

**ANÁLISIS DE DATOS APLICADO A IMÁGENES SATELITALES PARA CALCULAR  
LA DEFORESTACIÓN**

**HENRY JESÚS SOLARTE FAJARDO**

**FACULTAD DE INGENIERÍA  
MAESTRÍA EN INGENIERÍA DE SISTEMAS  
UNIVERSIDAD POLITÉCNICO GRANCOLOMBIANO  
BOGOTÁ D.C.  
2018**

## CONTENIDO

	Pág.
INTRODUCCIÓN.....	8
1. PLANTEAMIENTO DEL PROBLEMA .....	9
2. MARCO TEÓRICO .....	10
2.1 FUNDAMENTOS DE LA TELEDETECCIÓN .....	10
2.2 LANDSAT.....	11
2.2.1 Sensores Satelites Landsat .....	13
2.2.1.1 Bandas de Landsat 4 .....	13
2.2.1.2 Bandas de Landsat 5 .....	13
2.3 BANDAS DE LANDSAT .....	14
2.3.1 Banda 1 .....	<b>¡Error! Marcador no definido.</b>
2.3.2 Banda 2 .....	<b>¡Error! Marcador no definido.</b>
2.3.3 Banda 3 .....	<b>¡Error! Marcador no definido.</b>
2.3.4 Banda 4 .....	<b>¡Error! Marcador no definido.</b>
2.3.5 Banda 5 .....	<b>¡Error! Marcador no definido.</b>
2.3.6 Banda 6 .....	<b>¡Error! Marcador no definido.</b>
2.3.7 Banda 7 .....	<b>¡Error! Marcador no definido.</b>
2.3.8 Banda 8 .....	<b>¡Error! Marcador no definido.</b>
2.3.9 Banda 9 .....	<b>¡Error! Marcador no definido.</b>
2.3.10 Bandas 10 y 11.....	<b>¡Error! Marcador no definido.</b>
2.4 ANÁLISIS DE IMÁGENES .....	15
3. METODOLOGÍA.....	17
3.1 MATERIALES .....	17
3.2 DIAGRAMA DE FLUJO .....	19
4. RESULTADO .....	20
4.1 SELECCIÓN Y DESCARGA DE IMÁGENES .....	20
4.1.1. Preprocesamiento de imágenes .....	25
4.1.2. Apilamiento de Bandas .....	25
4.1.3. Tamaño de imágenes descargadas de Landsat .....	27
4.1.4. Errores de imágenes Landsat .....	30
4.1.5. Combinación de Bandas LANDSAT 8.....	33
4.1.5. Imágenes Ideam utilizadas para el modelo .....	35
4.1.6 Importación de imágenes a GRASS GIS .....	38
4.2 EXTRACCIÓN DE CARACTERÍSTICAS .....	40
4.2.1 Extracción de características imagen IDEAM .....	44

4.3 MINERÍA DE DATOS.....	47
5. CONCLUSIONES.....	<b>¡Error! Marcador no definido.</b>
REFERENCIAS .....	75

## LISTA DE FIGURAS

	Pág.
Fig. 2-1. Componentes de la Teledetección.....	10
Fig. 2-2. Satélite soviético Sputnik-1, réplica en el Museo de los Conquistadores del Espacio, Moscú, Rusia. Satélite artificial Sputnik. ....	11
Fig. 2-3. Landsat 8 en órbita. Fuente: NASA y OSC.....	12
Fig. 2-4. Model of Landsat 4 Satellite. ....	12
Fig. 3-1 Flujo de procesos en el desarrollo del estudio. ....	19
Fig. 4-1. Ficha metadatos Colombia. ....	21
Fig. 4-2. Búsqueda por fechas. ....	21
Fig. 4-3. Data Set Search. ....	22
Fig. 4-4. Search Results. ....	22
Fig. 4-5. Imagen Descargada. ....	23
Fig. 4-6. Ejemplo de Banda 1. ....	23
Fig. 4-7. Metadata. ....	23
Fig. 4-8. Apilamiento de Bandas imagen propia. ....	25
Fig. 4-9. Recolección de bandas figura propia. ....	26
Fig. 4-10. Layer. ....	26
Fig. 4-11, Reunión de bandas. ....	27
Fig. 4-12. Coordenadas.....	27
Fig. 4-13. Subset. ....	28
Fig. 4-14. Recorte de imagen imagen propia. ....	29
Fig. 4-15. Subset figura propia. ....	29
Fig. 4-16. Rescale imagen propia. ....	30
Fig. 4-17. Gapfill. ....	30
Fig. 4-18. Arreglo Gapfill. ....	31
Fig. 4-19. Arreglo Gapfill. ....	31
Fig. 4-20 Resultados Gapfill. ....	32
Fig. 4-21. Resultados Gapfill. ....	32
Fig. 4-22. Color natural. ....	33
Fig. 4-23. Falso color. ....	33
Fig. 4-24. Color infrarojo. ....	33
Fig. 4-25. Agricultura. ....	33
Fig. 4-26. Análisis de vegetación. ....	33
Fig. 4-27. Penetración atmosférica. ....	33
Fig. 4-28. Vegetación saludable. ....	34
Fig. 4-29. Agua. ....	34
Fig. 4-30. Eliminación atmosférica. ....	34
Fig. 4-31. Infrarojo onda corta. ....	34
Fig. 4-32. Imagen facilitada por IDEAM. ....	35
Fig. 4-33. Reproject. ....	36
Fig. 4-34. Proyección Espacial. ....	36
Fig. 4-35. Set Output Corners. ....	37
Fig. 4-36. Tamaño del Pixel. ....	37
Fig. 4-37. Valores imágenes IDEAM. ....	37
Fig. 4-38. Listado de Imágenes. ....	39
Fig. 4-39. Imágenes deforestación. ....	39
Fig. 4-40. EXECGRASS. ....	40
Fig. 4-41. READRAST. ....	40
Fig. 4-42. Resumen Ráster. ....	41
Fig. 4-43. Extracción final. ....	43
Fig. 4-44. Extracción de características. ....	43

Fig. 4-45. Datos de las imagenes. ....	43
Fig. 4-46. Valores. ....	45
Fig. 4-47. Imágenes deforestación. ....	46
Fig. 4-48. Imágenes de estudio. ....	46
Fig. 4-49. Imágenes deforestación y estudio. ....	46
Fig. 4-50. Comando para incluir deforestación. ....	47
Fig. 4-51. Inserción de deforestación en imagen. ....	47
Fig. 4-52. Datos totales. ....	51
Fig. 4-53. Datos de deforestación. ....	51
Fig. 4-54. Datos de no deforestación. ....	52
Fig. 4-55. Datos de deforestación y no deforestación. ....	52
Fig. 4-56. (70% Consolidado). ....	52
Fig. 4-57. (30 % consolidado). ....	53
Fig. 4-58. Consulta de datos 70% consolidado. ....	53
Fig. 4-59. Consulta de datos 30% consolidado. ....	53
Fig. 4-60. Finalización entrenamiento consolidado. ....	54
Fig. 4-61. Deforestación 1990. ....	55
Fig. 4-62. No deforestación 1990. ....	55
Fig. 4-63. Deforestación y no deforestación 1990. ....	56
Fig. 4-64. Deforestación 1990. ....	56
Fig. 4-65. No deforestación 1990. ....	56
Fig. 4-66. Resultados Esperados. ....	57
Fig. 4-67. Puntos de deforestación. ....	60
Fig. 4-68. Puntos sobrepuestos de no deforestación y deforestación. ....	60
Fig. 4-69. Puntos sobrepuestos de deforestación y no deforestación. ....	61
Fig. 4-70. Clasificación deseada. ....	61
Fig. 4-71. Clasificación obtenida. ....	61
Fig. 4-72. Bosque estable consolidado. ....	64
Fig. 4-73. Deforestación consolidado. ....	64
Fig. 4-74. Puntos de sin información consolidado. ....	65
Fig. 4-75. Puntos de regeneración consolidado. ....	65
Fig. 4-76. No bosque estable consolidado. ....	65
Fig. 4-77. Puntos totales consolidado. ....	66
Fig. 4-78. Resultados esperados. ....	66
Fig. 4-79. Resultados obtenidos. ....	66
Fig. 4-80. Deforestación PCA 1990. ....	68
Fig. 4-81. No deforestación 1990. ....	68
Fig. 4-82. No deforestación sobrepuesta deforestación PCA 1990. ....	68
Fig. 4-83. Clasificación deseada. ....	69
Fig. 4-84. Clasificación obtenida. ....	69
Fig. 4-85. Bosque estable PCA 1990. ....	70
Fig. 4-86. Deforestación PCA 1990. ....	70
Fig. 4-87. Sin información PCA 1990. ....	70
Fig. 4-88. Regeneración 1990. ....	71
Fig. 4-89. Todos los puntos PCA 1990. ....	71
Fig. 4-90. Resultados esperados. ....	71
Fig. 4-91. Resultados obtenidos. ....	72
Fig. 5-1. Deforestación y no deforestación. ....	<b>¡Error! Marcador no definido.</b>
Fig. 5-2. Puntos combinados. ....	<b>¡Error! Marcador no definido.</b>

## LISTA DE TABLAS

	Pág.
Tabla 1. Sensor MMS <i>Landsat</i> 4. ....	13
Tabla 2. Sensor TM <i>Landsat</i> 5. ....	13
Tabla 3. Sensor ETM + <i>Landsat</i> 7. ....	14
Tabla 4. Sensor OLI <i>Landsat</i> 8. ....	14
Tabla 5. Datos de imágenes. ....	24
Tabla 6. Imágenes recolectadas. ....	24
Tabla 7. Set Output Corners. ....	36
Tabla 8. Datos a investigar. ....	50
Tabla 9. Imágenes de estudio y deforestación para análisis. ....	50
Tabla 10. Deforestación y no deforestación. ....	51
Tabla 11. Datos y deforestación 1990. ....	55
Tabla 12. Consolidado de sensibilidad y especificidad. ....	57
Tabla 13. Datos y deforestación. ....	59
Tabla 14. Resultado PCA consolidado. ....	59
Tabla 15. Resultado PCA consolidado. ....	64
Tabla 16. Imágenes y deforestación 1990. ....	67
Tabla 17. Total perdida de datos. ....	67
Tabla 18. Clase de deforestación. ....	<b>¡Error! Marcador no definido.</b>
Tabla 19. Valores obtenidos. ....	<b>¡Error! Marcador no definido.</b>

## RESUMEN

El presente trabajo pretende calcular la progresión de la deforestación en un área geográfica específica y dentro de unos límites temporales dados, recurriendo para ello a imágenes satelitales *Landsat*. El trabajo emplea una metodología que inicia en la descarga de imágenes, procesadas con programas especializados tales como *Erdas*, - *Envi* . De igual manera, se empleó *Grass* para la importación de imágenes. se utilizó *rstudio* para la extracción de las características. Se elaboró una red neuronal *keras* utilizando para ello el lenguaje de programación *python* , desarrollo que permitió dar una conclusión respecto del área objeto de estudio.

**Palabras claves:** Deforestación, Imágenes, *Keras*, *Landsat*, Reconocimiento de patrones, Procesamiento de imágenes y Red Neuronal.

## INTRODUCCIÓN

Hoy en día atendiendo al progreso tecnológico, el mundo cuenta con múltiples insumos que permiten diagnosticar situaciones adversas como la deforestación y tomar medidas que permitan mitigar sus efectos. La deforestación se hace visible y se puede vislumbrar a través del procesamiento de las imágenes utilizando para esto software especializado en el tratamiento de imágenes satelitales. Esto se puede dar gracias a la evolución tecnológica que ha llevado al estudio de la teledetección, el espectro electromagnético, campos de estudio estos que desarrollan aspectos de tal magnitud e impacto que generan una percepción a gran escala respecto de los fenómenos que ocurren en el globo terráqueo. Disponible en <http://rimd.org/advf/documentos/4922edde159e6.pdf> [1]

A su vez la ingeniería de Sistemas mediante la minería de datos y la utilización de diferente software, permite detallar características particulares que se requieran extraer de estas imágenes. Las imágenes que se requieren para adelantar trabajos en materia de ingeniería de sistemas, no pueden ser utilizadas tal como son extraídas del satélite puesto que las mismas necesitan un procesamiento especial para que sirvan de insumo útil en la investigación. Así las cosas, el procesamiento de tales imágenes se traduce en un paso vital para que las mismas como base del trabajo a desarrollar arrojen resultados confiables y sujetos de validación y verificación.

El trabajo se desarrolla atendiendo a una metodología que inicia con la descarga de imágenes, el procesamiento de las mismas, la importación a programas especializados como envi, erdas Grass, la extracción de las características, el análisis en una red neuronal, esto con el fin de tomar decisiones. <https://ccc.inaoep.mx/~ariel/recpat.pdf> [2]

En este contexto este trabajo pretende encontrar la deforestación en un área determinada del país, con base en imágenes satelitales, a las cuales se aplicará reconocimiento de patrones.

Para lograr tal cometido el trabajo se desarrolla en seis capítulos, en el capítulo inicial se realiza una introducción respecto de la temática a abordar en la investigación, el segundo capítulo contiene el planteamiento del problema, se continua con un marco teórico consignado en el tercer capítulo, el capítulo cuarto abarca la metodología y el desarrollo mismo del objeto de investigación, finalmente el capítulo quinto se establecen los resultados de la investigación y finalmente el último capítulo hace alusión a la bibliografía que soporta el presente trabajo.



## 1. PLANTEAMIENTO DEL PROBLEMA

Hoy por hoy en un planeta en donde los recursos naturales se ven amenazados por diferentes factores entre ellos el cambio climático y el accionar humano, es de vital importancia que desde el campo de la ingeniería de sistemas se contribuya a diseñar herramientas que permitan conservar los recursos naturales y entender la dinámica de cambio que ellos han debido afrontar. Bajo este contexto, el trabajo plantea el siguiente problema de investigación, para contribuir a la construcción de conocimiento en la materia a partir de la disciplina de los sistemas.

Es posible determinar la deforestación en un área geográfica determinada, mediante el reconocimiento de patrones aplicado a imágenes satelitales.

Hipótesis:

1. Las imágenes satelitales deben ser convertidas en *raster* con miras a determinar características específicas, para ello se debe utilizar programas de procesamiento de imágenes y programas estadísticos.
2. Se pueden utilizar y conjugar resultados de diversos estudios de deforestación, utilizándolos como semilla, obteniendo a partir de ellos resultados deseados.
3. La deforestación de un área geográfica determinada, se calcula a través de reconocimiento de patrones aplicado a imágenes satelitales.

## 2. MARCO TEÓRICO

### 2.1 FUNDAMENTOS DE LA TELEDETECCIÓN

La capacidad que tiene el hombre de ver se produce debido a que los ojos reciben y traducen adecuadamente una energía luminosa. Es decir, se puede percibir una señal de cualquier objeto que no es originada por este, sino por un foco energético exterior que le ilumina, llámese sol o iluminación artificial. Esta es la razón del porque no se puede percibir los objetos en la oscuridad. [3]

El ojo humano es el sensor natural que percibe la información de la superficie terrestre, no obstante lo anterior, el ojo posee limitaciones, una de ellas es que sólo se puede ver a través de una región espectral visible, la otra limitación es que en la oscuridad el ojo humano no es capaz de percibir objetos y si lo hace es con mucha dificultad. Al contrario de esto los sensores que se encuentran en los satélites artificiales son multiespaciales e hiperespectrales esto permite registrar diferentes longitudes de onda o lo que se denomina bandas del espectro electromagnético. [4]

El sistema de teledetección se basa en ondas electromagnéticas, las cuales utilizan los siguientes elementos:

1. Fuente o fuentes de la radiación electromagnética.
2. Interacción de las ondas electromagnéticas con la superficie terrestre.
3. Interacción de las ondas electromagnéticas con la atmósfera.
4. Sensor de las ondas electromagnéticas.

Disponible en <http://fcf.unse.edu.ar/archivos/series-didacticas/SD-33-Principios-teledeteccion-REUTER.pdf> [7]

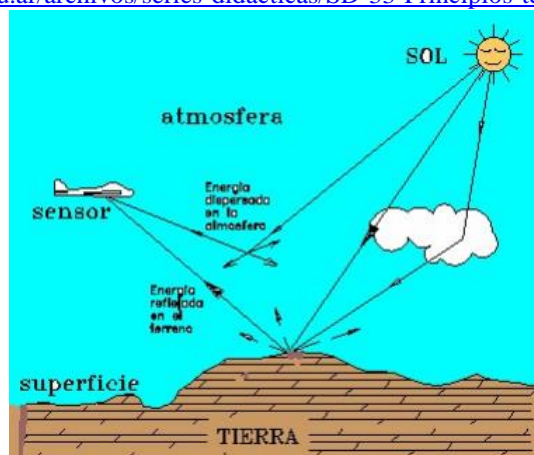


Fig. 2-1. Componentes de la Teledetección.

Disponible en <http://fcf.unse.edu.ar/archivos/series-didacticas/SD-33-Principios-teledeteccion-REUTER.pdf> [7]

## 2.2 LANDSAT

Para el desarrollo del presente estudio se utilizaron imágenes satelitales, tomadas por los sensores equipados en satélites *Landsat*. Disponible en <https://sputnik87.wordpress.com/tag/sputnik-1/> [5]

Con el desarrollo de la aviación, surgió la fotografía espacial, la cual ha evolucionado a un ritmo vertiginoso, permitiendo que la humanidad aumente los conocimientos sobre el planeta. Aquí nace el término *remote sensing* esto a inicio de la década de los años 60 y designa a cualquier medio de observación remota. [3]

En 1960, la NASA puso en órbita el primer satélite de observación de la Tierra, TIROS-1, pionero de la investigación meteorológica. Luego de ello se inició una serie de lanzamientos en la que se reconoce principalmente los satélites *LANDSAT*. [4]

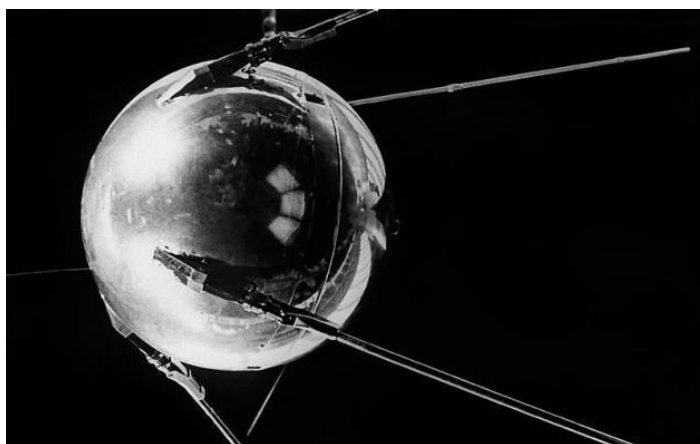


Fig. 2-2. Satélite soviético Sputnik-1, réplica en el Museo de los Conquistadores del Espacio, Moscú, Rusia. Satélite artificial Sputnik.

Disponible en <https://sputnik87.wordpress.com/tag/sputnik-1/> [5]

Los satélites *LANDSAT* fueron lanzados desde el año de 1972, hasta la fecha se han realizado 8 lanzamientos cinco de estos lanzamientos han finalizado, *Landsat 6* no pudo llegar a órbita, *Landsat 7* Pero desde 2003 con la línea del corrector de exploración defectuosa, y *Landsat 8* actualmente operativo.



Fig. 2-3. *Landsat 8* en órbita. Fuente: NASA y OSC..



Fig. 2-4. Model of *Landsat 4* Satellite.

## 2.2.1 Sensores Satelites *Landsat*

### 2.2.1.1 Bandas de *Landsat 4*

Tabla 1. Sensor MMS *Landsat 4*.

Sensor	Misión	Banda	Longitud de Onda ( $\mu$ )	Resolución espacial (m)	Resolución radiométrica
TM	4 – 5	1 Azul	0.45-0.52	30	8 bit
		2 Verde	0.52-0.60	30	8 bit
		3 Rojo	0.63-0.69	30	8 bit
		4 NIR	0.76-0.90	30	8 bit
		5 SWIR 1	1.55-1.75	30	8 bit
		6 TIR	10.4-12.5	120	8 bit
		7 SWIR2	2.8 -2.35	30	8 bit

Disponible en <http://documentacion.ideam.gov.co/openbiblio/bvirtual/022977/ProtocoloProcesamientoDigital2.pdf> [8]

### 2.2.1.2 Bandas de *Landsat 5*.

Tabla 2. Sensor TM *Landsat 5*

Sensor	Misión	Banda	Longitud de Onda ( $\mu$ )	Resolución espacial (m)	Resolución radiométrica
TM	4 – 5	1 Azul	0.45-0.52	30	8 bit
		2 Verde	0.52-0.60	30	8 bit
		3 Rojo	0.63-0.69	30	8 bit
		4 NIR	0.76-0.90	30	8 bit
		5 SWIR 1	1.55-1.75	30	8 bit
		6 TIR	10.4-12.5	120	8 bit
		7 SWIR2	2.8 -2.35	30	8 bit

Disponible en <http://documentacion.ideam.gov.co/openbiblio/bvirtual/022977/ProtocoloProcesamientoDigital2.pdf> [8]

### 2.2.1.3 Bandas de Landsat 7.

Tabla 3. Sensor ETM + *Landsat 7*.

Sensor	Misión	Banda	Longitud de Onda ( $\mu$ )	Resolución espacial (m)	Resolución radiométrica
ETM+	7	1 Azul	0.45-0.515	30	8 bit
		2 Verde	0.52-0.60	30	8 bit
		3 Rojo	0.63-0.69	30	8 bit
		4NIR	0.76-0.90	30	8 bit
		5 SWIR 1	1.55-1.75	30	8 bit
		6 TIR	10.4-12.5	60	8 bit
		7 SWIR-2	20.9-2.35	30	8 bit

Disponible en <http://documentacion.ideam.gov.co/openbiblio/bvirtual/022977/ProtocoloProcesamientoDigital2.pdf> [8]

### 2.2.1.3 Bandas de Landsat 8.

Tabla 4. Sensor OLI *Landsat 8*.

Sensor	Misión	Banda	Longitud de Onda ( $\mu$ )	Resolución espacial (m)	Resolución radiométrica
OLI	8	1Costa/Aero-sol	0.435-0.451	30	12 bit
		2 Azul	0.452-0.512	30	12 bit
		3 Verde	0.533-0.590	30	12 bit
		4Rojo	0.636-0.673	30	12 bit
		5NIR	0.851-0.879	30	12 bit
		6 SWIR 1	1.566-1.651	30	12 bit
		10 TIR 1	10.60-12.36	100	12 bit
		11TIR 2	11.5-12.51	100	12 bit
		7 SWIR-2	2.107-2.294	30	12 bit

Disponible en <http://documentacion.ideam.gov.co/openbiblio/bvirtual/022977/ProtocoloProcesamientoDigital2.pdf> [8]

## 2.3 BANDAS DE LANDSAT.

Para el desarrollo del siguiente trabajo se utilizan imágenes de *Landsat 4, 5, 7, 8* sin embargo se va a explicar lo que cada banda representa utilizando el sensor OLI de *Landsat 8*, por cuanto las bandas de este satélite compilan las bandas que emplean los satélites precedentes.

Bandas	Longitud de Onda (nm)	GSD (m)
1 - COASTAL AEROSOL	300	30
2 - AZUL	450 - 510	30
3 - VERDE	530 - 590	30
4 - ROJO	640 - 670	30
5- INFRAROJOS	850 - 880	30
6 - SWIR 1	1570 - 1650	30
7 - SWIR 2	2110 - 2290	30
8- PANCROMÁTICA	500 - 680	15
9- CIRRUS	1360 - 1380	30
10- INFRAROJOS TÉRMICOS 1	10600 - 11190	100 (*30)
11- INFRAROJOS TÉRMICOS 2	11500 - 12510	100 (*30)

\* Las bandas TIRS se obtienen a resoluciones de 60m pero son remuestados a 30m para información del producto final

Disponible en <http://geocento.es/galeria-de-satelites-para-buscar-y-adquirir-imagenes/satelite-imagenes-landsat-8/>

## 2.4 ANÁLISIS DE IMÁGENES

Las imágenes *LANDSAT* poseen en general 11 bandas pero para efectos de este estudio solo se van a utilizar 6 bandas como se lo va a explicar mas adelante, estas 6 bandas conforman una sola imagen pero cada una de estas bandas tiene información muy grande que permiten diferenciar unos atributos específicos; para analizar las imágenes se debe separar bandas y extraer sus características, para tal efecto el estudio se vale de una serie de programas especializados, una vez obtenidos los datos se procede a realizar el estudio.

Lo que se intenta es con el insumo de grandes cantidades de datos se descubra cúmulos de información a partir de las bases que posee esta investigación, es decir, se realizará minería de datos y se va a utilizar para ello aprendizaje supervisado. El aprendizaje supervisado es el análisis discriminante en el que un patrón de entrada está asignado para una clase predefinida, para realizar la minería de datos se va a utilizar una red neuronal.

- Redes Neuronales. Para la realización del trabajo se pretende utilizar una red neuronal con el objeto de que la misma permita analizar los datos y obtener los resultados esperados. Ver <https://www.xataka.com/robotica-e-ia/las-redes-neuronales-que-son-y-por-que-estan-volviendo> [11]

En este contexto, se investigó sobre una red neuronal de fácil implementación que contará con características que la hagan a su vez muy precisa y veloz en su procesamiento, el algoritmo que cumple con esos requisitos es keras.

- Keras es una API de redes neuronales de alto nivel, escrita en Python y capaz de ejecutarse sobre TensorFlow, CNTK o Theano. Fue desarrollado con un enfoque en permitir la experimentación rápida. “*Pasar de la idea al resultado con la menor demora posible es la clave para hacer una buena investigación*”. Ver en <https://keras.io/> [12]

- También se recurre a PCA (Principal Component Analysis), el cual se utiliza para la reducción de dimensión de los datos originales. [29]

- Para esta investigación se utilizaron varios programas de procesamiento de imágenes satelitales ellos son ERDAS, GRASS, ENVI se empleó phyton para el programa de importación de imágenes, para la implementación de la red neuronal keras, como también para la creación del programa PCA para graficar los puntos de las imágenes, la explicación de cada uno de estos se efectuará más adelante.



### 3.METODOLOGÍA

#### 3.1 MATERIALES

Para el presente estudio se recurrió a la descarga de las siguientes imágenes obtenidas de los satélites *LANDSAT* y descargadas de la página <http://earthexplorer.usgs.gov/>.

En esta investigación, se cuenta con 29 imágenes; en el apilamiento de bandas se pudo apreciar todas las imágenes dando como resultado que todas no son sensibles de procesamiento digital debido a la alta nubosidad, por tanto, en este paso se descartan imágenes quedando para el estudio las siguientes (en este punto se clasificaron los archivos quedando las imágenes con los siguientes nombres):

- laguna\_cocha\_1990\_abril
- laguna\_cocha\_1999\_julio
- laguna\_cocha\_2002\_octubre
- laguna\_cocha\_2005\_noviembre
- (laguna\_cocha\_2005\_noviembre\_23\_gapfill)
- laguna\_cocha\_2008\_septiembre
- (laguna\_cocha\_2008\_septiembre\_12\_gapfill)
- laguna\_cocha\_2010\_noviembre
- laguna\_cocha\_2012\_julio\_21
- (laguna\_cocha\_2012\_julio\_21\_gapfill)
- laguna\_cocha\_2013\_enero
- laguna\_cocha\_2014\_diciembre

De igual manera, se emplearon en el presente trabajo imágenes suministradas por el IDEAM en medio magnético.

1. Mapas de cambio de bosque para el Departamento de Nariño periodos 1990-2000, 2000-2005, 2005-2010, 2010-2012, 2012-2013 versión 5.0. Suministrado por IDEAM mediante respuesta calendada el día 6 de junio de 2017.

2. Mapa de cambio de bosque para el departamento de Nariño periodo 2013-2014 versión 6.0. Suministrado por IDEAM mediante respuesta calendada el día 6 de junio de 2017.

Una vez obtenidas las imágenes satelitales mediante descargas y a través del IDEAM en medio magnético, se procede a realizar un análisis de las mismas, para ello se recurre a un software especializado, el cual se describe de forma precedente:

**ERDAS.** Es un es el software geoespacial de teledetección que se centra en el procesamiento de imagen. Disponible en [http://www.erdas.com.ar/productos\\_imagine.htm](http://www.erdas.com.ar/productos_imagine.htm) [19]

**ENVI** Al igual que ERDAS es un software geoespacial, en el desarrollo de este trabajo se utilizó una herramienta muy importante que tiene este programa que es gapfill. Disponible en [http://www.esri.es/es/productos/aplicaciones-especializadas/envi\\_66/](http://www.esri.es/es/productos/aplicaciones-especializadas/envi_66/) [20]

En este contexto, se emplea el programa de computación llamado GRASS GIS, versión 7.2. GRASS GIS, comúnmente conocido como GRASS (Sistema de Soporte de Análisis de Recursos Geográficos). <https://grass.osgeo.org/> [23]

A. Keras es una red neuronal de alto nivel escrita en Python . Keras es una poderosa biblioteca Python fácil de usar para desarrollar y evaluar modelos de aprendizaje en profundidad. [deep learning](https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/) models. Véase <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/> [27]

PCA (Principal component analysis) Se utilizó en la reducción de dimensión de los datos originales para su posterior reconstrucción, con la menor pérdida de información posible. Disponible en [http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patronos-70038/contenidos/Ejercicios/trabajo\\_1\\_rpyrn\\_2009-10.pdf](http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patronos-70038/contenidos/Ejercicios/trabajo_1_rpyrn_2009-10.pdf) [29]

### 3.2 DIAGRAMA DE FLUJO

Para la realización de la metodología se diseñó un diagrama de flujo que refleja los pasos que se desarrollan a lo largo de la ejecución de este proyecto.

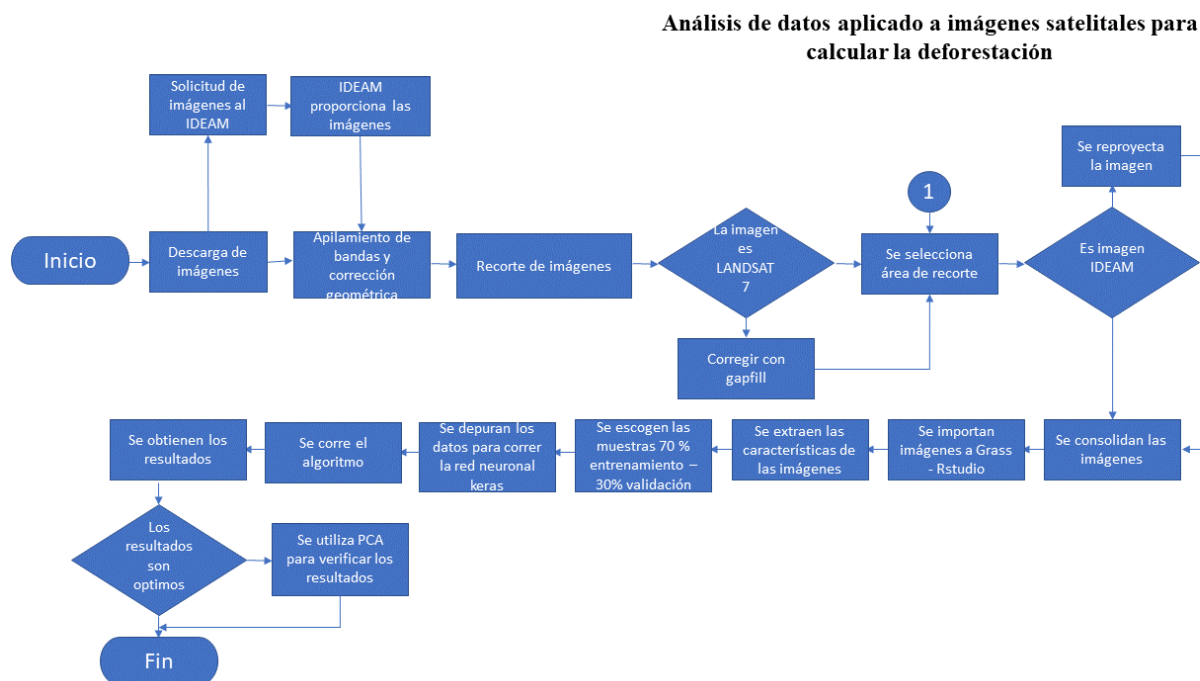


Fig. 3-1 Flujo de procesos en el desarrollo del estudio.

Para el desarrollo del trabajo se recurre en primera instancia a la selección y descarga de las imágenes, acto seguido se realiza el preprocesamiento de las mismas, esta etapa tiene dos pasos fundamentales que son el apilamiento de bandas y la corrección geométrica, este último paso se realiza con la ayuda de software especializado como *envi* y *erdas*, este software también permite recortar las imágenes de tal manera que queden con el mismo tamaño, así mismo permite realizar la corrección en las imágenes *Landsat 7* utilizando para esto la herramienta de GAPFILL. Adicionalmente se solicitó al IDEAM las imágenes que contiene la semilla del estudio, es decir, las imágenes que contienen la deforestación, en este punto es necesario advertir que a las imágenes del IDEAM deben ser reproyectadas. Se aclara que tanto las imágenes descargadas de internet como las imágenes suministradas por el IDEAM deben ser sometidas a extracción de características, se utiliza para esto la buena interacción entre GRASS y RSTUDIO, a la postre se realiza minería de datos empleando en este procedimiento una red neuronal llamada KERAS, que arroja resultados para ser evaluados, finalmente se recurre a PCA con el fin de validar los puntos diagramados.

## 4. RESULTADO

En primer término es menester indicar que el presente trabajo se edificó a partir de imágenes que constituyen el insumo sine quanon o punto de partida que permitió el desarrollo del mismo, es determinante precisar respecto a las mismas que si bien estos insumos disponibles en páginas de descarga como earthexplorer, glcf, Landsatlook, dgi, glcfapp, entre muchas otras, la calidad que las mismas poseen no es óptima para su análisis, entre otros factores porque no son visibles ya que son tomadas en la noche aunado al hecho del índice de nubosidad que se presenta, dado el clima del área de estudio; así pues y en este contexto la cantidad de imágenes óptimas para el estudio es muy reducida.

En este entendido, con miras a disponer de imágenes de calidad adecuada, se recurrió al satélite *Landsat 7*, que es una de las mejores opciones disponibles para obtener imágenes idóneas. No obstante, lo anterior, para la época en que se efectuó el proceso de descargue de imágenes, el sensor del satélite presentó fallas, pese a esta situación las imágenes obtenidas con el satélite continuaron siendo la mejor opción disponible. En consecuencia y con la finalidad de que estas imágenes sean útiles y pertinentes se recurrió al programa ENVI el cual cuenta con la herramienta gapfill que contribuyó a reparar la imagen. Se precisa que fue valioso en el trabajo el material entregado por el IDEAM, entidad en la cual reposaban imágenes del territorio y más específicamente del área y problema de estudio.

En consecuencia y para el progreso del siguiente trabajo tanto las imágenes descargadas de internet como las suministradas por el IDEAM debieron ser procesadas para que sean completamente homogéneas, esto para que los datos sean iguales, la extracción de características es un punto fundamental dentro de la investigación, características que se ubicaran utilizando RSTUDIO, estos insumos se ingresan en una red neuronal que utiliza un algoritmo llamado keras, el cual esta implementado en lenguaje Phyton y que arroja los resultados para ser analizada. Finalmente se emplea PCA para diagramar los puntos recolectados.

### 4.1 SELECCIÓN Y DESCARGA DE IMÁGENES

Para la descarga de imágenes se utilizó la página <http://earthexplorer.usgs.gov/>, esta página pertenece al servicio geológico de EEUU.

La Universidad de Maryland provee este servicio en su página (<http://glcf.umd.edu/data/>) [14]. También se puede descargar imágenes de <http://Landsatlook.usgs.gov/viewer.html> [15]. Para iguales efectos se cuenta con la página <http://www.dgi.inpe.br/CDSR/> [16].

Para desarrollar este trabajo se descargaron una gran cantidad de imágenes, buscando que en cada una de ellas haya el menor índice de nubosidad.

Se precisa que emplearon imágenes descargadas de la página <http://earthexplorer.usgs.gov/>, [15] enfocándolas en imágenes *LANDSAT*. En este contexto se procede a describir el proceso de descarga de imágenes.

El primer paso es ingresar a la página <http://earthexplorer.usgs.gov/> [15], de forma precedente se debe ir a Search Criteria y seleccionar las coordenadas del sitio objeto de investigación.

Otra manera de lograr tal resultado, consiste en especificar la localidad deseada formato WRS-2.

En la siguiente imagen se muestra la ficha de metadatos de imágenes grilla *Landsat*.

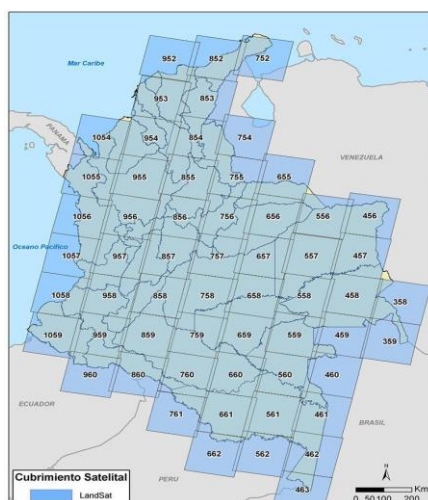


Fig. 4-1. Ficha metadatos Colombia.

Disponible en <http://www.biesimci.org/satelital/Auxiliar/Limites/Grillassatelite/Indices/B551L01-S000002A001.html> [17]

En el diseño de esta investigación se examinarán imágenes desde el año de 1985 hasta el año 2016, por tanto se ingresan estos rangos de búsqueda:

Fig. 4-2. Búsqueda por fechas.

Disponible en <https://earthexplorer.usgs.gov/> [13]

Data Sets: en esta pestaña se escoge el sensor que se desea para realizar la investigación, para el caso se escogió imágenes *LANDSAT* 4, 5, 7 y 8.

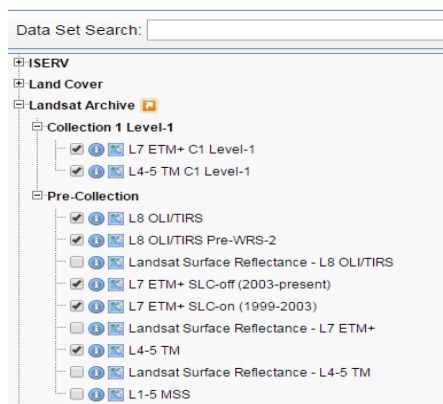


Fig. 4-3. Data Set Search

Disponible en <https://earthexplorer.usgs.gov/> [13]

Así, se procede a efectuar click en la tecla buscar, donde aparecen varios resultados de los que escogeremos el que sirva para el estudio

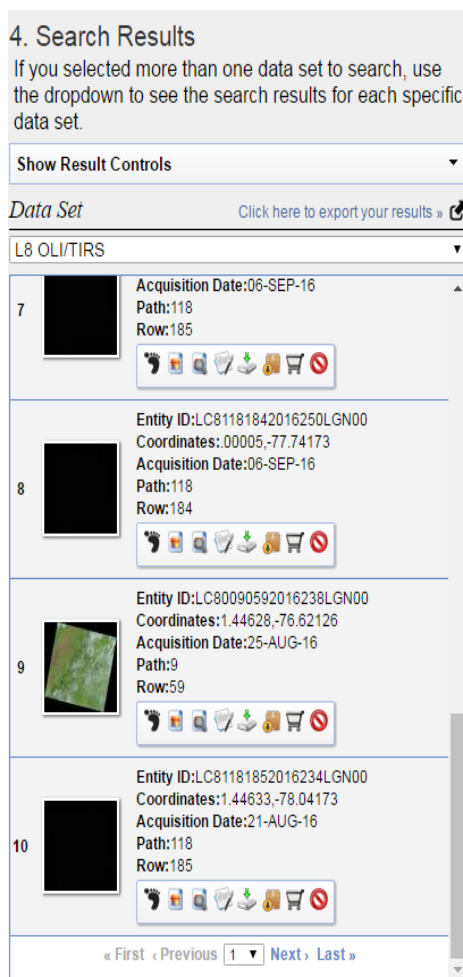


Fig. 4-4. Search Results.

Disponible en <https://earthexplorer.usgs.gov/> [13]

En consecuencia, se elige el mejor resultado y se descarga el archivo

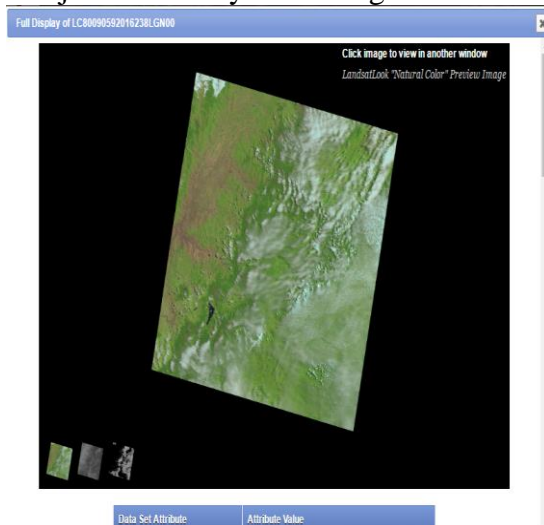


Fig. 4-5. Imagen Descargada.

Disponible en <https://earthexplorer.usgs.gov> [13]

Al final se tiene la descarga con todas las bandas de la imagen.

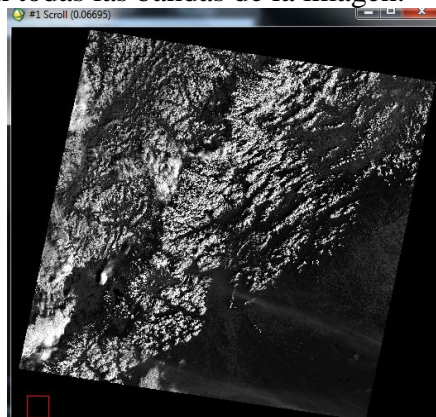


Fig. 4-6. Ejemplo de Banda 1.

Disponible en <https://earthexplorer.usgs.gov/> [13]

Se cuenta también con un archivo metadata, el cual contiene toda la información vertida en esta

```
GROUP = L1_METADATA_FILE
GROUP = METADATA_FILE_INFO
ORIGIN = "Image courtesy of the U.S. Geological Survey"
REQUEST_ID = "0701605161817_00009"
LANDSAT_SCENE_ID = "LC80090592016014LGN00"
FILE_DATE = 2016-05-18T00:32:33Z
STATION_ID = "LGN"
PROCESSING_SOFTWARE_VERSION = "LPGS_2.6.2"
END_GROUP = METADATA_FILE_INFO
GROUP = PRODUCT_METADATA
DATA_TYPE = "L1T"
ELEVATION_SOURCE = "GLS2000"
OUTPUT_FORMAT = "GEOTIFF"
SPACECRAFT_ID = "LANDSAT_8"
SENSOR_ID = "OLI_TIRS"
WRS_PATH = 9
WRS_ROW = 59
NADIR_OFFNADIR = "NADIR"
TARGET_WRS_PATH = 9
TARGET_WRS_ROW = 59
DATE_ACQUIRED = 2016-01-14
```

Fig. 4-7. Metadata.

Disponible en <https://earthexplorer.usgs.gov/> [13]

Para el presente estudio se recopilaron las siguientes imágenes:

Tabla 5. Datos de imágenes

Año	Imágenes	Nombre
1985	1	LT50090591985024XXX02.tar
1986	1	LT50090591986059AAA02.tar
1987	1	LT50090591987286XXX01.tar
1988	1	LT40090591988345XXX02.tar
1989	3	LT40090591989027XXX02.tar LT40090591989219XXX06.tar LT40090591989363XXX01.tar
1990	1	LT05_L1TP_009059_19900428_20170130_01_T1_ANG
1991	1	LT40090591991001XXX01.tar
1999	2	LT50090591999191XXX06.tarLT50090591999319CPE00.tar
2000	2	LE70090592000202EDC00.tarLE70090592000314EDC00.tar
2001	1	LE70090592001252EDC00.tar
2002	1	LE70090592002287EDC00.tar
2005	1	LE70090592005327EDC00.tar
2008	1	LE70090592008256ASN00.tar
2009	2	LE70090592009146EDC00.tarLE70090592009258EDC00.tar
2010	2	LE70090592010261ASN00.tarLE70090592010309EDC00.tar
2012	2	LE70090592012203ASN00.tar LE70090592012219ASN00.tar
2013	2	LC80090592013357LGN00.tar LE70090592013029EDC00.tar
2014	2	LC80090592014024LGN00.tar LC80090592014344LGN00.tar
2015	1	LE70090592008256ASN00.tar
2016	2	LC80090592016014LGN00.tar LC80090592016014LGN00.tar
Total	30	

En total se tomaron 30 imágenes de diferentes satélites *LANDSAT*

Tabla 6. Imágenes recolectadas.

<b>LANDSAT</b>	<b>SENSOR</b>	<b>IMÁGENES</b>
<i>LANDSAT 4</i>	MSS	5
<i>LANDSAT 5</i>	TM	6
<i>LANDSAT 7</i>	ETM+	14
<i>LANDSAT 8</i>	OLI - TIRS	5



**4.1.1. Preprocesamiento de imágenes.** La presente investigación toma como referencia algunos puntos de la publicación Protocolo de procesamiento digital de imágenes para la cuantificación de la deforestación en Colombia V.2, un libro desarrollado por el Ministerio de Ambiente, el IDEAM, la fundación MOORE, y la fundación Patrimonio Natural. Disponible en [http://www.ideam.gov.co/documents/13257/13817/Protocolo\\_para\\_la\\_cuantificaci%C3%B3n\\_De\\_forestaci%C3%B3n\\_Nivel\\_Nacional.pdf](http://www.ideam.gov.co/documents/13257/13817/Protocolo_para_la_cuantificaci%C3%B3n_De_forestaci%C3%B3n_Nivel_Nacional.pdf) [18]

En este contexto los pasos a seguir en la investigación de este trabajo son:

- Apilamiento de bandas
- Corrección geométrica

El área examinada se traduce en entornos de la Laguna de la Cocha. Para el pre procesamiento de la imagen y para el apilamiento de las bandas se trabaja con el Software ERDAS, ENVI.

**4.1.2. Apilamiento de Bandas.** Es un procedimiento fundamental dentro del procesamiento de imágenes, para esta función se emplea la herramienta ERDAS. El apilamiento de las bandas se realiza según el sensor que se esté utilizando, hay que recordar que todos los sensores no tienen la misma cantidad de bandas y las bandas entre sensores son diferentes. En esta investigación, se cuenta con 29 imágenes; en el apilamiento de bandas se pudo apreciar todas las imágenes dando como resultado que todas no son sensibles de procesamiento digital debido a la alta nubosidad, por tanto, en este paso se descartan imágenes quedando para el estudio 9 imágenes tres de ellas *Landsat 7*, las cuales es necesario arreglarlas con gapfill.

Esto arroja un total de nueve imágenes, sin embargo, tres imágenes muestran errores y es necesario arreglarlas con gapfill esto es un procesamiento especial, el cual será explicado más adelante. Para el desarrollo de apilamiento de bandas se utilizó el software ERDAS, es preciso aclarar que tanto en ENVI como en ARCMAP, se puede efectuar el mismo procedimiento y debe arrojar el mismo resultado.

Se trabajó en apilar las bandas que se van a utilizar, para ilustrar el ejemplo se utiliza una imagen *Landsat 8* en la que no se incluye ni la banda seis por ser termal, ni la banda 8 por ser pancromática ya que tiene una resolución de 15 x 15 metros.

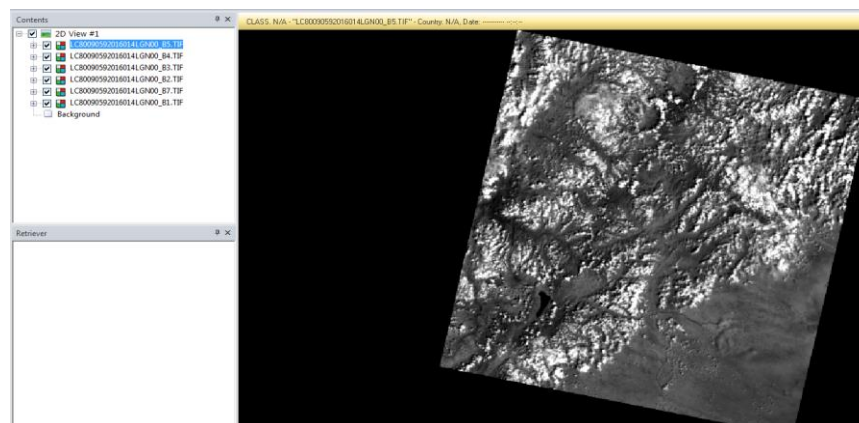


Fig. 4-8. Apilamiento de Bandas imagen propia.

La siguiente figura muestra las bandas a combinar, recurriendo para ello a la herramienta Layer Stack.

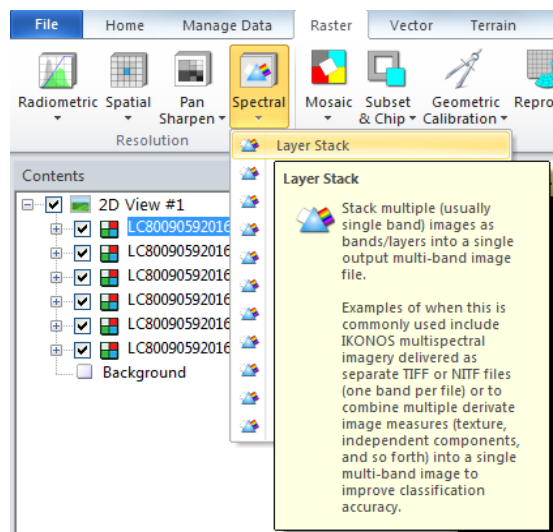


Fig. 4-9. Recolección de bandas figura propia.

A la postre se asigna un nombre de salida y se adiciona banda por banda en orden; hay que estar seguro de que la salida de la imagen sea de 8 bits y se presiona la opción ok.

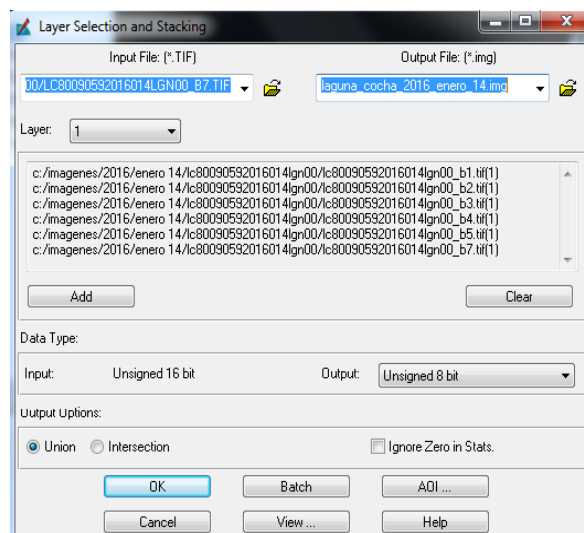


Fig. 4-10. Layer

Una vez realizado el paso anterior se obtiene como resultado la imagen con las bandas incluidas.

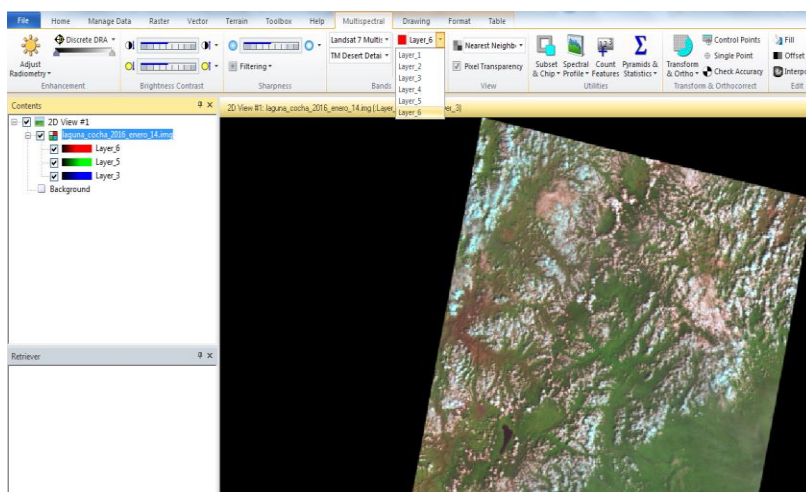


Fig. 4-11, Reunión de bandas

Derivado de lo anterior y como se aprecia en la figura se encuentra la imagen con las seis bandas incluidas de forma precedente. Entonces utilizando el software ERDAS se reúnen las bandas, excepto las bandas termales en cada uno de los sensores, esto por cuanto no se va a realizar investigación sobre la incidencia climática en el presente documento y en caso de *Landsat 7* y *Landsat 8* no se va a utilizar la banda Pancromática debido a que la resolución del pixel es de 15 X 15 metros y en este trabajo se va a emplea una medida por pixel de 30 X 30 metros.

**4.1.3. Tamaño de imágenes descargadas de *Landsat*.** Las imágenes originales son demasiado grandes ya que abarcan un gran espacio, por tanto lo que se requiere es recortar el área a trabajar para la elaboración del trabajo, utilizando para ello, el Software Erdas. Para realizar este procedimiento se recurre a la imagen procesada, mediante la utilización de la herramienta inquire box, esto con el fin de utilizar las mismas coordenadas y así asegurar trabajar siempre con las mismas dimensiones.

Las coordenadas que se usan para el recorte de la imagen son:



Fig. 4-12. Coordenadas

Con esta información el siguiente paso es el corte de la imagen, procedimiento que se realiza mediante la herramienta Create Subset Image que se encuentra dentro del menú Subset y Chip de ERDAS.

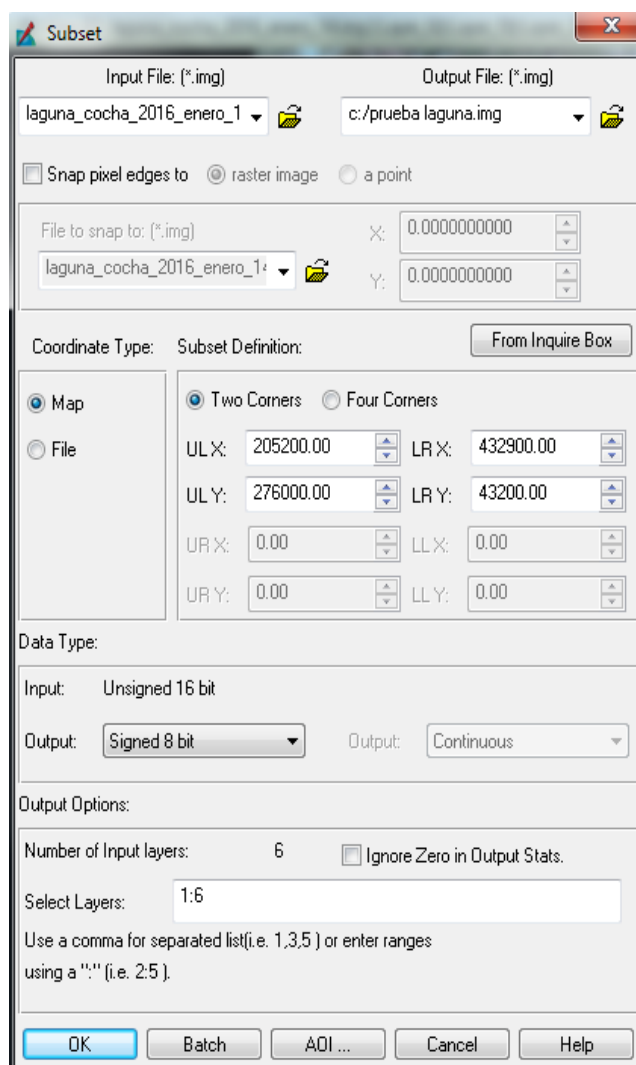


Fig. 4-13. Subset.

Se coloca el archivo de inicio para realizar el recorte, así como también el nombre con el cual quedará la imagen recortada, se sitúan las coordenadas obtenidas y se procede a pinchar la opción ok. Esta es la imagen resultante del recorte, tal procedimiento se debe hacer con todas las imágenes que van a ser objeto de estudio.

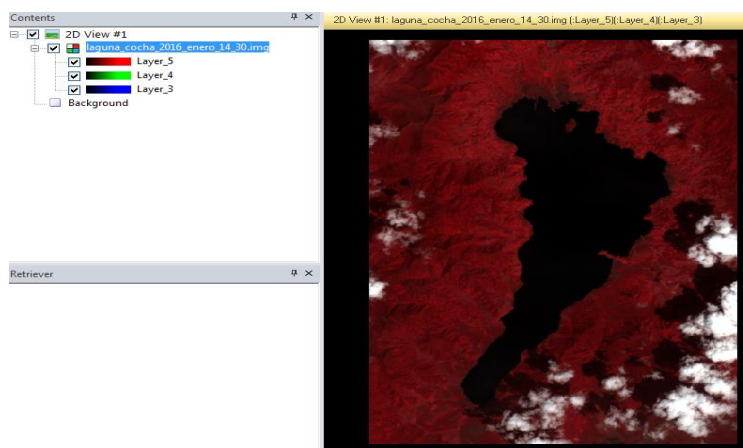


Fig. 4-14. Recorte de imagen propia.

Como se ha manifestado de manera precedente y a efectos de que las imágenes a procesar sean homogéneas y que el estudio pueda partir de iguales bases y supuestos, se debe transformar imágenes de *Landsat 8* de 16 a 8 bits, máxime si se considera que la mayoría de imágenes procesadas son de 8 bits.

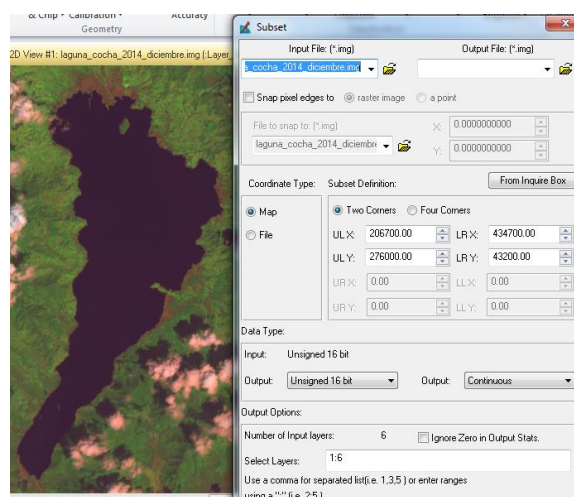


Fig. 4-15. Subset figura propia.

Este procesamiento es posible, gracias a la opción Rescale que se puede encontrar en el software Erdas, logrando así que toda la información quede en imágenes de 8 bits.

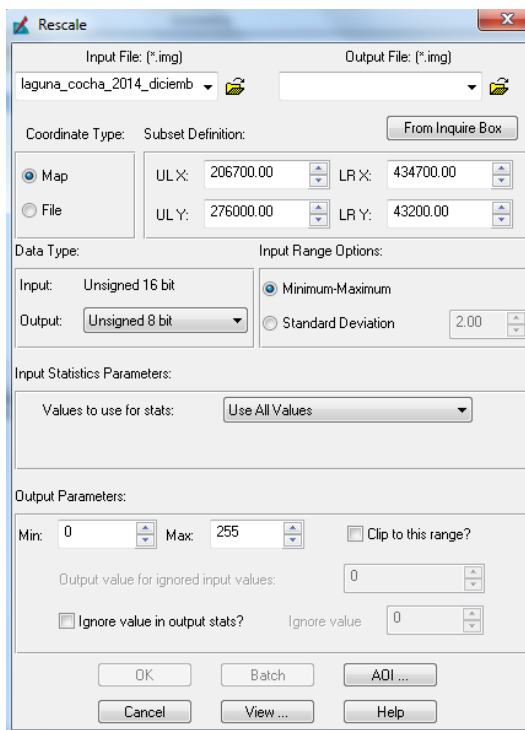


Fig. 4-16. Rescale imagen propia.

**4.1.4. Errores de imágenes *Landsat*.** Desde el 31 de mayo de 2003, el instrumento SLC del sensor ETM+, tuvo una grave avería que no pudo ser resuelta esto produce que las imágenes que capta el satélite presenten vacíos fuera del área central de la escena. Disponible en <http://www.geografiafisica.org/2013/02/11/lanzamiento-del-satelite-ldcm-landsat-data-continuity-mission/> [21]

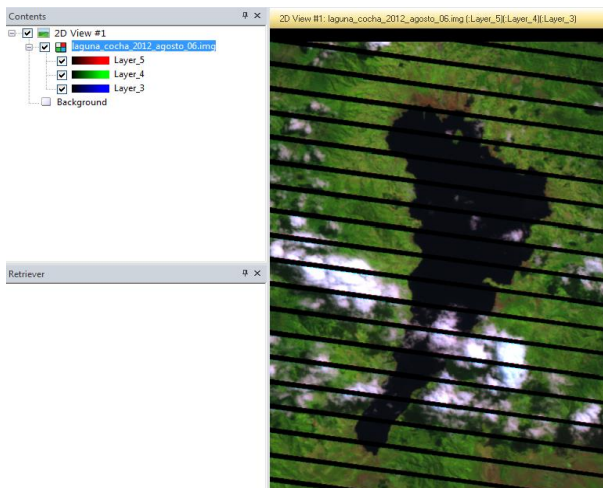


Fig. 4-17. Gapfill.

Disponible en [http://Landsat.usgs.gov/products\\_slcoffbackground.php](http://Landsat.usgs.gov/products_slcoffbackground.php) [22]

Para superar esta falencia existen varios métodos, para el caso particular, se acude al método Gapfill que será implementado en el Software ENVI en la versión 4.7. ENVI no posee esta librería por lo que hay que se debe descargarla de Internet, una vez instalada se abre la imagen en el software.

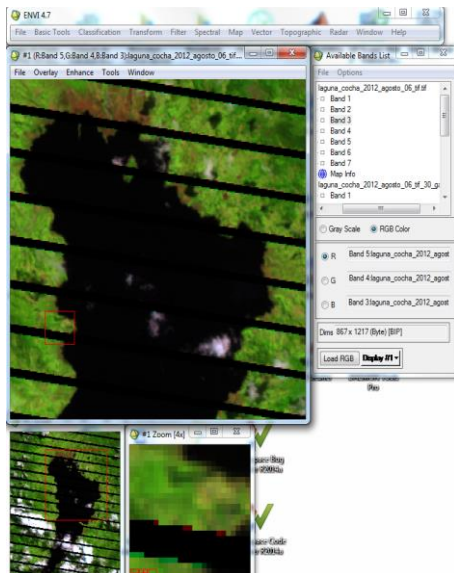


Fig. 4-18. Arreglo Gapfill

Posteriormente se debe ir a menú Basic Tools donde se encontrara la opción gapfill.

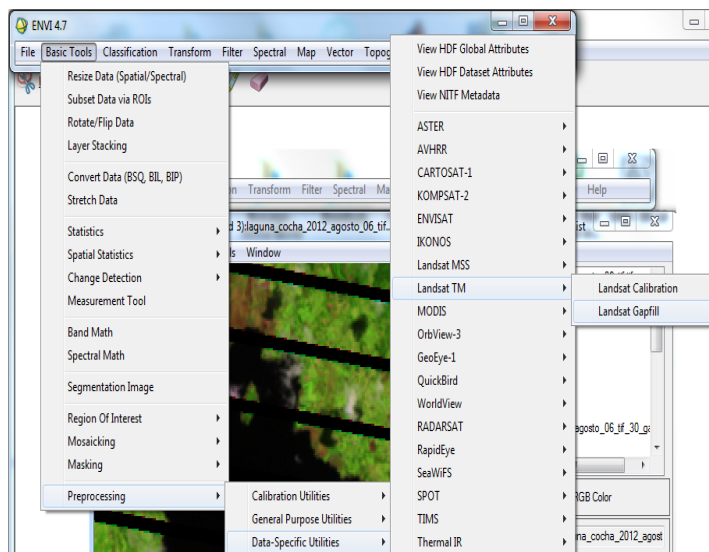


Fig. 4-19. Arreglo Gapfill.

Después de este procedimiento se obtiene los siguientes resultados.

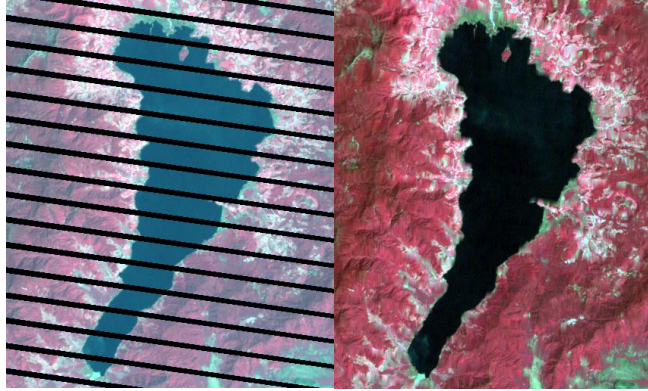




Fig. 4-20 Resultados Gapfill.

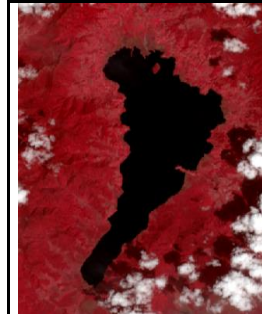



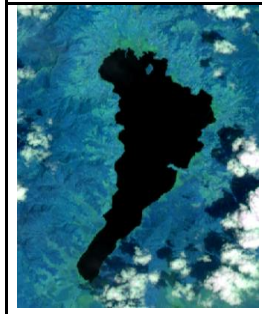
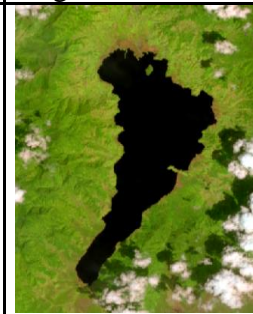
Fig. 4-21. Resultados Gapfill

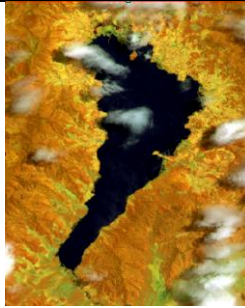





#### 4.1.5. Combinación de Bandas *LANDSAT 8*.

432 Color Natural	764 Falso Color
	
Fig. 4-22. Color natural.	Fig. 4-23. Falso color.

543 Color infrarrojo (vegetación)	652 Agricultura
	
Fig. 4-24. Color infrarrojo.	Fig. 4-25. Agricultura.

765 Penetración atmosférica	654 Análisis de Vegetación
	
Fig. 4-26. Análisis de vegetación.	Fig. 4-27. Penetración atmosférica

BANDA 562 Vegetación saludable	Tierra Agua 564
	
Fig. 4-28. Vegetación saludable.	Fig. 4-29. Agua.

753 Natural con eliminación atmosférica	754 Infrarrojo de onda corta
	
Fig. 4-30. Eliminación atmosférica	Fig. 4-31. Infrarrojo onda corta

Las imágenes antes procesadas fueron descargadas de la página <http://earthexplorer.usgs.gov/> [13]

No obstante, lo anterior, para el trabajo se requiere imágenes de modelo que sirvan de entrenamiento para los patrones de reconocimiento, es decir, un modelo que permita a un programa de minería de datos, efectuar un proceso comparativo entre las imágenes inicialmente almacenadas en él y las que se le suministren de forma posterior como destino, para obtener el dato a investigar.

Para efectos de modelo de entrenamiento, se toma en el presente trabajo imágenes suministradas por el IDEAM, imágenes que fueron procesadas de la siguiente manera:

**4.1.5. Imágenes Ideam utilizadas para el modelo.** Para el procesamiento de las imágenes satelitales es necesario realizar una clasificación, para diferenciar la zona de bosque, con la zona con ausencia del mismo o zona de deforestación. Con las fotos que actualmente cuenta la investigación se puede efectuar, una clasificación manual pero por la falta de experticia se podría incurrir en errores de clasificación, otro inconveniente es que llevaría mucho tiempo, por tal motivo, se realizó una solicitud al Instituto de Hidrología, Meteorología y estudios Ambientales, IDEAM en la que se requirió información cartográfica, esta institución amablemente facilitó:

1. Mapas de cambio de bosque en el Departamento de Nariño periodos 1990-2000, 2000-2005, 2005-2010, 2010-2012, 2012. Suministrado por IDEAM mediante respuesta calendada el día 6 de junio de 2017.

2. Mapas de cambio de bosque en el departamento de Nariño periodo 2013-2014 versión 6. Suministrado por IDEAM mediante respuesta calendada el día 6 de junio de 2017. Esta información, se torna como semilla para realizar el análisis, así lo que se debe efectuar en primer lugar es un tratamiento a las imágenes que suministro el IDEAM, por cuanto las imágenes entregadas son del Departamento de Nariño y para el desarrollo del presente trabajo solamente se necesita una sección del departamento, por lo que hay que hacer un recorte de la imagen.

También se deben proyectar las imágenes, es decir, hay que cambiar la proyección espacial ya que las imágenes del IDEAM tienen una proyección EPSG 3116 y las imágenes objeto de estudio tienen una proyección espacial EPSG 32618. Por último, hay que dejar el tamaño de las celdas a 30.00 mts ya que las imágenes que las imágenes originales no están exactamente cortadas a 30.00 mts.

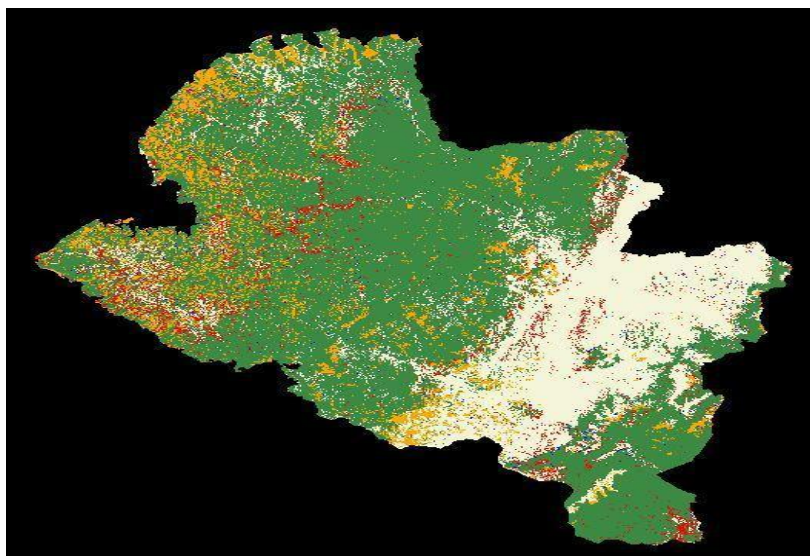


Fig. 4-32. Imagen facilitada por IDEAM.

Para ello se recurre a ERDAS , el mismo software que ya se utilizó antes para el recorte de imágenes, en el se encuentra una opción llamada Reproject Images.

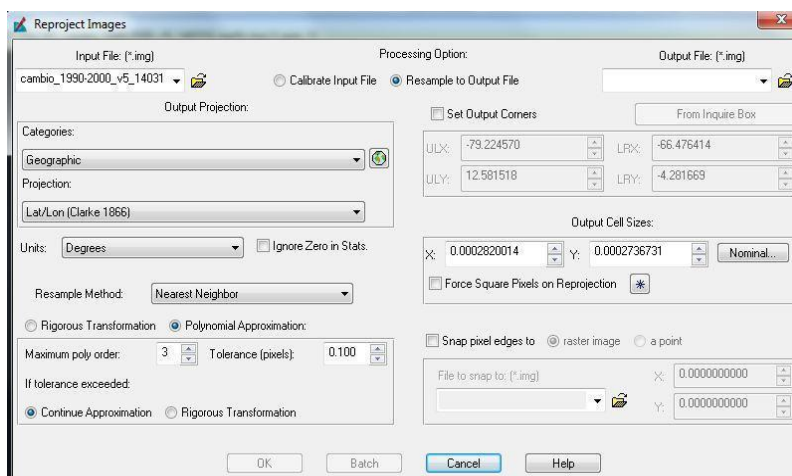


Fig. 4-33. Reproject

Se procede a realizar el cambio de la proyección espacial ya que el código que poseen las imágenes suministradas por el IDEAM tiene una proyección EPSG Código 3116 mientras que la proyección espacial del trabajo corresponde a EPSG Código 32618.

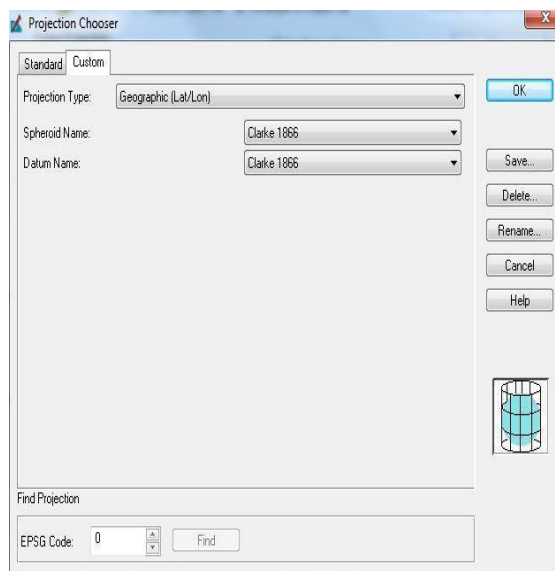


Fig. 4-34. Proyección Espacial

Una vez realizada la reproyección espacial es necesario indicar las coordenadas para realizar el corte en la imagen, acudiendo a la función Set Output Corners, procediendo a colocar las coordenadas.

Tabla 7. Set Output Corners.

ULX	=	ULY	=
253680.00		128760.00	
LRX	=	LRY	=
266670.00		110520.00	

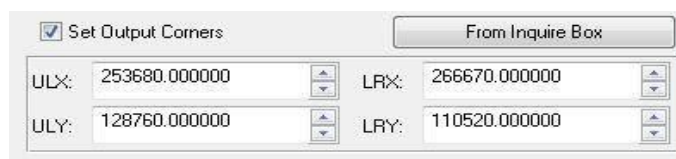


Fig. 4-35. Set Output Corners

Por último, se debe dejar el tamaño del pixel de 30 metros por 30 metros ya que las imágenes que suministro el IDEAM tienen una dimensión diferente, para esto se aplica la función Output Cell Sizes.

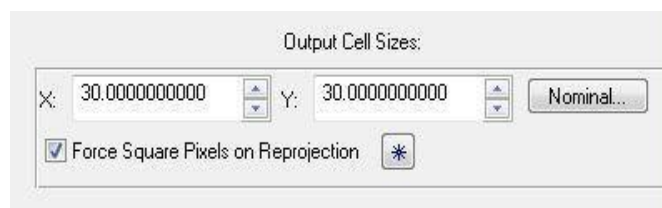


Fig. 4-36. Tamaño del Pixel.

Este procedimiento nos da como resultado la imagen clasificada de deforestación con las dimensiones exactas a las de las imágenes de estudio.

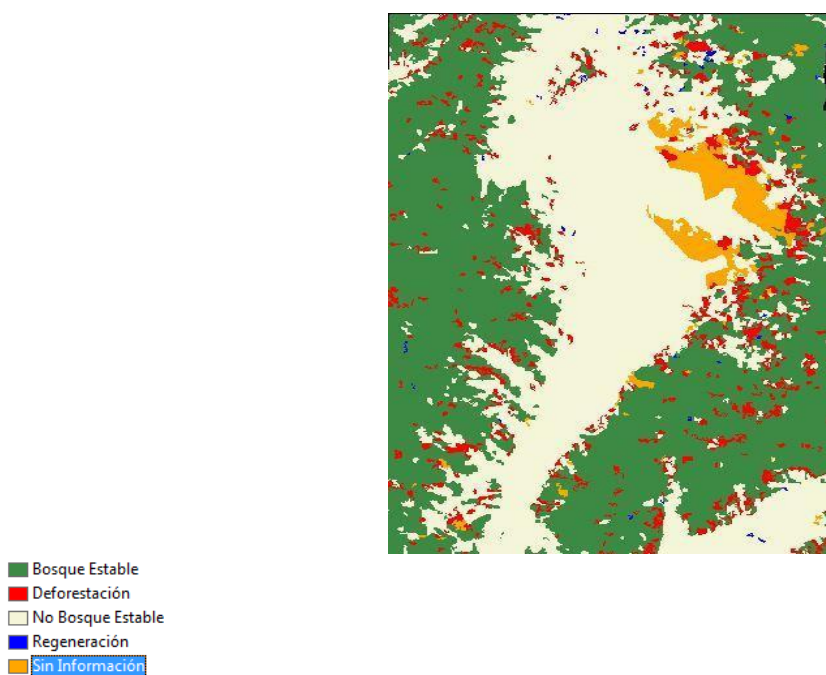


Fig. 4-37. Valores imágenes IDEAM

Se debe realizar el mismo tratamiento para las imágenes de deforestación que el realizado para las imágenes de la laguna de la cocha, es decir la importación y la extracción de sus características.

**4.1.6 Importación de imágenes a GRASS GIS.** Llegando a este punto del proceso se efectuó la descarga de las imágenes útiles, aplicando a las mismas diferentes software (Envi y Erdas) para el corte y la combinación de bandas. Con el insumo antes descrito el problema radica en extraer los valores de nivel digital que tienen estas bandas en cada uno de sus pixeles, como un valor numérico para ser empleado como dato en un proceso posterior de minería de datos, a efectos de analizarlos y procesarlos computacionalmente.

En este procedimiento, se emplea el programa de computación llamado GRASS GIS, versión 7.2. GRASS GIS, comúnmente conocido como GRASS. <https://grass.osgeo.org/> [23]

El programa enunciado se puede manipular en varios sistemas operativos, para la investigación que me ocupa, el mismo se aplicará al sistema operativo LINUX (Manjaro) de donde es nativo el software, razón por la cual proporciona un mejor desempeño y compatibilidad. La ventaja del Software antes enunciado, se traduce en que este posee una interface con R; R es un lenguaje de programación, de distribución libre, bajo licencia de GNU, que es muy utilizado en el cómputo estadístico, básico para extraer la información de datos de las imágenes a procesar. [24] [ftp://cran.r-project.org/pub/R/doc/contrib/Santana\\_El\\_arte\\_de\\_programar\\_en\\_R.pdf](ftp://cran.r-project.org/pub/R/doc/contrib/Santana_El_arte_de_programar_en_R.pdf)

Para la importación de archivos se desarrolló un script en python; con la finalidad de evitar pérdida de información puesto que el script procesa la totalidad del lote de imágenes. El script, busca las imágenes que se encuentra en el directorio, agotado este paso las importa directamente a un directorio en grass previamente establecido, tal como se describe a continuación

Script de importación de archivos

```

1. import os
2. import sys
3. import glob
4. import grass.script as grass
5.
6. def import_tifs(dirpath):
7.     for dirpath, dirname, filenames in os.walk(dirpath):
8.         for tif_file in filenames:
9.             if tif_file.lower().endswith('.tif') or tif_file.lower().endswith('.img'):
10.                full_path = os.path.join(dirpath, tif_file)
11.                grass.message('Importing %s -> %s@%s...' % (full_path, tif_file, dirpath))
12.                grass.run_command('r.in.gdal',input = full_path, output = tif_file[:-4])
13.
14. def main():
15.     path = easygui.diropenbox("Seleccione folder con imagenes")
16.     import_tifs(path)
17.
18. if __name__ == "__main__":
19.     main()

```

Una vez se corre el script se derivan los siguientes resultados:

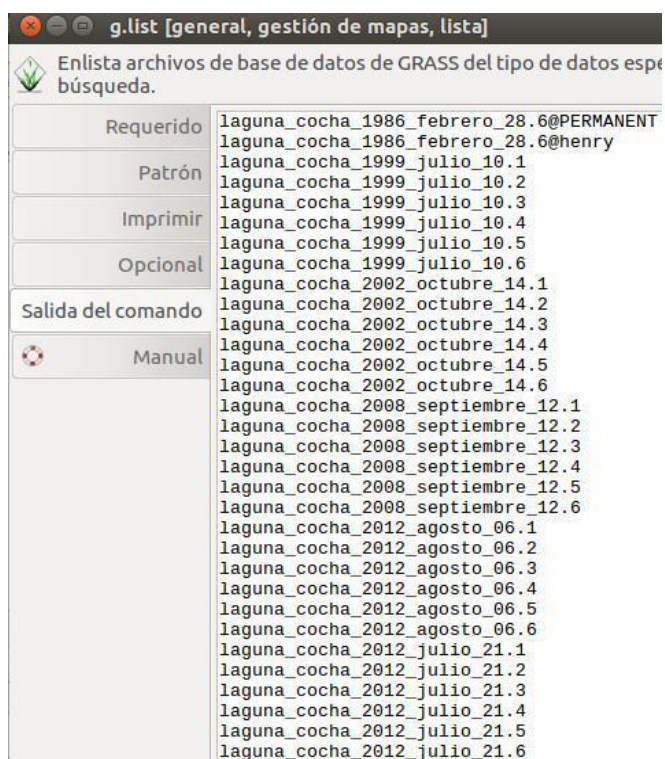


Fig. 4-38. Listado de Imágenes.

Surtida esta etapa se importan las imágenes del IDEAM, utilizando el script que se empleó para importar las imágenes precedentes.

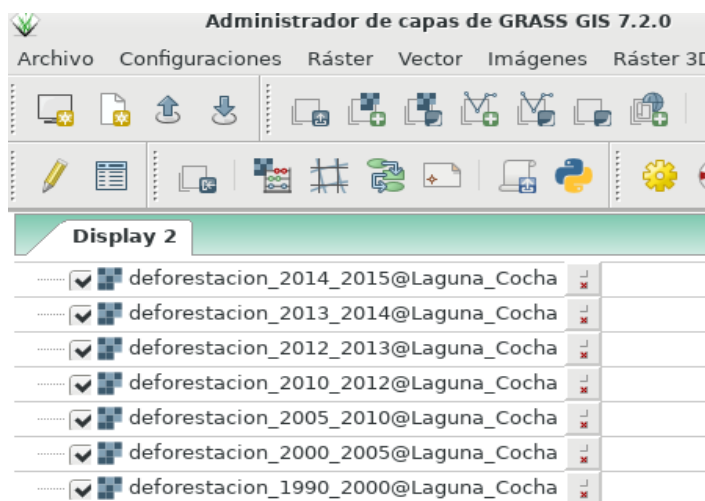


Fig. 4-39. Imágenes deforestación.

## 4.2 EXTRACCIÓN DE CARACTERÍSTICAS

Con la finalidad de extraer las características de las imágenes importadas con anterioridad, se recurrirá a programas de Información Geográfica, para el caso GRASS 7 y RStudio, El paquete proporciona facilidades para usar todos los comandos de GRASS desde la línea de comandos R. A continuación, se extraen las características de la imagen, desarrollando para ello un código que se basa en los comandos que se describen a continuación.

- EXECGRASS permite ejecutar comandos de GRASS desde R,
- READRAST es una función que permite leer archivos de grass en R cuando lo lee a R se vuelve SPATIALGRIDDATAFRAME , que se puede escribir otra vez a GRASS con la función denominada WRITERAST.

```
> execGRASS("g.list", parameters = list(type = "raster", pattern = "**1999**"))
laguna_cocha_1999_julio_10.1
laguna_cocha_1999_julio_10.2
laguna_cocha_1999_julio_10.3
laguna_cocha_1999_julio_10.4
laguna_cocha_1999_julio_10.5
laguna_cocha_1999_julio_10.6
>
```

Fig. 4-40. EXECGRASS

El comando readRAST es muy importante debido a que permite la extracción del ráster de la imagen. Ejemplo de readRAST:

```
henry@HenryPC: ~
> lco1986 <-readRAST(c("laguna_cocha_1986_febrero_28.1","laguna_cocha_1986_febre
ro_28.2","laguna_cocha_1986_febrero_28.3","laguna_cocha_1986_febrero_28.4","lagu
na_cocha_1986_febrero_28.5","laguna_cocha_1986_febrero_28.6"), cat=c(NULL))
> str(lco1986)
Formal class 'SpatialGridDataFrame' [package "sp"] with 4 slots
 ..@ data      :'data.frame': 264306 obs. of  6 variables:
 .. ..$ laguna_cocha_1986_febrero_28.1: int [1:264306] 57 57 58 56 54 55 54 55
55 54 ...
 .. ..$ laguna_cocha_1986_febrero_28.2: int [1:264306] 24 24 24 23 22 23 23 22
22 21 ...
 .. ..$ laguna_cocha_1986_febrero_28.3: int [1:264306] 18 20 21 19 17 17 18 18
16 16 ...
 .. ..$ laguna_cocha_1986_febrero_28.4: int [1:264306] 62 61 57 51 60 64 59 56
65 64 ...
 .. ..$ laguna_cocha_1986_febrero_28.5: int [1:264306] 36 39 43 38 31 28 33 37
29 30 ...
 .. ..$ laguna_cocha_1986_febrero_28.6: int [1:264306] 13 13 16 13 11 11 13 13
9 11 ...
 ..@ grid      :Formal class 'GridTopology' [package "sp"] with 3 slots
 .. ..@ cellcentre.offset: num [1:2] 253680 110520
 .. ..@ cellsize        : num [1:2] 30 30
 .. ..@ cells.dim       : int [1:2] 434 609
 ..@ bbox      : num [1:2, 1:2] 253665 110505 266685 128775
 .. - attr(*, "dinnames")=List of 2
 .. ..$ : NULL
 .. ..$ : chr [1:2] "min" "max"
 ..@ proj4string:Formal class 'CRS' [package "sp"] with 1 slot
 .. ..@ projargs: chr "+proj=utm +no_defs +zone=18 +a=6378137 +rf=298.257223
563 +towgs84=0,0,0,0,0,0,0 +to_meter=1"
```

Fig. 4-41. READRAST

En esta imagen se resalta el SPATIALGRIDDATAFRAME que permite almacenar en un objeto puntos, grillas, coordenadas, límites de datos que posean componente espacial.



La siguiente figura muestra un resumen del raster logrado con el comando anterior.

```
> str(lng1986F)
Formal class 'SpatialGridDataFrame' [package "sp"] with 4 slots
..@ data      : 'data.frame': 264306 obs. of  6 variables:
.. ..$ laguna_cocha_1986_febrero_28.1: Factor w/ 211 levels "45", "46", "47",...: 13 1
.. ..$ laguna_cocha_1986_febrero_28.2: int [1:264306] 24 24 24 23 22 23 22 22 21 ..
.. ..$ laguna_cocha_1986_febrero_28.3: int [1:264306] 18 28 21 19 17 17 18 16 16 ..
.. ..$ laguna_cocha_1986_febrero_28.4: int [1:264306] 62 61 57 51 60 64 59 56 65 64 ..
.. ..$ laguna_cocha_1986_febrero_28.5: int [1:264306] 36 39 43 38 31 28 33 37 29 30 ..
.. ..$ laguna_cocha_1986_febrero_28.6: int [1:264306] 13 13 16 13 11 11 13 13 9 11 ..
..@ grid      : Formal class 'GridTopology' [package "sp"] with 3 slots
.. ..@ cellcentre.offset: num [1:2] 253680 110520
.. ..@ cellsize       : num [1:2] 30 30
.. ..@ cells.dim      : int [1:2] 434 609
..@ bbox      : num [1:2, 1:2] 253665 110505 266685 128775
.. .. attr(*, "dimnames")=List of 2
.. .. ..$ : NULL
.. .. ..$ : chr [1:2] "min" "max"
..@ proj4string: Formal class 'CRS' [package "sp"] with 1 slot
.. ..@ projargs: chr "+proj=utm +no_defs +zone=18 +a=6378137 +rf=298.257223563 +tow
```

Fig. 4-42. Resumen Ráster.

La figura anterior indica cada una de las bandas y estas componen la imagen; las bandas se distinguen por número específico, en la parte entera se muestra que el archivo recorre del pixel 1 hasta el 264.306, lo cual se produce debido a que las dimensiones de la imagen fueron recortadas para que el programa las procesara de manera homogénea. Así, cada imagen presenta una dimensión de 434 filas por 609 columnas, la dimensión de cada pixel es de 30 x 30 metros, para un total de 264.306 pixel.

Se destaca que en este punto y gracias al comando READRAST se ha transformado cada banda de la imagen en un vector con 264306, vectores que permitirán analizar y consolidar el resultado del presente trabajo.

Luego de analizar los comandos que se emplean en el código de extracción de características, se procede a transcribir el mismo.

```
1. #se cargan las librerias necesarias
2. library("rgrass7")
3. library("rgdal")
4. setwd("/home/henrysolarte/rgrass_cocha")
5. #se crea una lista con los patrones de las imagenes importadas a grass
6. annos <- c("1990_abril", "1999_julio", "2002_octubre", "2005_noviembre",
7.           "2008_septiembre", "2010_noviembre", "2012_julio", "2013_enero", "2014_diciem
8. bre")
9. #se hacen las instrucciones para un spatialgriddataframe generico
10. p <- matrix(data=NA, nrow=8, ncol=8)
11. df = data.frame(z = c(1:6,NA,8,9),
12.                xc = c(1,1,1,2,2,2,3,3,3),
13.                yc = c(rep(c(0, 1.5, 3),3)))
14. coordinates(df) = ~xc+yc
15. gridded(df) = TRUE
16. df = as(df, "SpatialGridDataFrame")
17. #se copia ese spatialgriddataframe en las variables que van a guardar la información de
18. las imagenes dentro de el
19. abril_1990 <- df
20. julio_1999 <- df
21. octubre_2002 <- df
```

```

21. noviembre_2005 <- df
22. septiembre_2008 <- df
23. noviembre_2010 <- df
24. julio_2012 <- df
25. enero_2013 <- df
26. diciembre_2014 <- df
27.
28. #se crea una lista con esas variables (spatialgrid dataframes) para poderlo iterar
29. dfs = c(abril_1990, julio_1999, octubre_2002, noviembre_2005, septiembre_2008, noviembre_2010, julio_2012, enero_2013, diciembre_2014)
30. #se itera sobre cada uno de los años y la lista de variables para extraer la información masivamente de grass
31. for (i in seq(1:9)){
32.   ptrn=paste("\*", annos[i], "*\")
33.   ptrn=gsub(" ", "", ptrn)
34.   print(ptrn)
35.   #se lista la imagen con el patron que se ingreso en la variable annos
36.   a <- execGRASS("g.list", parameters = list(type = "raster", pattern = ptrn))
37.   b <- attributes(a)$resOut
38.   print(annos[i])
39.   dfs[i] <- readRAST(c(b[1],b[2],b[3],b[4],b[5],b[6]),
40.                     cat=c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE),
41.                     ignore.stderr = get.ignore.stderrOption(),
42.                     NODATA=0, plugin=get.pluginOption(), mapset=NULL,
43.                     useGDAL=get.useGDALOption(), close_OK=TRUE, drivename="HFA",
44.                     driverFileExt=".img", return_SGDF=TRUE)
45. }
46. #se extraen los spatialgriddataframe de la lista que se creo originalmente
47.
48. abril_1990 <- dfs[[1]]
49. julio_1999 <- dfs[[2]]
50. octubre_2002 <- dfs[[3]]
51. noviembre_2005 <- dfs[[4]]
52. septiembre_2008 <- dfs[[5]]
53. noviembre_2010 <- dfs[[6]]
54. julio_2012 <- dfs[[7]]
55. enero_2013 <- dfs[[8]]
56. diciembre_2014 <- dfs[[9]]

```

Con el código del script, se procede a correrlo en RStudio, extrayendo las características de las nueve imágenes con que se cuentan para realizar el estudio. Como resultado de la extracción de características, se obtienen las siguientes variables.

Variable	Class	Size
agosto_2012	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
enero_2016	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
febrero_1986	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
julio_1999	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
julio_2012	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
noviembre_2005	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
octubre_2002	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)
septiembre_2008	Large SpatialGridDataFrame	(264306 elements, 6.1 Mb)

Fig. 4-43. Extracción final

Se observa que las características fueron extraídas como `spatialGridDataFrame`., es decir, con características espaciales.

`SpatialGridDataFrame` es el tipo de objeto que R utiliza para almacenar capas ráster.

```

Global Environment
Values
agosto_2012      Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
..@ data : 'data.frame': 264306 obs. of 6 variables:
.. ..$ laguna_cocha_2012_agosto_06.1: int [1:264306] 36 35 36 36 36 35 35 35 34 34 ...
.. ..$ laguna_cocha_2012_agosto_06.2: int [1:264306] 27 26 28 28 27 27 27 26 26 26 ...
.. ..$ laguna_cocha_2012_agosto_06.3: int [1:264306] 24 22 22 23 22 21 23 20 20 21 ...
.. ..$ laguna_cocha_2012_agosto_06.4: int [1:264306] 67 68 68 65 61 63 62 62 60 60 ...
.. ..$ laguna_cocha_2012_agosto_06.5: int [1:264306] 39 31 33 34 33 31 33 33 25 26 ...
.. ..$ laguna_cocha_2012_agosto_06.6: int [1:264306] 22 17 17 17 17 19 20 17 16 14 ...
..@ grid :Formal class 'GridTopology' [package "sp"] with 3 slots
.. ..@ cellcentre.offset: num [1:2] 253680 110520
.. ..@ cellsize : num [1:2] 30 30
.. ..@ cells.dim : int [1:2] 434 609
..@ bbox : num [1:2, 1:2] 253665 110505 266685 128775
.. .. attr(*, "dimnames")=List of 2
.. .. ..$ : NULL
.. .. ..$ : chr [1:2] "min" "max"
..@ proj4string:Formal class 'CRS' [package "sp"] with 1 slot
.. ..@ projargs: chr "+proj=utm +no_defs +zone=18 +a=6378137 +rf=298.257223563 +towgs84=0,0,0,0,0,0 +t..."

```

Fig. 4-44. Extracción de características

A continuación, se aprecian los datos extraídos como `SpatialGridDataFrame`, en los cuales se observan las seis bandas de la imagen y dos campos adicionales que corresponden a los datos espaciales. Estos dos datos se refieren a la información espacial de la imagen.

juna_cocha_2012_agosto_06.1	laguna_cocha_2012_agosto_06.2	laguna_cocha_2012_agosto_06.3	laguna_cocha_2012_agosto_06.4	laguna_cocha_2012_agosto_06.5	laguna_cocha_2012_agosto_06.6	s1	s2
36	27	24	67	39	22	253680	128760
35	26	22	68	31	17	253710	128760
36	28	22	68	33	17	253740	128760
36	28	23	65	34	17	253770	128760
36	27	22	61	33	17	253800	128760
35	27	21	63	31	19	253830	128760
35	27	23	62	33	20	253860	128760
35	26	20	62	33	17	253890	128760

Fig. 4-45. Datos de las imágenes.

### 4.2.1 Extracción de características imagen IDEAM.

Una vez importadas las imágenes con RSTUDIO se extraen con el mismo script las características, tal como se hizo con las imágenes anteriores.

#### 4.2.1.1 Script de extracción de características de deforestación.

```

1. #se cargan las librerías necesarias
2. library("rgrass7")
3. library("rgdal")
4. setwd("~/rgrass_cocha")#se crea una lista con los patrones de las imágenes importadas a
grass
5. annos <-
c("deforestacion_1990_2000", "deforestacion_2000_2005", "deforestacion_2005_2010", "
deforestacion_2010_2012", "deforestacion_2012_2013", "deforestacion_2013_2014", "def
orestacion_2014_2015")
6. #se hacen las instrucciones para un spatialgriddataframe genérico
7. p <- matrix(data=NA, nrow=7, ncol=7)
8. df = data.frame(z = c(1:6,NA,8,9),
9. xc = c(1,1,1,2,2,2,3,3,3), yc = c(rep(c(0, 1.5, 3),3)))
10. coordinates(df) = ~xc+yc
11. gridded(df) = TRUE
12. df = as(df, "SpatialGridDataFrame")
13. #se copia ese espacialgriddataframe en las variables que van a guardar la información de
las imágenes dentro de el
14. deforestacion_1990_2000 <- df
15. deforestacion_2000_2005 <- df
16. deforestacion_2005_2010 <- df
17. deforestacion_2010_2012 <- df
18. deforestacion_2012_2013 <- df
19. deforestacion_2013_2014 <- df
20. deforestacion_2014_2015 <- df
21. #se crea una lista con esas variables (spatialgrid dataframes) para poderlo iterar
22. dfs = c(def_2000,def_2000_2005,def_2005_2010, def_2010_2012, def_2012_2013, def_2
013_2014, def_2014_2015)
23. #se itera sobre cada uno de los años y la lista de variables para extraer la información
masivamente de grass
24. for (i in seq(1:7)){
25. ptrn=paste("\"*" ,annos[i], "*"")
26. ptrn=gsub(" " , "" ,ptrn)
27. print(ptrn)
28. #se lista la imagen con el patron que se ingreso en la variable annos
29. a <- execGRASS("g.list", parameters = list(type = "raster", pattern = ptrn))
30. b <- attributes(a)$resOut
31. print(annos[i])
32. dfs[i] <- readRAST(c(b[1],b[2],b[3],b[4],b[5],b[6]),

```

```

33.     cat=c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE),
34.     ignore.stderr = get.ignore.stderrOption(),
35.     NODATA=0, plugin=get.pluginOption(), mapset=NULL,
36.     useGDAL=get.useGDALOption(), close_OK=TRUE, drivename="HFA",
37.     driverFileExt=".img", return_SGDF=TRUE)
38. }
39. #se extraen los spatialgridadataframe de la lista que se creó originalmente
40. deforestacion_1990_2000 <- dfs[[1]]
41. deforestacion_2000_2005 <- dfs[[2]]
42. deforestacion_2005_2010 <- dfs[[3]]
43. deforestacion_2010_2012 <- dfs[[4]]
44. deforestacion_2012_2013 <- dfs[[5]]
45. deforestacion_2013_2014 <- dfs[[6]]
46. deforestacion_2014_2015 <- dfs[[7]]

```

En este momento de la investigación se extrajeron las características de las imágenes a estudiar y de las imágenes del modelo a tomar, características estas que son dadas directamente por la clasificación efectuada previamente por el IDEAM. Así y por tratarse de imágenes procesadas por el IDEAM, esta entidad otorga cinco valores diferentes que indican diferentes contextos, así:



Fig. 4-46. Valores.

En consecuencia, los valores dados por el IDEAM, se transformaron según la necesidad de la investigación conforme a numeración que representa características particulares en la imagen, obteniendo en el *raster* los siguientes valores:

- 1) Bosque estable
- 2) Deforestación
- 3) No bosque estable
- 4) Regeneración

Se obtienen datos de la deforestación de las imágenes suministradas por el IDEAM,

Data	
def1990_2000	264306 obs. of 1 variable
def2000_2005	264306 obs. of 1 variable
def2005_2010	264306 obs. of 1 variable
def2010_2012	264306 obs. of 1 variable
def2012_2013	264306 obs. of 1 variable
def2013_2014	264306 obs. of 1 variable
def2014_2015	264306 obs. of 1 variable

Fig. 4-47. Imágenes deforestación.

Entonces, de las imágenes referidas se generan las imágenes a estudiar

abril_1990	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
diciembre_2014	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
enero_2013	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
julio_1999	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
julio_2012	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
noviembre_2005	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
noviembre_2010	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
octubre_2002	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
septiembre_2008	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)

Fig. 4-48. Imágenes de estudio.

Se observa la imagen antes de adjuntar el vector de deforestación

Data	
def1990_2000	264306 obs. of 1 variable
def2000_2005	264306 obs. of 1 variable
def2005_2010	264306 obs. of 1 variable
def2010_2012	264306 obs. of 1 variable
def2012_2013	264306 obs. of 1 variable
def2013_2014	264306 obs. of 1 variable
def2014_2015	264306 obs. of 1 variable

Values	
abril_1990	Large SpatialGridDataFrame (264306 elements, 6.1 Mb)
..@ data : 'data.frame': 264306 obs. of 6 variables:	
.. ..\$ laguna_cocha_1990_abril_28.1:	int [1:264306] 255 255 255 255 255 255 255 ...
.. ..\$ laguna_cocha_1990_abril_28.2:	int [1:264306] 148 149 150 152 153 149 144 ...
.. ..\$ laguna_cocha_1990_abril_28.3:	int [1:264306] 159 159 159 163 165 162 156 ...
.. ..\$ laguna_cocha_1990_abril_28.4:	int [1:264306] 157 158 159 162 162 159 155 ...
.. ..\$ laguna_cocha_1990_abril_28.5:	int [1:264306] 177 178 178 180 180 177 176 ...
.. ..\$ laguna_cocha_1990_abril_28.6:	int [1:264306] 96 96 97 95 97 98 95 98 93 9...
..@ grid :	Formal class 'GridTopology' [package "sp"] with 3 slots
.. ..@ cellcentre.offset:	num [1:2] 253680 110520
.. ..@ cellsize :	num [1:2] 30 30
.. ..@ cells.dim :	int [1:2] 434 609

Fig. 4-49. Imágenes deforestación y estudio.

En Rstudio se utiliza un pequeño comando, ilustrado a continuación para incluir la deforestación dentro de la imagen abril\_1990

```
abril_1990$def1990=def1990_2000$deforestacion_1990_2000
```

Fig. 4-50. Comando para incluir deforestación

Se genera la imagen de abril\_1190 mas la deforestación del año 1990\_2000

abril_1990		Large SpatialGridDataFrame (264306 elements, 7.1 Mb)	
..@ data :	'data.frame':	264306 obs. of 7 variables:	
.. ..\$ laguna_cocha_1990_abril_28.1:	int [1:264306]	255 255 255 255 255 255 255 ...	
.. ..\$ laguna_cocha_1990_abril_28.2:	int [1:264306]	148 149 150 152 153 149 144 ...	
.. ..\$ laguna_cocha_1990_abril_28.3:	int [1:264306]	159 159 159 163 165 162 156 ...	
.. ..\$ laguna_cocha_1990_abril_28.4:	int [1:264306]	157 158 159 162 162 159 155 ...	
.. ..\$ laguna_cocha_1990_abril_28.5:	int [1:264306]	177 178 178 180 180 177 176 ...	
.. ..\$ laguna_cocha_1990_abril_28.6:	int [1:264306]	96 96 97 95 97 98 95 98 93 9...	
.. ..\$ def1990 :	int [1:264306]	1 1 1 1 1 1 1 1 1 1 ...	

Fig. 4-51. Inserción de deforestación en imagen.

En este momento la información está lista para realizar el análisis correspondiente, al contar con las seis bandas y el vector de deforestación que se constituye en la clase.

### 4.3 MINERÍA DE DATOS

Lo que se intenta es que a partir de grandes cantidades de datos se descubra cúmulos de información a partir de las bases que posee esta investigación, es decir, se realizará minería de datos y se va a utilizar para ello aprendizaje supervisado. El aprendizaje supervisado es el análisis discriminante en el que un patrón de entrada está asignado para una clase predefinida Disponible en

<https://repository.javeriana.edu.co/bitstream/handle/10554/3826/RodriguezCampoJuanCarlos2008.pdf?sequence=1> [25]

Para el caso de esta investigación del aprendizaje supervisado es la clase deforestación que fue realizada por el IDEAM, al suministrar la imagen segmentando la deforestación.

Los resultados previstos con antelación, se pueden categorizar de la siguiente manera:

- True Positive (Verdaderos Positivos)
- Toda la deforestación que fue clasificada como deforestación.
- False Positive (Falsos Positivos)
- Todos los diferentes de deforestación que fueron clasificados como deforestación.
- True Negative (Verdadero Negativo)
- Toda la deforestación que fue clasificados como diferentes de deforestación.
- False Negative (Falso Negativo)
- Todos los diferentes de deforestación que fueron clasificados como diferentes de deforestación.

Aunado a la categorización anterior se debe tener en cuenta como elementos relacionales dos factores adicionales que son la sensibilidad y la especificidad.

Sensibilidad es igual a los verdaderos positivos entre la sumatoria de los verdaderos positivos mas los falsos negativos.

La especificidad es la resultante del cociente entre verdaderos negativos (VN) dividido entre la suma de los verdaderos negativos mas los falsos positivos (VN + FP). Disponible en [http://www.scielo.edu.uy/scielo.php?script=sci\\_arttext&pid=S1688-04202009000100005](http://www.scielo.edu.uy/scielo.php?script=sci_arttext&pid=S1688-04202009000100005) [26]

El aprendizaje se desarrollara a través de una red neuronal por la característica de los datos, muchas entradas y una clase. La red neuronal a emplear se trata de KERAS un algoritmo muy poderoso y muy fácil de implementar

B. Keras es una red neuronal de alto nivel escrita en Python . Keras es una poderosa biblioteca Python fácil de usar para desarrollar y evaluar modelos de aprendizaje en profundidad. [deep learning](#) models. Véase <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/> [27]

Envuelve las eficientes bibliotecas de cálculo numérico Theano y TensorFlow y le permite definir y entrenar modelos de redes neuronales en unas breves líneas de código. Puede consultarse en <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/> [27]

Para correr la librería Keras se utiliza anaconda. Anaconda es el ecosistema de Python datos más popular y confiable del mundo. Disponible en <https://www.continuum.io/what-is-anaconda> [28]

Para correr el algoritmo en keras , y crear un adecuado modelo es necesario seguir los siguientes pasos:

- 1) Carga de datos.
- 2) Definir Modelo.
- 3) Compile el modelo.
- 4) Modelo apto.
- 5) Evaluar el modelo.
- 6) Atelo todo junto.

Instalación de Keras en anaconda in Windows

- 1 Instalar Python y configurarlo.
- 2 Instalar SciPy y configurarlo.
- 3 Instalar Keras y backend que puede ser Theano o TensorFlow y configurarlo.
- 4 Instalar Git para Windows

Una vez instalado anaconda y todas las librerías necesarias para que corra keras iniciamos la consola para el estudio de este proyecto utilizamos spider.



Programa:

```

1. # Create your first MLP in Keras
2. from keras.models import Sequential
3. from keras.layers import Dense
4. import numpy
5. import rp
6. # fix random seed for reproducibility
7. numpy.random.seed(7)
8. # load pima indians dataset
9. ds_train = numpy.loadtxt("setenta18.csv", delimiter=";")
10. # split into input (X) and output (Y) variables
11. X_tr = ds_train[:,0:6]
12. Y_tr = ds_train[:,6]
13. # create model
14. model = Sequential()
15. model.add(Dense(12, input_dim=6, activation='relu'))
16. model.add(Dense(8, activation='relu'))
17. model.add(Dense(1, activation='sigmoid'))
18. # Compile model
19. model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
20. # Fit the model
21. model.fit(X_tr, Y_tr, epochs=150, batch_size=10)
22. # evaluate the model
23. scores = model.evaluate(X_tr, Y_tr)
24. print("entrenamiento terminado")
25. print("\n%s: %.2f%%" % (model.metrics_names[0], scores[0]*100))
26. print("\n%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
27. print("scores")
28. print(scores)
29. ds_test = numpy.loadtxt("treinta18.csv", delimiter=";")
30. # split into input (X) and output (Y) variables
31. X_ts = ds_test[:,0:6]
32. Y_ts = ds_test[:,6]
33. r = model.predict(X_ts)
34. r2 = zip(Y_ts, r)
35. print(r2)
36. result = rp.calcular_indices(r2, [0, 2])
37. print("result\n")
38. print(result)
39. #print("Y_ts\n")
40. #print(Y_ts)
41. #print("r\n")
42. #print(r)

```

### A. Entrenamiento del algoritmo

Se dispone de seis bandas que proporciona la imagen satelital, también se cuenta con una clase que la proporciona la imagen de deforestación. En total se posee seis bandas y una clase que mide la deforestación, en la matriz precedente las seis columnas iniciales representan la banda que se aplica a cada dato, según la clasificación *Landsat*, en tanto que la séptima columna refleja la semilla, para el caso la deforestación. Se precisa que estas columnas que representan bandas fueron extraídas de una multiplicidad de columnas, entre otros, ya que las mismas representan el objeto de estudio

Tabla 8. Datos a investigar.

	b1	b2	b3	b4	b5	b6	df
1	78	32	28	56	46	21	5
2	78	34	31	60	50	21	5
3	83	37	33	64	55	24	5
4	103	45	41	72	71	36	5
5	126	55	55	84	84	42	5
6	128	56	56	84	83	41	5
7	116	51	50	79	75	35	5
8	100	44	43	73	65	29	5
9	89	38	35	70	57	24	5

De tal manera que se realizó la siguiente paridad para estas imágenes de la siguiente manera

Tabla 9. Imágenes de estudio y deforestación para análisis.

•Laguna_cocha_1990_abril_28	Cambio_1990-2000
•Laguna_cocha_1999_julio_10	
•Laguna_cocha_2002_octubre_14	Cambio_2000-2005
•Laguna_cocha_2005_noviembre_23	
•Laguna_cocha_2008_septiembre_12	Cambio_2005-2010
•Laguna_cocha_2010_noviembre_11	
•Laguna_cocha_2012_julio_21	Cambio_2010-2012
•Laguna_cocha_2013_enero_29	Cambio_2012-2013
•Laguna_cocha_2014_diciembre_10	Cambio_2013-2014

Los datos de la deforestación van desde el número 1 al número 5

- 1) Bosque estable
- 2) Deforestación
- 3) No bosque estable
- 4) Regeneración
- 5) Sin Información

Para efectos del ejercicio estos cinco datos se los ha cambiado por deforestación y no deforestación. Es decir, el valor de 2 de deforestación fue cambiado por el número 1 y los demás valores fueron cambiados por cero. Deforestación = 1, No deforestación = 0

Tabla 10. Deforestación y no deforestación.

	b1	b2	b3	b4	b5	b6	df		b1	b2	b3	b4	b5	b6	df
1	78	32	28	56	46	21	0	1	36	33	32	29	61	39	1
2	78	34	31	60	50	21	0	2	36	33	32	29	58	39	1
3	83	37	33	64	55	24	0	3	38	35	33	31	57	41	1
4	103	45	41	72	71	36	0	4	36	33	32	29	61	39	1
5	126	55	55	84	84	42	0	5	36	33	32	30	59	40	1
6	128	56	56	84	83	41	0	6	36	33	31	29	58	38	1
7	116	51	50	79	75	35	0	7	37	34	32	29	64	39	1
8	100	44	43	73	65	29	0	8	37	33	31	28	58	38	1
9	89	38	35	70	57	24	0	9	38	35	34	33	61	51	1

La tabla que integra los años donde se introdujeron todos los datos de la investigación, contiene un total de 2.378.754 datos que corresponden a nueve imágenes procesadas con 264306 datos por cada una de ellas.

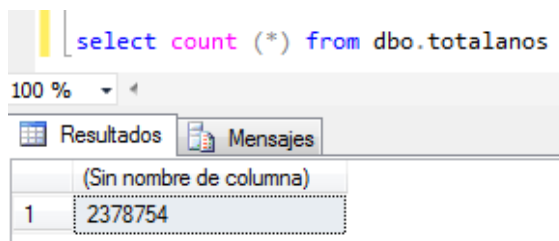


Fig. 4-52. Datos totales.

De los cuales 47822 son datos de deforestación

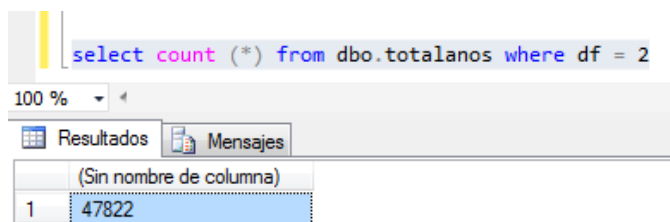



Fig. 4-53. Datos de deforestación.

Por tanto al contar con 47822 datos de deforestación, se deben tomar aleatoriamente otros 47822 datos que pertenecen a datos de no deforestación.

```

select top 5314 newid() randomId, * into #temp from dbo.totalanos where origen = '1990' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5314 newid() randomId, * from dbo.totalanos where origen = '2014' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5314 newid() randomId, * from dbo.totalanos where origen = '2013' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5314 newid() randomId, * from dbo.totalanos where origen = '1999' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5314 newid() randomId, * from dbo.totalanos where origen = '2012' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5313 newid() randomId, * from dbo.totalanos where origen = '2005' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5313 newid() randomId, * from dbo.totalanos where origen = '2010' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5313 newid() randomId, * from dbo.totalanos where origen = '2002' and df <> 2 order by randomId
insert into #temp (randomId,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 5313 newid() randomId, * from dbo.totalanos where origen = '2008' and df <> 2 order by randomId
select count (*) from #temp

```



(Sin nombre de columna)
47822

Fig. 4-54. Datos de no deforestación.

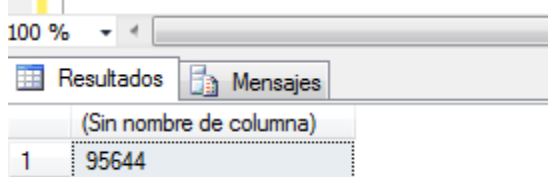
Existen 47822 df = 1 se tomaron otros 47822 df = 0 estos se toman de manera aleatoria.

En total se genera una tabla con 95644 elementos

```

select count (*) from defynodef

```



(Sin nombre de columna)
95644

Fig. 4-55. Datos de deforestación y no deforestación.


De estos datos se toma un 70% para entrenamiento y un 30% para validación, por cuanto se está realizando un aprendizaje supervisado.

Se selecciona el 70% que da un total de 66942 registros

```

select top 8956 Id,s1,s2,origen,b1,b2,b3,b4,b5,b6,df into #temp from defynodef where origen = '1990'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 9482 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '1999'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 9530 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2002'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2005'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2008'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2010'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2012'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2013'
insert into #temp (Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2) select top 7438 Id,b1,b2,b3,b4,b5,b6,df,origen,s1,s2 from defynodef where origen = '2014'
select count (*) from #temp

```



(Sin nombre de columna)
66942

Fig. 4-56. (70% Consolidado).

Con el script se encuentra el restante 30%.

```
select d.* into #resultantes from defynodef d left join #temp t on d.id=t.id where t.id is null;
```

30 %

Mensajes

(28702 filas afectadas)

Fig. 4-57. (30 % consolidado).

Por tanto, 70% genera 66942 registros

Id	b1	b2	b3	b4	b5	b6	df	s1	s2	origen	
1	64DD15BE-A951-4093-B0CA-C7D270D76D95	43	39	40	76	71	40	2	264660	117360	2002
2	022E65F-06DC-4D97-8DD2-C7D3AFE8755A	41	33	32	77	63	38	2	265710	123330	2002
3	9BC74DE2-F949-4597-A156-C7E408E57924	40	33	31	69	54	30	2	254340	124110	2002
4	153CD8F9-99E7-4EC7-9C50-C7ED74ED3F5F	42	34	35	60	69	46	2	266490	120870	2002
5	68EA23AB-1AE4-4E63-B03F-C7F853DAD487	41	36	36	73	56	33	2	255930	128760	2002
6	FFF42F6E-FA26-48ED-9A78-C7F87C850E5D	37	31	25	77	45	23	2	266190	127380	2002
7	0881C877-C89F-49D1-BAF5-C800B6147249	42	39	34	79	62	35	2	265140	115410	2002
8	AC8A1536-AC6B-4A08-BD58-C8125DF34A3A	52	44	43	77	67	41	2	264150	112080	2002
9	DA685DAF-1F6A-4600-B522-C818E1ACD87E	36	28	21	43	26	15	2	257250	114060	2002
10	650BA4F5-0331-485D-B037-C82E73D67B4A	44	41	38	89	80	44	2	258660	112440	2002
11	D39A63C3-E6B5-4814-BD85-C835218634A	38	30	25	65	46	25	2	262290	128040	2002
12	19B899B5-7384-4978-A8FE-C861655D50D6	42	33	32	52	54	31	2	257220	114480	2002

Consulta ejecutada correctamente. DD-PC (12.0 RTM) UACT\SoporteUact (52) bd\_completa\_4 | 00:00:01 | 66942 filas

Fig. 4-58. Consulta de datos 70% consolidado.

30% genera 28702 registros

Id	b1	b2	b3	b4	b5	b6	df	s1	s2	origen	
1	A61DAAB1-2CD1-4087-80A1-833954E171AF	194	85	89	97	107	55	2	265140	115560	1990
2	6B16B95E-AD42-4F28-951E-83C804C7AE5D	195	88	94	112	131	79	2	255270	124680	1990
3	E4F43691-DC48-4ED7-A0DC-84180F60743D	188	87	89	105	129	75	2	264060	112110	1990
4	E894BE98-A1F0-4ADA-BA2B-847E7D4D3734	71	31	27	80	59	22	2	263940	124500	1990
5	6076B8CA-465F-4BD4-926C-84C880A39E47	85	36	36	60	71	33	2	262770	127680	1990
6	3C2E146E-4157-4C06-A5FA-852955FE45D3	120	53	50	78	67	38	2	253950	120990	1990
7	7F68A81B-7E9D-4D13-8AA3-852D293014A4	85	36	32	57	52	25	2	265410	123270	1990
8	24E31361-7AE0-46FB-809D-854A6FA0B0CC	92	40	36	73	62	28	2	262800	118590	1990
9	2D1F0B4F-4940-4D72-9127-859A7D18ED50	84	40	36	105	95	36	2	263220	123990	1990
10	E5DA8070-CC86-48C6-ACCA-85CA97E9E517	85	37	33	81	63	25	2	261780	126120	1990
11	ADA3CE35-BF5D-42DF-92AC-85FF238A12CD	184	85	86	116	123	69	2	257190	119550	1990
12	44FD97E2-BC32-41E2-BFED-861DC9F0C443	87	42	36	104	92	34	2	263160	124230	1990
13	879489F3-5B80-4539-A216-86233F2A6CCE	83	38	31	106	74	27	2	262920	125340	1990
14	F8DFA9F3-889A-48F7-ARAD-86528651D2FR	114	49	47	78	80	38	2	266610	111780	1990

Consulta ejecutada correctamente. DD-PC (12.0 RTM) UACT\SoporteUact (52) bd\_completa\_4 | 00:00:00 | 28702 filas

Fig. 4-59. Consulta de datos 30% consolidado.

En este momento se proceden a eliminar los datos que no se necesitan en este caso las columnas Id, s1, s2, origen.

Con los resultados anteriores, es decir, la muestra que corresponde tanto a los porcentajes del 70 y 30%, se procede a correr el algoritmo.

Después de hacer un recorrido a los datos, así

```
66942/66942 [=====] - 10s - loss: 0.3421 - acc: 0.8178
Epoch 146/150
66942/66942 [=====] - 9s - loss: 0.3416 - acc: 0.8174
Epoch 147/150
66942/66942 [=====] - 11s - loss: 0.3424 - acc: 0.8163
Epoch 148/150
66942/66942 [=====] - 8s - loss: 0.3419 - acc: 0.8165
Epoch 149/150
66942/66942 [=====] - 11s - loss: 0.3415 - acc: 0.8182
Epoch 150/150
66942/66942 [=====] - 11s - loss: 0.3415 - acc: 0.8169
65760/66942 [=====>.] - ETA: 0s
entrenamiento terminado
```

Fig. 4-60. Finalización entrenamiento consolidado.

Los resultados de los estudios una vez ejecutado el algoritmo keras y conforme a lo establecido en la parte inicial de este paso (Minería de datos) son:

- True Positive (Verdaderos Positivos)
- Toda la deforestación que fue clasificada como deforestación.
- False Positive (Falsos Positivos).
- Todos los diferentes de deforestación que fueron clasificados como deforestación.
- True Negative (Verdadero Negativo)
- Toda la deforestación que fue clasificados como diferentes de deforestación.
- False Negative (Falso Negativo)

Todos los diferentes de deforestación que fueron clasificados como diferentes de deforestación.

Esto para el caso de estudio, se refleja en los siguientes valores:

VP = 294  
 VN = 14116  
 FP = 0  
 FN = 14292  
 Sensibilidad = 0.02  
 Especificidad = 1

Debido a que los resultados no fueron los mejores se decide hacer un intento diferente. El procedimiento que se realiza es el mismo solamente que esta vez se utiliza la deforestación del año 1990 para todas las imágenes y según esto analizar los resultados.

Tabla 11. Datos y deforestación 1990.

• Laguna_cocho_1990_abril_28	Cambio_1990-2000
• Laguna_cocho_1999_julio_10	Cambio_1990-2000
• Laguna_cocho_2002_octubre_14	Cambio_1990-2000
• Laguna_cocho_2005_noviembre_23	Cambio_1990-2000
• Laguna_cocho_2008_septiembre_12	Cambio_1990-2000
• Laguna_cocho_2010_noviembre_11	Cambio_1990-2000
• Laguna_cocho_2012_julio_21	Cambio_1990-2000
• Laguna_cocho_2013_enero_29	Cambio_1990-2000
• Laguna_cocho_2014_diciembre_10	Cambio_1990-2000

Los datos de la deforestación van desde el número 1 al número 5

- 1) Bosque estable
- 2) Deforestación
- 3) No bosque estable
- 4) Regeneración
- 5) Sin Información

Para efectos del ejercicio estos cinco datos se los ha cambiado por deforestación y no deforestación. Es decir el valor de 2 de deforestación fue cambiado por el número 1 y los demás valores fueron cambiados por cero.

Deforestación = 1  
No deforestación = 0

The screenshot shows a SQL query window with the following text: `select count (*) from defynodef where df = 2`. Below the query, there are tabs for 'Resultados' and 'Mensajes'. The 'Resultados' tab is active, showing a table with one column labeled '(Sin nombre de columna)' and one row with the value '146493'.

Fig. 4-61. Deforestación 1990.

En este caso los datos de deforestación son 146.493.

Se observa que los datos que se tienen de deforestación son mayores por que a medida que ha avanzado el tiempo la deforestación ha ido disminuyendo, esto, según los datos recolectados y analizados.

Por tanto, se toman otros valores aleatoriamente que representen la no deforestación, este dato debe ser igual a 146493

The screenshot shows a SQL query window with the following text: `select count (*) from defynodef where df <> 2`. Below the query, there are tabs for 'Resultados' and 'Mensajes'. The 'Resultados' tab is active, showing a table with one column labeled '(Sin nombre de columna)' and one row with the value '146493'.

Fig. 4-62. No deforestación 1990.

Esto conduce a un total de 292.986 datos

id	b1	b2	b3	b4	b5	b6	df	origen	s1	s2	
190542	70995474-19E4-4B09-9E83-05C1A2F32C94	62	54	49	1...	71	38	5	2013	258690	118680
190543	DBE4996B-3D54-44F9-AF81-05C1A6832470	41	33	25	92	42	21	1	2013	265290	114300
190544	F7481A1F-5B1A-4C85-891C-05C1BF888C26	41	28	23	16	11	10	5	2013	261150	119490
190545	FFB04005-B680-4583-98A2-05C1C0CEFB01	62	48	44	76	45	28	1	2013	257070	118200
190546	5C6C2D4E-CCC8-4FDC-9835-05C230E2E568	1...	86	92	1...	90	70	1	2013	254160	111060
190547	EF6A480C-FC3E-4683-9CDA-05C24887E172	48	31	27	24	17	14	5	2013	263850	121680
190548	E0D298FD-46FE-4367-9EBC-05C260947F2D	43	36	30	84	51	27	5	2013	264450	121590
190549	80EA1CCB-194F-4AFB-AD51-05C2658B83F3	40	32	26	71	42	22	5	2013	253710	127320
190550	08256AA3-8C26-4021-8358-05C2B8B8950C	40	31	27	75	33	19	1	2013	254850	123870
190551	107A7B67-519A-4DE8-84ED-05C2E081455D	42	32	27	78	41	21	1	2013	265950	116520
190552	1EF2177B-1ADA-4C8E-8E9A-05C312BA28CD	64	56	52	1...	78	44	5	2013	258750	118920
190553	85D27BBA-3868-4E43-81D7-05C31A89012D	1...	84	86	1...	86	67	1	2013	254130	111150

Fig. 4-63. Deforestación y no deforestación 1990.

De igual manera que en la ilustración que antecede se toma aleatoriamente el 70% para entrenamiento y el 30% restante para validación, generando setenta por ciento, es decir, 205.083 Datos

b1	b2	b3	b4	b5	b6	df	
7239	57	45	37	65	63	2	
7240	60	52	41	1...	89	43	2
7241	62	55	52	85	96	54	2
7242	59	46	41	72	85	46	2
7243	59	49	36	90	70	34	2
7244	60	54	43	1...	96	43	2
7245	56	47	34	1...	86	40	2
7246	58	46	36	76	67	33	2
7247	62	53	48	79	103	60	2
7248	57	44	35	76	66	31	2
7249	58	48	37	1...	91	41	2

Fig. 4-64. Deforestación 1990.

En consecuencia, el treinta por ciento restante, equivale a 87.903 datos

b1	b2	b3	b4	b5	b6	df	
1	46	33	28	69	39	21	1
2	42	34	26	101	53	24	1
3	98	80	72	84	69	52	1
4	47	34	31	32	22	19	3
5	40	25	20	14	10	9	5
6	66	54	52	88	48	32	1
7	42	35	30	90	56	32	1
8	214	190	207	165	162	139	1
9	42	33	26	100	48	23	1
10	52	48	49	98	92	55	5
11	50	40	37	80	49	28	1
12	47	39	37	79	53	30	1

Fig. 4-65. No deforestación 1990.

Contando con los datos precedente, se corre el algoritmo gkeras, los resultados son similares al ejercicio anterior

$$VP = 1$$

$$VN = 28$$

$$FP = 0$$

$$FN = 87874$$

$$\text{Sensibilidad} = 0$$

$$\text{Especificidad} = 1$$

Datos de sensibilidad y especificidad corrido algoritmo keras



Tabla 12. Consolidado de sensibilidad y especificidad.

DATOS	DEFORESTACIÓN CONSOLIDADA	DEFORESTACIÓN 1990-2000
VP	294	1
VN	14116	28
FP	0	0
FN	14292	87874
SENSIBILIDAD	0.02	0
ESPECIFICIDAD	1	1
MUESTRA	66942	205083

Todos los resultados arrojados suministraron un alto número de falsos negativos.

Se recurre entonces a realizar un PCA para verificar si las variables son o no aptas para su análisis.

PCA se utiliza para la reducción de dimensión de los datos originales para su posterior reconstrucción. Cuantificar el error de reconstrucción en función de la reducción de dimensiones aplicada. Disponible en [http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patrones-70038/contenidos/Ejercicios/trabajo\\_1\\_rpyrn\\_2009-10.pdf](http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patrones-70038/contenidos/Ejercicios/trabajo_1_rpyrn_2009-10.pdf) [29]

Lo que se busca es disminuir el número de bandas para poder diagramar los puntos de la no deforestación y la deforestación y observar donde aplica los puntos. El siguiente sería un ejemplo de un resultado esperado:

El clasificador al que se acudió, es KERAS busca clasificar los datos, con el mínimo error de clasificación, similar a lo que muestra la siguiente gráfica.

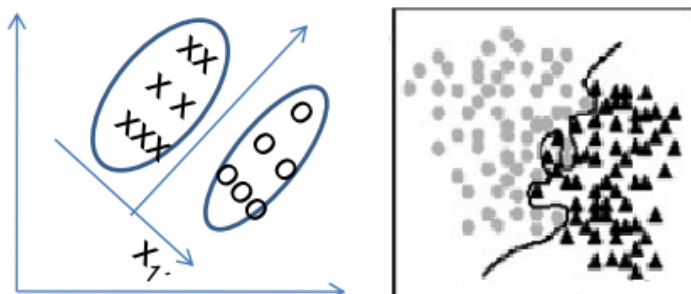


Fig. 4-66. Resultados Esperados.

Disponible en <http://slideplayer.es/slide/5390954/>

En el análisis del trabajo se efectuará clasificación determinando áreas de deforestación y no deforestación, pese a ello y como los resultados no fueron los esperados es necesario recurrir a PCA, con el cual se procede a graficar los puntos para determinar cual es el error que se presenta en el clasificador KERAS, lo que se pretende con PCA es reducir la dimensionalidad de un conjunto de datos que consiste en muchas variables correlacionadas entre si.

El siguiente es el código del programa:

```

1. # Create your first MLP in Keras
2. # from keras.models import Sequential
3. # from keras.layers import Dense
4. import numpy
5. import csv
6. import matplotlib.pyplot as plt
7. from sklearn.decomposition import PCA as sklearnPCA
8. # import rp
9.
10. def traer_pca(lista, componentes=3):
11.     sklearn_pca = sklearnPCA(n_components=componentes)
12.
13.     # entrenar objeto de PCA
14.     sklearn_pca.fit(lista)
15.
16.     # aplicar transformacion PCA a la matriz provista
17.     sklearn_transf = sklearn_pca.transform(lista)
18.     return sklearn_transf, sklearn_pca
19. # fix random seed for reproducibility
20. numpy.random.seed(7)
21. # load pima indians dataset
22. ds_train = numpy.loadtxt("setentatodos1.csv", delimiter=";")
23. X_tr = ds_train[:,0:6]
24. Y_tr = ds_train[:,6]
25.
26. trn_pca, trn_pca_obj = traer_pca(X_tr, 2)
27.
28. x = zip(trn_pca, Y_tr)
29. arr0 = []
30. arr2 = []
31. # f = open('output.csv', 'w')
32. for itm in x:
33.     if itm[1] == 0:
34.         arr0.append([itm[0][0], itm[0][1], itm[1]])
35.     if itm[1] == 2:
36.         arr2.append([itm[0][0], itm[0][1], itm[1]])
37.
38.     # f.write("{};{};{}\n".format(itm[0][0], itm[0][1], itm[1]))
39.
40. # f.close()
41. a0 = numpy.array(arr0)
42. a2 = numpy.array(arr2)
43.

```

```

44. plt.scatter(a2[:,0], a2[:,1], color='b')
45. #plt.scatter(a0[:,0], a0[:,1], color='r')
46. plt.show()
47.
48. print("trn_pca: {}".format(trn_pca))
49. print("x: {}".format(x))
50. print("trn_pca_obj: {}".format(trn_pca_obj))
51. print("explained_variance: {}".format(trn_pca_obj.explained_variance_))
52. print("explained_variance_ratio: {}".format(trn_pca_obj.explained_variance_ratio_))
53.

```

Primero se aplicó el PCA en los resultados obtenidos luego de mezclar las imágenes satelitales combinados con los resultados de la deforestación de sus años correspondientes.

Tabla 13. Datos y deforestación.

• Laguna_cocha_1990_abril_28	Cambio_1990-2000
• Laguna_cocha_1999_julio_10	
• Laguna_cocha_2002_octubre_14	Cambio_2000-2005
• Laguna_cocha_2005_noviembre_23	
• Laguna_cocha_2008_septiembre_12	Cambio_2005-2010
• Laguna_cocha_2010_noviembre_11	
• Laguna_cocha_2012_julio_21	Cambio_2010-2012
• Laguna_cocha_2013_enero_29	Cambio_2012-2013
• Laguna_cocha_2014_diciembre_10	Cambio_2013-2014

Lo primero que se observa es que el algoritmo PCA convierte a dos columnas las seis anteriores a efectos de graficar los datos obtenidos en el estudio.

Tabla 14. Resultado PCA consolidado.

<b>28.4625593</b>	<b>-4.29430143</b>
<b>-9.78437095</b>	1.45721565
...	...
<b>-42.24653576</b>	18.84974346

En total se obtienen dos columnas y 274306 filas, se detecta que la pérdida de información fue mínima.

Total perdida de datos

Al correr el software PCA presenta los resultados de información recuperada de 0.92944. Esto significa que un 0.07056 de información se perdió en el proceso PCA aproximadamente un 7%.

Analizando con los datos en donde se escogieron varios años de deforestación los siguientes son los resultados:

Se analizan los puntos de la deforestación dando como resultado:

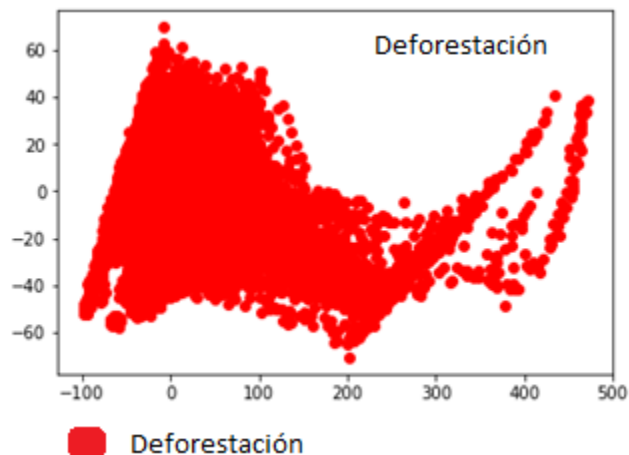


Fig. 4-67. Puntos de deforestación.

Se analizan los puntos de no deforestación, obteniendo:

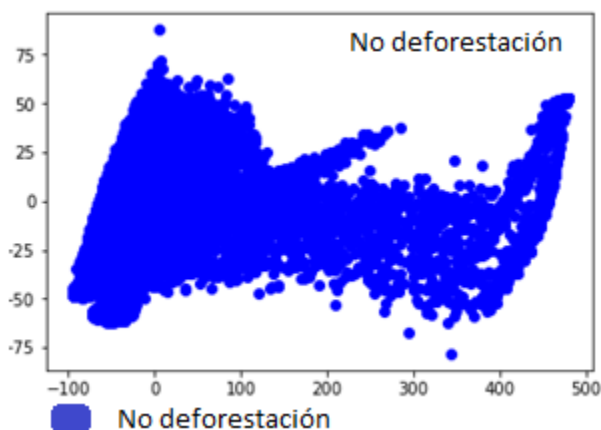


Fig. 4-68. Puntos sobrepuestos de no deforestación y deforestación.

Se coloca ahora los puntos de la deforestación sobre la no deforestación

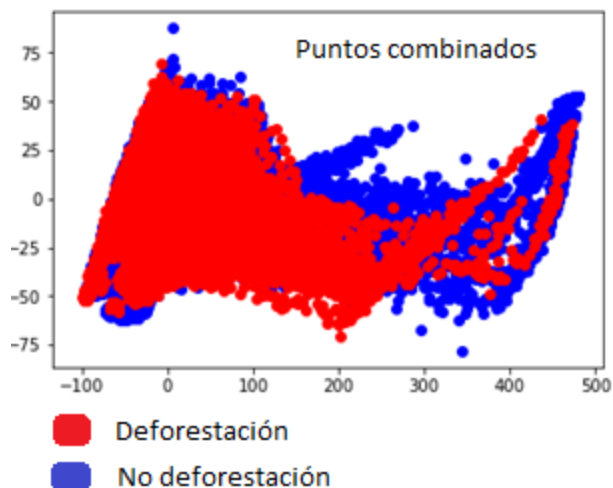


Fig. 4-69. Puntos sobrepuestos de deforestación y no deforestación.

Lo que el presente estudio pretende observar es una frontera de decisión para realizar la clasificación.

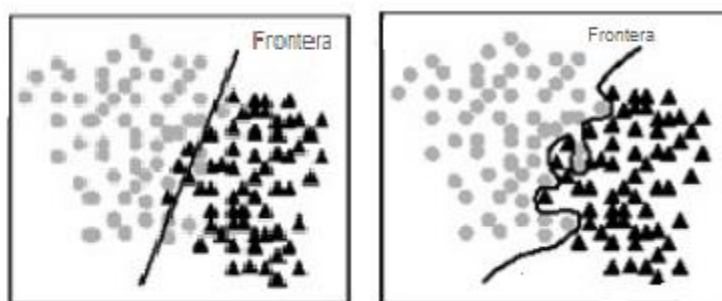


Fig. 4-70. Clasificación deseada

Sin embargo, se puede observar que no es posible determinar ninguna frontera por que los puntos a clasificar es decir la deforestación y la no deforestación se sobreponen.

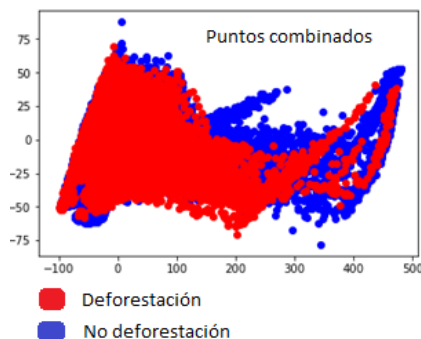


Fig. 4-71. Clasificación obtenida

Encontramos que los puntos son imposibles de analizar por qué se sobreponen, de esta manera, ningún algoritmo al no encontrar frontera de clasificación, podría clasificar la deforestación de la no deforestación.

Los resultados anteriores son tomados en base a deforestación y no deforestación, sin embargo los datos que se encuentra en base a las imágenes del IDEAM son:

1. Bosque estable
2. Deforestación
3. Sin información
4. Regeneración
5. No bosque estable

Se decide entonces graficar los puntos de todas las variables. Para esto hay que hacer unos cambios en el programa

```

1. # Create your first MLP in Keras
2. # from keras.models import Sequential
3. # from keras.layers import Dense
4. import numpy
5. import csv
6. import matplotlib.pyplot as plt
7. from sklearn.decomposition import PCA as sklearnPCA
8.
9. # import rp
10.
11. def traer_pca(lista, componentes=3):
12.     sklearn_pca = sklearnPCA(n_components=componentes)
13.
14.     # entrenar objeto de PCA
15.     sklearn_pca.fit(lista)
16.
17.     # aplicar transformacion PCA a la matriz provista
18.     sklearn_transf = sklearn_pca.transform(lista)
19.     return sklearn_transf, sklearn_pca
20.
21. # fix random seed for reproducibility
22. numpy.random.seed(7)
23. # load pima indians dataset
24. ds_train = numpy.loadtxt("consolidado70pociento.csv", delimiter=";")
25. X_tr = ds_train[:,0:6]
26. Y_tr = ds_train[:,6]
27.
28. trn_pca, trn_pca_obj = traer_pca(X_tr, 2)
29.
30. x = zip(trn_pca, Y_tr)
31. arr1 = []
32. arr2 = []
33. arr3 = []
34. arr4 = []

```

```

35. arr5 = []
36. # f = open('output.csv', 'w')
37. for itm in x:
38.     if itm[1] == 1:
39.         arr1.append([itm[0][0], itm[0][1], itm[1]])
40.     if itm[1] == 2:
41.         arr2.append([itm[0][0], itm[0][1], itm[1]])
42.     if itm[1] == 3:
43.         arr3.append([itm[0][0], itm[0][1], itm[1]])
44.     if itm[1] == 4:
45.         arr4.append([itm[0][0], itm[0][1], itm[1]])
46.     if itm[1] == 5:
47.         arr5.append([itm[0][0], itm[0][1], itm[1]])
48.
49.
50. # f.write("{};{};{}\n".format(itm[0][0], itm[0][1], itm[1]))
51.
52. # f.close()
53. a1 = numpy.array(arr1)
54. a2 = numpy.array(arr2)
55. a3 = numpy.array(arr3)
56. a4 = numpy.array(arr4)
57. a5 = numpy.array(arr5)
58.
59. plt.scatter(a1[:,0], a1[:,1], color='b')
60. plt.scatter(a2[:,0], a2[:,1], color='r')
61. plt.scatter(a3[:,0], a3[:,1], color='y')
62. plt.scatter(a4[:,0], a4[:,1], color='m')
63. plt.scatter(a5[:,0], a5[:,1], color='c')
64. plt.show()
65.
66. print("trn_pca: {}".format(trn_pca))
67. print("x: {}".format(x))
68. print("trn_pca_obj: {}".format(trn_pca_obj))
69. print("explained_variance: {}".format(trn_pca_obj.explained_variance_))
70. print("explained_variance_ratio: {}".format(trn_pca_obj.explained_variance_ratio_))

```

Se procede a implementar todas las variables en una misma gráfica.

Lo primero que se observa es que el algoritmo PCA convierte a dos columnas las seis anteriores a efectos de graficar los datos obtenidos en el estudio.

Tabla 15. Resultado PCA consolidado.

<b>-12.25431905</b>	<b>14.63507777</b>
<b>-22.84957605</b>	15.80351159
...	...
<b>-35.21038901</b>	5.96076968

En total se obtienen dos columnas y 274306 filas.

Se detecta que la pérdida de información fue mínima  
Total pérdida de datos

Al correr el software PCA presenta los resultados de información recuperada de 0.92583, esto significa que un 0.07417 de información se perdió en el proceso PCA aproximadamente un 7.4%.

Se procede a graficar los puntos de bosque estable

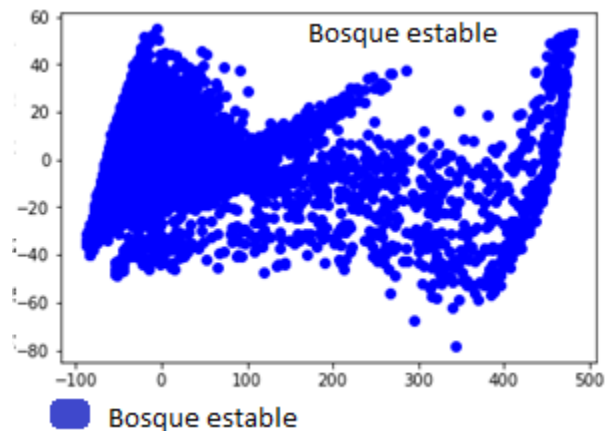


Fig. 4-72. Bosque estable consolidado.

Se procede a graficar los puntos de la deforestación.

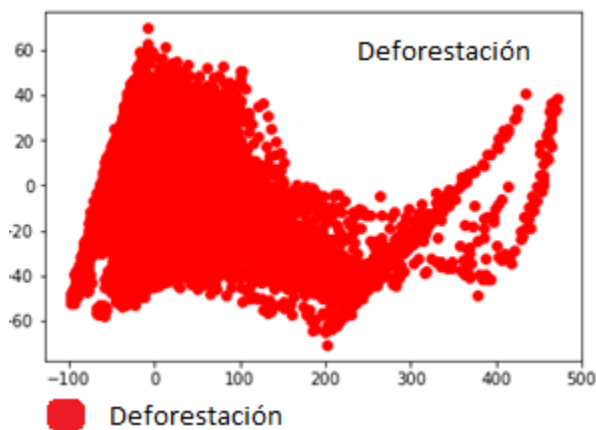


Fig. 4-73. Deforestación consolidado.



Se procede a graficar los puntos de sin información

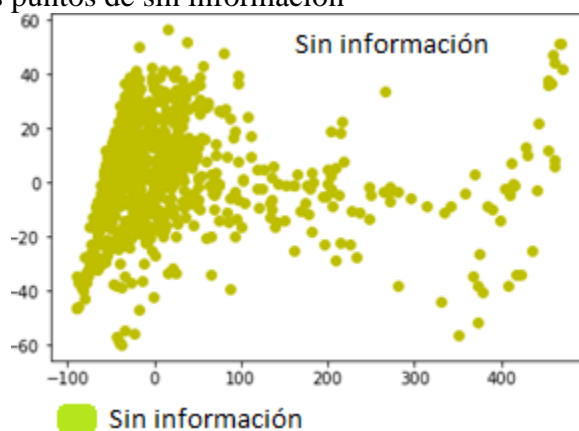


Fig. 4-74. Puntos de sin información consolidado.

Se procede a graficar los puntos de regeneración

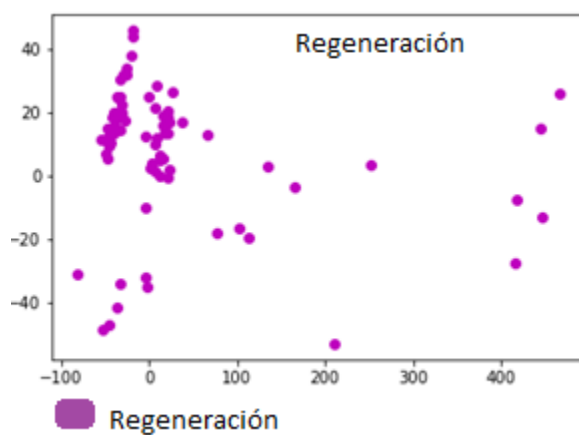


Fig. 4-75. Puntos de regeneración consolidado.

Se procede a graficar los puntos de no bosque estable.

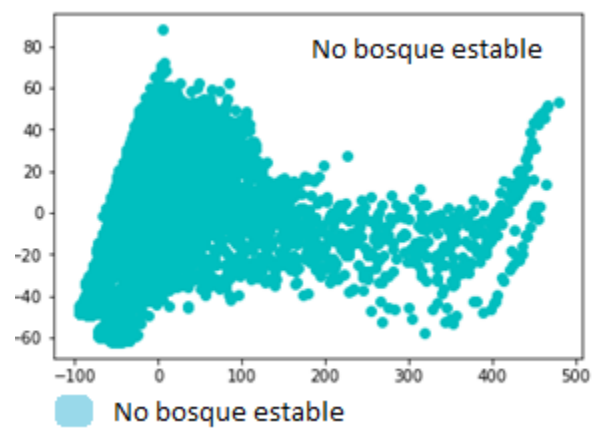


Fig. 4-76. No bosque estable consolidado.

La grafica obtenida es la siguiente:

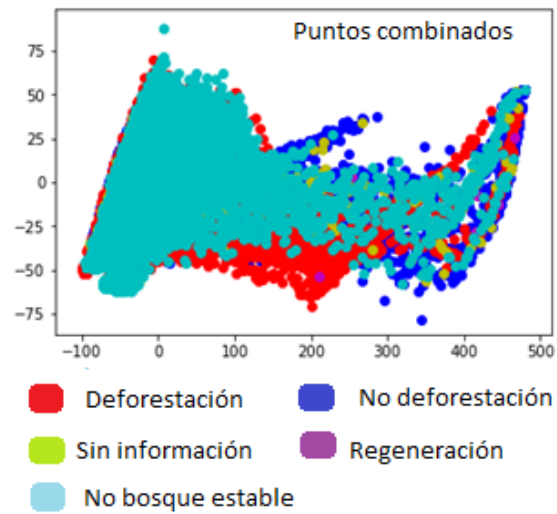


Fig. 4-77. Puntos totales consolidado.

Como se puede observar no se observa una frontera de clasificación lo que hace la clasificación.

Resultados esperados

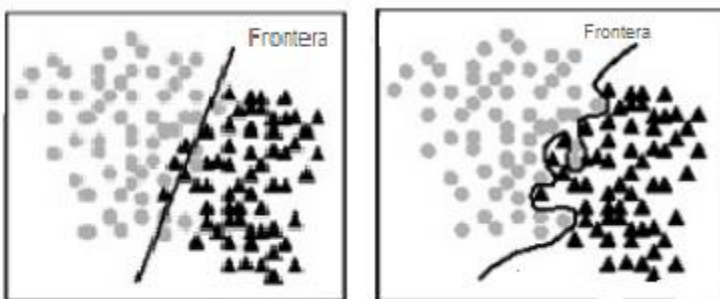


Fig. 4-78. Resultados esperados.

Resultados obtenidos

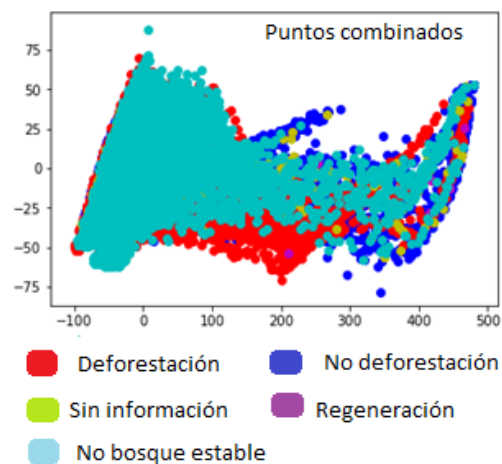


Fig. 4-79. Resultados obtenidos.

Ahora se analizan los puntos obtenidos con un solo año es decir con los datos de deforestación del año de 1990.

Tabla 16. Imágenes y deforestación 1990.

•Laguna_cocha_1990_abril_28	Cambio_1990-2000
•Laguna_cocha_1999_julio_10	Cambio_1990-2000
•Laguna_cocha_2002_octubre_14	Cambio_1990-2000
•Laguna_cocha_2005_noviembre_23	Cambio_1990-2000
•Laguna_cocha_2008_septiembre_12	Cambio_1990-2000
•Laguna_cocha_2010_noviembre_11	Cambio_1990-2000
•Laguna_cocha_2012_julio_21	Cambio_1990-2000
•Laguna_cocha_2013_enero_29	Cambio_1990-2000
•Laguna_cocha_2014_diciembre_10	Cambio_1990-2000

Corriendo el algoritmo, se obtienen los siguientes resultados: la transformación de siete bandas en dos, tal como se muestra a continuación:

Tabla 17. Total perdida de datos.

<b>-4.90680108</b>	<b>9.5701588</b>
<b>4.81725304</b>	16.06774011
...	...
<b>-4.615666659</b>	18.03833304

En total se obtienen dos columnas y 274306 filas.

Se detecta que la perdida de información fue mínima

Al correr el software PCA presenta los resultados de información recuperada de 0.92583. Esto significa que un 0.07693 de información se perdió en el proceso PCA aproximadamente un 7.6%.

Entonces se grafica los puntos que muestra la deforestación.

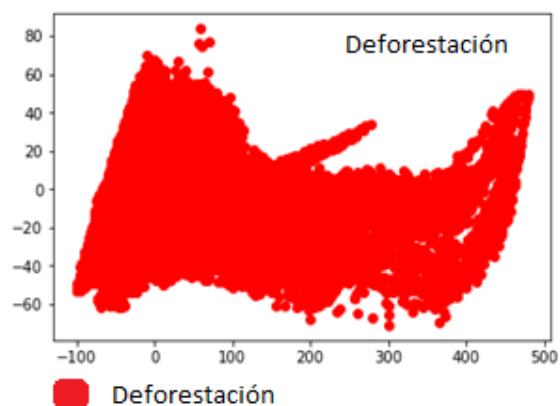


Fig. 4-80. Deforestación PCA 1990.

De igual manera, se grafica los puntos que muestra la no deforestación

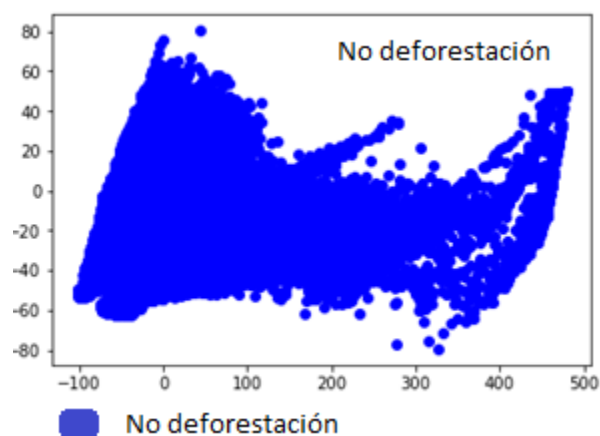


Fig. 4-81. No deforestación 1990.

Se toman los puntos de la deforestación y la no deforestación en un mismo gráfico, colocando la deforestación debajo.

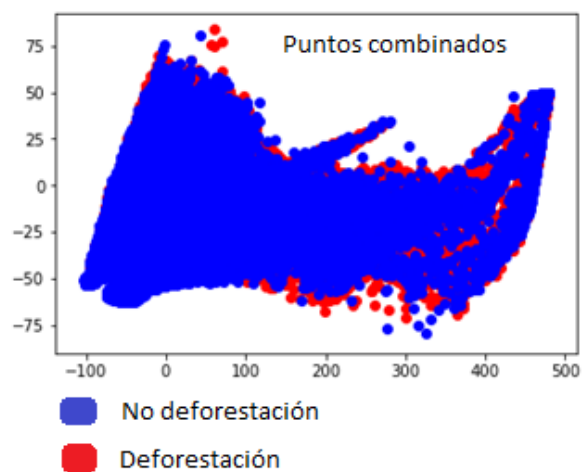


Fig. 4-82. No deforestación sobrepuesta deforestación PCA 1990.

Resultados esperados

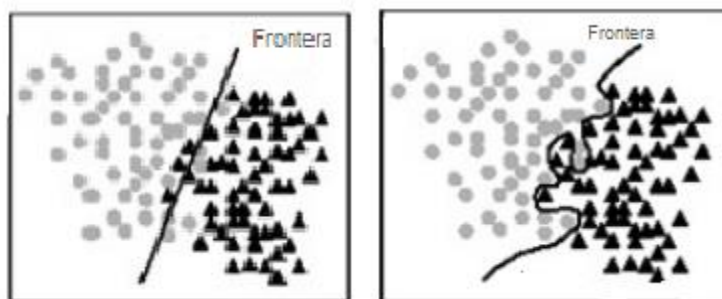


Fig. 4-83. Clasificación deseada.

Resultados obtenidos

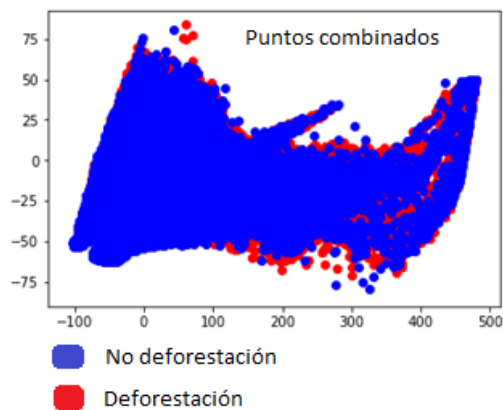


Fig. 4-84. Clasificación obtenida.

Se realiza un último intento de clasificación para esto se va a trabajar con todas las variables que nos proporciona las imágenes del IDEAM.

1. Bosque estable
2. Deforestación
3. Sin información
4. Regeneración
5. No bosque estable

Se toma la decisión de graficar los puntos de todas las variables.

Se procede a graficar los puntos de bosque estable

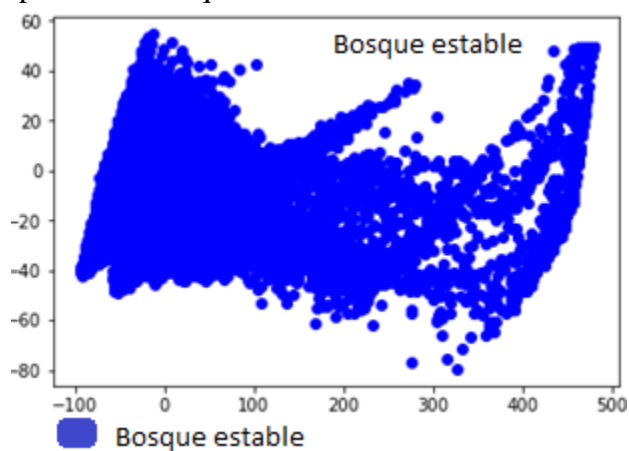


Fig. 4-85. Bosque estable PCA 1990.

Se procede a graficar los puntos de deforestación

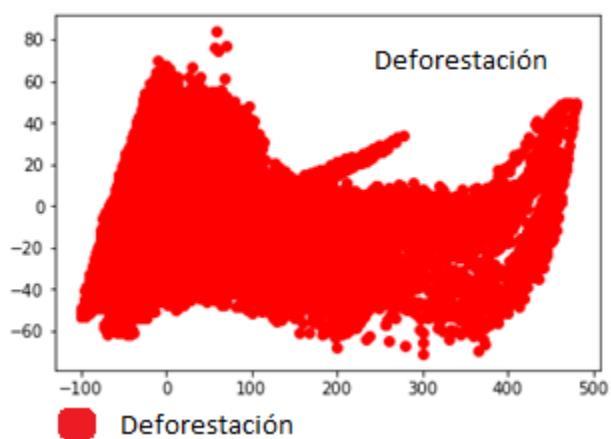


Fig. 4-86. Deforestación PCA 1990.

Se procede a graficar sin información

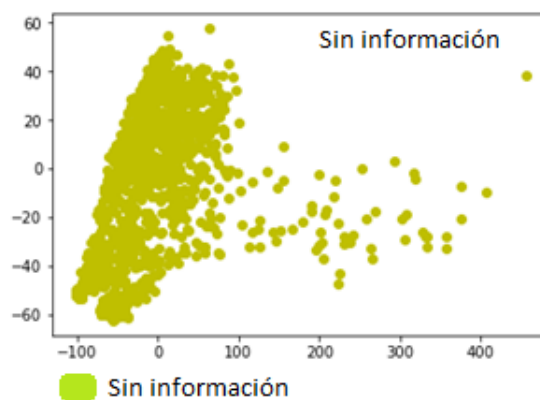


Fig. 4-87. Sin información PCA 1990.

Se procede a graficar regeneración

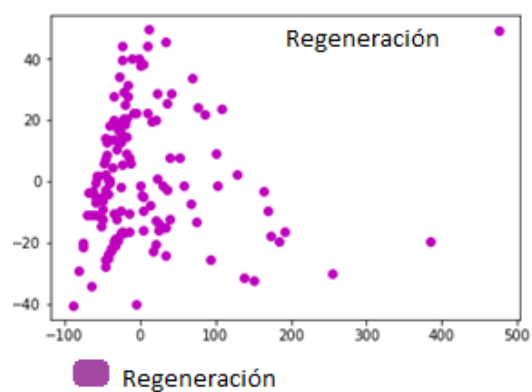


Fig. 4-88. Regeneración 1990.

Se procede a graficar no bosque estable

Ahora graficamos todas las variables

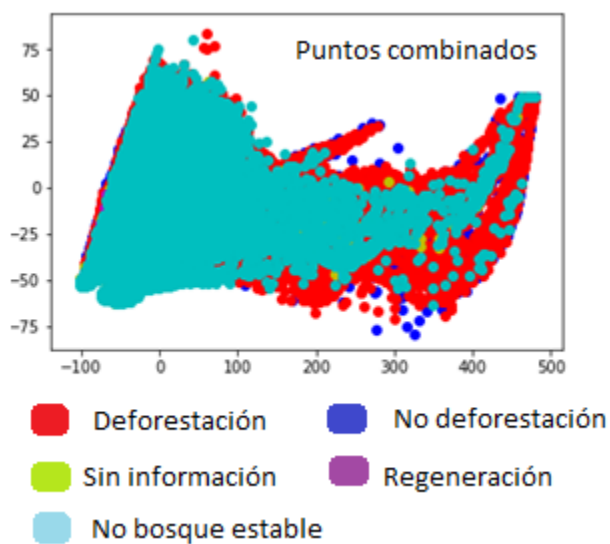


Fig. 4-89. Todos los puntos PCA 1990.

Resultados esperados



Fig. 4-90. Resultados esperados.

## Resultados obtenidos

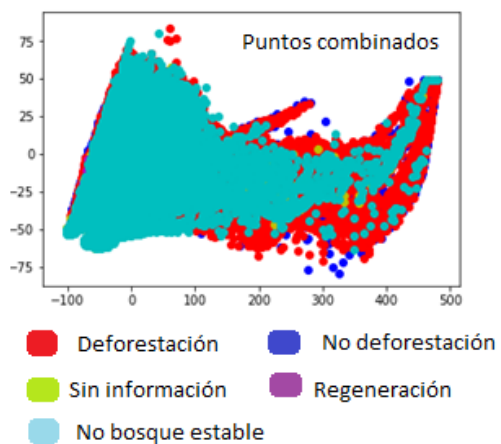


Fig. 4-91. Resultados obtenidos.

Como se puede observar al realizar la clasificación con los datos que arroja el presente estudio de varios años o de un solo año, los datos reflejados son siempre los mismos no se puede observar una frontera para poder realizar la clasificación.



## 5. CONCLUSIONES

En los datos extraídos se encuentran dos columnas s1 y s2 que corresponde a la notación o componente espacial, en el proceso de extracción de las características es arrojado en el SPATIALGRIDDDATAFRAME, este permite tener almacenados en un objeto puntos, grillas, coordenadas y límites de datos que posean componente espacial, esto permite hacer minería de datos espaciales, dicho aporte puede ser de utilidad para ahondar mas en el tema haciendo uso de estas herramientas para investigaciones futuras. En esta investigación se hizo minería de datos convencional, es posible que debido a esto los resultados no fueran tan concluyentes como se esperaba en la hipótesis inicial, es posible que de haber usado el componente geográfico y haber hecho minería de datos espacial los resultados arrojaran una mejor aproximación a la hipótesis inicial.

Esta investigación buscaba medir las bondades de la minería de datos frente a los métodos y herramientas convencionales (argis, envi, erdas, idrisi), el resultado fue diferente al esperado puesto que los datos al no incorporar la componente geográfica (como lo hacen las herramientas convencionales) se vio afectado. Es imposible comparar estas dos alternativas sin que ambas utilicen el componente geográfico como insumo, la técnica propuesta en esta investigación se puede decir que se vio afectada enormemente por la falta de este componente muy posiblemente de haberlo incorporado las bondades habrían sido mayores y seria comparable fácilmente con los métodos tradicionales.

La utilización de la red neuronal utilizando keras, que recurre a las bondades de Tensorflow y Theano hace que los resultados sean muy concisos y que permite su análisis de una manera muy rápida.

La utilización de PCA permite reducir la dimensionalidad del conjunto de datos, es decir las seis bandas originales compactarlas en dos componentes que agrupan aproximadamente un 92% de la información aportada, además de reducir la correlación entre los datos, de esta manera poder graficar los puntos que representan las clases de los niveles digitales. Este análisis arroja resultados que no son del todo concluyentes pero que permiten entender el comportamiento de los datos, como por ejemplo que al hacer el PCA y solo obtener dos clases que representen la mayor cantidad de información quiere decir que los datos originales estaban fuertemente correlacionados motivo por el cual no constituyen por si mismos un insumo óptimo para la construcción de un modelo predictivo.

El estudio puede entonces extenderse y pulirse a futuro combinando los datos de PCA con nuevos insumos, por ejemplo complementar estos con muestras obtenidas en campo haciendo uso de equipos sofisticados como puede ser un espectroradiómetro para recolectar firmas espectrales precisas de deforestación que permitan tener semillas más acordes para que sean adicionadas al proceso de entrenamiento del algoritmo.

Los datos proporcionados por el IDEAM fueron de gran utilidad para el desarrollo del proyecto, pero no se contaba que al ser usados como semilla en el algoritmo de entrenamiento aportarían una baja variabilidad en el modelo generado por el proceso de minería de datos. Dado que el insumo del modelo y los datos usados como semilla compartían una base que se correspondía demasiado bien, es decir, las clases bosque estable y no bosque estable aportaban demasiados individuos al modelo que incrementaban la presencia de valores VERDADERO NEGATIVO (deforestación que fue clasificados como diferentes de deforestación), lo que quiere decir que sin esta redundancia de información es posible que los valores VERDADEROS POSITIVOS (deforestación que fueron clasificados como deforestación – mas bajos) y todos los FALSOS NEGATIVO (diferentes de deforestación que fueron clasificados como diferentes de deforestación - mas altos) fueran mas acordes con la realidad.

RGRASS se constituye en una buena herramienta para el análisis de las imágenes ya que posee un puente entre GRASS, R y RSTUDIO, este es un potente programa estadístico que permite hacer uso de algoritmos complejos, además generar muy buenos gráficos, esto apporto mucho en el desarrollo del proyecto siendo el motor para poder extender los análisis a mejores resultados que facilitaron su interpretación.

Es procedente establecer que conforme a los resultados obtenidos la investigación se torna en un insumo que puede ser empleado como una herramienta de análisis de datos en trabajos relacionados con ecología y medio ambiente.

## REFERENCIAS

- [1] Á. Soldano, "Rimd," 20 marzo 2009. [Online]. Available: <http://rimd.org/advf/documentos/4922edde159e6.pdf>. [Accessed 10 noviembre 2017].
- [2] Instituto Nacional de Astrofísica Óptica y Electrónica, "Reconocimiento de patrones," 2016. [Online]. Available: <https://ccc.inaoep.mx/~ariel/recpat.pdf>. [Accessed 8 Noviembre 2017].
- [3] E. Chuvieco, Fundamentación de Teledetección Espacial., Segunda ed., Madrid - España.: E. S. RIALP, Ed. , 1995.
- [4] J. Martínez Vega, Guía Didáctica de Teledetección y Medio Ambiente., Red Nacional de Teledetección Ambiental, 2010.
- [5] F. Reuter, "Principios de Teledetección," 2009. [Online]. Available: <http://fcf.unse.edu.ar/archivos/series-didacticas/SD-33-Principios-teledeteccion-REUTER.pdf>.
- [6] R. Sputnik, "El futuro empezó en 1997," 11 Noviembre 1980. [Online]. Available: <https://sputnik87.wordpress.com/>. [Accessed 03 Octubre 2017].
- [7] IDEAM, "Protocolo de procesamiento Digital," s.f. [Online]. Available: <http://documentacion.ideam.gov.co/openbiblio/bvirtual/022977/ProtocoloProcesamientoDigital2.pdf>.
- [8] "Las redes neuronales: qué son y por qué están volviendo," Xataka, 30 Diciembre 2014. [Online]. Available: <https://www.xataka.com/robotica-e-ia/las-redes-neuronales-que-son-y-por-que-estan-volviendo>. [Accessed 12 Octubre 2017].
- [9] Keras. [Online]. Available: <https://keras.io/>.
- [10] "Técnicas Clásicas en Reconocimiento de Patrones", s.f. [Online]. Available: [http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patrones-70038/contenidos/Ejercicios/trabajo\\_1\\_rpyrn\\_2009-10.pdf](http://ocw.upm.es/ciencia-de-la-computacion-e-inteligencia-artificial/redes-neuronales-y-reconocimiento-de-patrones-70038/contenidos/Ejercicios/trabajo_1_rpyrn_2009-10.pdf).
- [11] ERDAS, "ERDAS Imagine, módulos y funcionalidades," 2017. [Online]. Available: [http://www.erdas.com.ar/productos\\_imagine.htm](http://www.erdas.com.ar/productos_imagine.htm). [Accessed 11 Noviembre 2017].
- [12] ESRI, "The Science of where," 2017. [Online]. Available: <http://www.esri.es/>. [Accessed 8 Noviembre 2017].
- [13] GRASS, "Newcomers: How to start with GRASS GIS?," 2017. [Online]. Available: <https://grass.osgeo.org/>. [Accessed 1 Diciembre 2017].
- [14] J. Brownlee, "Develop Your First Neural Network in Python With Keras Step-By-Step," Machine Learning Mastery, 24 Mayo 2016. [Online]. Available: <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/>. [Accessed 13 Noviembre 2017].
- [15] GLCF, "Data & Products," University of Maryland, 2016. [Online]. Available: <http://glcf.umd.edu/data/>. [Accessed 4 Septiembre 2017].
- [16] USGS, "LandsatLook," 2017. [Online]. Available: <http://landsatlook.usgs.gov/viewer.html>. [Accessed 3 Octubre 2017].
- [17] INPE, "Image Catalog," Instituto Nacional de Pesquisas Espaciais, 2017. [Online]. Available: <http://www.dgi.inpe.br/CDSR/>. [Accessed 7 Octubre 2017].
- [18] UNODC, "Norma Técnica de metadatos, Documentación límite grilla de imágenes," Oficina de las Naciones Unidas contra la Droga y el Delito, 2017. [Online]. Available: <http://www.biesimci.org/satelital/Auxiliar/Limites/Grillassatelite/Indices/B551L01-S000002A001.html>. [Accessed 22 Noviembre 2017].
- [19] USGS, "Earth Explorer," 2017. [Online]. Available: <https://earthexplorer.usgs.gov/>.
- [20] E. Cabrera, Vargas and D. M. Galindo, "Protocolo de Procesamiento Digital de Imágenes para la Cuantificación de la Deforestación en Colombia Nivel Nacional Escala Gruesa y Fina," 2011. [Online]. Available: [http://www.ideam.gov.co/documents/13257/13817/Protocolo\\_para\\_la\\_cuantificaci%C3%B3n\\_Deforestaci%C3%B3n\\_Nivel\\_Nacional.pdf](http://www.ideam.gov.co/documents/13257/13817/Protocolo_para_la_cuantificaci%C3%B3n_Deforestaci%C3%B3n_Nivel_Nacional.pdf).

- [21 J. R. Martínez, "Geografía Física," 13 Febrero 2013. [Online]. Available:  
] <http://www.geografiafisica.org/2013/02/11/lanzamiento-del-satelite-ldcm-landsat-data-continuity-mission/>.  
] [Accessed 25 Noviembre 2017].
- [22 USGS, "SLC-off Products: Background," Department of the Interior, U.S. Geological Survey, 22 Junio 2017.  
] [Online]. Available: <https://landsat.usgs.gov/slc-products-background>. [Accessed 28 Noviembre 2017].
- [23 J. Santana and E. Farfán, El arte de programar en R: un lenguaje para la estadística, Vols. ISBN 978-607-9368 ,  
] 2014.
- [24 J. C. Rodríguez Campo, "Estudio comparativo de técnicas estadística multivariada versus las redes neuronales  
] artificiales en el análisis de datos de calidad de agua," 2008. [Online]. Available:  
<https://repository.javeriana.edu.co/bitstream/handle/10554/3826/RodriguezCampoJuanCarlos2008.pdf?sequence=1>. [Accessed 22 Noviembre 2017].
- [25 E. Sandoya, "Diagnóstico basado en evidencia," *Revista Uruguaya de Cardiología*, vol. 24, no. 1, pp. 028-040,  
] 9 Marzo 2009.
- [26 Ananconda, "Ananconda," 2017. [Online]. Available: <https://www.anaconda.com/what-is-anaconda/>.  
] [Accessed 3 noviembre 2017].
- [27 L. Garcia, M. Évora Brondo, J. Antonio and A. Perez, Satélites de Teledetección para la Gestión del Territorio,  
] España.: P. y aguas. G. de C. Consejería de Agricultura, Ganadería,, 2012.
- [28 A. Ariza, Descripción y Corrección de Productos Landsat 8 LDCM (Landsat Data Continuity Mission), CIAF,  
] Ed., Centro de Investigación Y Desarrollo En Información Geográfica Del IGAC, 2013.
- [29 M. Javier, Multiespectrales, Satelitales, Multiespectrales, (2014).  
]
- [30 INEGI, "Aspectos técnicos de las imágenes Landsat," 1999. [Online]. Available:  
] [http://www.inegi.org.mx/geo/contenidos/imgpercepcion/imgsatelite/doc/aspectos\\_tecnicos\\_de\\_imagenes\\_landat.pdf](http://www.inegi.org.mx/geo/contenidos/imgpercepcion/imgsatelite/doc/aspectos_tecnicos_de_imagenes_landat.pdf). [Accessed 19 Octubre 2017].