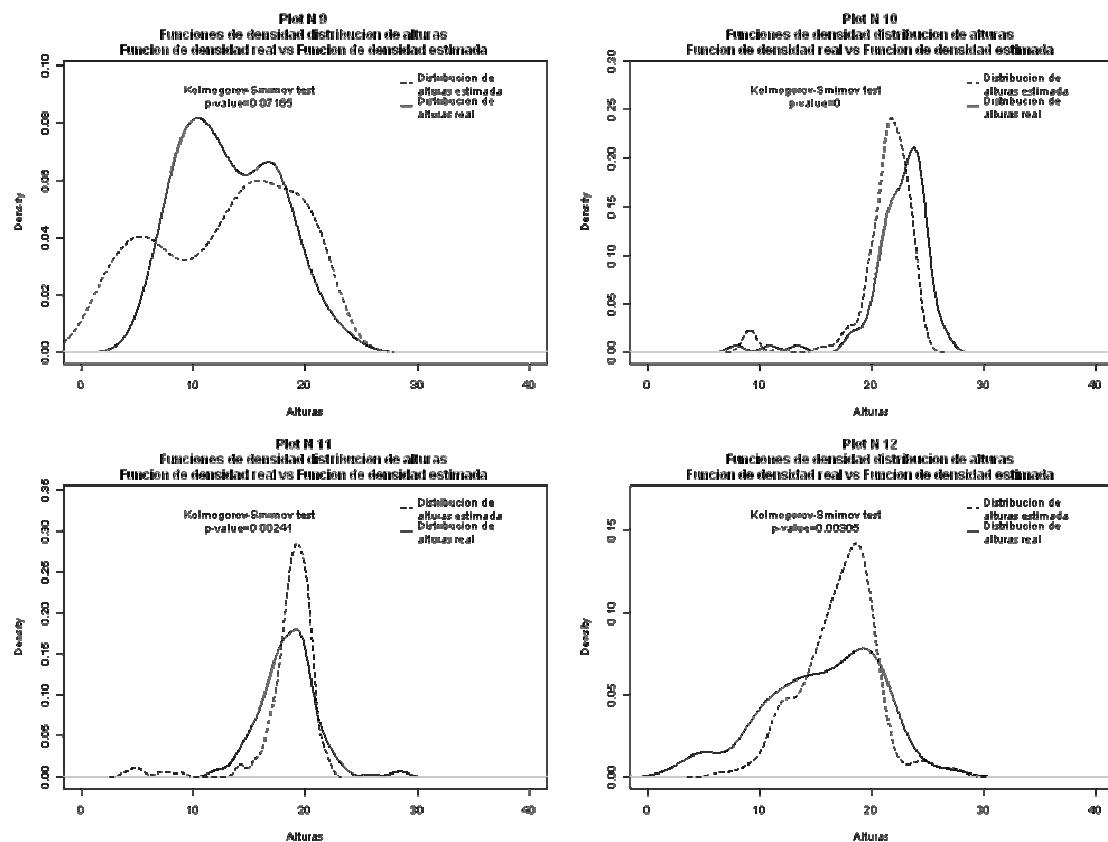


Figura 60 C. Funciones de densidad suavizada. Parcelas 9-12.

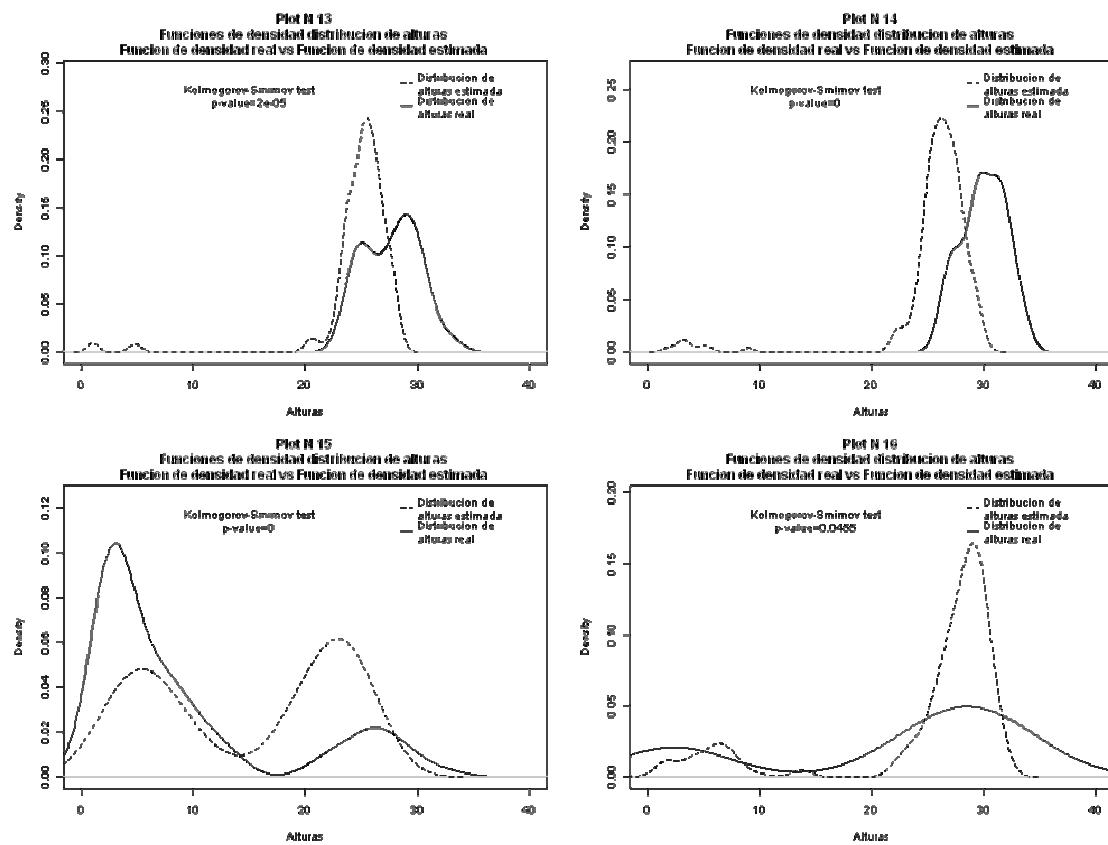


Figura 60 D. Funciones de densidad suavizada. Parcelas 13-16.

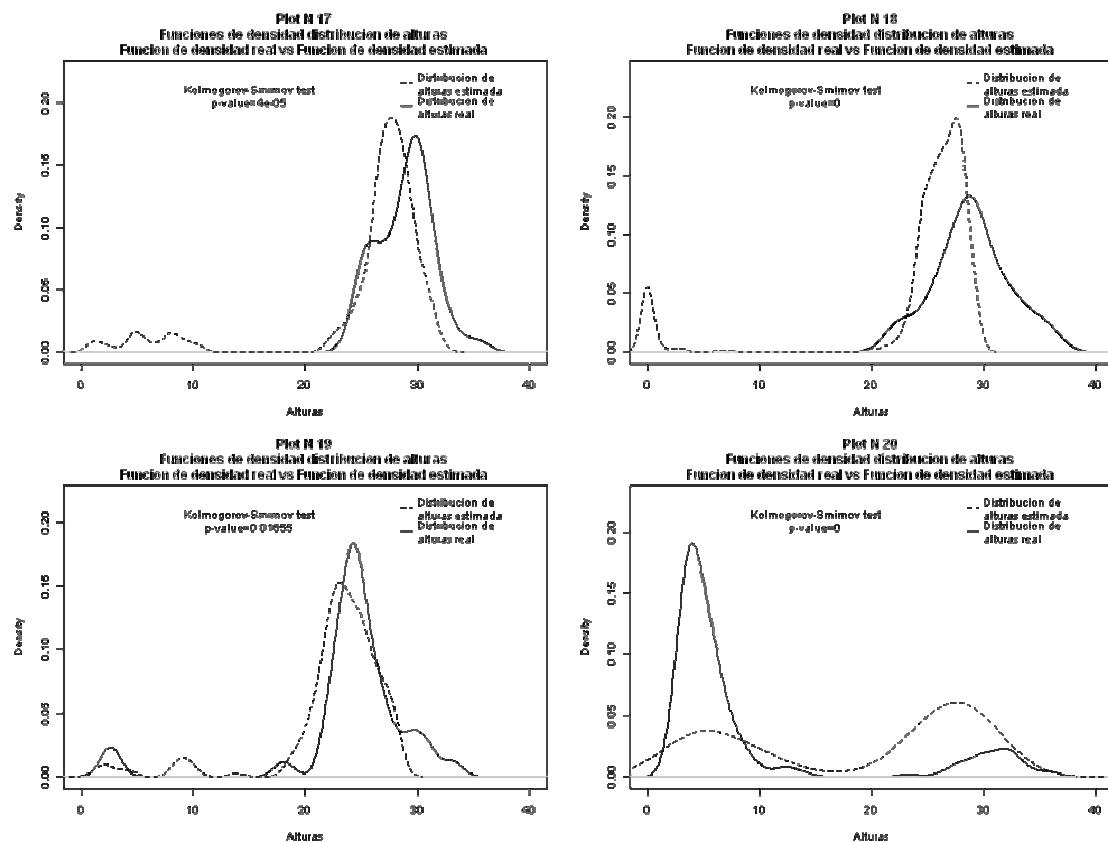


Figura 60 E. Funciones de densidad suavizada. Parcelas 17-20.

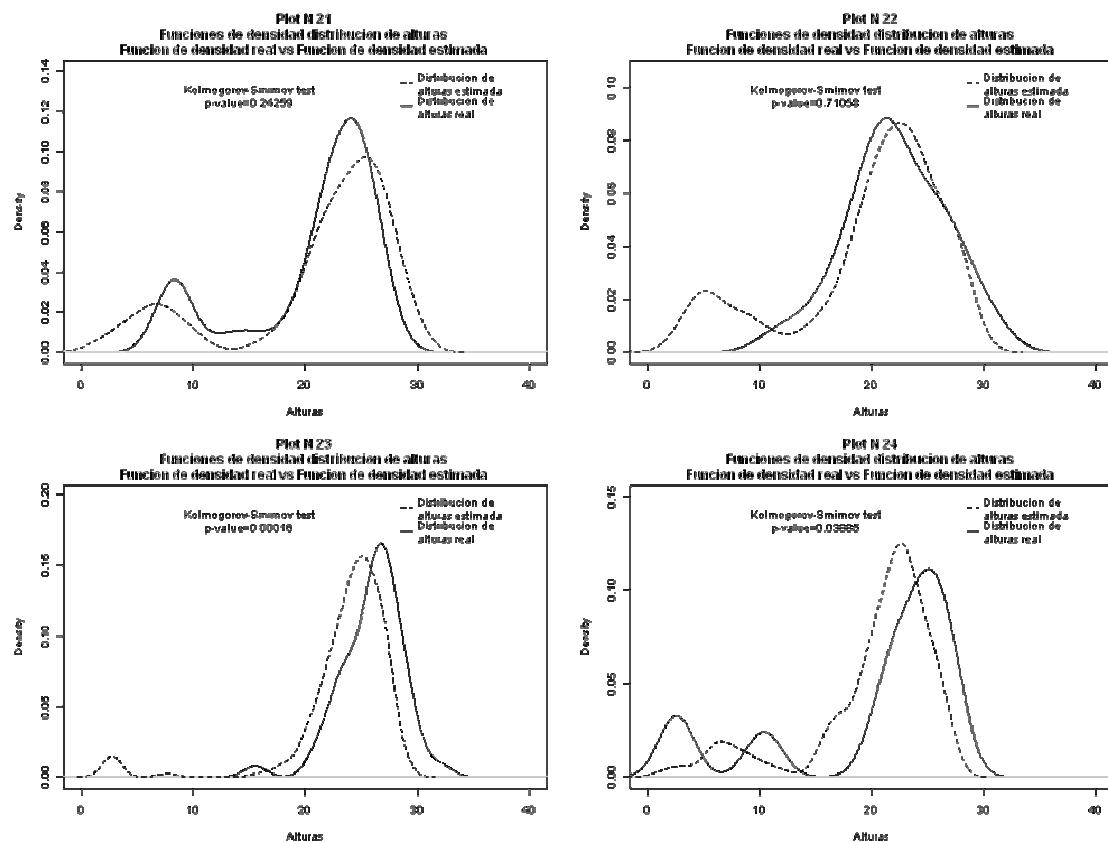


Figura 60 F. Funciones de densidad suavizada. Parcelas 21-24.

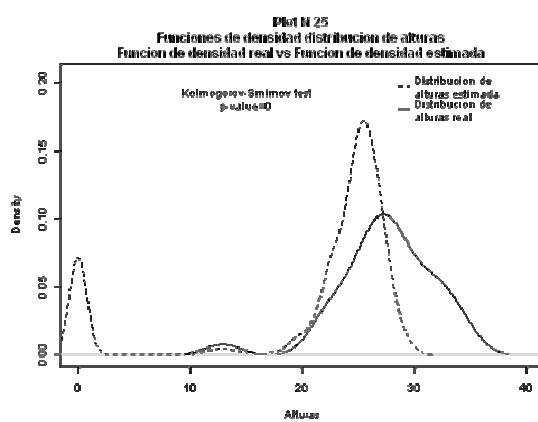


Figura 60 G. Funciones de densidad suavizada. Parcela25.

Las conclusiones que se obtienen al observar este grupo de gráficas y los correspondientes valores de los test es que si bien, hay diferencias estadísticamente significativas entre la curva



de densidad real y la curva de densidad estimada en la mayoría de los casos, los valores observados para los principales descriptores de las funciones de densidad (mediana y quartiles 1 y 3) son en general válidos. Los mayores diferencias se observan en aquellas parcelas donde hay dos estratos de árboles de distintas edades, concretando se puede apreciar en las parcelas 4, 15, 20 y 24. Los árboles grandes no permiten que lleguen pulsos suficientes a los estratos inferiores con lo que en general hay una estimación a la baja de la función de densidad de alturas para los tamaños menores y una sobre estimación de la misma para los valores de altura elevados. Un artefacto que se detectó durante este proceso es que puntos de la nube de puntos que caen muy cerca del suelo provocan que el algoritmo anote como árboles puntos que deben ser matorrales o rocas. Resolver este problema no es difícil dado que simplemente filtrando esos resultados erróneos por la altura, el problema desaparece. Para filtrar estos árboles erróneos se eliminaron en este análisis todos los puntos con altura menor de 1.3 metros que es la altura del árbol más pequeño que se puede encontrar en el campo.

Para ver la validación completa de las 25 parcelas consultar el anexo B.

8.5 Evaluación de la precisión estimando las funciones de distribución de tamaños de arbolado.

Finalmente los resultados obtenidos al predecir las tres variables estudiadas en la sección 8.3 directamente del algoritmo sin ningún tipo de calibración se compararon con los resultados proporcionados por otros algoritmos.(Algoritmo de Hyyppä and algoritmo watershed). Las siguientes gráficas muestran las diferencias respecto al valor real de las estimaciones realizadas en cada parcela.

En los gráficos 61 A, 61 B y 61 C se muestra la comparación de los algoritmos de Hyyppä y Watershed con el nuestro y las mediciones de campo para la altura media del arbolado, número de pies por hectárea y biomasa por hectárea. La gráfica de la parte de arriba muestra el error y la magnitud real de la medida, mientras que en la de abajo se muestra el error absoluto.

Para la altura media del arbolado se observa a simple vista que el cálculo de la altura media es bastante acertada, estimando la mayoría de veces en exceso, salvo alguna excepción en la que los árboles son excesivamente altos y se deben de eliminar en exceso por la razón que se comentó previamente (el filtrado para evitar contar árboles que no crecen rectos varias veces). Para alguna parcela árbol la diferencia de altura es muy alta, aunque para la mayoría la diferencia es pequeña.

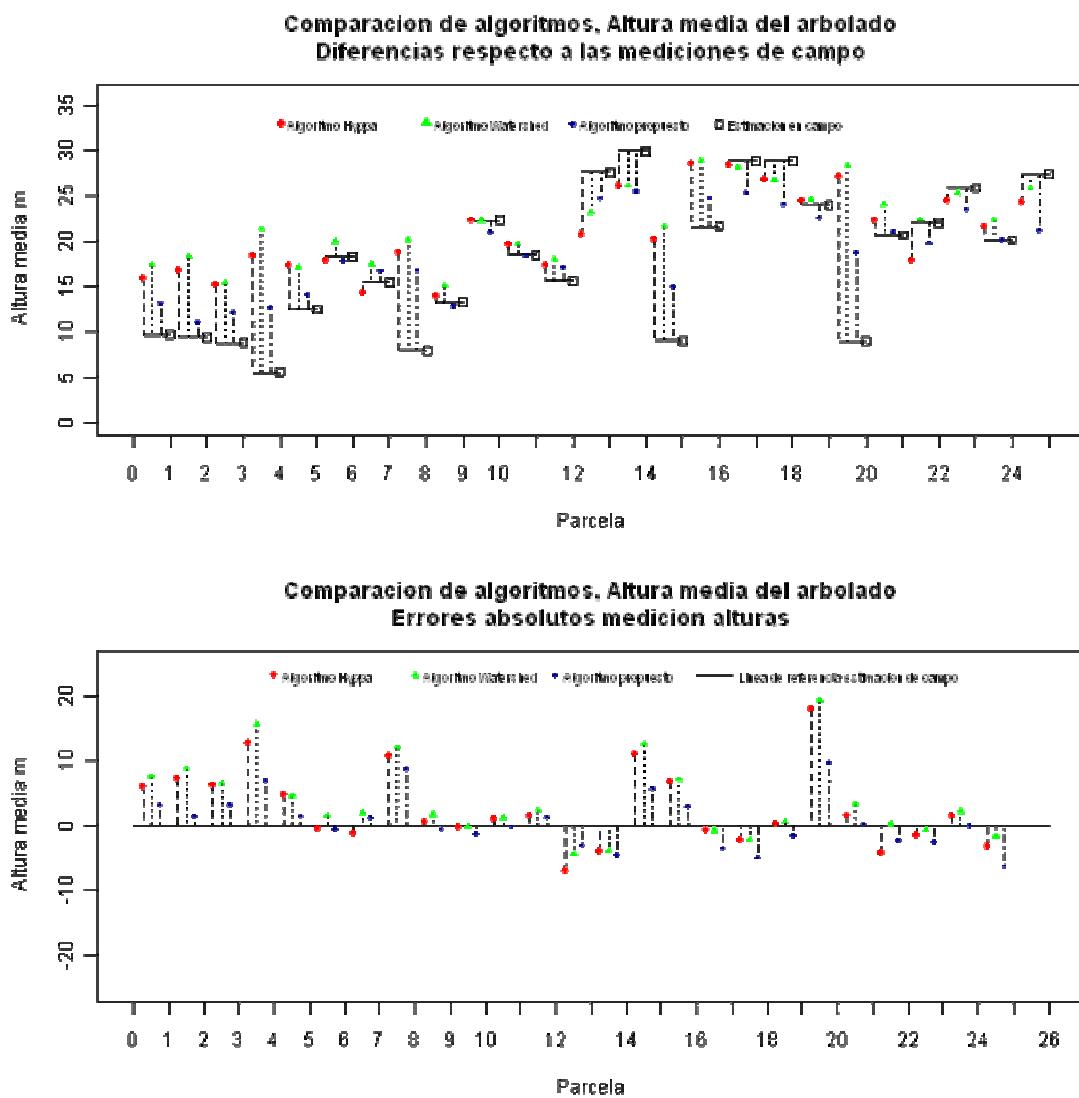


Figura 61 A. Comparación Algoritmos Altura media del arbolado. Parcelas 1-25.

En las gráficas 61 B se observa que el número de pies por hectárea es muy malo, se obtienen errores excesivamente grandes, en general se aprecia una sobreestimación en el número de pies.

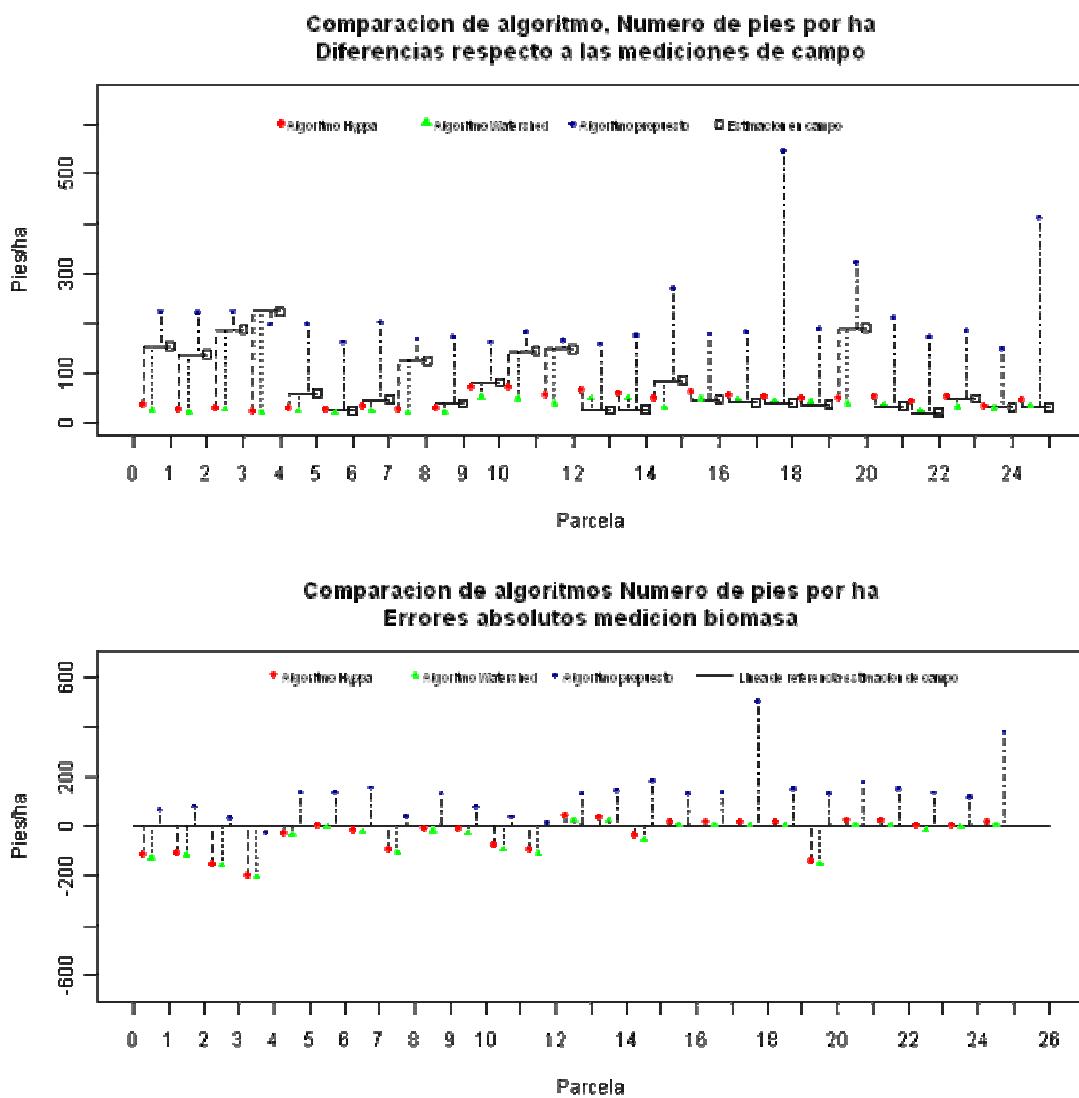


Figura 61 B. Comparación Algoritmos Número de Pies por ha. Parcelas 1-25.

En las gráficas 61 C se compara la biomasa por hectárea, se obtienen bastantes errores, puesto que en el cálculo de pies por hectárea el resultado mostraba errores altos, y al no calibrar el modelo se obtienen bastantes resultados malos como se puede observar para la mayoría de parcelas.

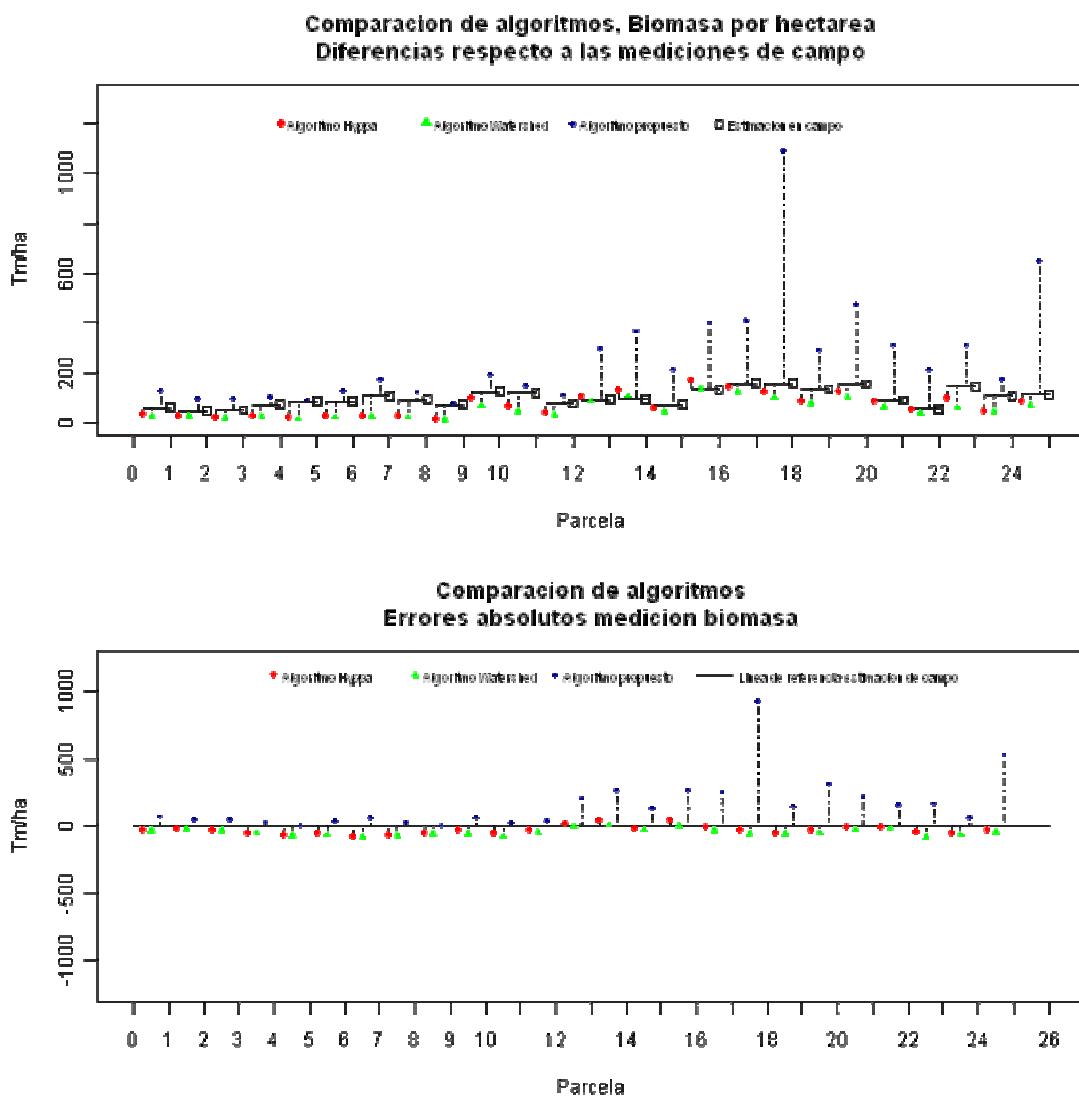


Figura 61 C. Comparación Algoritmos Biomasa por hectárea. Parcelas 1-25.

Por último se comparó como de bien estimaban cada uno de los citados algoritmos los principales descriptores de las funciones de densidad de alturas del arbolado. Las siguientes gráficas muestran de forma conjunta los gráficos de cajas y bigotes para los datos de campo así como para las estimaciones realizadas con los distintos algoritmos. Fundamentalmente se observa que el algoritmo propuesto proporciona mejores estimaciones del tamaño medio de altura. Proporciona mejores estimaciones en el rango intercuartílico, es decir que los rangos de árboles estimados por nuestro algoritmo son mejores que los de Hyyppä y Watershed. Aunque existen excepciones debido a la pérdida de árboles bajos tapados por árboles más altos, lo que nos impide



detectarlos, a pesar de ello nuestro algoritmo se comporta mejor que los otros dos en líneas generales, puesto que trabajamos con la nube de puntos y esto nos permite detectar algunos de los árboles bajos aunque no sean todos. Mientras que los métodos que trabajan con datos rasterizados pierden todos los árboles bajos que quedan ocultos bajo los árboles más altos como se explicó en la sección 2.2.1

A continuación se muestran los gráficos 62 A, 62 B y 62 C en los que se puede ver detalladamente los resultados. En la parte de debajo de la gráfica se muestra aparece un número con el siguiente formato X.Y, donde X representa el número del método e Y representa el número de parcela.

Haciendo un análisis más detallado de los resultados obtenidos en la figura 62 A se observa que para todas las parcelas analizadas (1-6) el algoritmo que mejor se comporta es el descrito en este trabajo, tanto para el cálculo de la altura media como para el cálculo de los rangos de alturas.

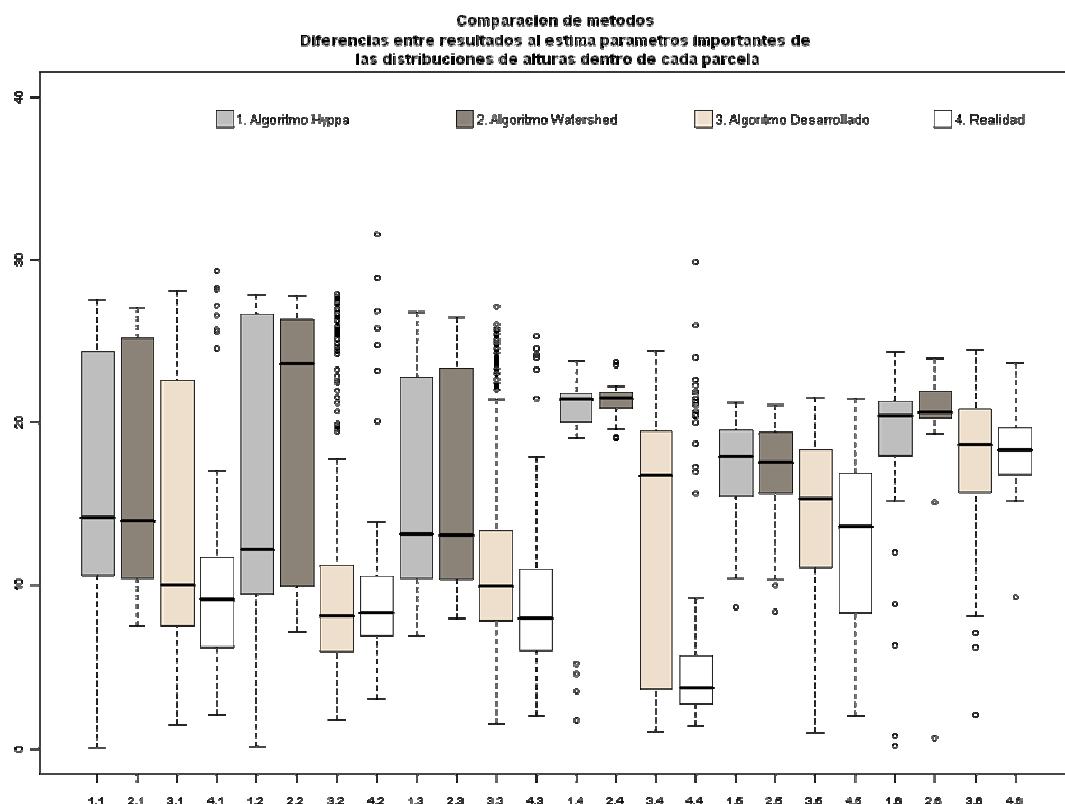


Figura 62 A. Comparación de métodos. Distribución de altura dentro de cada parcela.

Parcelas 1-6.



Analizando las parcelas de la 7 a la 12 en la figura 62 B se observa que en general es también nuestro algoritmo el que mejor se comporta para todas las parcelas a excepción de la parcela 10, para la cual se comporta mejor Hyppä.

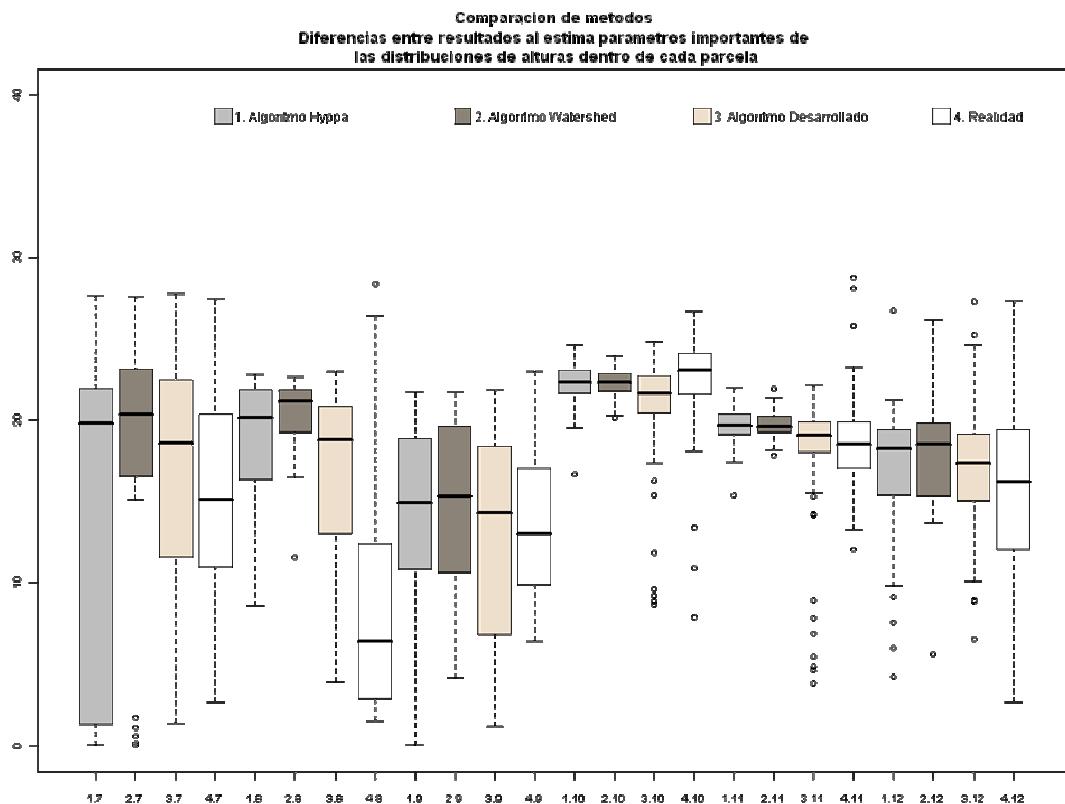


Figura 62 B. Comparación de métodos. Distribución de altura dentro de cada parcela.
Parcelas 1-12.

Fijándonos en la figura 62 C, se observa que hay varias parcelas para las que nuestro algoritmo no es el que mejor se comporta, concretamente para las parcelas 14, 17 y 18, en las tres parcelas coincide que el rango de alturas de árbol es muy grande, con lo cual es posible que como se explicó anteriormente el filtro de borrado de árboles altos que cumplen unas determinadas condiciones, con el fin de no contabilizar árboles altos que no crecen rectos varias veces, no esté bien ajustado y borre más árboles de los deseados.

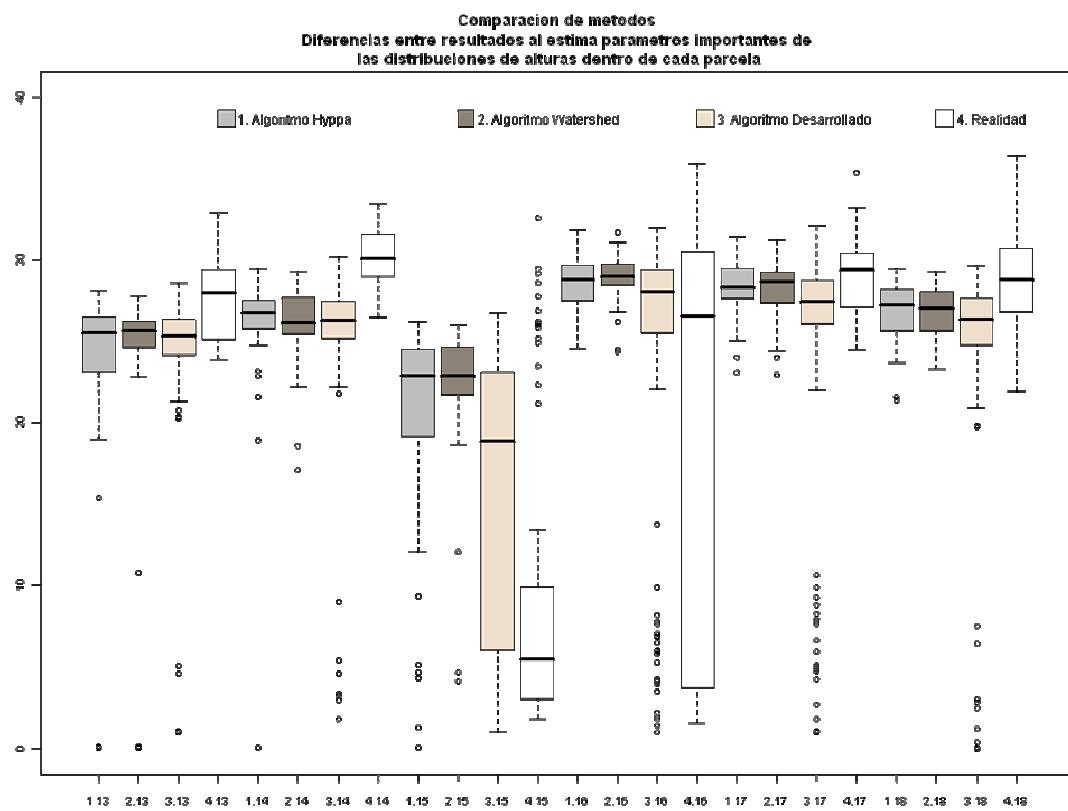


Figura 62 C. Comparación de métodos. Distribución de altura dentro de cada parcela.
Parcelas 13-18.

Por último si analizamos la figura 62 D se observa que para alguna parcelas con predominio de árboles muy altos nuestro algoritmo no es el que mejor se comporta, por ejemplo para las parcelas 19, 23 y 25. Sin embargo nuestro algoritmo se comporta mejor para el resto de parcelas.

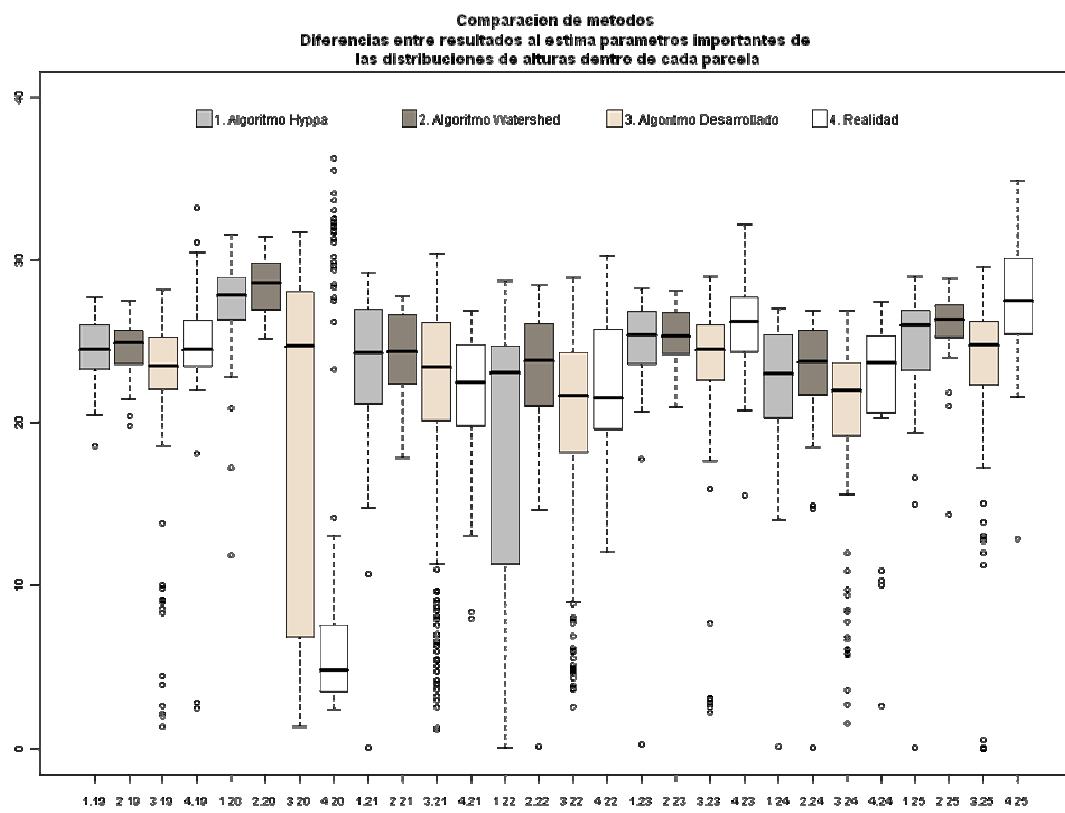


Figura 62 D. Comparación de métodos. Distribución de altura dentro de cada parcela.
Parcelas 19-25.



9 Conclusiones

De los resultados obtenidos por el algoritmo se podría concluir lo siguiente:

- Se obtiene una sobreestimación del número de pies por hectárea. Esto se debe a que la fusión de véxeles no está funcionando como es debido, sería necesario realizar un ajuste de estos parámetros. Además toda predicción de variables cuyo valor se derive de la suma de la aportación de cada uno de los distintos árboles (La biomasa por hectárea es una de estas variables) se verán seriamente afectadas por este error. Produciendo estimaciones completamente erróneas debido al ruido introducido al sobre estimar el número de pies. Las estimaciones directas del número de pies introducen incertidumbre y su uso está completamente desaconsejado. Por otro lado la calibración con los datos de campo tampoco mejora los resultados.
- La altura media oscilan la mayoría entre $\pm 5m$. En general se observa una ligera mayoría de estimaciones a la alta. La altura media no es muy sensible al problema general de la sobre-estimación del número de pies mostrada por este algoritmo, siempre y cuando las proporciones de árboles detectados por encima y debajo de la media medida en campo aparezcan en las mismas proporciones con las que estos aparecen en la realidad. El algoritmo que aquí se presenta ha mostrado resultados satisfactorios estimando los parámetros más importantes de las funciones de distribución de los árboles y esto implica estimar de forma aproximada las proporciones de pies por encima y debajo de la media real.
- La biomasa se ve afectada por la sobreestimación del número de pies, de modo que estará notablemente sesgadas debido al efecto de esta sobre-estimación. La biomasa de cada árbol está relacionada con el tamaño de este. El algoritmo que se presenta, produce estimaciones correctas del tamaño del arbolado. Así tenemos que los dos principales factores que influyen en el cálculo final de la biomasa por hectárea, son calculados de forma desigual por el algoritmo. El tamaño medio del arbolado es estimado con precisión por el algoritmo, sin embargo, para la cantidad de pies por hectárea el algoritmo no proporciona ninguna mejora. Este parámetro mejorará ajustando el algoritmo para mejorar el número de pies por hectárea.



En este apartado se presentan las aportaciones de este trabajo (apartado 9.1) y las futuras líneas de trabajo (apartado 9.2).

9.1 Aportaciones

Dados los abordajes previos hechos por Hyppä [10], Morsdorf [14], Wang [22] para solucionar el problema de estimación de la masa forestal, en este trabajo se introducen las aportaciones que describen los siguientes párrafos.

Primero se describe un método que proporciona una aportación importante en cuanto a la fusión de árboles. Esta fusión se hace no sólo por proximidad entre los árboles de vox con puntos sino que en cierta manera también se estima la forma del árbol aunque no existan todos los puntos y con todos esos vox (los que tienen puntos y los que no) se realiza la fusión entre dos árboles. Esto es muy necesario. En las primeras versiones del algoritmo se hicieron pruebas con las parcelas de entrenamiento sin realizar esta fusión y el número de árboles era bastante mayor. Además la fusión entre dos árboles cuando se localiza un vox conflictivo con un número de puntos alto es también importante. A pesar de que los resultados obtenidos actualmente no son los deseados se pueden mejorar fácilmente haciendo pruebas y ajustando estos parámetros.

Otra aportación es la consideración que se hace para árboles que crecen inclinados. Esto es muy importante para la zona de estudio en la que se han realizado las pruebas y como ya se explicó hay árboles que crecen inclinados se añade otra novedad que es establecer una limitación a la hora de asignar un árbol ficticio. La limitación que se añade es que si un árbol sólo tiene un vox o tiene menos de tres puntos y la altura es mayor que 10 no lo consideramos árbol y se borra de la lista de árboles ficticios. Antes de realizar esta implementación nos dimos cuenta que obteníamos gran cantidad de árboles en zonas en las que realmente no había árboles y lo que había eran huecos. Observamos esas zonas con detenimiento y observamos que a pesar de no haber árboles había puntos LiDAR en capas altas. De modo que introdujimos esta nueva contemplación, mejorando considerablemente este comportamiento. Aunque habría que mejorar o mejorar dicho precisar más estas operaciones viendo los comportamientos obtenidos de las pruebas, ya que esta parte aún puede mejorar mucho.

Otra de las principales aportaciones de este trabajo es la del uso de lógica fuzzy para discernir a qué árbol pertenece cada punto de un vox. De esta forma dejamos de tratar



siempre la información de vox y pasamos a tratarla con puntos para afinar en los tamaños de las copas.

Otra aportación es el proporcionar un método implementado bajo una herramienta GIS open source, para que cualquiera lo pueda usar y realizar nuevas aportaciones y mejoras.

9.2 Futuras líneas de trabajo

Durante el desarrollo del siguiente trabajo y una vez obtenidos los resultados del mismo han surgido las siguientes líneas de trabajo.

La correcta selección de los contornos es uno de los problemas que pueden contribuir a la calidad de los resultados. Por lo que se propone trabajar en ellos.

Otro punto importante es el crecimiento del árbol tanto a lo ancho como hacia abajo. Ambos crecimientos son importantes para los resultados e influyen significativamente en los resultados. Por ello es otro de los puntos importantes para trabajar. Se podría mejorar considerando el crecimiento de un árbol como un elipsoide para árboles del tipo que estamos estudiando en nuestras parcelas, tal y como hacen Riaño en su artículo. Esta línea de trabajo puede ser interesante y dar sus frutos, porque a pesar que el crecimiento y formas de los árboles es muy dispar, en muchos casos se podría encontrar un patrón.

Otra línea de investigación quizás la más inmediata es ir ajustando más algunos valores que se emplean en el algoritmo. Como se puede apreciar de forma más detallada en el capítulo 8 de este trabajo la estimación de la altura media es bastante buena, pero el algoritmo falla considerablemente en el cálculo del número de pies por hectárea y en el cálculo de biomasa. Estos resultados se pueden mejorar observando y analizando los resultados para cada una de las parcelas y estudiando para cada una de ellas la distribución de los árboles. Cuando se realizaron las pruebas de entrenamiento con las parcelas entrenamiento 1 y parcelas entrenamiento 6, los ajustes de parámetros para fusionar, asignar puntos a un árbol u a otro, o la decisión a la hora de eliminar un árbol alto, afectaba de forma muy distinta a una y otra parcela. Esto se debe a las diferencias de distribución de árboles dentro de ellas (una presenta árboles muy juntos), el rango de alturas de los árboles. En una de ellas se encuentran árboles de todas las alturas, mientras que en la otra los árboles son prácticamente de la misma edad con lo que tienen alturas muy similares. Como las parcelas difieren tanto unas de otras, hay que



encontrar un equilibrio que no beneficie a unas y perjudique a otras, para ello es necesario entrenar con muchas parcelas e ir ajustando. Todo ello requiere mucho tiempo, ya que la ejecución del algoritmo es larga, aunque varía en función del número de puntos de la parcela pero podríamos hablar de varias horas, de modo que probar un pequeño ajuste puede llevar muchas horas de trabajo.

Una vez que ya se han analizado todos los resultados y se observa que los fallos se producen principalmente para árboles de alturas pequeñas, con lo cual habría que trabajar de algún modo en ello para resolver este problema. También el algoritmo estima en algunas ocasiones árboles altos que no existen o por el contrario no contabiliza árboles altos que sí que existen. Esto puede ser por los árboles que crecen torcidos por ejemplo. Esto ya se está contemplando pero sería interesante ajustar mejor los valores.

Otra interesante línea de investigación sería usar utilizar y comparar nuevos métodos, hibridando métodos clásicos, como realiza María Guijarro [8] en su tesis.







10 Anexo A: Resultados obtenidos con Statgraphics Comparación de dos muestras- Alturas y ZMax.

A continuación se muestra un análisis realizado para cada una de las 25 parcelas con Statgraphics contrastando los datos de campo con los datos obtenidos con el algoritmo. En el análisis se muestra un resumen estadístico, histograma, función de densidad y un gráfico de cajas y bigotes. Cabe destacar que para las funciones de densidad se ha usado el parámetro de suavizado por defecto de Statgraphics, este parámetro produce un sobre suavizado, de modo que se realizaron nuevamente estas funciones empleando un kernel gaussiano y el ancho de banda fue seleccionado empleando la regla de Silverman (1986), que se muestra en la sección 8.4. Para poder observar mejor el comportamiento del algoritmo es deseable fijarse en las funciones de suavizado de ficha sección 8.4.

10.1 Parcela Volante 1

Resumen Estadístico

	Alturas	ZMax
Frecuencia	154	224
Media	9,72078	13,1802
Varianza	28,5908	58,9837
Desviación típica	5,34704	7,68008
Mínimo	2,1	1,48603
Máximo	29,3	28,1397
Rango	27,2	26,6537
Asimetría tipi.	8,11597	4,01615
Curtosis típificada	9,01471	-3,08563

Tabla 4. Resumen Estadístico Parcela 1.



Nota: Como se puede observar los resultados obtenidos con el algoritmo nos dan una cantidad de árboles algo mayor que con los resultados obtenidos de campo. En este resumen estadístico se observa que la media que estimamos con el algoritmo se sobreestima 3,5 metros aproximadamente, ya que sobreestimábamos en exceso árboles altos entre 22 y 26 metros.

Histograma

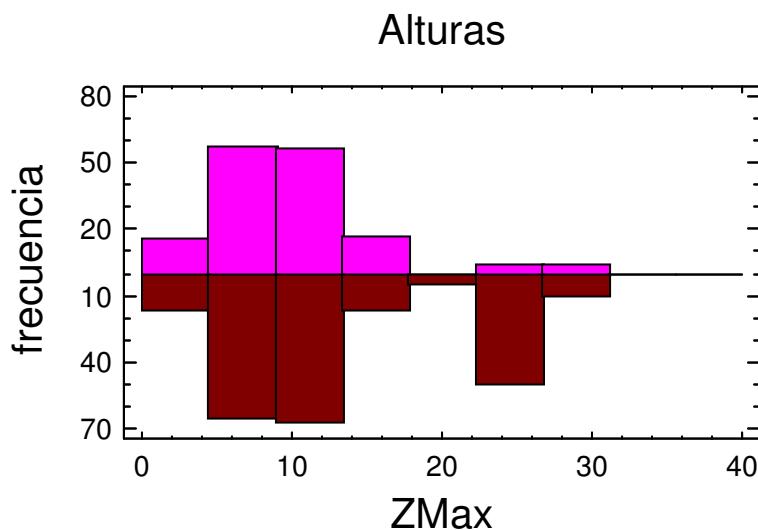


Figura 63 A. Histograma Parcela 1.

Nota: Al obtenerse mayor cantidad de árboles, se refleja también en el histograma que aparecen más árboles en la parte de abajo del mismo, es decir en los que obtenemos con el algoritmo. Estos árboles en exceso que se obtienen oscilan entre 22 y 26 metros. El número de árboles obtenidos entre 6 y 12 metros también es ligeramente mayor.



Densidades Suavizadas

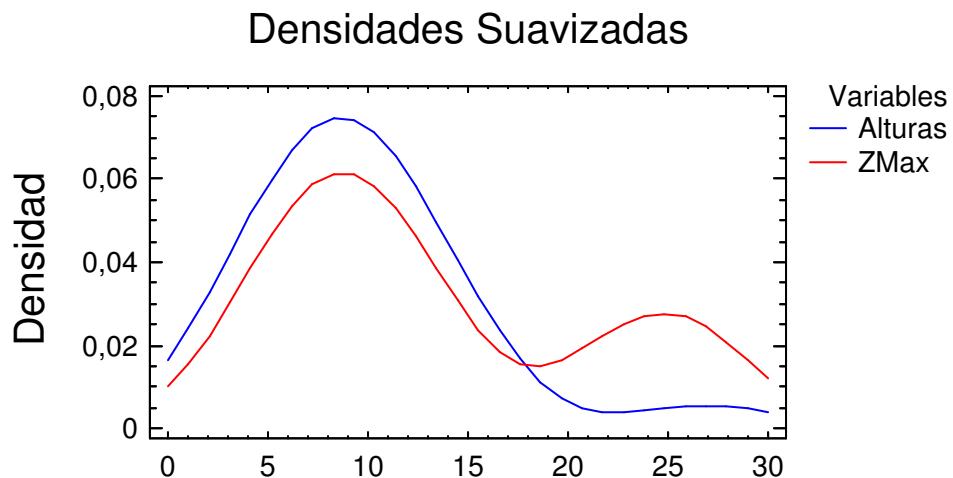


Figura 63 B. Densidades Suavizadas. Parcela 1.

Con las densidades suavizadas vamos a conocer como se distribuyen los tamaños de los árboles dentro de la parcela. Se pierde la información de la frecuencia absoluta pero se obtiene una gráfica mas suavizada.

Nota: Se observa que los resultados son diferentes para árboles altos, que en los datos obtenidos de campo apenas hay, mientras que en los resultados obtenidos por el algoritmo se muestran algunos.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

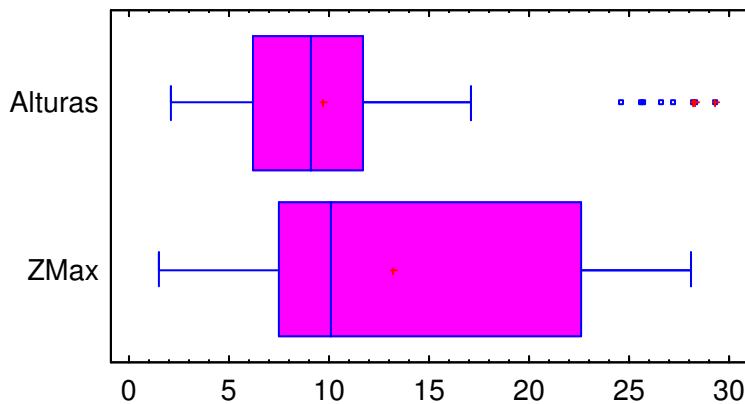


Figura 63 C. Gráfico de Cajas y Bigotes. Parcela 1.

Nota: Las Cajas delimitan la mitad de la muestra, dejando fuera el 25% más pequeño y el 25% más alto. La línea vertical es el valor normal en una población. En los datos de campo se obtiene un valor de 8 metros como altura media, mientras que en los datos del algoritmo se obtiene un valor de 10 metros. Se observa que los árboles por el algoritmo son más grandes que los obtenidos en campo, con una diferencia bastante alta.



10.2 Parcela Volante 2

Resumen Estadístico

	Alturas	ZMax
Frecuencia	138	221
Media	9,4029	11,0098
Varianza	24,7937	57,53
Desviación típica	4,97933	7,58486
Mínimo	3,0	1,76538
Máximo	31,6	27,9363
Rango	28,6	26,1709
Asimetría tipi.	11,4082	7,03824
Curtosis típificada	17,2317	-0,356755

Tabla 5. Resumen Estadístico Parcela 2.

Nota: Como se puede observar los resultados obtenidos con el algoritmo nos dan una cantidad de árboles algo mayor que con los resultados obtenidos de campo. En este resumen estadístico se observa que la media que estimamos con el algoritmo se sobreestima 1,5 metros aproximadamente.



Histograma

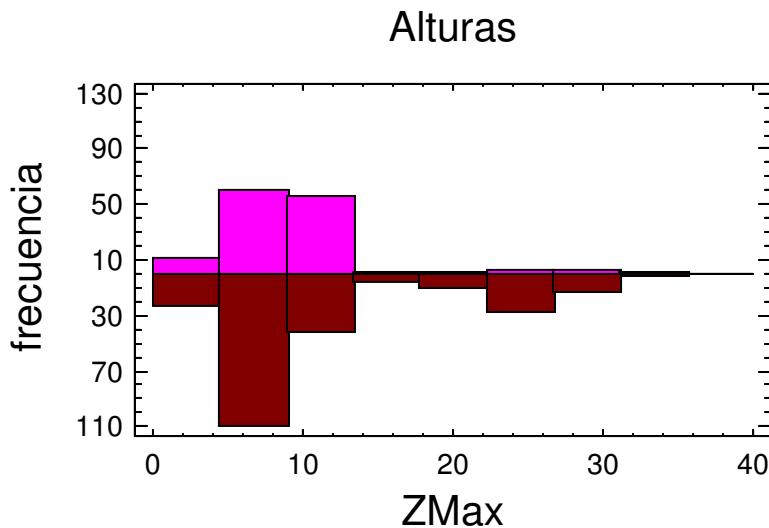


Figura 64 A. Histograma. Parcela 2.

Nota: El histograma muestra una distribución bastante coincidente para alturas bajas entre 0 y 5 metros, aunque nuestro algoritmo localiza árboles en exceso, para árboles entre 5 y 9 metros el algoritmo localiza muchos más árboles de los existentes en la realidad. Para alturas entre 9 y 13 metros el algoritmo estima menos árboles de los que existen en realidad y para alturas mayores de 13 en general el algoritmo localiza árboles en exceso.



Densidades Suavizadas

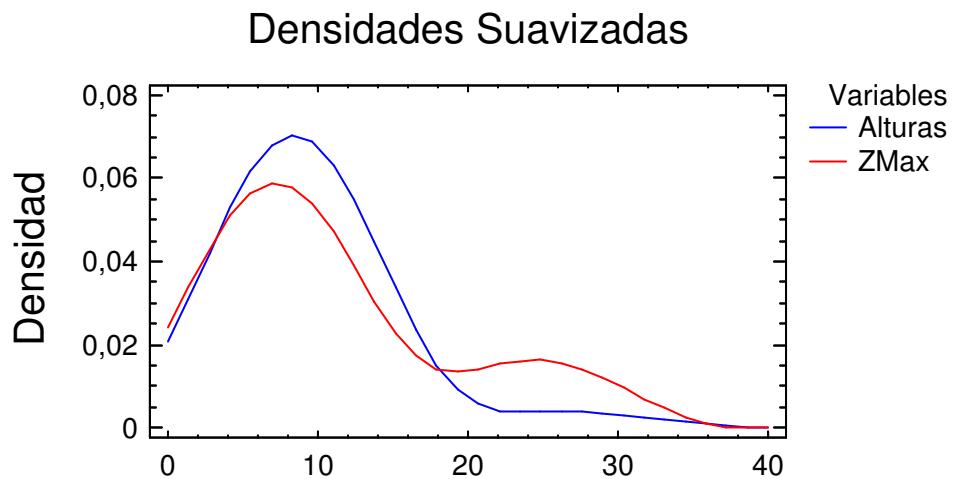


Figura 64 B. Densidades Suavizadas. Parcela 2.

Nota: Se observa que para alturas bajas nuestro algoritmo muestra menos árboles de los que existen en realidad, mientras que para alturas altas localiza árboles en exceso.



Gráfico de cajas y bigotes

Gráfico de Cajas y Bigotes

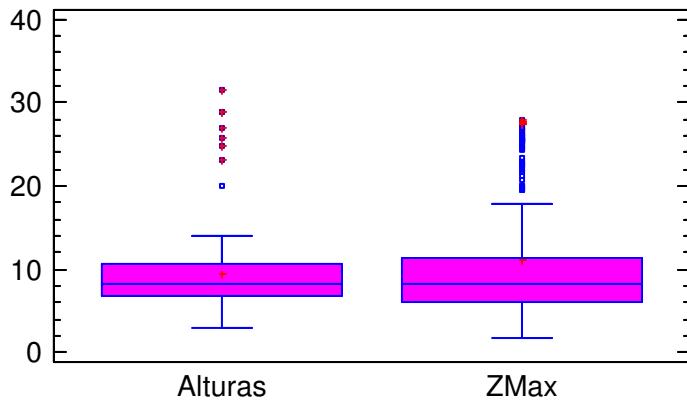


Figura 64 C. Gráfico de Cajas y Bigotes. Parcela 2.

Nota: En el gráfico de cajas y bigotes se muestran que los resultados pertenecen a la misma población puesto que la media normal de altura en ambas poblaciones es prácticamente coincidente. Además los árboles entre ambas muestras coinciden también en alturas, aunque se puede decir que nuestro algoritmo encuentra árboles de un metro por encima y por debajo de los que existen en campo.



10.3 Parcela Volante 3

Resumen Estadístico

	Alturas	ZMax
Frecuencia	187	223
Media	8,81711	12,2272
Varianza	18,1727	43,9187
Desviación típica	4,26295	6,62712
Mínimo	2,0	1,51095
Máximo	25,3	27,1707
Rango	23,3	25,6597
Asimetría tipi.	7,48302	5,36469
Curtosis típificada	8,9038	-1,30707

Tabla 6. Resumen Estadístico Parcela 3.

Nota: Se observa que la muestra de campo tiene menos árboles que los obtenidos con el algoritmo, aunque la diferencia no es excesivamente grande. En cuanto a la media se aprecia que con el algoritmo se obtiene una media de altura bastante superior del orden de 3,5 metros.



Histograma

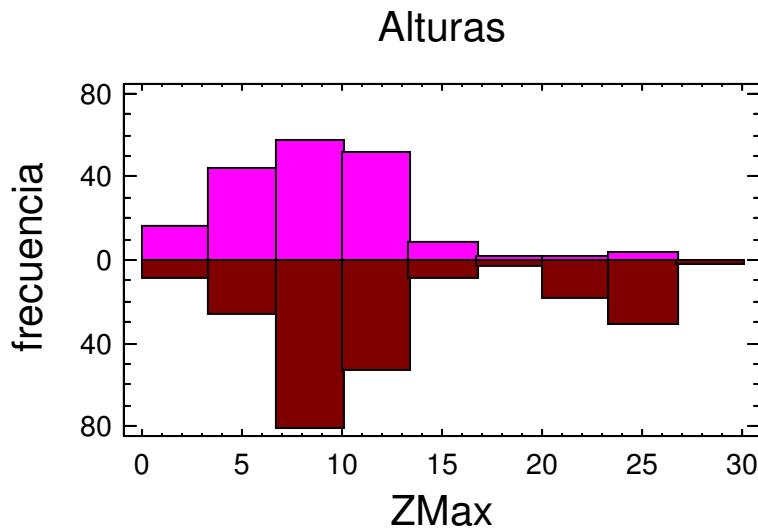


Figura 65 A. Histograma. Parcela 3.

Nota: En el histograma se aprecia que alturas bajas (0-7 metros) el algoritmo encuentra menos árboles de los que en realidad hay. De 7 a 10 metros localiza árboles en exceso, mientras que de 10 a 17 metros los árboles localizados por el algoritmo son bastante exactos a los que hay en campo. Para alturas mayores nuestro algoritmo difiere, ya que localiza árboles en exceso.



Densidades suavizadas

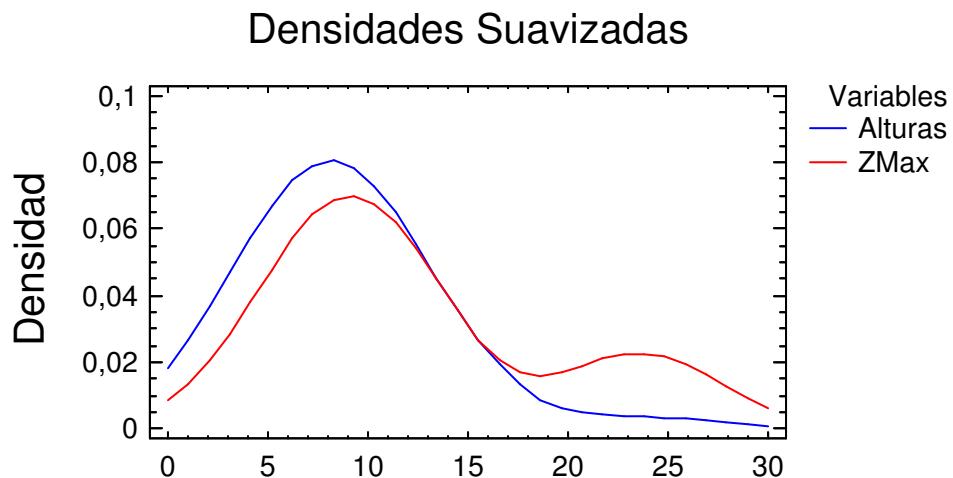


Figura 65 B. Densidades Suavizadas. Parcela 3.

Nota: Al igual que en parcelas anteriores se aprecia que encuentra menos árboles pequeños mientras que la estimación de árboles grandes es en superior a la que hay en realidad.



Gráfico de cajas y bigotes

Gráfico de Cajas y Bigotes

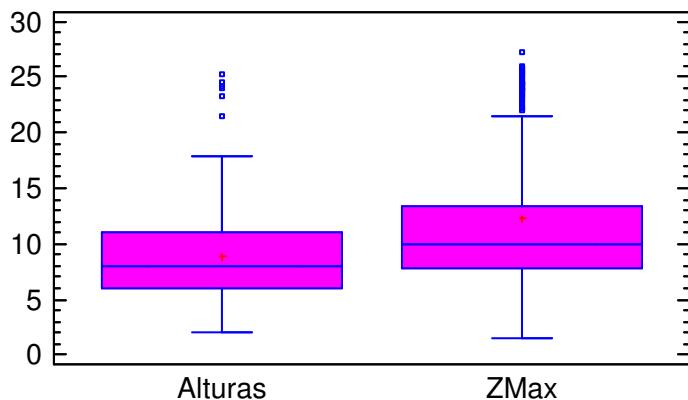


Figura 65 C. Gráfico de Cajas y Bigotes. Parcela 3.

Nota: Se observa que la media de altura para árboles de campo es aproximadamente de 8 metros, mientras que para los árboles obtenidos con el algoritmo es de 10 metros y los árboles localizados por los algoritmos son mayores que los existentes en campos.



10.4 Parcela Volante 4

Resumen Estadístico

	Alturas	ZMax
Frecuencia	224	198
Media	5,56339	12,7584
Varianza	29,0006	63,0971
Desviación típica	5,38522	7,94337
Mínimo	1,4	1,02941
Máximo	29,9	24,4345
Rango	28,5	23,4051
Asimetría tipi.	15,9245	-1,75284
Curtosis típificada	18,276	-4,73118

Tabla 7. Resumen Estadístico Parcela 4.

Nota: En esta parcela los árboles obtenidos por el algoritmos son ligeramente menores que los árboles obtenidos de campo, a diferencia de lo que había ocurrido hasta ahora con el resto de parcelas. La diferencia entre la media para las dos muestras es muy alta, aproximadamente de 7 metros, ya que el algoritmo sobreestima árboles altos y por el contrario no detecta árboles bajos que sí que hay en esta parcela.



Histograma

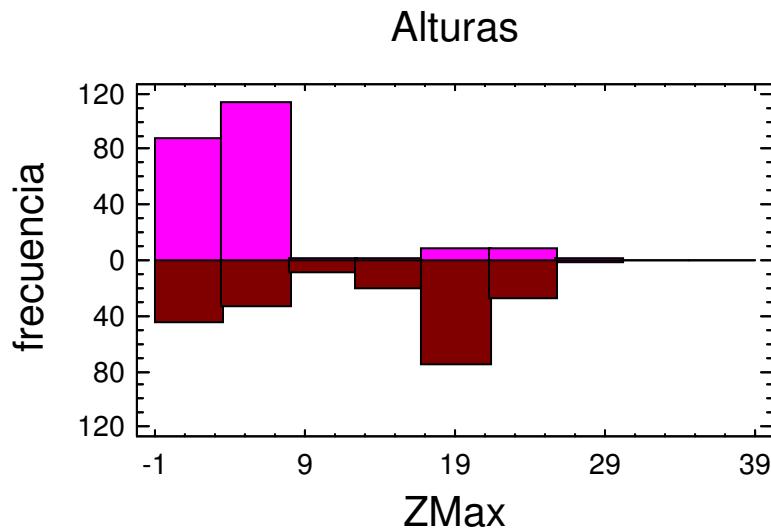


Figura 66 A. Histograma. Parcela 4.

Nota: En el histograma se observa que en la muestra de campo los árboles que predominan tienen una altura de 3 a 8 metros aproximadamente y árboles más bajos incluso, mientras que los datos obtenidos por el algoritmo se muestra un predominio de árboles altos de 18 a 21 metros, sobreestimados con respecto a la muestra de campo. Esto hace suponer que nuestro algoritmo no se está comportando muy bien para poblaciones con árboles bajos. Ya que no los detecta todos.



Densidades suavizadas

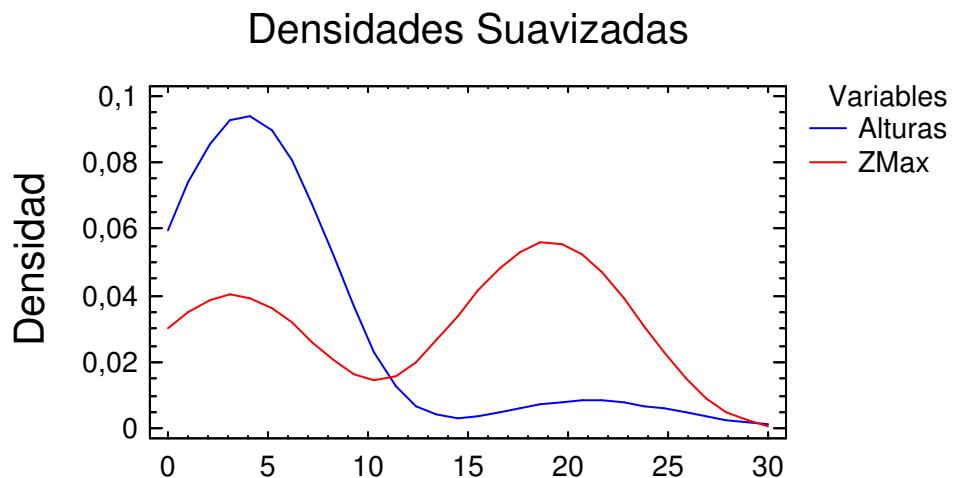


Figura 66 B. Densidades Suavizadas. Parcela 4.

Nota: En este gráfico se observa claramente que mientras que en la muestra de campo hay mayor cantidad de árboles bajos y muy pocos árboles altos, nuestro algoritmo no detecta esos árboles bajos en su totalidad y por el contrario localiza árboles altos en exceso.

Gráfico de cajas y bigotes

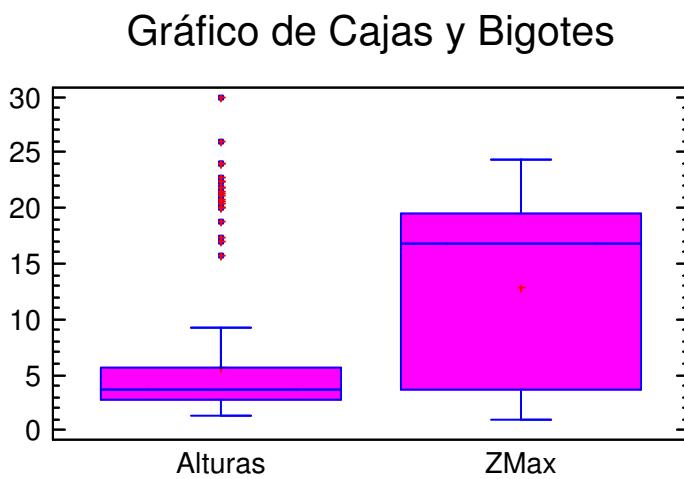


Figura 66 C. Gráfico de Cajas y Bigotes. Parcela 4.



Nota: En el gráfico se observan resultados muy dispares entre ambas muestras. Mientras que en campo la altura normal de los árboles es de 3,5 metros aproximadamente. El algoritmo estima que la altura normal es de 17 metros.

Los resultados para esta parcela son realmente malos, esto es debido al predominio de árboles bajos en la misma.



10.5 Parcela Volante 5

Resumen Estadístico

	Alturas	ZMax
Frecuencia	60	199
Media	12,4917	14,141
Varianza	28,3737	25,3606
Desviación típica	5,32669	5,03593
Mínimo	2,0	1,00242
Máximo	21,5	21,5832
Rango	19,5	20,5808
Asimetría tipi.	-1,9532	-4,47831
Curtosis típificada	-0,98055	-0,953623

Tabla 8. Resumen Estadístico Parcela 5.

Nota: Los resultados obtenidos del algoritmo contienen más árboles que los resultados procedentes de campo. Se observa que la media de las dos poblaciones difiere en menos de 2 metros en exceso por parte del algoritmo.



Histograma

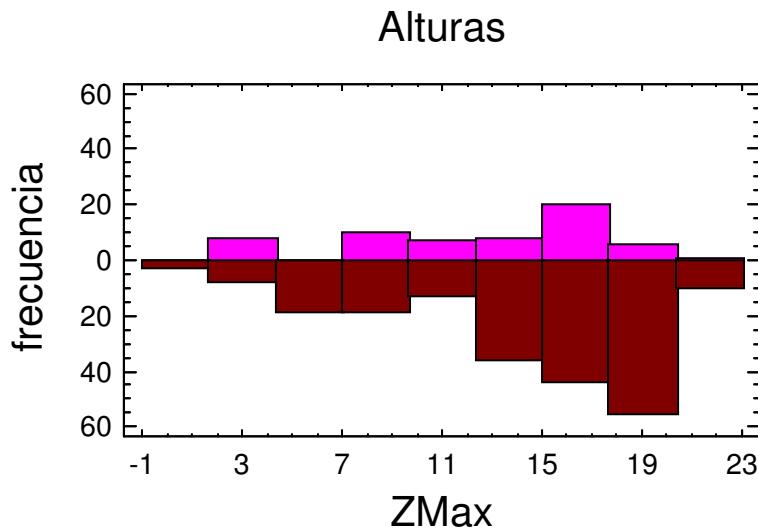


Figura 67 A. Histograma. Parcela 5.

Densidades Suavizadas

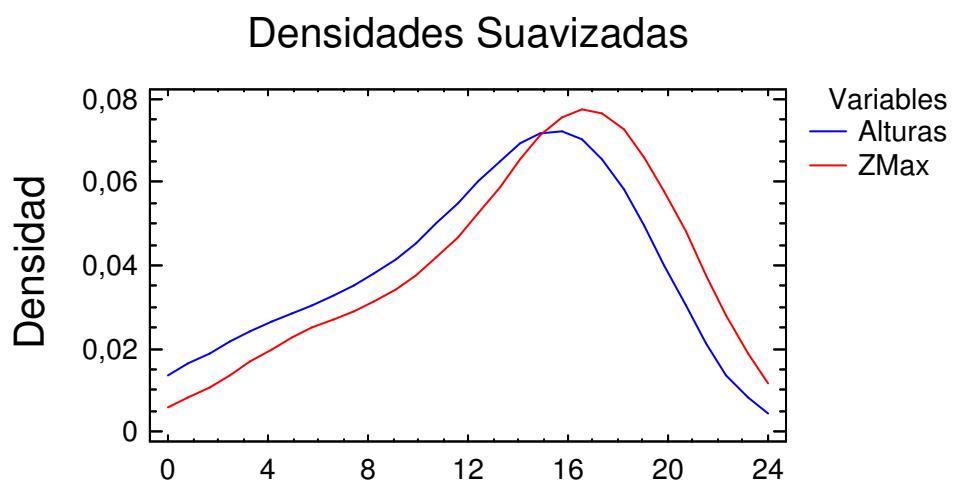


Figura 67 B. Densidades Suavizadas. Parcela 5.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

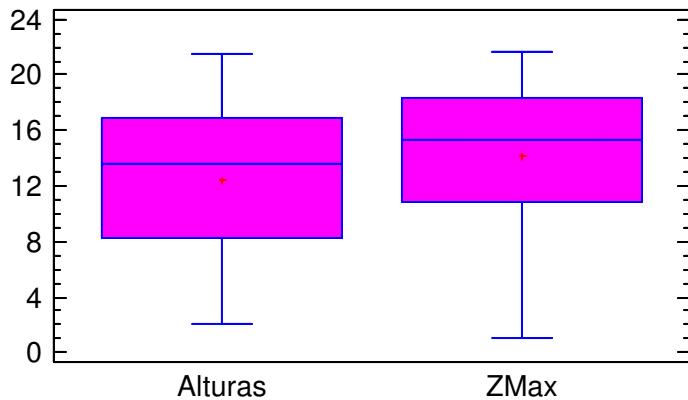


Figura 67 C. Gráfico de Cajas y Bigotes. Parcela 5.

Nota: La altura media de los árboles de la población uno es de 14 metros aproximadamente, mientras que la altura media de los árboles de la población dos es de 15 metros. Ambas difieren en 1 metro aproximadamente. El rango de valores es más amplio en la muestra de campo que en la muestra obtenida por el algoritmo, en concreto la primera oscila entre 8 y 16 metros, mientras que la segunda oscila entre 11 y 17 metros.



10.6 Parcela Volante 6

Resumen Estadístico

	Alturas	ZMax
Frecuencia	25	163
Media	18,3432	17,8693
Varianza	7,71099	15,5252
Desviación típica	2,77687	3,94021
Mínimo	9,3	2,08879
Máximo	23,7	24,532
Rango	14,4	22,4432
Asimetría tipi.	-2,32988	-5,2871
Curtosis típificada	3,8593	3,28308

Tabla 9. Resumen Estadístico Parcela 6.

Nota: El número de muestras de ambas poblaciones es muy dispar, la muestra 1 tiene 25 árboles mientras que la muestra 2 está sobreestimada, puesto que tienen 163. Se observa que la media de la muestra 1 es de 18,34 metros frente a los 17,86 metros de la muestra 2, en realidad hay menos de medio metro de diferencia por defecto.



Histograma

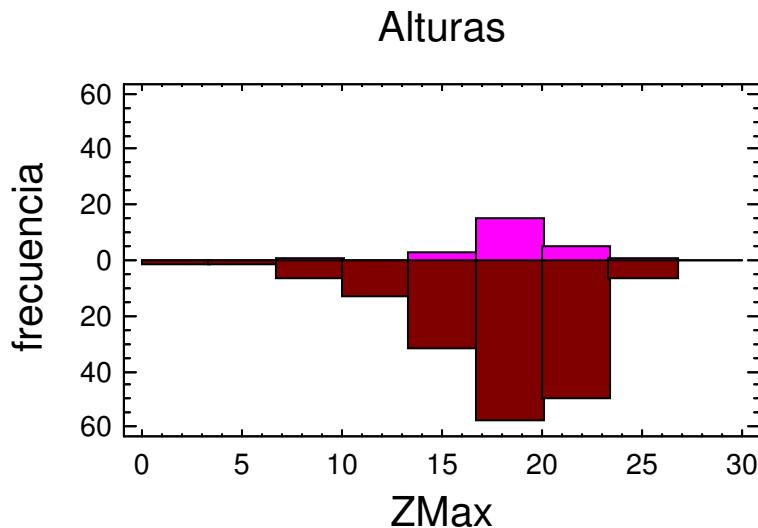


Figura 68 A. Histograma. Parcela 6.

Nota: En el histograma se observa claramente la diferencia de árboles entre ambas muestras, pero se aprecia que en la parte donde hay árboles según el inventario de campo, alturas entre 13 y 23 metros se observan mayor cantidad de árboles que en el resto de zonas.

Densidades Suavizadas

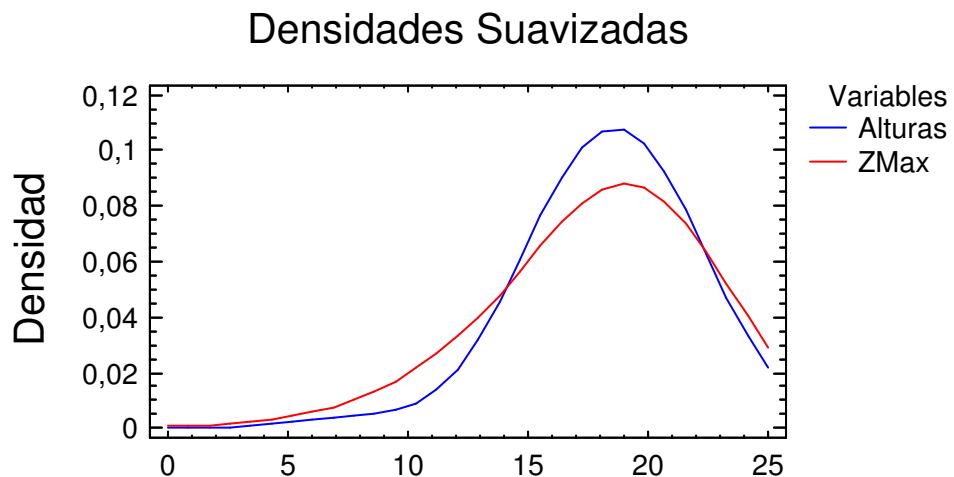


Figura 68 B. Densidades Suavizadas. Parcela 6.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

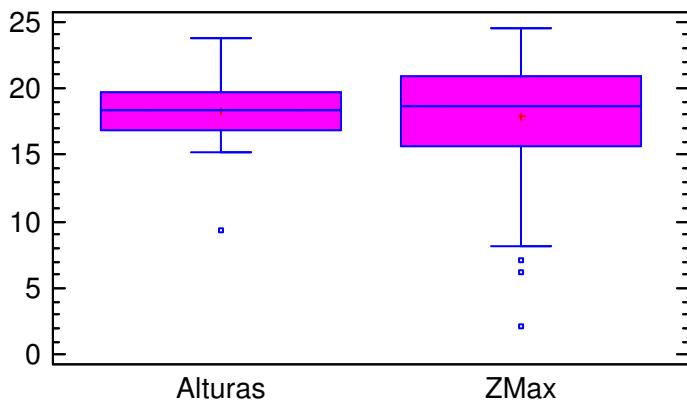


Figura 68 C. Gráfico de Cajas y Bigotes. Parcela 6.

Nota: En este gráfico se observa que la media de ambas muestras es muy similar, aproximadamente 18 metros. Y se observa que el rango de alturas de los árboles de la muestra 2 es mayor que el rango de alturas de la muestra 1. El de la muestra dos va de 16 a 21 metros, mientras que el de la muestra 1 va de 17 a 19 metros. Es decir que hay una diferencia de un metro por debajo y otra diferencia de 2 metros en exceso.



10.7 Parcela Volante 7

Resumen Estadístico

	Alturas	ZMax
Frecuencia	47	204
Media	15,4574	16,7497
Varianza	43,4129	55,4281
Desviación típica	6,58885	7,44501
Mínimo	2,7	1,42513
Máximo	27,4	27,7739
Rango	24,7	26,3487
Asimetría tipi.	-0,247488	-3,94729
Curtosis típificada	-0,710721	-2,04364

Tabla 10. Resumen Estadístico Parcela 7.

Nota: Se observa que la diferencia de árboles entre ambas muestras es muy alta, en la muestra obtenida de campo hay 47 árboles, frente a los 204 obtenidos por el algoritmo. Las medias de ambas muestras difieren en poco más de 1 metro en exceso por parte del algoritmo.



Histograma

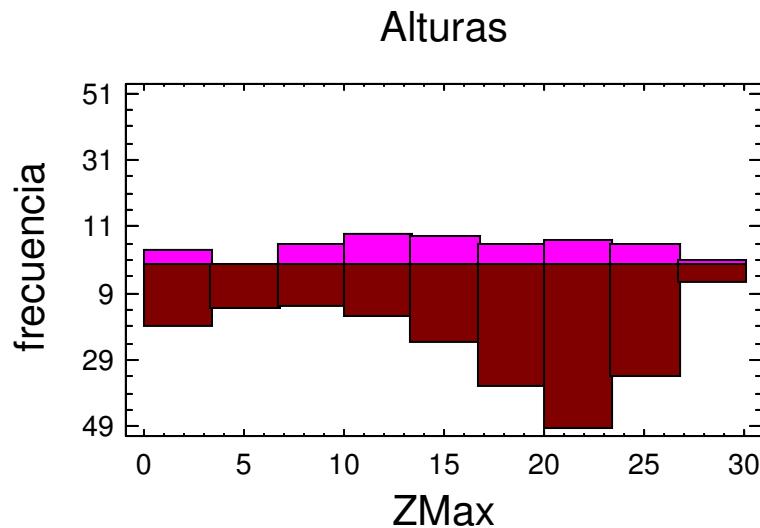


Figura 69 A. Histograma. Parcela 7.

Densidades Suavizadas

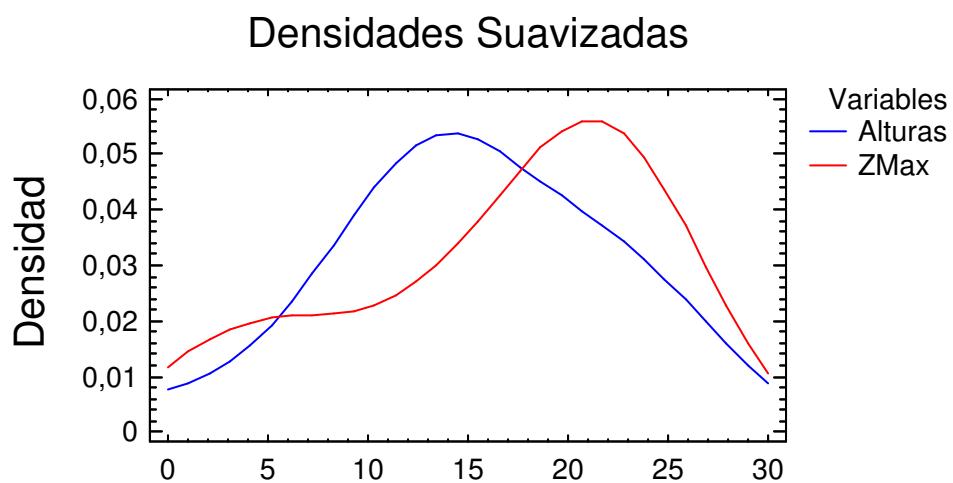


Figura 69 B. Densidades Suavizadas. Parcela 7.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

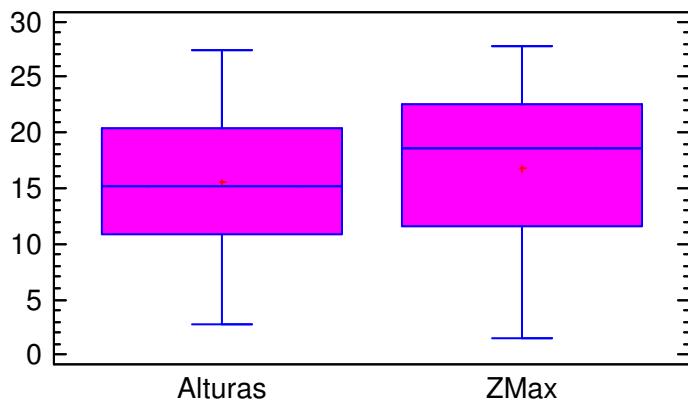


Figura 69 C. Gráfico de Cajas y Bigotes. Parcela 7.

Nota: En el gráfico de cajas y bigotes se observa que la altura normal de los datos de campo es 15 metros aproximadamente frente a los 17 metros en el caso de los datos obtenidos con el algoritmo. En el caso de los datos obtenidos de campo se observa que se encuentran árboles más bajos entorno a 11 metros mientras que en la muestra del algoritmo los más pequeños serían 12 metros, pero por el contrario se localizan en exceso árboles más altos.



10.8 Parcela Volante 8

Resumen Estadístico

	Alturas	ZMax
Frecuencia	126	170
Media	7,97619	16,8804
Varianza	35,6589	25,1948
Desviación típica	5,97151	5,01944
Mínimo	1,5	3,90286
Máximo	28,4	23,5904
Rango	26,9	19,6875
Asimetría tipi.	5,09011	-3,94431
Curtosis típificada	1,90594	-2,06677

Tabla 11. Resumen Estadístico Parcela 8.

Nota: La diferencia de árboles entre la muestra de campo y la muestra estimada por el algoritmo es de 44 árboles en exceso. La media entre la parcelas es muy grande en concreto de 8 metros.



Histograma

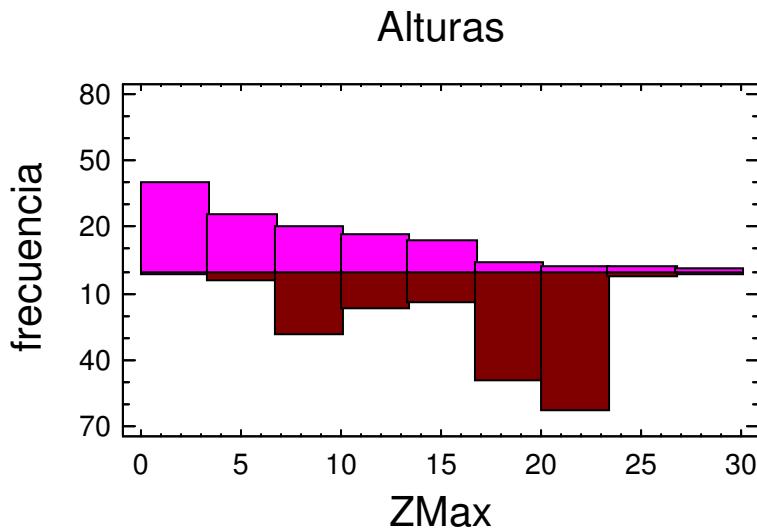


Figura 70 A. Histograma. Parcela 8.

Nota: En el histograma se observa que esta parcela contiene principalmente árboles bajos, y se observa que para alturas menores de 7 metros nos ha resultado imposible localizar esos árboles, mientras que para alturas mayores de 17 metros hemos encontrado árboles en exceso.

Densidades Suavizadas

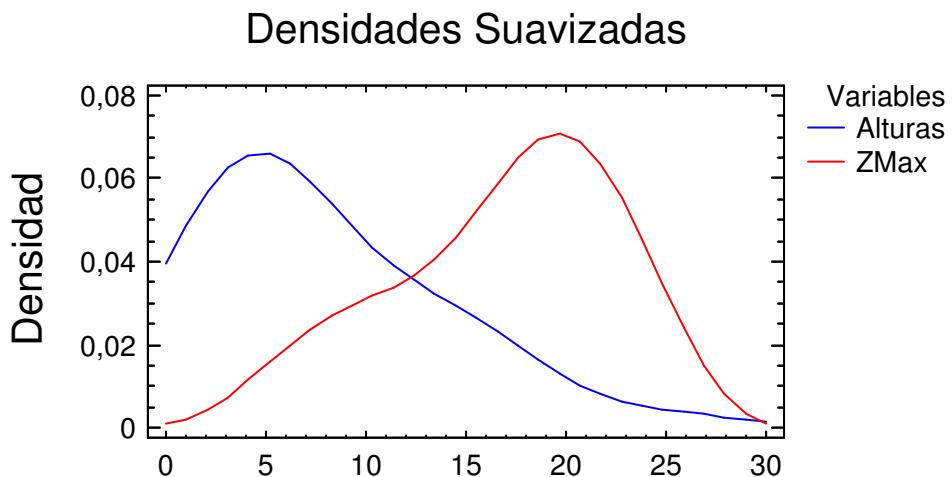


Figura 70 B. Densidades Suavizadas. Parcela 8.



Nota: En esta parcela al igual que en la parcela 4, predominan los árboles bajos, y nuestro algoritmo no los localiza todos. Por el contrario encuentra más árboles altos de los que en realidad existen.

Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

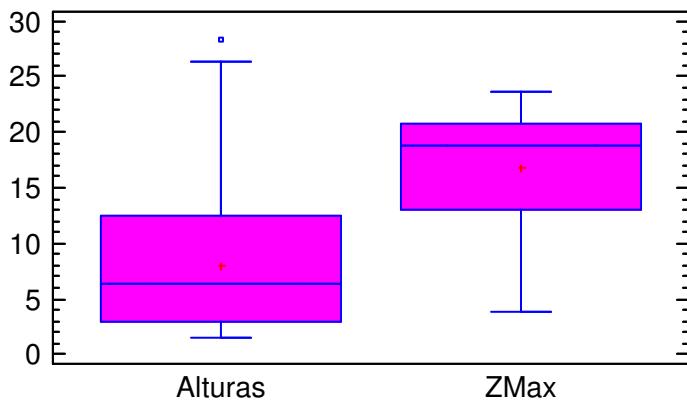


Figura 70 C. Gráfico de Cajas y Bigotes. Parcela 8.

Nota: Se observa que la altura normal de los árboles de campo es de 6 metros frente a los 19 metros de los árboles estimados por el algoritmo. De igual forma hay gran diferencia en torno a las alturas de los árboles obtenidos del inventario frente a los valores obtenidos por el algoritmo.



10.9 Parcela Volante 9

Resumen Estadístico

	Alturas	ZMax
Frecuencia	39	173
Media	13,2923	12,8914
Varianza	17,9713	38,3443
Desviación típica	4,23925	6,19227
Mínimo	6,4	1,21835
Máximo	23,0	21,8682
Rango	16,6	20,6498
Asimetría tipi.	0,78965	-1,68078
Curtosis típica	-1,07483	-3,08865

Tabla 12. Resumen Estadístico Parcela 9.

Nota: Se observa que los árboles obtenidos por el algoritmo son excesivamente mayores que los que existen en realidad. La media obtenida por el algoritmo es ligeramente menor que la que hay en realidad.

Histograma

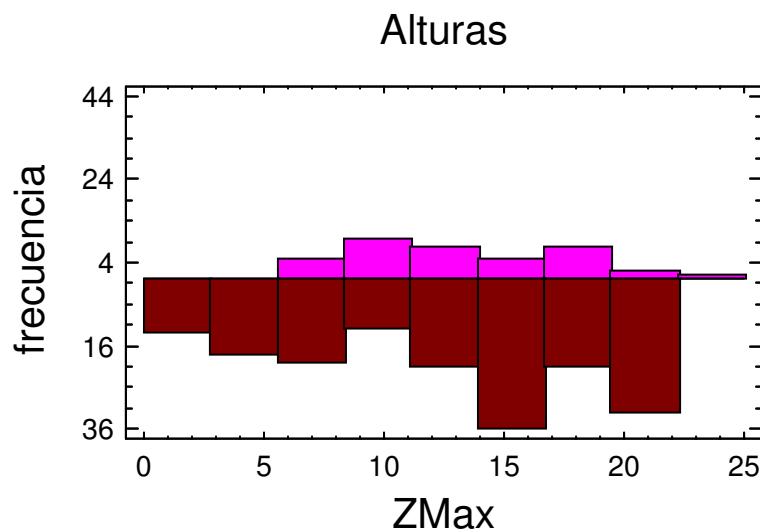


Figura 71 A. Histograma. Parcela 9.



Densidades Suavizadas

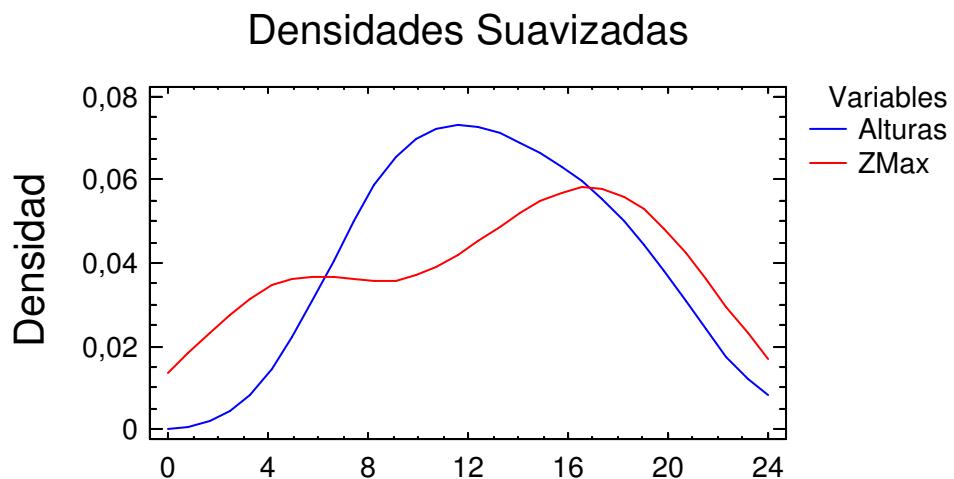


Figura 71 B. Densidades suavizadas. Parcela 9

Gráfico de Cajas y Bigotes

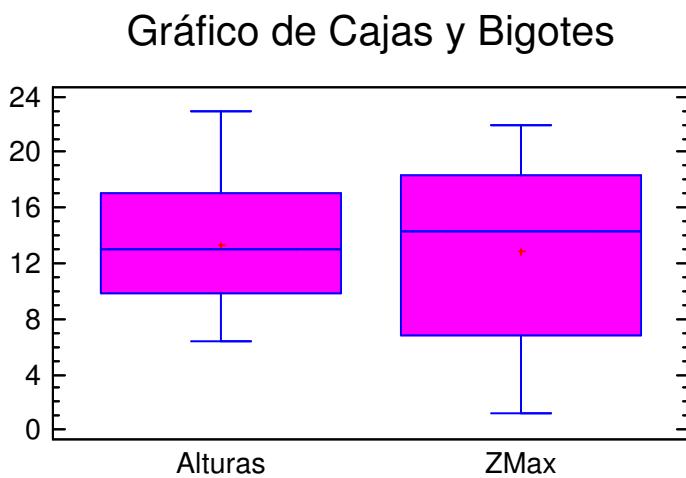


Figura 71 C. Gráfico de Cajas y Bigotes. Parcela 9.

Nota: Se observa que la altura media de los árboles de campo es de 13 metros aproximadamente, mientras que la obtenida por el algoritmo es de 14 metros. El rango de alturas para la muestra de campo es de 10 metros a 17 metros, mientras que el rango es superior en los datos obtenidos por el algoritmo, desde 7 metros a 18 metros.



10.10 Parcela Volante 10

Resumen Estadístico

	Alturas	ZMax
Frecuencia	81	163
Media	22,3914	21,1296
Varianza	8,5673	8,71751
Desviación típica	2,92699	2,95254
Mínimo	7,9	8,66731
Máximo	26,7	24,8487
Rango	18,8	16,1814
Asimetría tipi.	-9,66532	-14,0632
Curtosis típicada	18,0162	21,9855

Tabla 13. Resumen Estadístico Parcela 10.

Nota: El número de árboles estimado por el algoritmo es el doble que el de campo. La media de campo es ligeramente mayor que la estimada por el algoritmo, en concreto 1 metro superior.

Histograma

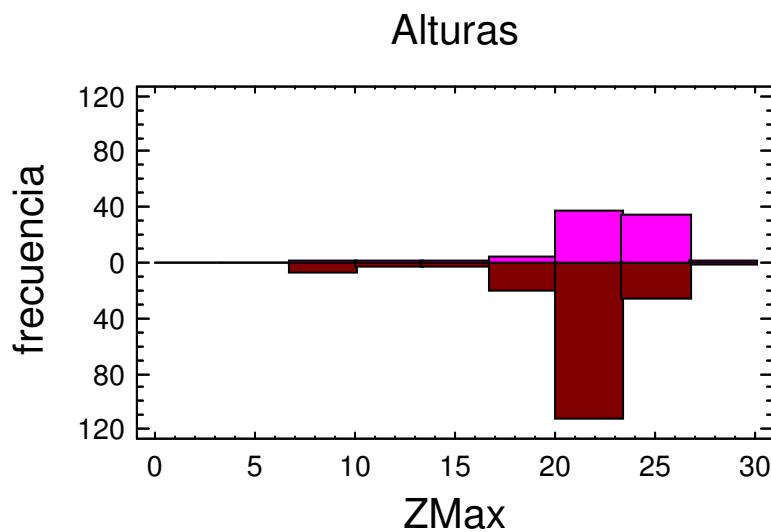


Figura 72 A. Histograma. Parcela 10.

Nota: En el histograma se observa que la gran diferencia está en árboles de 20 a 23 metros que el algoritmo estima muchos más de los que existen. Entre 16 y 30 metros el algoritmo también estima más árboles de los que existen y por el contrario para altura de 23 a 27 metros el algoritmo estima ligeramente menos árboles de los que



existen.

Densidades Suavizadas

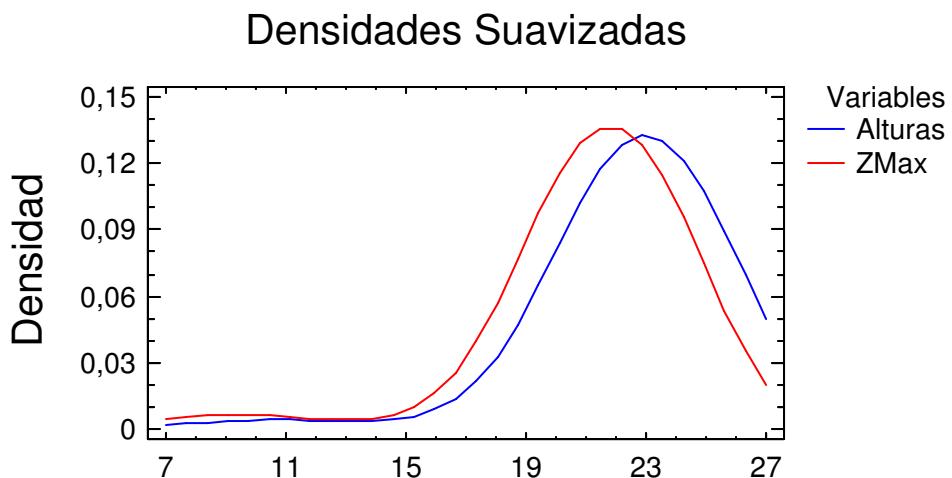


Figura 72 B. Densidades Suavizadas. Parcela 10.

Gráfico de Cajas y Bigotes

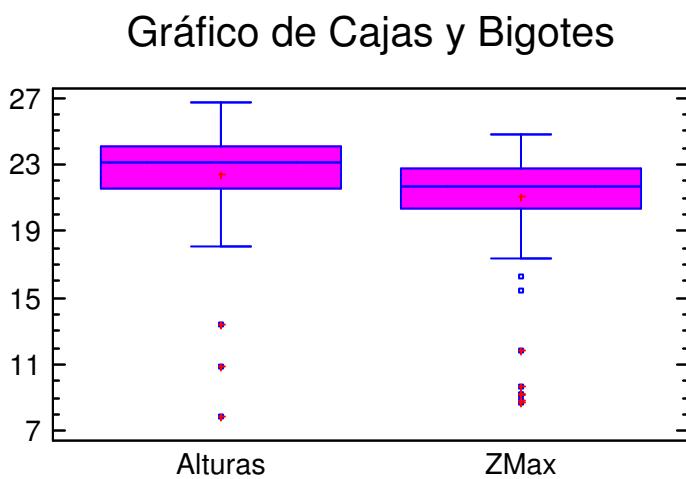


Figura 72 C. Gráfico de Cajas y Bigotes. Parcela 10.

Nota: Se observa que la media de los árboles obtenidos de campo es de 23 metros, mientras que la media obtenida por el algoritmo es de 22 metros. El rango de altura de los árboles de campo es superior al rango de altura de los árboles estimados por el algoritmo, aunque la diferencia no es muy grande.



10.11 Parcela Volante 11

Resumen Estadístico

	Alturas	ZMax
Frecuencia	144	182
Media	18,5014	18,4757
Varianza	6,38266	8,29691
Desviación típica	2,52639	2,88044
Mínimo	12,1	3,84
Máximo	28,8	22,1441
Rango	16,7	18,3041
Asimetría tipi.	3,1903	-17,9304
Curtosis típica	6,8041	34,196

Tabla 14. Resumen Estadístico Parcela 11.

Nota: Se observa que el número de árboles obtenidos por el algoritmo es mayor que los existentes en campo, aproximadamente 40 árboles más de los que hay en realidad. La media es bastante similar en ambas muestras.

Histograma

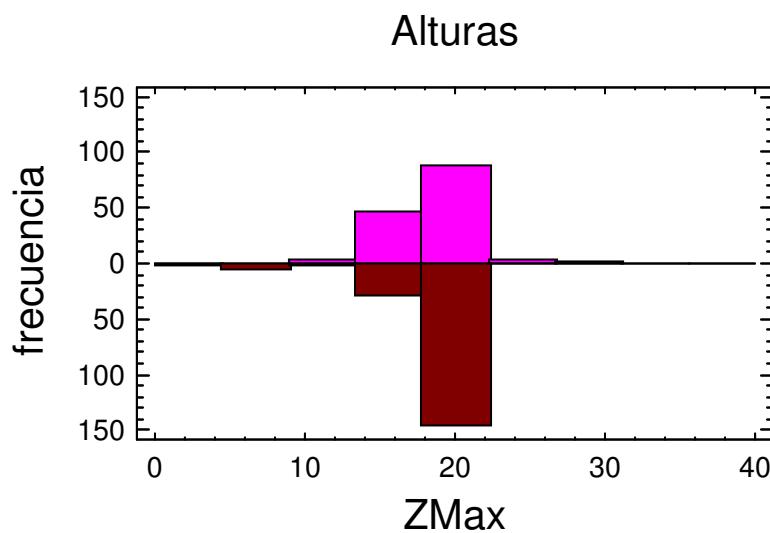


Figura 73 A. Histograma. Parcela 11.



Nota: Se observa que para la altura de 14 a 18 metros, el algoritmo estima menos árboles de los que existen, mientras que para árboles de entre 18 y 22 metros el algoritmo estima bastantes más árboles de los que existen en realidad.

Densidad Suavizada

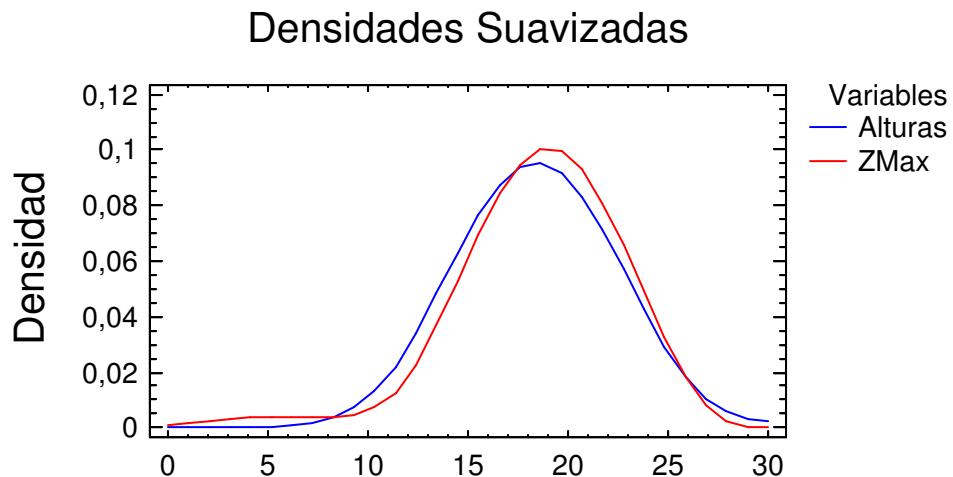


Figura 73 B. Densidades Suavizadas. Parcela 11.

Nota: El gráfico de densidad es bastante coincidente en ambas muestras.

Gráfico de Cajas y Bigotes

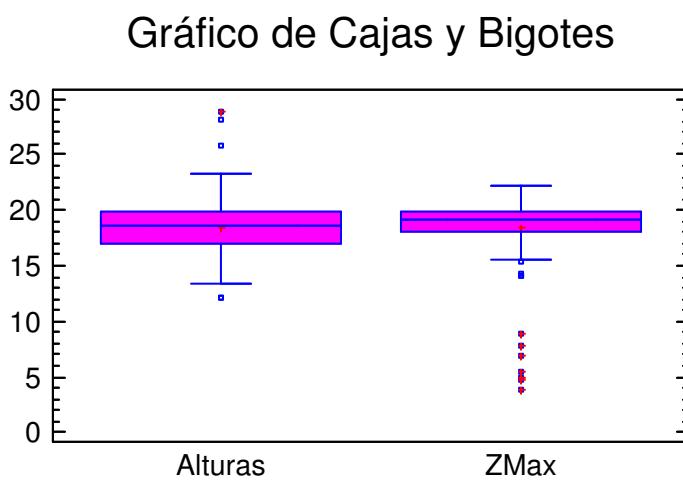


Figura 73 C. Gráfico de Cajas y Bigotes. Parcela 11.

Nota: Se observa que la media entre ambas muestras es muy similar, difiere en menos



de un metro en exceso en lo estimado por el algoritmo que en los datos de campo. Y se observa que con el algoritmo se pierden árboles de alturas más bajas.

10.12 Parcela Volante 12

Resumen Estadístico

	Alturas	ZMax
Frecuencia	149	166
Media	15,6369	17,0842
Varianza	25,5603	11,2634
Desviación típica	5,05572	3,35609
Mínimo	2,7	6,56843
Máximo	27,3	27,2878
Rango	24,6	20,7194
Asimetría tipi.	-2,09243	-0,557135
Curtosis típicada	-0,518345	2,29367

Tabla 15. Resumen Estadístico Parcela 12.

Nota: Se observa que el número de árboles estimado por el algoritmo es ligeramente superior que los árboles existentes en campo, en concreto 17 árboles más. La media obtenida por el algoritmo difiere con la de campo en 1,5 metros aproximadamente.

Histograma

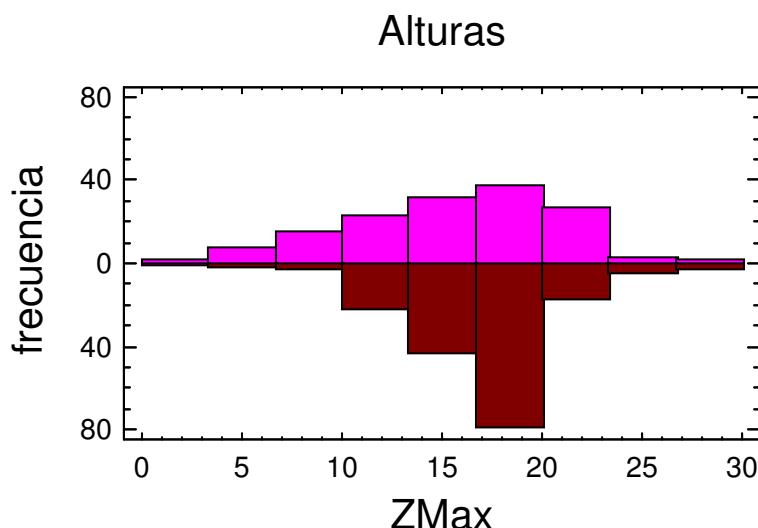


Figura 74 A. Histograma. Parcela 12.



Nota: Se observa que para árboles bajos de 4 a 10 metros es donde difiere principalmente el algoritmo, que localiza muy pocos árboles comparado con los que existen en la muestra de campo. Para árboles de entre 17 y 20 metros el algoritmo obtiene más árboles de los que existen en realidad.

Densidades Suavizadas

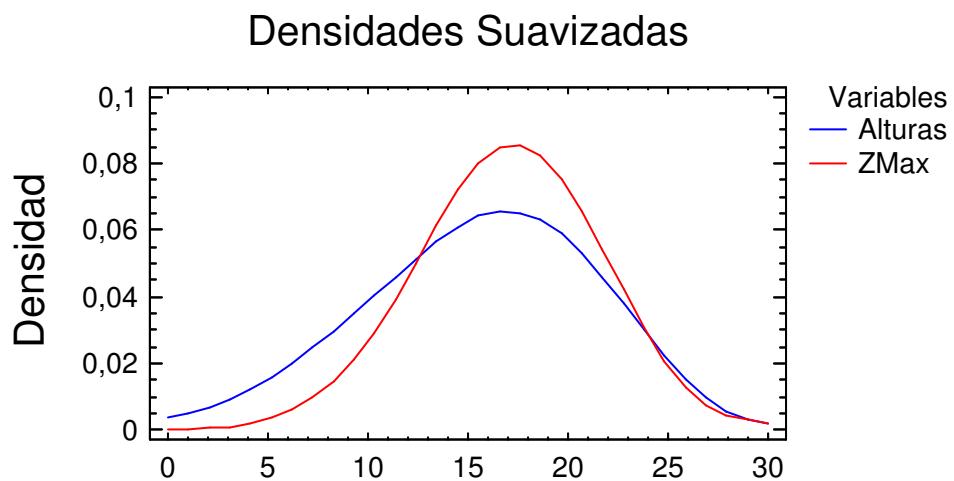


Figura 74 B. Densidades Suavizadas. Parcela 12.

Gráfico de Cajas y Bigotes

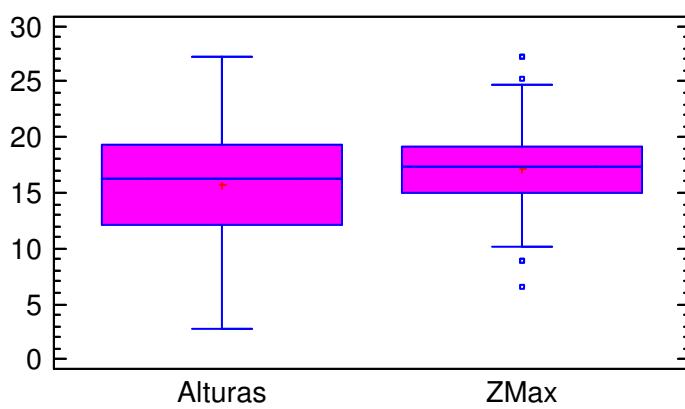


Figura 74 C. Gráfico de Cajas y Bigotes. Parcela 12.



Nota: Se observa que la altura media es ligeramente mayor para los datos obtenidos por el algoritmo que para los datos obtenidos de campo. Esto se debe a que no encontramos todos los árboles bajos y por el contrario encontramos más árboles altos de los que en realidad existen.

10.13 Parcela Volante 13

Resumen Estadístico

	Alturas	ZMax
Frecuencia	26	159
Media	27,65	24,7516
Varianza	5,9978	14,8772
Desviación típica	2,44904	3,8571
Mínimo	23,9	1,00418
Máximo	32,9	28,602
Rango	9,0	27,5979
Asimetría tipi.	0,173363	-24,2344
Curtosis típica	-0,925876	64,924

Tabla 16. Resumen Estadístico Parcela 13.

Nota: Se observa que el número de árboles obtenido por el algoritmo es excesivamente mayor que los árboles existentes en campo. La media obtenida por el algoritmo es 3 metros inferior a la media obtenida de los datos de campo.



Histograma

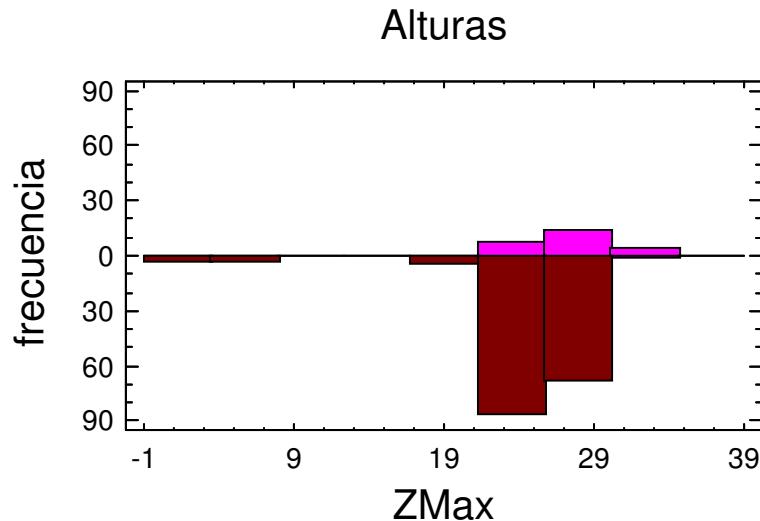


Figura 75 A. Histograma. Parcela 13.

Nota: Se aprecia la gran diferencia de árboles obtenidos con el algoritmo frente a la muestra de campo.

Densidades suavizadas

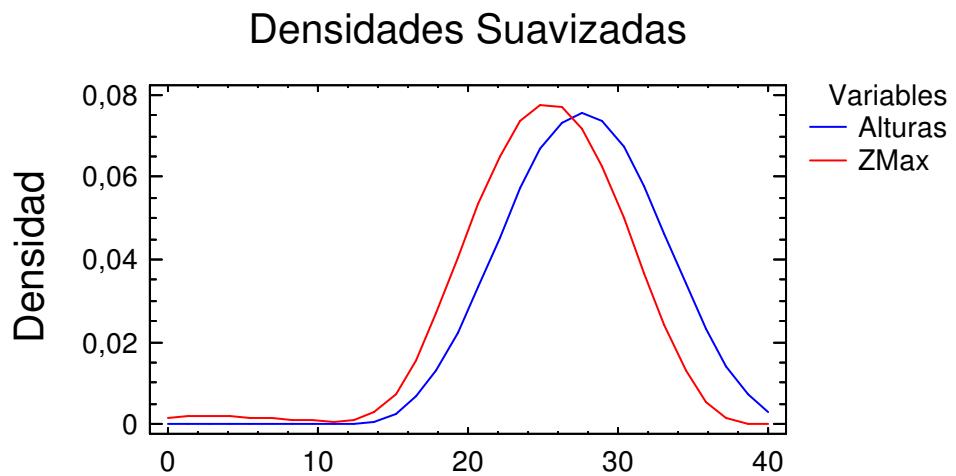


Figura 75 B. Densidades Suavizadas. Parcela 13.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

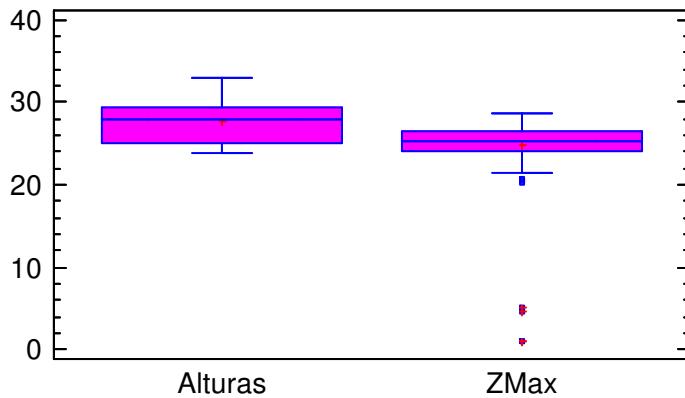


Figura 75 C. Gráfico de Cajas y Bigotes. Parcela 13.

Nota: La altura media de los árboles obtenidos con el algoritmo es inferior a la altura media de los árboles existentes en campo, la diferencia es aproximadamente de 2 metros. El rango de altura de árboles encontrados por el algoritmo es menor que los encontrados en campo, curiosamente en ese caso la diferencia radica en árboles altos de 28 a 30 metros.



10.14 Parcela Volante 14

Resumen Estadístico

	Alturas	ZMax
Frecuencia	27	175
Media	29,9889	25,5182
Varianza	3,84333	21,5364
Desviación típica	1,96044	4,64073
Mínimo	26,5	1,82542
Máximo	33,4	30,2315
Rango	6,9	28,4061
Asimetría tipi.	-0,393322	-21,156
Curtosis típificada	-0,907685	43,5602

Tabla 17. Resumen Estadístico Parcela 14.

Nota: Se observa que el número de árboles obtenido por el algoritmo es excesivamente mayor que los árboles existentes en campo. La media obtenida por el algoritmo es 4,5 metros inferior a la media obtenida de los datos de campo.



Histograma

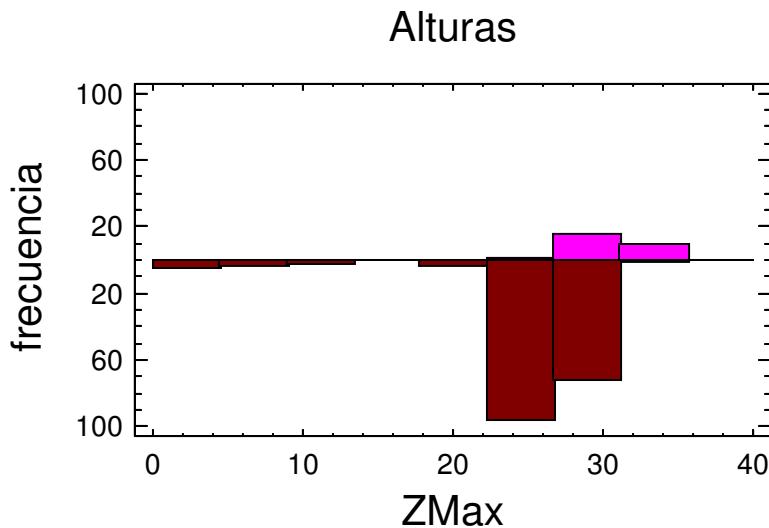


Figura 76 A. Histograma. Parcela 14.

Nota: En el histograma se aprecia que el algoritmo localiza bastantes más árboles de los que existen en realidad. Localiza árboles bajos que ni siquiera existen en la parcela real.



Densidades Suavizadas

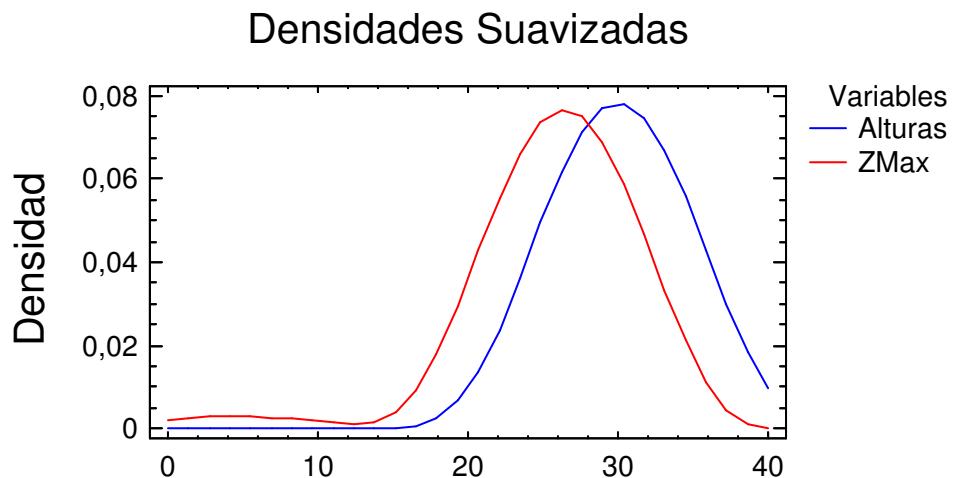


Figura 76 B. Densidades Suavizadas. Parcela 14.

Gráfico de Cajas y Bigotes

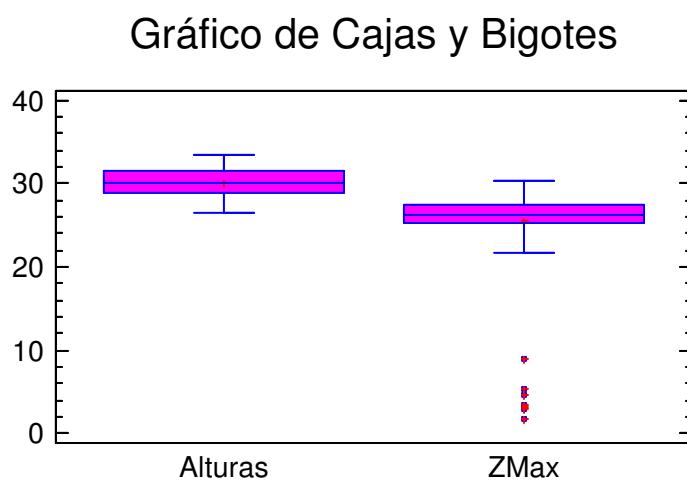


Figura 76 A. Histograma. Parcela 14.



10.15 Parcela Volante 15

Resumen Estadístico

	Alturas	ZMax
Frecuencia	85	271
Media	9,03882	14,9787
Varianza	78,5695	74,8385
Desviación típica	8,86395	8,65092
Mínimo	1,8	1,00191
Máximo	32,6	26,7604
Rango	30,8	25,7585
Asimetría tipi.	5,05866	-1,33117
Curtosis típificada	0,71212	-5,63862

Tabla 18. Resumen Estadístico Parcela 15.



Histograma

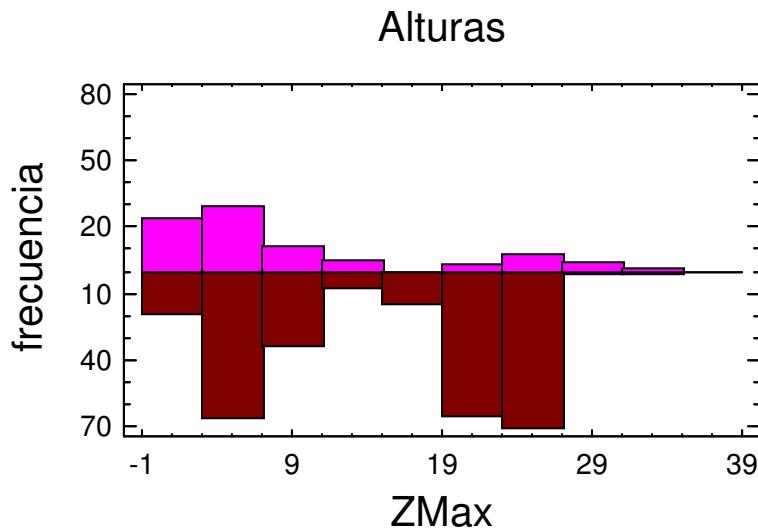


Figura 77 A. Histograma. Parcela 15.

Densidades Suavizadas

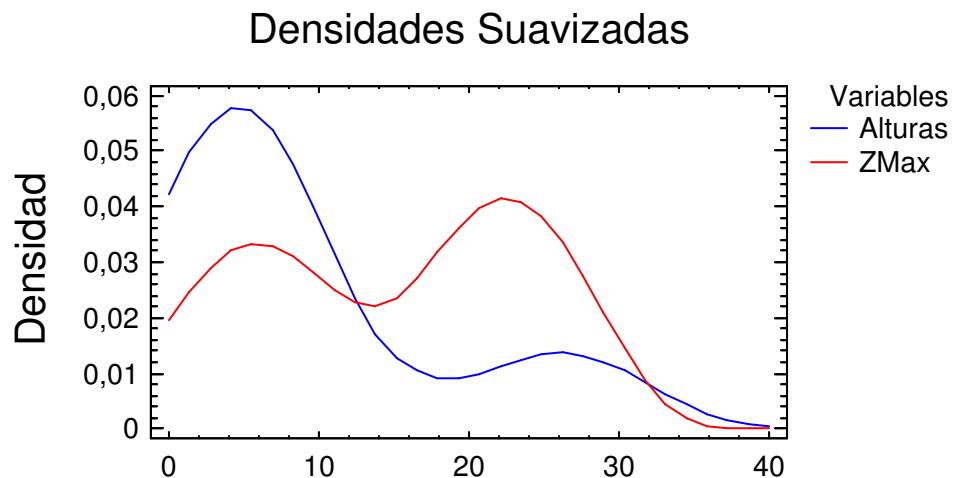


Figura 77 B. Densidades suavizadas. Parcela 15.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

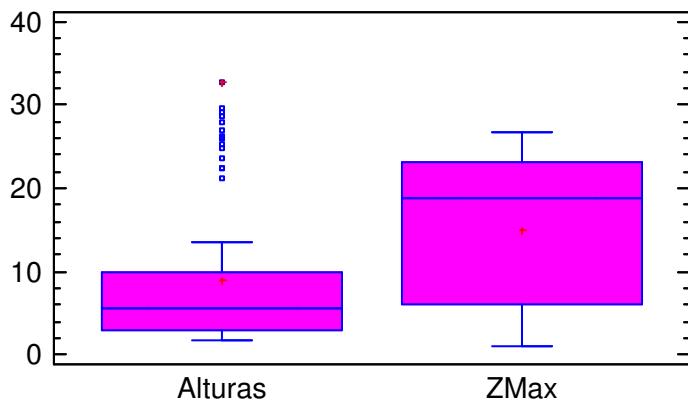


Figura 77 C. Gráfico de Cajas y Bigotes. Parcela 15.



10.16 Parcela Volante 16

Resumen Estadístico

	Alturas	ZMax
Frecuencia	46	179
Media	21,6837	24,7882
Varianza	141,942	68,9913
Desviación típica	11,9139	8,3061
Mínimo	1,5	1,01708
Máximo	35,9	31,9605
Rango	34,4	30,9434
Asimetría tipi.	-2,62802	-10,0507
Curtosis típificada	-1,21819	5,1382

Tabla 19. Resumen Estadístico Parcela 16.



Histograma

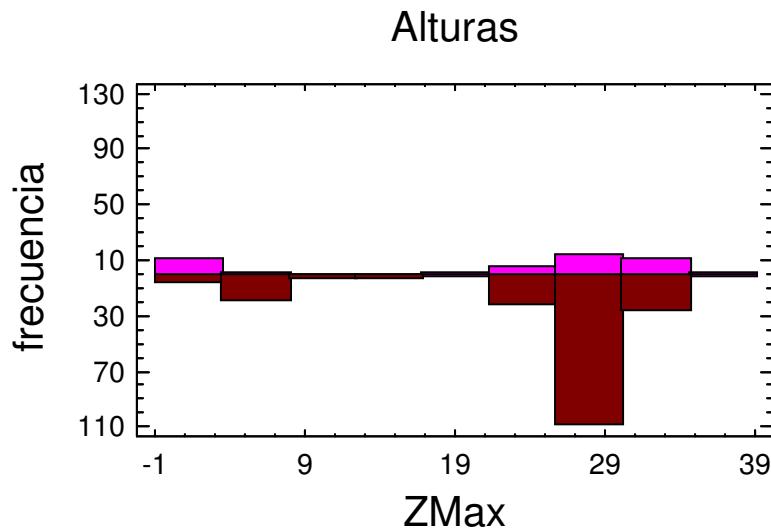


Figura 78 A. Histograma. Parcela 16.

Nota: Se observa que a pesar de que la estimación de árboles por el algoritmo es excesivamente mayor, las alturas de los árboles son bastante coincidentes. Para las alturas bajas el algoritmo localiza árboles unos metros más altos de los que en realidad hay.

Densidades Suavizadas

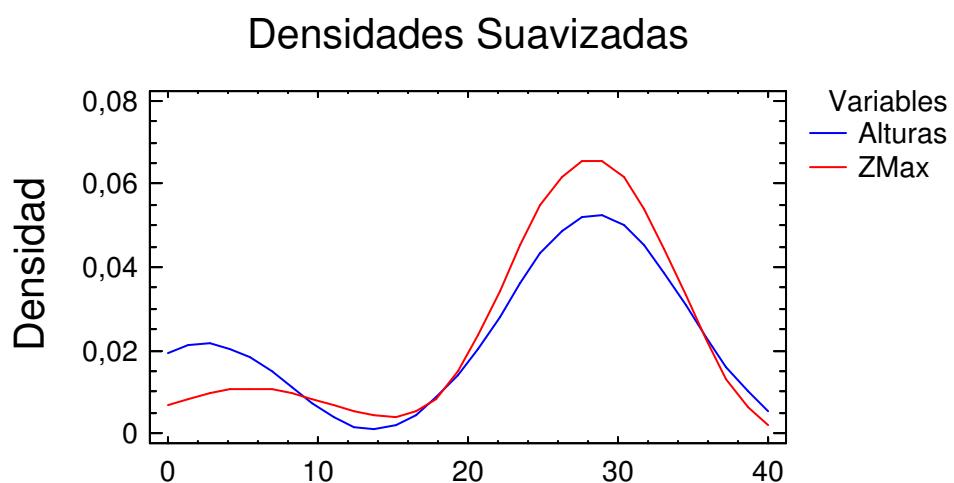


Figura 78 B. Densidades Suavizadas. Parcela 16.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

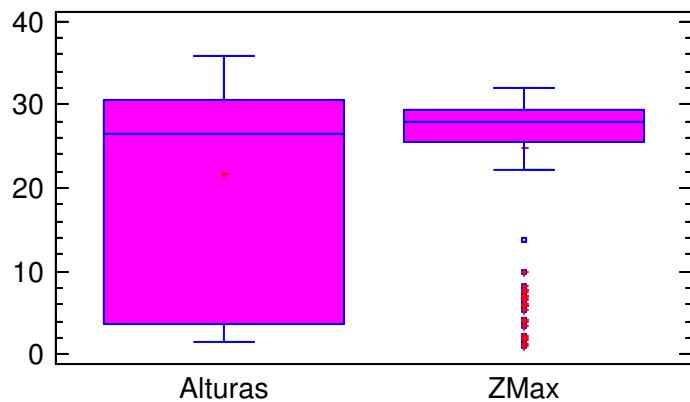


Figura 78 C. Gráfico de Cajas y Bigotes. Parcela 16.



10.17 Parcela Volante 17

Resumen Estadístico

	Alturas	ZMax
Frecuencia	41	182
Media	28,8951	25,4106
Varianza	6,01748	48,5681
Desviación típica	2,45305	6,96908
Mínimo	24,5	1,02092
Máximo	35,4	32,1148
Rango	10,9	31,0939
Asimetría tipi.	0,297151	-13,0202
Curtosis típificada	-0,0402056	12,314

Tabla 20. Resumen Estadístico Parcela 17.



Histograma

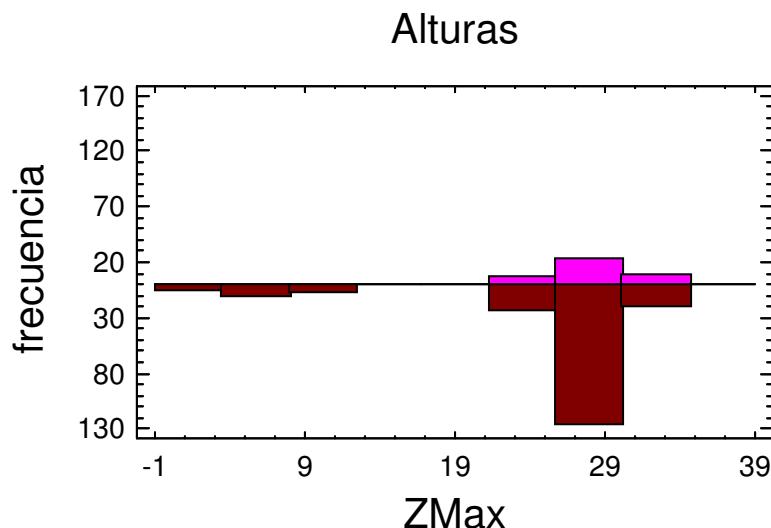


Figura 79 A. Histograma. Parcela 17.

Densidades Suavizadas

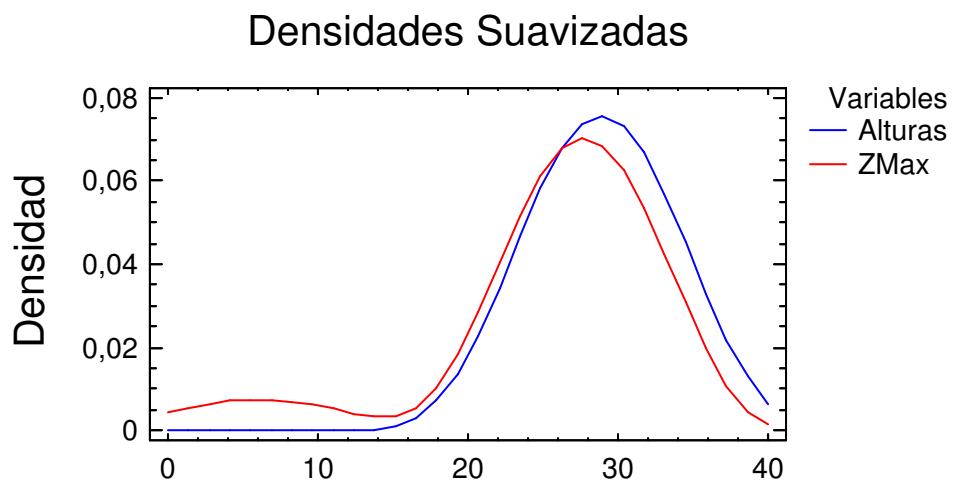


Figura 79 B. Densidades suavizadas. Parcela 17.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

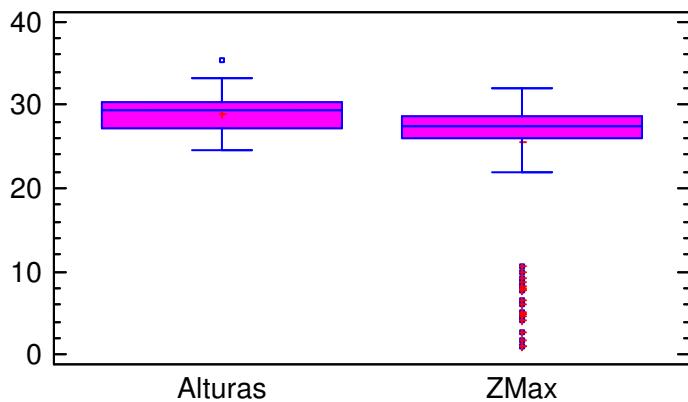


Figura 79 C. Gráfico de Cajas y Bigotes. Parcela 17.



10.18 Parcela Volante 18

Resumen Estadístico

	Alturas	ZMax
Frecuencia	40	546
Media	28,915	24,1023
Varianza	11,3721	57,7576
Desviación típica	3,37225	7,59984
Mínimo	21,9	0,00002
Máximo	36,4	29,6573
Rango	14,5	29,6573
Asimetría tipi.	0,173424	-25,4209
Curtosis típificada	0,00510838	26,8303

Tabla 21. Resumen Estadístico Parcela 18.



Histograma

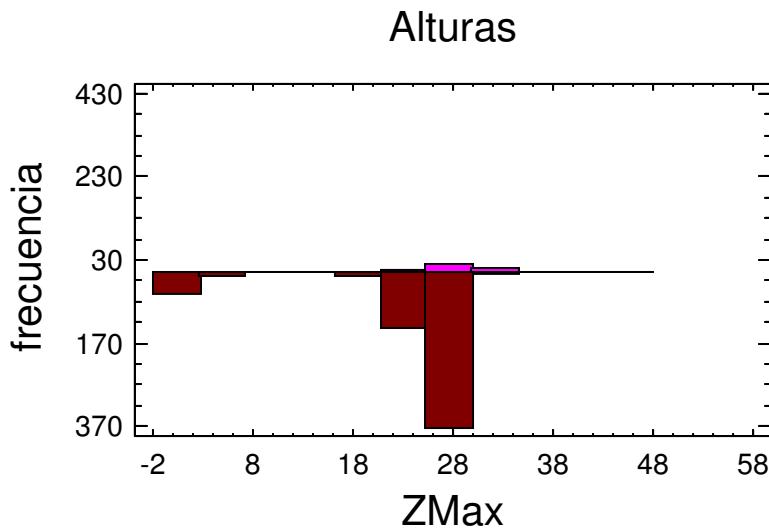


Figura 80 A. Histograma. Parcela 18.

Densidades Suavizadas

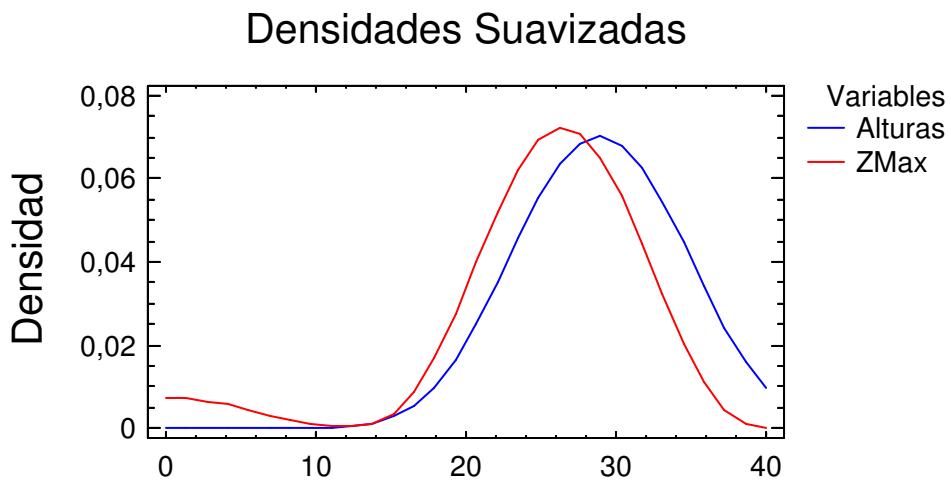


Figura 80 B. Densidades Suavizadas. Parcela 19.



Gráfico de Cajas y Bigotes

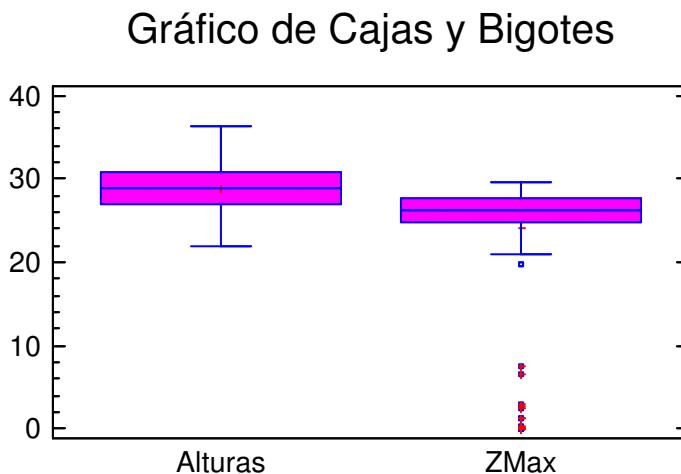


Figura 80 C. Gráfico de Cajas y Bigotes. Parcela 19.

10.19 Parcela Volante 19

Resumen Estadístico

	Alturas	ZMax
Frecuencia	37	188
Media	24,0351	22,6187
Varianza	34,8723	26,1145
Desviación típica	5,90528	5,11023
Mínimo	2,5	1,29982
Máximo	33,2	28,1792
Rango	30,7	26,8793
Asimetría tipi.	-6,34814	-14,0023
Curtosis típificada	10,4583	18,9337

Tabla 22. Resumen Estadístico Parcela 19.



Histograma

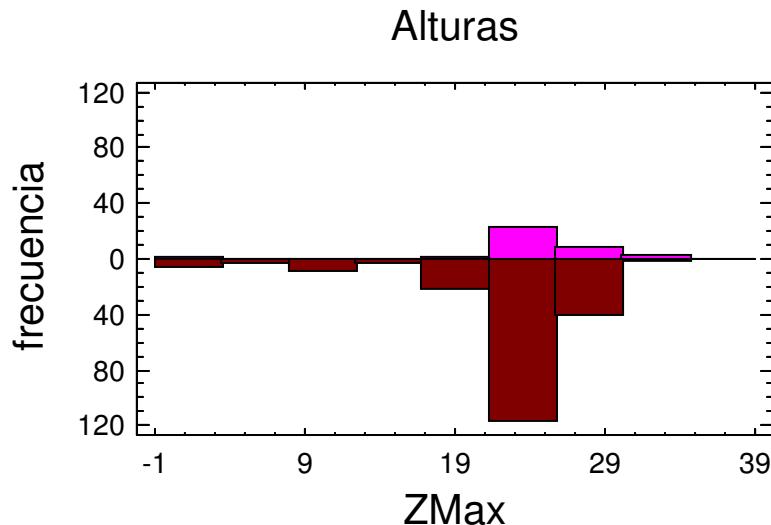


Figura 81 A. Histograma. Parcela 19.

Densidades Suavizadas

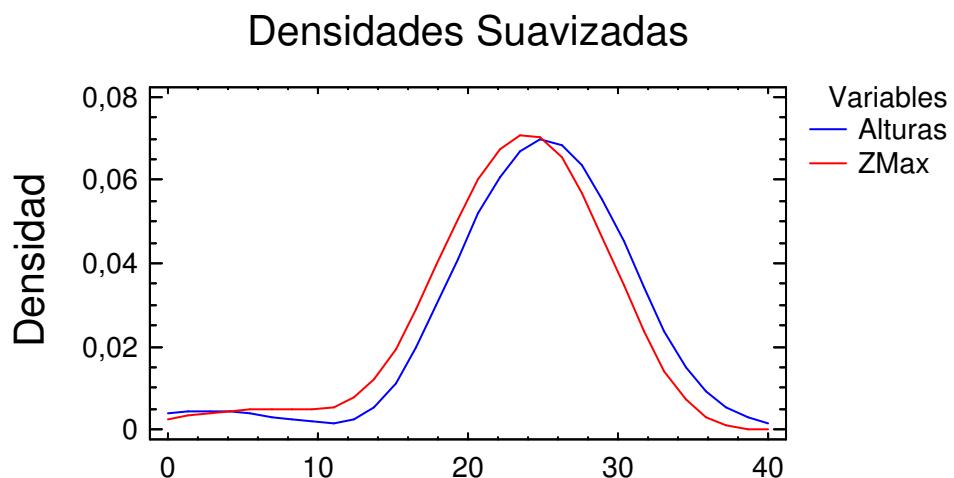


Figura 81 B. Densidades Suavizadas. Parcela 19.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

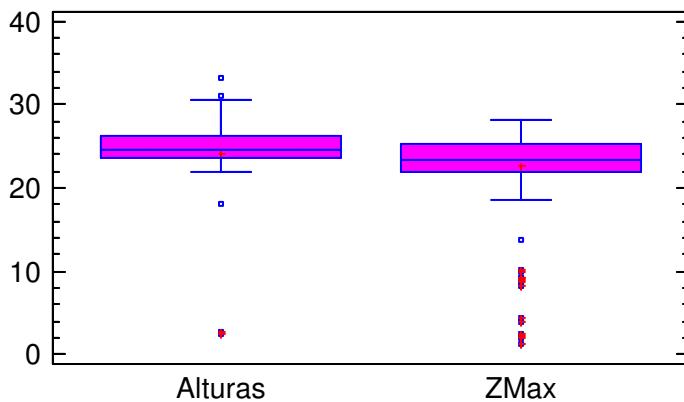


Figura 81 C. Gráfico de Cajas y Bigotes. Parcela 19.

10.20 Parcela Volante 20

Resumen Estadístico

	Alturas	ZMax
Frecuencia	190	322
Media	8,96	18,7888
Varianza	90,5814	113,821
Desviación típica	9,51742	10,6687
Mínimo	2,4	1,32495
Máximo	36,3	31,69
Rango	33,9	30,365
Asimetría tipi.	10,3497	-2,92933
Curtosis típificada	4,84873	-5,96705

Tabla 23. Resumen Estadístico Parcela 20.



Histograma

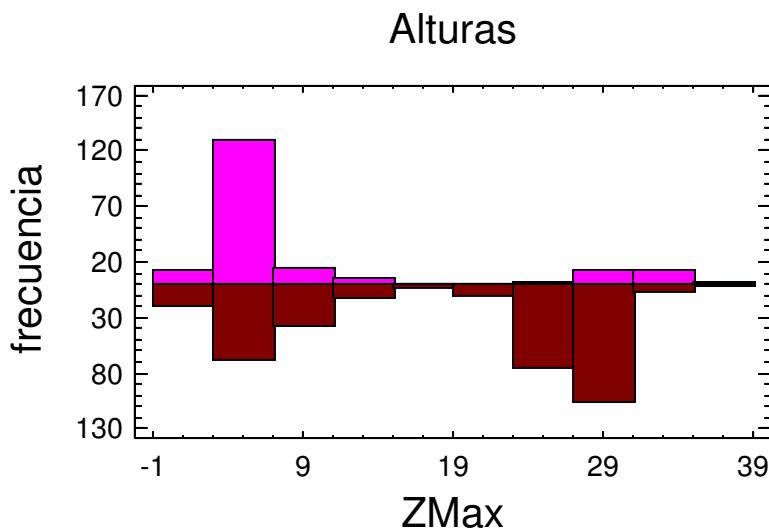


Figura 82 A. Histograma. Parcela 20.

Densidades Suavizadas

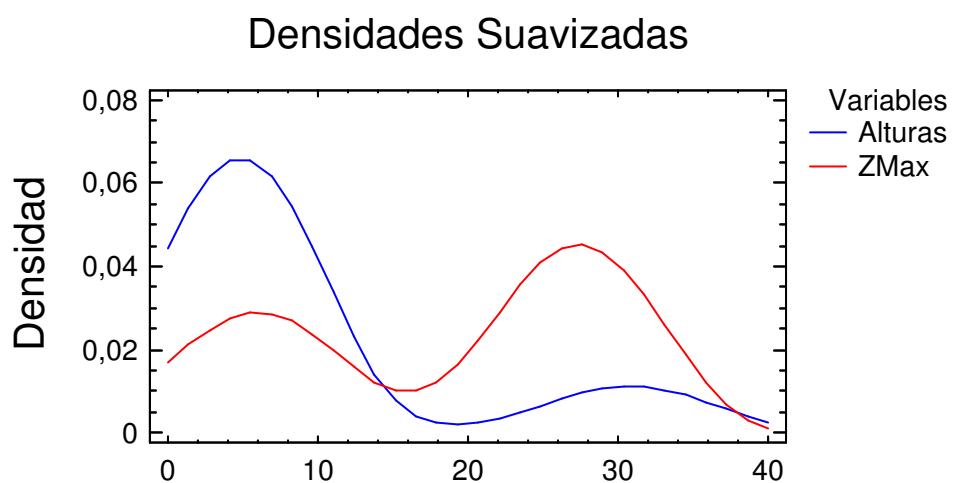


Figura 82 B. Densidades Suavizadas. Parcela 20.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

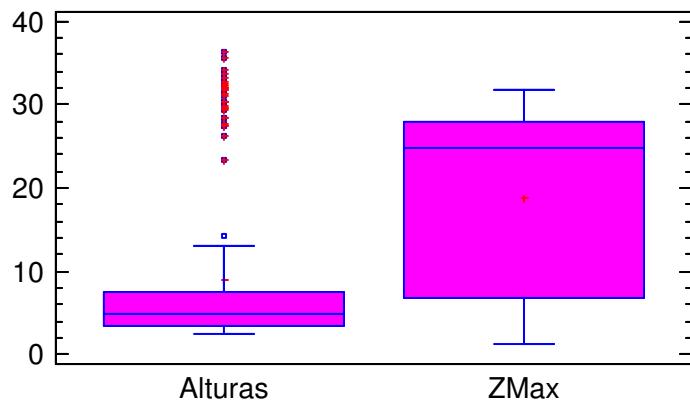


Figura 82 C. Gráfico de Cajas y Bigotes. Parcela 20.



10.21 Parcela Volante 21

Resumen Estadístico

	Alturas	ZMax
Frecuencia	33	212
Media	20,7394	21,1228
Varianza	36,7375	54,6735
Desviación típica	6,06114	7,39415
Mínimo	8,0	1,09287
Máximo	26,9	30,4197
Rango	18,9	29,3268
Asimetría tipi.	-3,05137	-7,75593
Curtosis típificada	0,39302	1,50174

Tabla 24. Resumen Estadístico Parcela 21.



Histograma

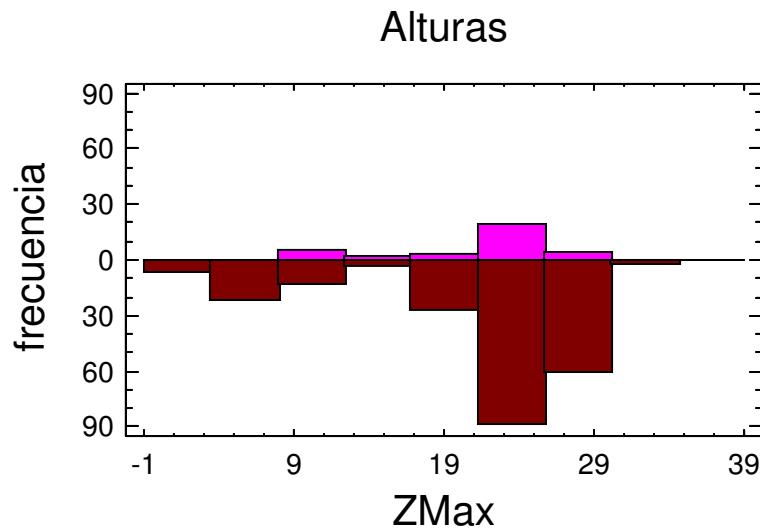


Figura 83 A. Histograma. Parcela 21.

Densidades Suavizadas

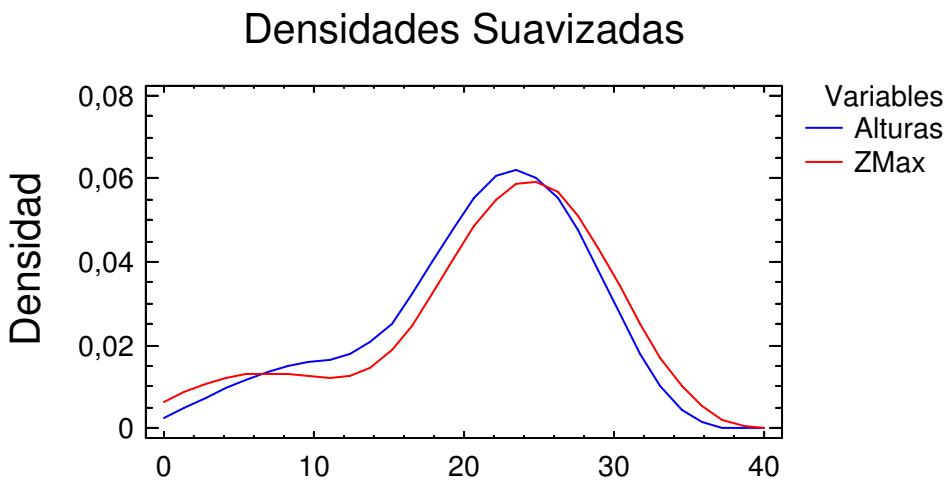


Figura 83 B. Densidades suavizadas. Parcela 21.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

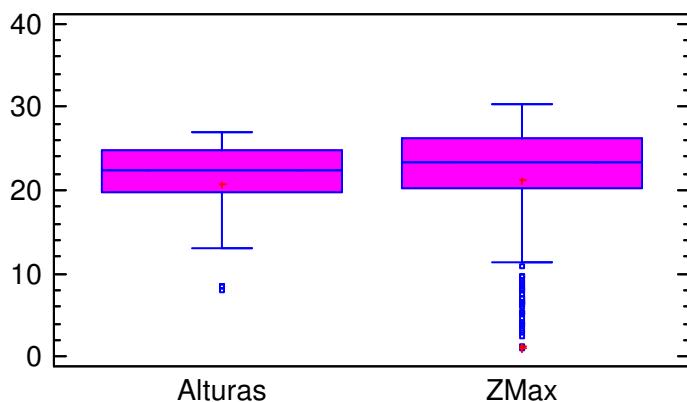


Figura 83 C. Gráfico de Cajas y Bigotes. Parcela 21.

10.22 Parcela Volante 22

Resumen Estadístico

	Alturas	ZMax
Frecuencia	21	173
Media	22,1143	19,9055
Varianza	19,3173	46,451
Desviación típica	4,39514	6,8155
Mínimo	12,1	2,58084
Máximo	30,3	28,9383
Rango	18,2	26,3574
Asimetría tipi.	-0,388167	-6,01253
Curtosis típificada	0,154758	0,823622

Tabla 25. Resumen Estadístico Parcela 22.



Histograma

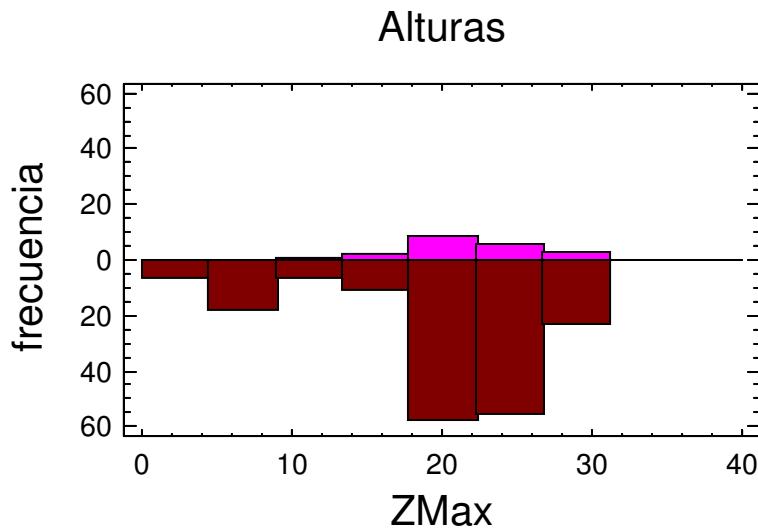


Figura 84 A. Histograma. Parcela 22.

Densidades Suavizadas

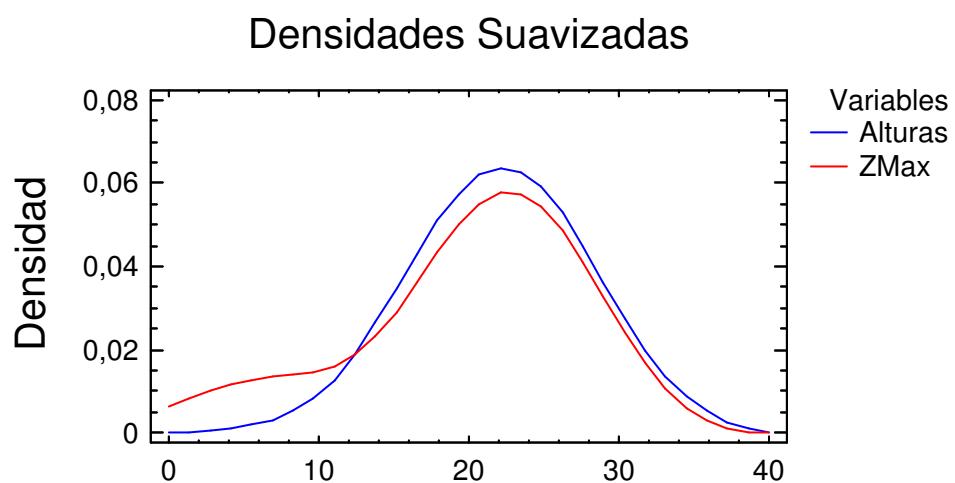


Figura 84 B. Densidades Suavizadas. Parcela 22.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

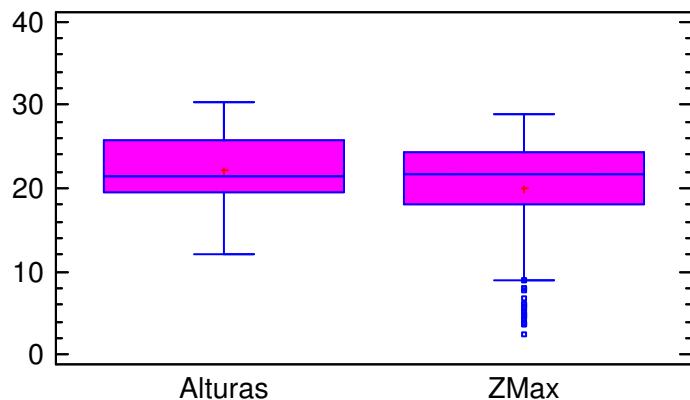


Figura 84 C. Gráfico de Cajas y Bigotes. Parcela 22.



10.23 Parcela Volante 23

Resumen Estadístico

	Alturas	ZMax
Frecuencia	49	185
Media	25,9122	23,5755
Varianza	7,96735	21,5724
Desviación típica	2,82265	4,64461
Mínimo	15,5	2,18767
Máximo	32,2	29,019
Rango	16,7	26,8313
Asimetría tipi.	-2,83551	-17,422
Curtosis típificada	4,03501	32,0171

Tabla 26. Resumen Estadístico Parcela 23.



Histograma

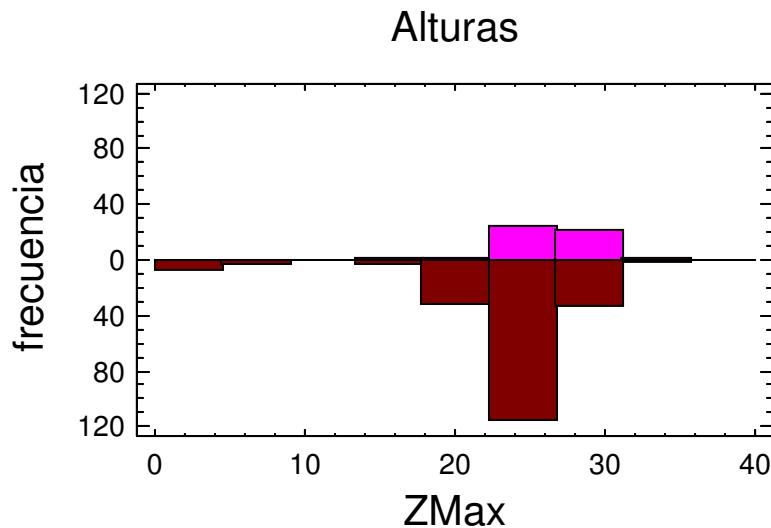


Figura 85 A. Histograma. Parcela 23.

Densidades Suavizadas

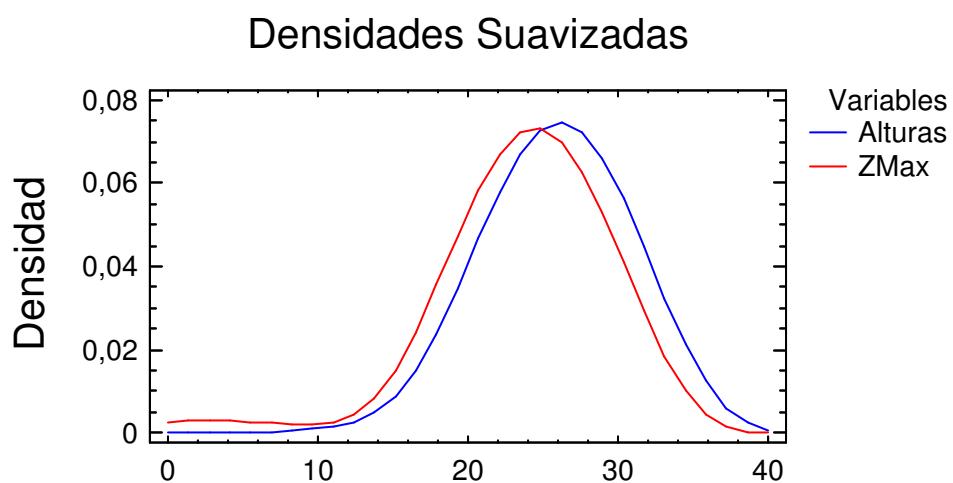


Figura 85 B. Densidades Suavizadas. Parcela 23.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

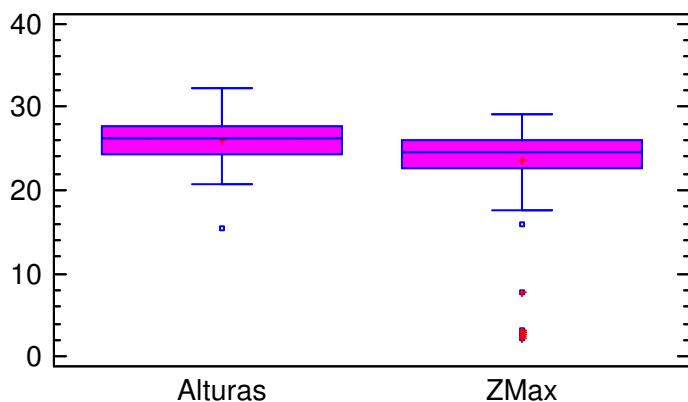


Figura 85 C. Gráfico de Cajas y Bigotes. Parcela 23.



10.24 Parcela Volante 24

Resumen Estadístico

	Alturas	ZMax
Frecuencia	31	149
Media	20,1903	20,225
Varianza	68,1442	31,8449
Desviación típica	8,25495	5,64313
Mínimo	2,6	1,53796
Máximo	27,4	26,8911
Rango	24,8	25,3531
Asimetría tipi.	-3,11584	-7,95701
Curtosis típificada	0,492111	4,90831

Tabla 27. Resumen Estadístico Parcela 24.



Histograma

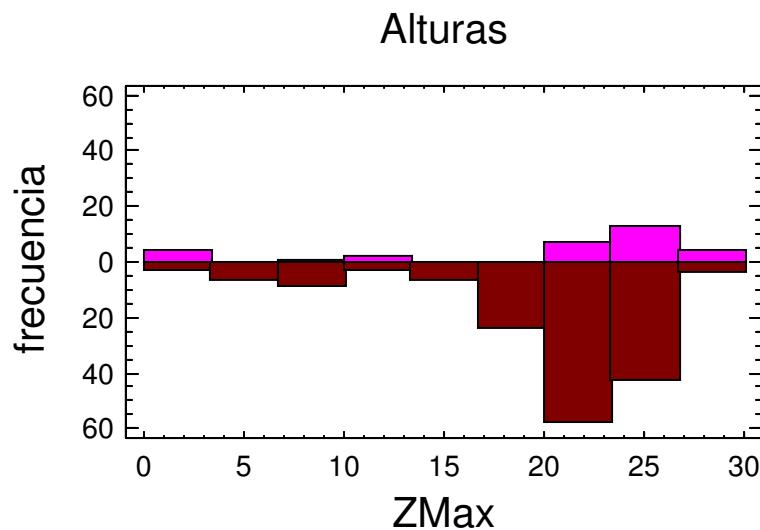


Figura 86 A. Histograma. Parcela 24.

Densidades Suavizadas

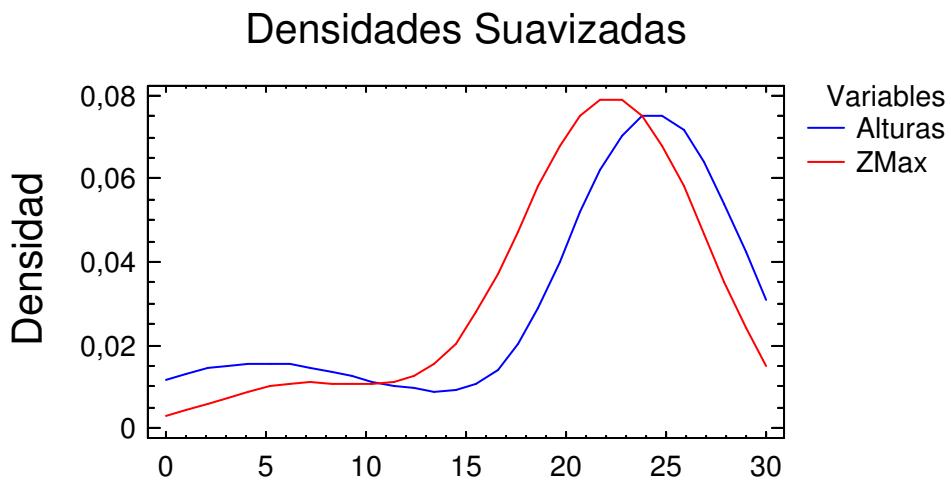


Figura 86 B. Densidades Suavizadas. Parcela 24.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

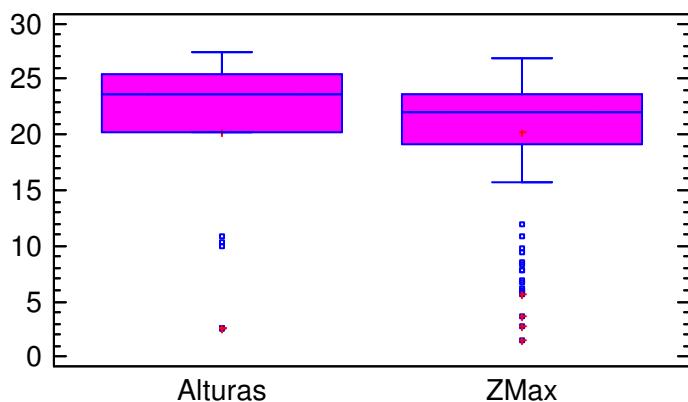


Figura 86 C. Gráfico de Cajas y Bigotes. Parcela 24.



10.25 Parcela Volante 25

Resumen Estadístico

	Alturas	ZMax
Frecuencia	31	413
Media	27,4161	21,2181
Varianza	19,0501	79,6639
Desviación típica	4,36464	8,92546
Mínimo	12,9	0,00008
Máximo	34,9	29,5815
Rango	22,0	29,5814
Asimetría tipi.	-2,34043	-14,9359
Curtosis típificada	3,21178	6,69235

Tabla 28. Resumen Estadístico Parcela 25.



Histograma

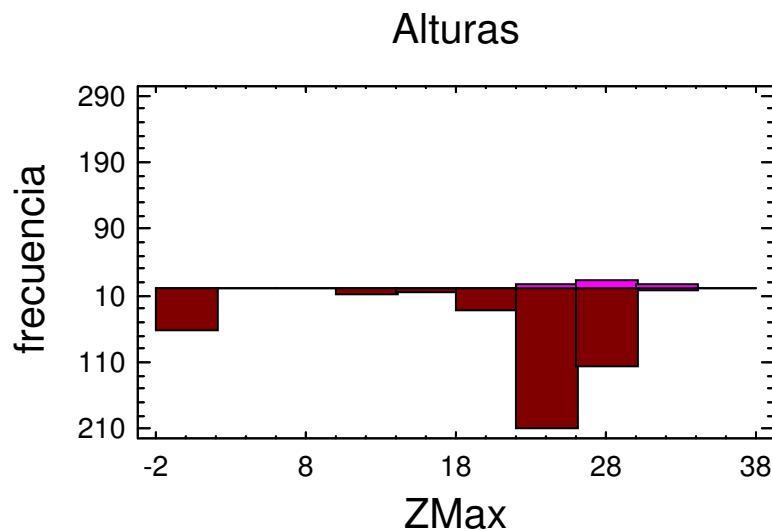


Figura 87 A. Histograma. Parcela 25.

Densidades Suavizadas

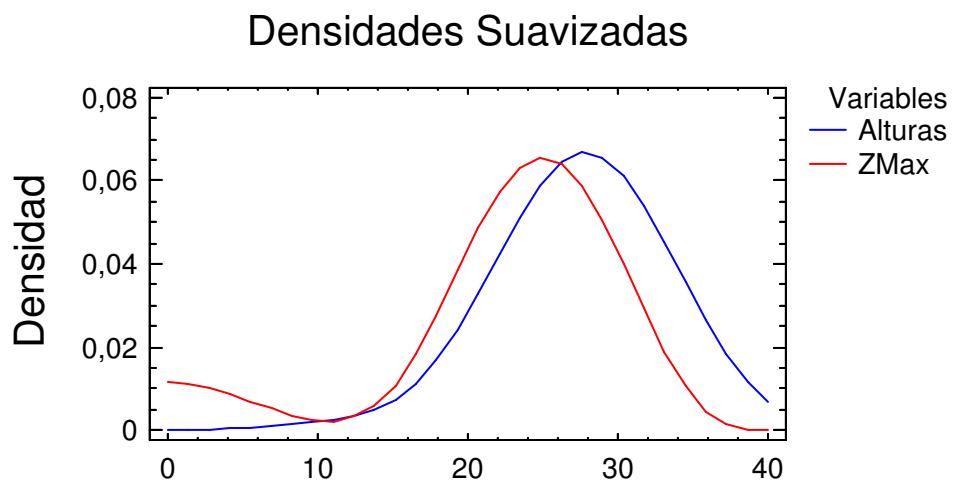


Figura 87 B. Densidades Suavizadas. Parcela 25.



Gráfico de Cajas y Bigotes

Gráfico de Cajas y Bigotes

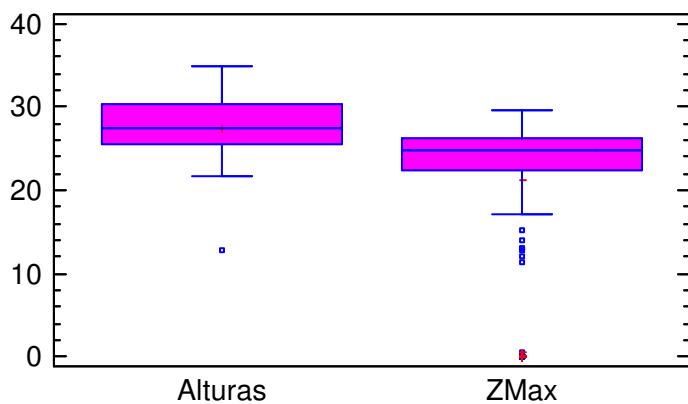


Figura 87 C. Gráfico de Cajas y Bigotes. Parcela 25.



11 Anexo B: Modelos de regresión para la calibración de las predicciones

Regresión simple - RealAltura frente a Algoritmo_Altura

Análisis de Regresión - Modelo Lineal $Y = a + b*X$

Variable dependiente: RealAltura

Variable independiente: Algoritmo_Altura

Error Estadístico

Parámetro	Estimación	estándar	T	P-Valor
Ordenada	-10.3518	2.98702	-3.46559	0.0021
Pendiente	1.51215	0.154174	9.80808	0.0000

Tabla 29 A. Análisis de Regresión- Modelo lineal. Altura-Real frente Altura algorítmico



Análisis de la Varianza

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	1147.8	1	1147.8	96.20	0.0000
Residuo	274.425	23	11.9315		
Total (Corr.)	1422.22	24			

Tabla 29 B. Análisis de la varianza.

Coeficiente de Correlación = 0.898357

R-cuadrado = 80.7045 porcentaje

R-cuadrado (ajustado para g.l.) = 79.8655 porcentaje

Error estándar de est. = 3.4542

Error absoluto medio = 2.33043

Estadístico de Durbin-Watson = 1.9762 (P=0.4242)

Autocorrelación residual en Lag 1 = -0.0470256

Nota: La salida muestra los resultados del ajuste al modelo lineal para describir la relación entre RealAltura y Algoritmo_Altura. La ecuación del modelo ajustado es:

$$\text{RealAltura} = -10.3518 + 1.51215 * \text{Algoritmo_Altura}$$

Dado que el p-valor en la tabla 29 A es inferior a 0.01, existe relación estadísticamente significativa entre RealAltura y Algoritmo_Altura para un nivel de confianza del 99%.



El estadístico R-cuadrado indica que el modelo explica un 80.7045% de la variabilidad en RealAltura. El coeficiente de correlación es igual a 0.898357, indicando una relación moderadamente fuerte entre las variables. El error estándar de la estimación muestra la desviación típica de los residuos que es 3.4542. Este valor puede usarse para construir límites de la predicción para las nuevas observaciones seleccionando la opción Predicciones del menú del texto.

El error absoluto medio (MAE) de 2.33043 es el valor medio de los residuos. El estadístico Durbin-Watson (DW) examina los residuos para determinar si hay alguna correlación significativa basada en el orden en el que se han introducido los datos en el fichero. Dado que el p-valor es superior a 0.05, no hay indicio de autocorrelación serial en los residuos.

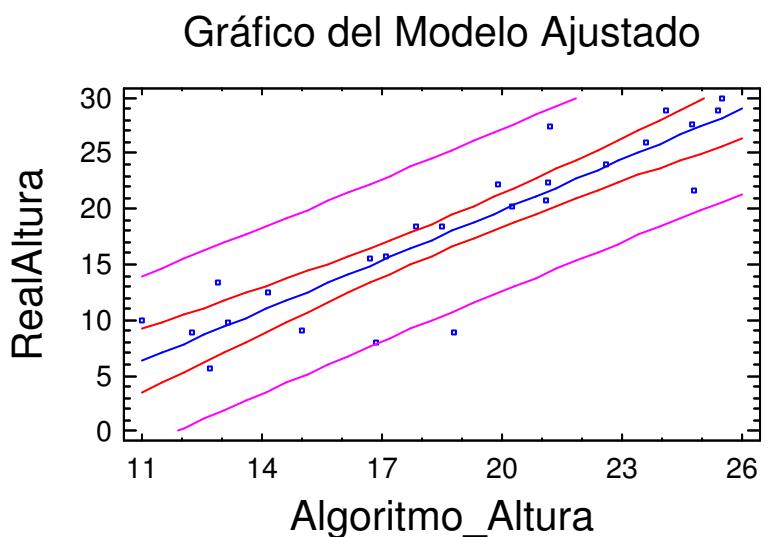


Figura 88 A. Gráfico del modelo ajustado. Altura Real / Altura algoritmo



Análisis de Varianza con Falta de ajuste

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	1147.8	1	1147.8	96.20	0.0000
Residuo	274.425	23	11.9315		
Falta de ajuste	274.425	23	11.9315		
Error puro	0.0	0			
Total (Corr.)	1422.22	24			

Tabla 29 C. Análisis de varianza con falta de ajuste

Nota: El test de Falta de Ajuste está diseñado para determinar si el modelo seleccionado es adecuado para describir los datos observados, o si se debería utilizar un modelo más complicado. El test se realiza comparando la variabilidad de los residuos del modelo actual con la variabilidad entre las observaciones a valores duplicados de la variable independiente X. Desgraciadamente, el test no se puede ejecutar en este caso porque no hay observaciones duplicadas a los mismos valores de Algoritmo_Altura.

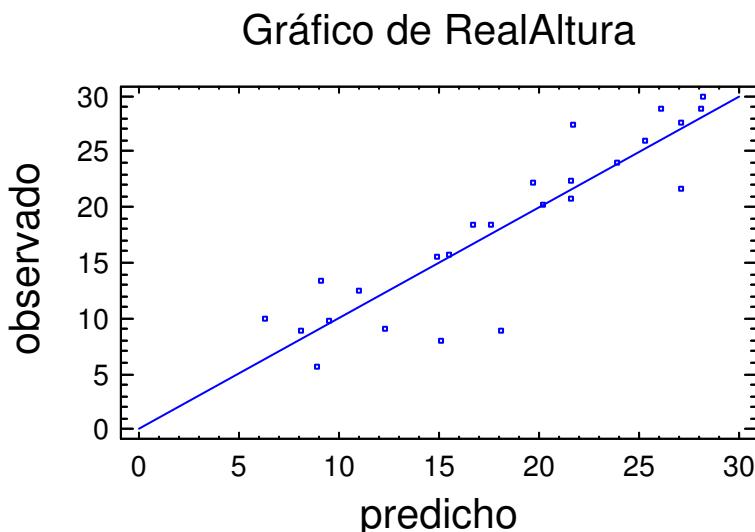


Figura 88 B. Gráfico de Altura Real. Observado / predicho.

Comparación de Modelos Alternativos

Modelo	Correlación	R-cuadrado
Lineal	0.8984	80.70%
Raiz cuadrada-X	0.8910	79.40%
Raiz cuadrada-Y	0.8849	78.31%
Logarítmico-X	0.8804	77.51%
Exponencial	0.8602	74.00%
Multiplicativo	0.8526	72.70%
Inverso-X	-0.8497	72.20%
curva-S	-0.8319	69.21%
Inverso-Y	-0.7762	60.25%
Doble inverso	0.7667	58.78%



Logístico	<sin ajuste>
Log Probit	<sin ajuste>

Figura 29 D. Comparación de modelos alternativos.

Nota: Esta tabla muestra los resultados de ajuste a los datos de varios modelos curvilíneos. De los modelos ajustados, el modelo lineal procura el valor de R-cuadrado más alto con 80.7045%. Este es el modelo seleccionado actualmente.

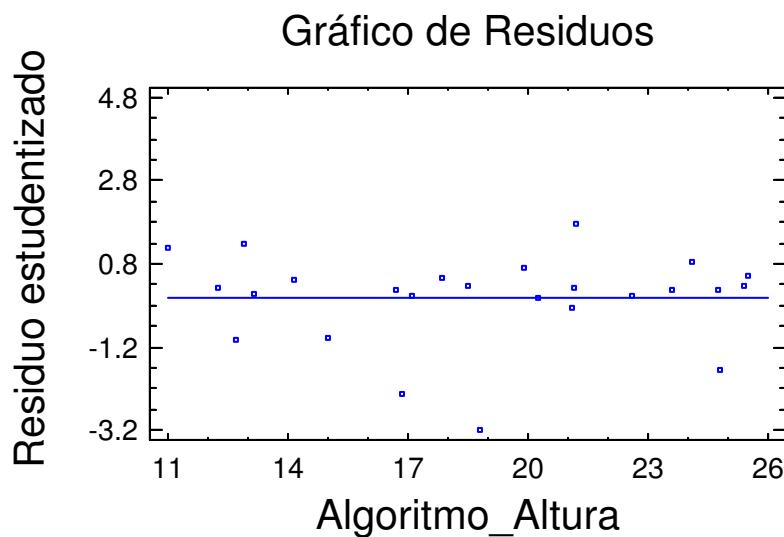


Figura 88 C. Gráfico de Residuos. Residuo estudiantizado /Altura algoritmo.

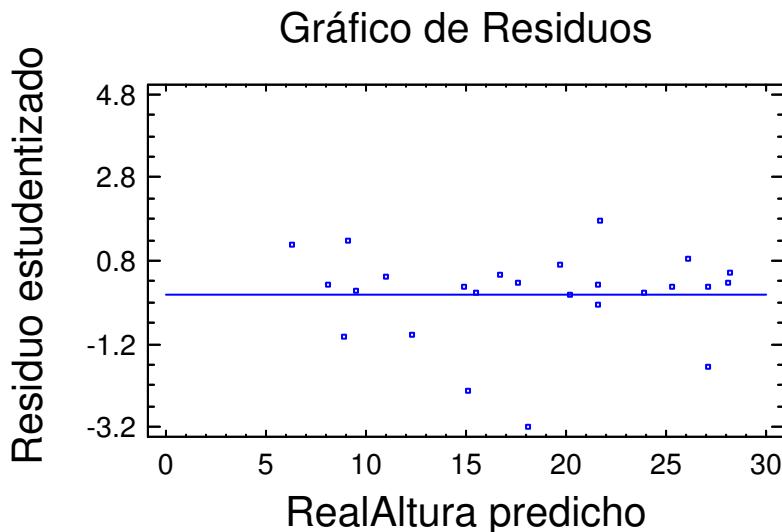


Figura 88 D. Gráfico de Residuos. Residuo estudiantizado / Altura Real predicha.

Regresión simple - RealBT frente a PilarBT

Análisis de Regresión - Modelo Lineal $Y = a + b*X$

Variable dependiente: RealBT

Variable independiente: PilarBT

Parámetro	Estimación	Error Estadístico		
		estándar	T	P-Valor
Ordenada	17907.9	2005.9	8.9276	0.0000
Pendiente	0.099483	0.0241453	4.12019	0.0004

Tabla 30 A. Análisis de Regresión- Modelo lineal. Real BT frente a Algoritmo BT



Análisis de la Varianza

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	6.82831E8	1	6.82831E8	16.98	0.0004
Residuo	9.2514E8	23	4.02235E7		
Total (Corr.)	1.60797E9	24			

Tabla 30 B. Análisis de la Varianza.

Coeficiente de Correlación = 0.651655

R-cuadrado = 42.4654 porcentaje

R-cuadrado (ajustado para g.l.) = 39.9639 porcentaje

Error estándar de est. = 6342.2

Error absoluto medio = 5326.47

Estadístico de Durbin-Watson = 1.64759 (P=0.1593)

Autocorrelación residual en Lag 1 = 0.136558

Nota: La salida muestra los resultados del ajuste al modelo lineal para describir la relación entre RealBT y PilarBT. La ecuación del modelo ajustado es:

$$\text{RealBT} = 17907.9 + 0.099483 * \text{PilarBT}$$

Dado que el p-valor en la tabla ANOVA es inferior a 0.01, existe relación estadísticamente significativa entre RealBT y PilarBT para un nivel de confianza del 99%.



El estadístico R-cuadrado indica que el modelo explica un 42.4654% de la variabilidad en RealBT. El coeficiente de correlación es igual a 0.651655, indicando una relación moderadamente fuerte entre las variables. El error estándar de la estimación muestra la desviación típica de los residuos que es 6342.2. Este valor puede usarse para construir límites de la predicción para las nuevas observaciones seleccionando la opción Predicciones del menú del texto.

El error absoluto medio (MAE) de 5326.47 es el valor medio de los residuos. El estadístico Durbin-Watson (DW) examina los residuos para determinar si hay alguna correlación significativa basada en el orden en el que se han introducido los datos en el fichero. Dado que el p-valor es superior a 0.05, no hay indicio de autocorrelación serial en los residuos.

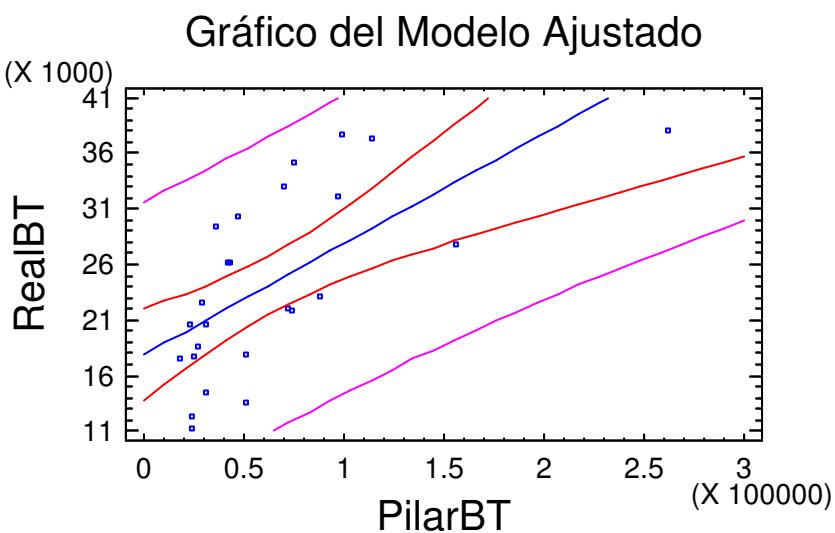


Figura 89 A. Gráfico del modelo ajustado. Real BT / BT algoritmo



Análisis de Varianza con Falta de ajuste

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	6.82831E8	1	6.82831E8	16.98	0.0004
Residuo	9.2514E8	23	4.02235E7		
Falta de ajuste	9.2514E8	23	4.02235E7		
Error puro	0.0	0			
Total (Corr.)	1.60797E9	24			

Tabla 30 C. Análisis de Varianza con Falta de ajuste

Nota: El test de Falta de Ajuste está diseñado para determinar si el modelo seleccionado es adecuado para describir los datos observados, o si se debería utilizar un modelo más complicado. El test se realiza comparando la variabilidad de los residuos del modelo actual con la variabilidad entre las observaciones a valores duplicados de la variable independiente X. Desgraciadamente, el test no se puede ejecutar en este caso porque no hay observaciones duplicadas a los mismos valores de PilarBT.

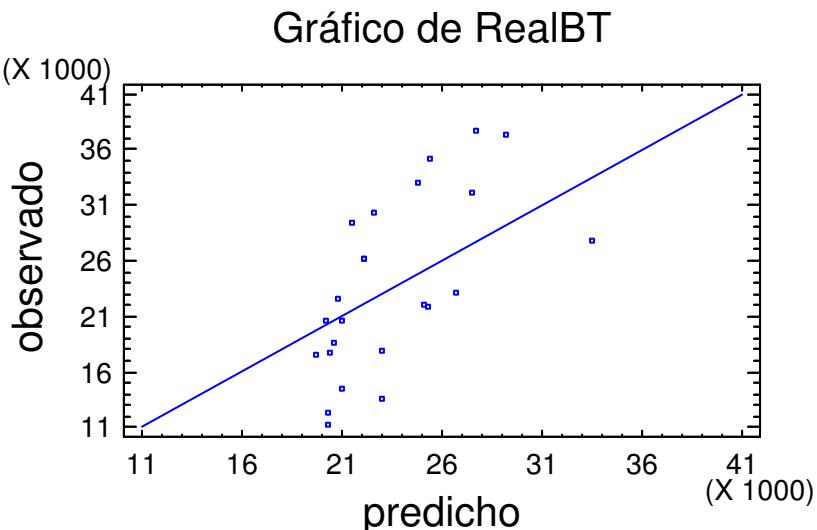


Figura 89 B. Gráfico de Real BT. observado / predicho.

Valores predichos

95.00% Límites de Predicción Límites de Confianza					
Predicho	Y	Inferior	Superior	Inferior	Superior
X					
18200.5	19718.5	6141.63	33295.4	16225.6	23211.4
261896.0	43962.1	27337.8	60586.4	33752.4	54171.8

Tabla 30 D. Valores predichos

Nota: Esta tabla muestra los valores predichos para RealBT usando el modelo ajustado. Además de las mejores predicciones, la tabla muestra:

- (1) 95.0% intervalos de predicción para las nuevas observaciones
- (2) 95.0% intervalos de confianza para la media de muchas observaciones

Los intervalos de predicción y confianza correspondientes a los límites internos y externos en el gráfico del modelo ajustado.

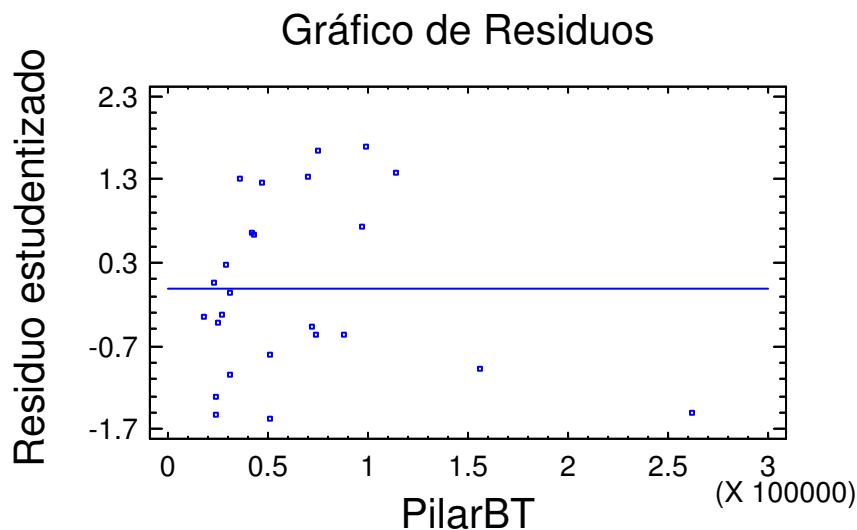


Figura 89 C. Gráfico de Residuos. Residuo estudiantizado / BT algoritmo

Comparación de Modelos Alternativos

Modelo	Correlación	R-cuadrado
Logarítmico-X	0.7313	53.48%
Raiz cuadrada-X	0.7063	49.88%
Inverso-X	-0.7056	49.79%
Multiplicativo	0.7006	49.09%
curva-S	-0.6921	47.90%
Doble inverso	0.6541	42.79%
Lineal	0.6517	42.47%
Raiz cuadrada-Y	0.6305	39.76%
Exponencial	0.6048	36.58%
Inverso-Y	-0.5418	29.36%
Logístico	<sin ajuste>	



Log Probit

<sin ajuste>

Tabla 30 E. Comparación de modelos alternativos

Nota: Esta tabla muestra los resultados de ajuste a los datos de varios modelos curvilíneos. De los modelos ajustados, el modelo logarítmico-X procura el valor de R-cuadrado más alto con 53.4766%. Este es 11.0112% más alto que el modelo lineal seleccionado actualmente. Para cambiar los modelos, seleccione el cuadro de diálogo Opciones del Análisis.

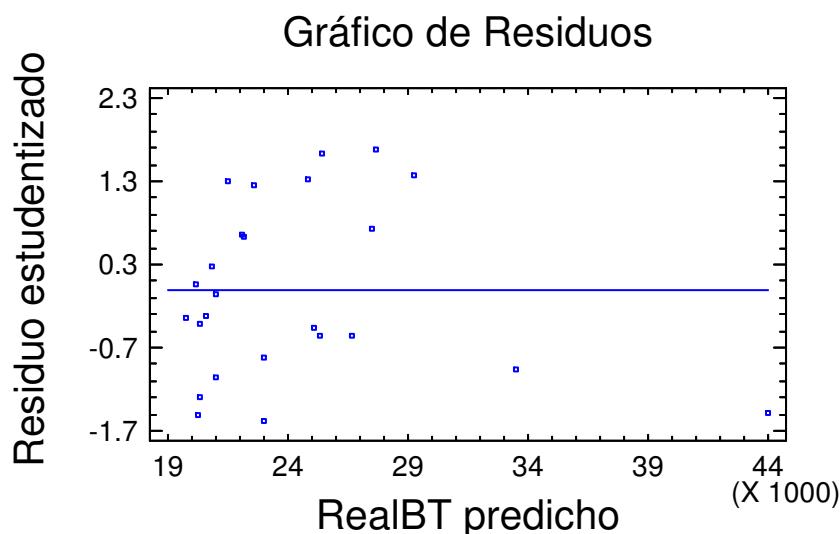


Figura 89 D. Gráfico de Residuos. Residuo estudiantizado / BT Real predicho.

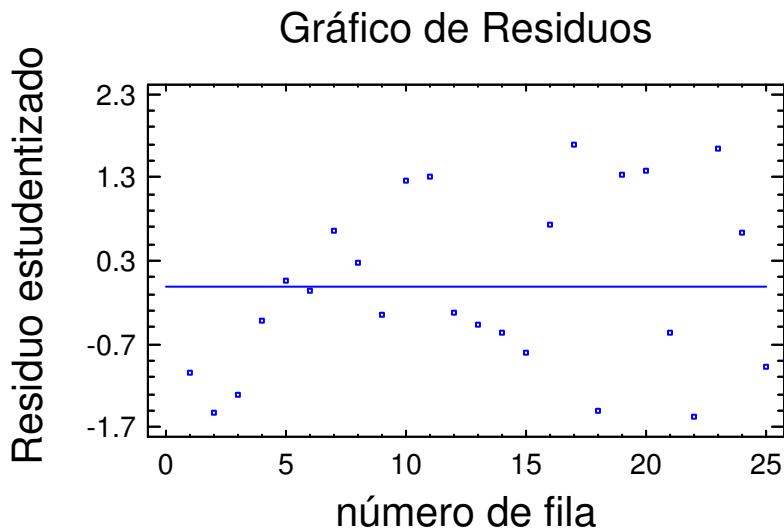


Figura 89 E. Gráfico de Residuos. Residuo estudiantizado / número de fila.

Puntos Influyentes

Fila	X	Y	Residuo		Influencia
			Predicha	Estudiantizado	
18	261896.0	38146.2	43962.1	-1.50	0.605574

Influencia media de un punto = 0.08

Tabla 30 F. Puntos influyentes

Nota: La tabla de puntos influyentes lista todas las observaciones que tienen valores de influencia superior a 3 veces la del punto medio de los datos. Leverage es un estadístico que mide la influencia de cada observación en la determinación de los coeficientes del modelo estimado. En este caso, un punto medio tendría un valor de influencia igual a 0.08. Hay un punto superior a 5 veces la influencia media. Debería



examinar cuidadosamente los puntos superiores a 5 veces la influencia media y determinar cuánto cambiaría el modelo si estos se eliminaran.

Regresión simple - RealN frente a PilarN

Análisis de Regresión - Modelo Lineal $Y = a + b*X$

Variable dependiente: RealN

Variable independiente: PilarN

	Error	Estadístico		
Parámetro	Estimación	estándar	T	P-Valor
Ordenada	79.6063	34.1663	2.32997	0.0289
Pendiente	0.00769726	0.145752	0.0528107	0.9583

Tabla 31 A. Error estadístico. Real N frente a algoritmo N

Análisis de la Varianza

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	11.3605	1	11.3605	0.00	0.9583
Residuo	93687.7	23	4073.38		
Total (Corr.)	93699.0	24			

Tabla 31 B. Análisis de la Varianza



Coeficiente de Correlación = 0.0110111

R-cuadrado = 0.0121245 porcentaje

R-cuadrado (ajustado para g.l.) = -4.33517 porcentaje

Error estándar de est. = 63.823

Error absoluto medio = 53.319

Estadístico de Durbin-Watson = 1.28839 (P=0.0303)

Autocorrelación residual en Lag 1 = 0.313306

Nota: La salida muestra los resultados del ajuste al modelo lineal para describir la relación entre RealN y PilarN. La ecuación del modelo ajustado es:

$$\text{RealN} = 79.6063 + 0.00769726 * \text{PilarN}$$

Dado que el p-valor en la tabla ANOVA es mayor o igual a 0.01, no existe relación estadísticamente significativa entre RealN y PilarN para un nivel de confianza del 90% o superior.

El estadístico R-cuadrado indica que el modelo explica un 0.0121245% de la variabilidad en RealN. El coeficiente de correlación es igual a 0.0110111, indicando una relación relativamente débil entre las variables. El error estándar de la estimación muestra la desviación típica de los residuos que es 63.823. Este valor puede usarse para construir límites de la predicción para las nuevas observaciones seleccionando la opción Predicciones del menú del texto.

El error absoluto medio (MAE) de 53.319 es el valor medio de los residuos. El estadístico Durbin-Watson (DW) examina los residuos para determinar si hay alguna correlación significativa basada en el orden en el que se han introducido los datos en el fichero. Dado que el p-valor es inferior a 0.05, hay indicio de una posible correlación serial. Represente los residuos frente al orden de fila para ver si hay algún modelo que pueda verse.

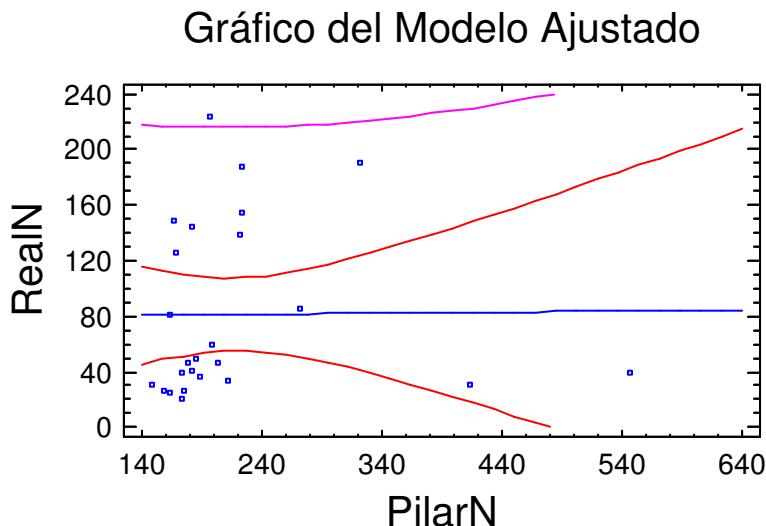


Figura 90 A. Gráfico del Modelo Ajustado. Real N / N algoritmo

Análisis de Varianza con Falta de ajuste

Fuente	Suma de cuadrados	GL	Cuadrado medio	Cociente-F	P-Valor
Modelo	11.3605	1	11.3605	0.00	0.9583
Residuo	93687.7	23	4073.38		
Falta de ajuste	86108.7	19	4532.04	2.39	0.2065
Error puro	7579.0	4	1894.75		
Total (Corr.)	93699.0	24			

Tabla 31 C. Análisis de Varianza con Falta de ajuste.

Nota: El test de Falta de Ajuste está diseñado para determinar si el modelo seleccionado es adecuado para describir los datos observados, o si se debería utilizar un modelo más complicado. El test se realiza comparando la variabilidad de los



residuos del modelo actual con la variabilidad entre las observaciones a valores duplicados de la variable independiente X. Dado que el p-valor para la Falta de Ajuste en la tabla ANOVA es mayor o igual a 0.10, el modelo parece ser adecuado para los datos observados.

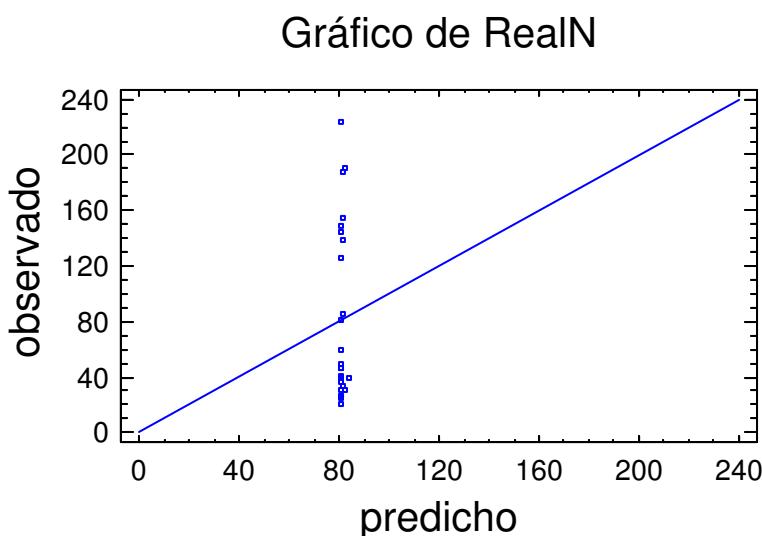


Figura 90 B. Gráfico de Real N. observado / predicho



Valores predichos

		95.00%		95.00%	
		Predicho	Límites de Predicción	Límites de Confianza	
X	Y	Inferior	Superior	Inferior	Superior
18200.5	219.7	-5204.06	5643.46	-5202.45	5641.85
261896.0	2095.49	-76803.8	80994.8	-76803.7	80994.7

Tabla 31 D. Valores predichos

Nota: Esta tabla muestra los valores predichos para RealN usando el modelo ajustado. Además de las mejores predicciones, la tabla muestra:

- (1) 95.0% intervalos de predicción para las nuevas observaciones
- (2) 95.0% intervalos de confianza para la media de muchas observaciones

Los intervalos de predicción y confianza correspondientes a los límites internos y externos en el gráfico del modelo ajustado.

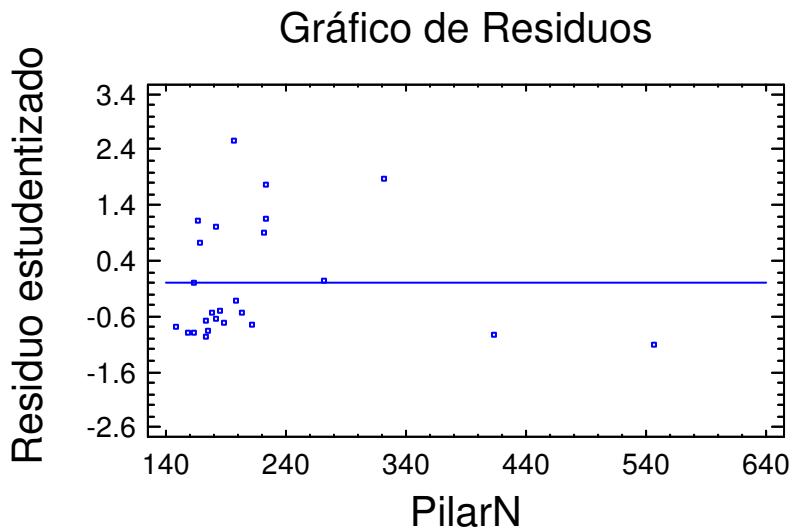


Figura 90 C. Gráfico de Residuos. Residuo estudiantizado / N algoritmo

Comparación de Modelos Alternativos

Modelo	Correlación	R-cuadrado
Doble inverso	0.2111	4.46%
curva-S	-0.1835	3.37%
Inverso-X	-0.1720	2.96%
Multiplicativo	0.1011	1.02%
Logarítmico-X	0.0933	0.87%
Raiz cuadrada-X	0.0515	0.26%
Inverso-Y	-0.0486	0.24%
Exponencial	0.0167	0.03%
Lineal	0.0110	0.01%
Raiz cuadrada-Y	0.0108	0.01%



Logístico	<sin ajuste>
Log Probit	<sin ajuste>

Tabla 31 E. Comparación de modelos alternativos

Nota: Esta tabla muestra los resultados de ajuste a los datos de varios modelos curvilíneos. De los modelos ajustados, el modelo doble recíproco procura el valor de R-cuadrado más alto con 4.45538%. Este es 4.44326% más alto que el modelo lineal seleccionado actualmente.

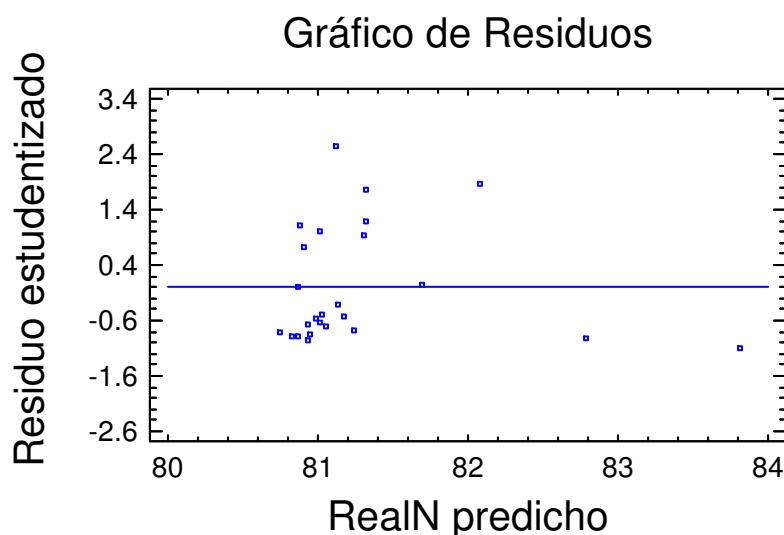


Figura 90 D. Gráfico de Residuos. Residuo estudiantizado / Real N predicho.

Residuos Atípicos

Fila	X	Y	Y	Residuo	
			Predicha	Residuo Estudentizado	
4	197.0	224.0	81.1227	142.877	2.55

Tabla 31 F. Residuos atípicos



Nota: La tabla de residuos atípicos lista todas las observaciones que tienen residuos estandarizados superiores a 2.0 en valor absoluto. Los residuos estandarizados miden cuánta desviación típica de cada valor observado de RealN se desvía del modelo de ajuste, utilizando todos los datos excepto esa observación. En este caso hay un residuo estandarizado superior a 2.0, pero ninguno superior a 3.0.

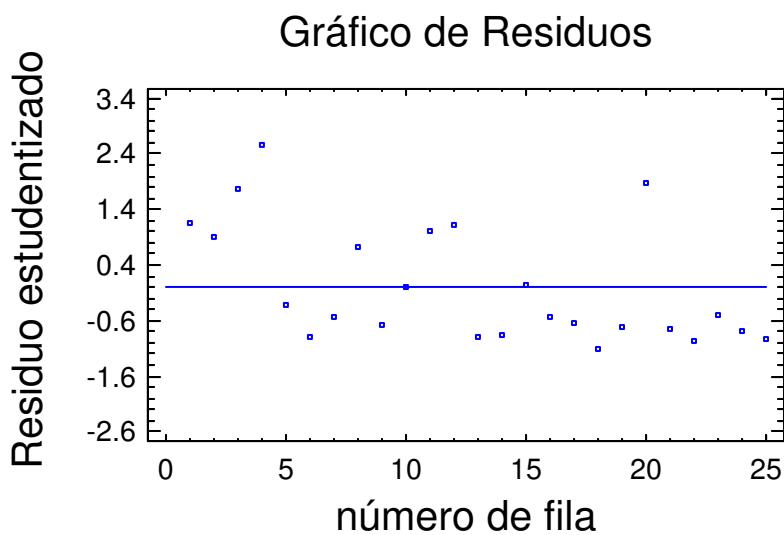


Figura 90 E. Gráfico de Residuos. Residuo estandarizado / número de fila.

Puntos Influyentes

Fila	X	Y	Y	Residuo	Influencia
			Predicha	Estandarizado	
18	546.0	40.0	83.809	-1.09	0.602993

Influencia media de un punto = 0.08

Tabla 31 G. Tabla de puntos Influyentes



Nota: La tabla de puntos influyentes lista todas las observaciones que tienen valores de influencia superior a 3 veces la del punto medio de los datos. Leverage es un estadístico que mide la influencia de cada observación en la determinación de los coeficientes del modelo estimado. En este caso, un punto medio tendría un valor de influencia igual a 0.08. Hay un punto superior a 5 veces la influencia media. Debería examinar cuidadosamente los puntos superiores a 5 veces la influencia media y determinar cuánto cambiaría el modelo si estos se eliminaran.



12 Bibliografía

- [1] Alexander C, Tansey K, Tate N J., Smith-Voysey S and Kaduk J. Extraction of vegetation for topographic mapping from full-waveform airborne laser scanning data.
- [2] Brandtberg T, Warner T A., Landenberger R E., McGraw James B. Detection and analysis of individual leaf-off tree crowns in small footprint, high sampling density lidar data from the eastern deciduous forest in North America.
- [3] Andersen, H-E, Reutebuch, S., Schreuder, G., 2002. Bayesian Object Recognition for the Analysis of Complex Forest Scenes in Airborne Laser Scanner Data. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol 34, part 3A, pp- 35-41
- [4] FUSION/LDV: Software for LIDAR Data Analysis and Visualization. Robert Mc Gaughey. U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station, University of Washington, Box 352100, Seattle, WA 98195-2100. Mayo 2009
- [5] García-Gutiérrez J., Martínez-Álvarz F., Laguna-Ruiz D. and Riquelme J. C. Remote mining: from clustering to DTM.
- [6] Gaulton R & Malthus T J. LiDAR mapping of canopy gaps in continuous cover forest; a comparison of canopy height model and point cloud based techniques.
- [7] Gougeon, F., Leckie, D., 2003. Forest information extractionfrom high spatial resolution images using an individual tree crown approach. Information report BC-X-396, Natural Resources Canada, Canadian Forest Service, Pacific Forestry Centre, 26 p.
- [8] Guijarro Mata-García María. Combinación de clasificadores para identificación de texturas en imágenes naturales: nuevas estrategias locales y globales. Tesis.
- [9] Hyppä J. Feasibility for estimation of single tree characteristics using laser scanner.
- [10] Hyppä J, Hyppä H, Litkey P, Yu X, Haggren H, Rönnholm P, Pyysalo U, Pitkänen J and Maltamo M. Algoritms and Methods of airborne laser scanning for forest measurements.



- [11] Hyppä, J., Inkinen, M. 1999. Detecting and estimating attributes for single trees using laser scanner. Photogrammetric Journal of Finland 1999; 16(2):27-42.
- [12] Kraus, K., Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS J. Photogramm. Remote Sens. 53,193–203.
- [13] Leckie, D., Gougeon, F., Hill, D., Quinn, R., Armstrong, L., Shreenan, R., 2003. Combined high-density lidar and multispectral imagery for individual tree crown analysis. Canadian Journal of Remote Sensing 29, No. 5, pp. 633–649.
- [14] Morsdorf Felix, Meier Erich, Allgöwer Brita and Nüesch Daniel. Clustering in airborne laser scanning raw data for segmentation of single trees.
- [15] Naesset E, J. T. .1 August 2002. Assessing Point Accuracy of DGPS Under Forest Canopy Before Data Acquisition, in the Field and After Postprocessing. Scandinavian Journal of Forest Research, 17(8):351-358.
- [16] Nilsson, M. .1996. Estimation of tree heights and stand volume using an airborne lidar system. Remote Sensing of Environment, 56(1):1-7. Available from <http://www.sciencedirect.com/science/article/B6V6V-3VVT36S-1/2/354e40a2c7e134d5a37e72432e2efae5> [accesed 20 August 2009]
- [17] Palomino, P; Garmendia, L; Mauro, FAlgoritmo para la estimación de la masa forestal, cálculo de características de árboles individuales partiendo de imágenes LiDAR. Junio 2009.
- [18] Persson., Å, Holmgren, J., Söderman, U., 2002. Detecting and measuring individual trees using an airborne laser scanner. Photogrammetric Engineering and Remote Sensing 68, No 9, pp.925-932.
- [19] Popescu, S., Wynne, R., Nelson, R., 2003. Measuring individual tree crown diameter with lidar and assessing its influence on estimating forest volume and biomass. Canadian Journal of Remote Sensing. 29, No. 5, pp. 564–577
- [20] Straub. B., 2003. A top-down operator for the automatic extraction of trees - concept and performance evaluation. Proceedings of the ISPRS working group III/3 workshop '3-D reconstruction from airborne laserscanner and InSAR data' Dresden, Germany 8-10 October 2003, pp.34-39.
- [21] Wack, R., Schardt, Lohr, U., Barrucho, L., Oliveira, T., 2003. Forest inventory for Eucalyptus plantations based on airborne laser scanner data. International Archives of



the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol 34, part 3/W13, pp. 40-46.

[22] Wang Y, Weinacker H, Koch B, Sterenczak. A Lidar Point Cloud Based fully automatic 3D Single Tree Modelling in Forest and evaluations of the procedure.

[23] Wang Y, Weinacker H and Koch B and Sterenczak .A Lidar Point Cloud Based Procedure for Vertical Canopy Structure Analysis and 3D Single Tree modelling in Forest. Sensors 2008.

[24] Zadeh, L.A.: Fuzzy sets. Proceedings of the IEEE (Information and Control), Vol:8: 338-353, 1965

[25] <http://www.directionsmag.es/articles/an%C3%A1lisis-de-datos-lidar-sobre-gvsig:-dielmoopenlidar/16>. Dielmo OpenLidar.

[26] <http://es.wikipedia.org>

[27] R Development Core Team (2008). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.