ISBN 978-987-4417-90-9

SMOTE, Algoritmo para balanceo de clases en un estudio aplicado a la ganadería.

Osvaldo Spositto[1], Gabriel Blanco[1], Lorena Matteo[1], Marcelo Levi[1] y Julio Bossero[1]

[1]Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigaciones Tecnológicas.
{sposito, g2blanco, lmatteo, mlevi,jbossero}@unlam.edu.ar

Abstract. En el estudio de los algoritmos de Minería de Datos del tipo supervisados surge el problema del desbalance de clases, que implica que la información no se encuentre distribuida equitativamente entre todas las clases que la componen, por lo que se generan efectos no deseados en el proceso de clasificación. Este trabajo considera el caso de conjuntos de datos que solamente tiene dos clases y una de ellas cuenta con una mayor cantidad de ejemplos que la otra. El interés principal del trabajo es la aplicación de la técnica de balanceo de clases SMOTE (Synthetic Minority Oversampling Technique), que con algoritmos de interpolación incrementa en forma "sintética" los ejemplos de la clase minoritaria. Los resultados experimentales muestran que algunas técnicas, en el proceso de entrenamiento, obtienen mejores porcentajes de clasificación, cuando se usan estos datos artificiales. El dataset utilizado registra la Diferencia Esperada entre Progenie de animales de la raza Aberdeen Angus.

Keywords: Desbalance de clases, SMOTE, Algoritmos Supervisados, Weka, DEP

1 Introducción

Se ha observado que algunas de las técnicas de MD del tipo supervisadas, si utilizan un conjunto de datos desequilibrado para la clasificación, presentan un rendimiento de generalización deficiente, debido a un fuerte sesgo hacia las clases mayoritarias [1]. Las clases con el mayor número de instancias se denominan clases mayoritarias y las clases con el menor número de instancias son referidas como las clases minoritarias. Intuitivamente, dado que hay una gran cantidad de ejemplos de clases mayoritarias, un modelo de clasificación tiende a favorecer las clases mayoritarias mientras que clasifica incorrectamente los ejemplos de las clases minoritarias [2].

Las técnicas supervisadas, se expresan mediante algoritmos capaces de tratar y analizar datos de forma automática, con el objeto de extraer cualquier tipo de información subyacente en dichos datos. Como se sabe, en el aprendizaje supervisado, los algoritmos trabajan con datos "*etiquetados*", intentado encontrar una función que, dadas las variables de entrada, les asigne la etiqueta de salida adecuada.

El algoritmo se entrena con un "histórico" de datos y así "aprende" al asignar la etiqueta de salida a un nuevo valor, es decir, predice el valor de salida [1].

El problema del desbalanceo de las clases, entonces, consiste en la predominancia de ciertos valores en los datos de entrenamiento y la escasez de otros.

Este trabajo, es parte de un proyecto de investigación que se lleva adelante en la Universidad Nacional de La Matanza, denominado "*Uso de Minería de Datos para Mejoramiento Genético en la Raza Aberdeen Angus*", donde se estudia la aplicación de distintas técnicas de MD con el objeto de encontrar, a partir de los valores genéticos de animales de la raza Aberdeen Angus, patrones o grupos de características que puedan determinar a priori, el peso de los terneros al nacer. En esta investigación se estudiaron tanto algoritmos del tipo No Supervisados como Supervisados. En la aplicación de los primeros, se utilizaron las siguientes técnicas: EM (Expectation Maximization), FarthestFirst, Simple K-Means y Mapas AutoOrganizados (Redes SOM). Ese trabajo titulado "*Peso al Nacer de Terneros Aberdeen Angus mediante Algoritmos No Supervisados*", fue publicado en CoNaIISI 2019 [3]. En este trabajo se puede leer el procedimiento realizado para obtener los datos usados para confeccionar la vista minable [4]. Estos mismos datos fueron utilizados en los experimentos con métodos supervisados y en particular en el estudio que es motivo de esta presentación.

En cuanto al estudio, sobre el empleo de técnicas Supervisadas, se realizó una comparación entre tres clasificadores: Árbol de Decisión (AD), Red Neuronal Artificial (RNA), del tipo Perceptron Multicapa y una Máquina de Soporte Vectorial (MSV). El trabajo se tituló: "*Clasificación del Peso al Nacer de Terneros Aberdeen Angus mediante Algoritmos Supervisados*", y fue presentado y aprobado en JAIOO 2019, pero no llegó a publicarse.

En ambos trabajos se utilizó un modelo de datos o vista minable, compuesto con datos provenientes de las evaluaciones genéticas de los toros: conocidos como la Diferencia Esperada entre Progenie (DEP) [5], que permite a los productores tomar decisiones de selección en base a información objetiva. Los DEP's anticipan, cómo será el comportamiento promedio de las futuras crías de un toro. Esos datos se complementaron, además, con datos de las hembras (edad, cantidad de partos, etc.) y genéticos de los padres de las hembras. Con este conjunto de datos, se alcanzaron valores aceptables de predicción, respecto de la variable a clasificar: Peso al nacer (PN), valor que de acuerdo a los veterinarios es determinante del potencial del desarrollo futuro del animal. Dicha variable se estableció, para los trabajos mencionados anteriormente, en "*Alta*" y "*Baja*". Un PN promedio es de 38 kilogramos, por tal motivo, a los pesos mayores o iguales a 38 kg., se los clasificó como Alto y al resto como Bajo [3].

Se concluyó que, con los algoritmos utilizados, en este último estudio, se obtuvieron valores aceptables (porcentaje de aciertos superior al 60%), en referencia a las métricas: *precisión, sensibilidad y especificidad*. Y que, para estos datos y con la configuración propuesta, para cada técnica, por el software WEKA[1] (Waikato Environment for Knowledge Analysis, en español "*entorno para análisis del conocimiento de la Universidad de Waikato"*) [7], el algoritmo Árbol de Decisión

[1] Es una plataforma de software para el aprendizaje automático y la minería de datos escrito en Java y desarrollado en la Universidad de Waikato. Weka es software libre distribuido bajo la licencia GNU-GPL. www.cs.waikato.ac.nz/~ml/weka/

tuvo la mejor precisión, es decir, la mejor probabilidad de discriminar correctamente, debido a que el valor de su media muestral fue del 72.5%. Cabe resaltar que el indicador precisión es una de las medidas principales, para establecer el desempeño de un algoritmo de clasificación en el área de la MD [1].
El principal interés en este trabajo es mejorar la clasificación obtenida en el trabajo expuesto en el párrafo anterior. Para ello, se modificaron, en la vista minable, los objetos de la clase minoritaria, sin eliminar objetos de la clase mayoritaria, lo cual, según la literatura actual, puede producir pérdida de información importante [2][8], mediante la aplicación del filtro SMOTE (cuya traducción al español es "*técnica de sobre muestreo de minorías sintéticas*") [9]. Luego, se realizó una nueva comparación, utilizando ese nuevo set de datos, a través del mismo software WEKA, usando siempre la configuración por defecto. A continuación, se hace una reseña de algunos trabajos relacionados con el uso del algoritmo SMOTE. No se han encontrado trabajos en donde se aplique la técnica SMOTE relacionado con la ganadería.

Antecedentes y Trabajos Relacionados

El problema del desbalanceo de clases es una cuestión que se está abordando en la actualidad de forma activa y son muchos los investigadores que estudian y proponen nuevas técnicas para poder hacerle frente a este problema. La mayoría de ellos solo se han concentrado en resolver situaciones como la nuestra de clasificación que tienen que ver con dos clases. En [8], se encuentra un detalle de varios trabajos relacionados con el desbalanceo de clases ordenados cronológicamente. Algunos otros trabajos encontrados respecto al uso del algoritmo SMOTE son los siguientes:

- De Jesús, Juan. (2016). *Técnicas de muestreo para mejorar el rendimiento del algoritmo back-propagation en problemas de desbalance de clases: Un estudio empírico sobre la clasificación en imágenes de percepción remota*.
  Resumen: En este trabajo se analizaron las diferentes técnicas de re muestreo para tratar el desbalance de clases en dominios de dos clases con ayuda de datos de imágenes de percepción, para determinar que técnica arroja un mejor clasificador. Los resultados mostraron aquellos algoritmos que mejor funcionaron al momento de clasificar, de acuerdo a un análisis estadístico aplicado a los algoritmos.
- J. Monroy de Jesús y otros. (2018). *Algoritmo de aprendizaje eficiente para tratar el problema del desbalance de múltiples clases*.
  Resumen: Los resultados demostraron que la diversidad de versiones de algoritmos y cuan competitivos resultaron en el desempeño de la clasificación con respecto a los métodos de sobre-muestreo y sub-muestreo (ROS, SMOTE y RUS).
- David Municio Duran. (2019). *Técnicas de oversampling aplicadas al análisis de imágenes hiperespectrales.*
  Resumen: En este trabajo se ha estudiado el impacto de diferentes algoritmos de oversampling en el proceso de clasificación de imágenes hiperespectrales. El objetivo del trabajo fue mejorar los resultados de clasificación en imágenes con alta dimensionalidad y gran desbalanceo entre clases.

- Rosa María Valdovinos Rosas. (2006). *Técnicas de Submuestreo, Toma de Decisiones y Análisis de Diversidad en Aprendizaje Supervisado con Sistemas Múltiples de Clasificación.*
  Resumen: Este trabajo se encarga del estudio de Sistemas Múltiples de Clasificación (SMC), para el reconocimiento de patrones. El trabajo se centró en la limpieza de un conjunto de datos, se emplearon para la reducción del tamaño del conjunto de entrenamiento, un algoritmo de subconjunto selectivo modificado, y para la generación de patrones sintéticos, se utilizó el algoritmo SMOTE.

2. Algoritmo de balanceo de clases

Para este trabajo se hace uso del algoritmo SMOTE como parte de los métodos basados en muestreo (sampling), para balanceo de clases [2]. Se ha elegido SMOTE debido a que es uno de los algoritmos más utilizados para tratar el problema de desbalanceo de clases [8][10]. SMOTE es una técnica basada en sobremuestreo (oversampling), que genera instancias "*sintéticas*" o artificiales en el espacio de atributos con el objetivo de equilibrar la muestra de datos basado en la regla del Vecino más cercano [1]. Esta regla consiste en suponer que instancias próximas entre sí, tienen mayor probabilidad de pertenencia a la misma clase. La generación se realiza interpolando nuevas instancias en lugar de duplicarlas como hacen los algoritmos del tipo re-muestreo (resampling) [1]. Para cada una de las instancias minoritarias se buscan las instancias minoritarias vecinas (más cercanas). Se crean N o α (alfa) instancias sobre el segmento que une la instancia original y cada una de las vecinas. Un trabajo de Rodríguez Torres [11], realizó una descripción algorítmica de esta técnica, que puede ser reescrita en estos tres pasos:

1) Se determina la cantidad promedio de interpolaciones que debe aportar cada elemento minoritaria: α (ya sea calculado o elegido).

2) Se asigna a cada muestra minoritario el número $\lfloor\alpha$ o $\lceil\alpha$[2], cantidad de interpolaciones en las cuales intervendrá ya sea con ayuda de azar o determinístico de modo tal de lograr la cantidad de muestras artificiales deseada.

3) Luego se procesa cada elemento de la muestra calculando según el número asignado la cantidad de vecinos cercanos interpolando por azar un punto sobre el segmento que los une.

La Figura 1 muestra un ejemplo de procedimiento que utiliza el algoritmo SMOTE, a continuación, se explica el proceso:

A) Para cada ejemplo de la clase minoritaria ***k***, se calcula el vecino más cercano (***i, j, l, n. m***).

B) Se elige aleatoriamente un ejemplo de 5 puntos más cercanos.

C) Se genera sintéticamente el evento k_1, de modo que k_1 se encuentra entre k e i.

D) En esta imagen se ve el conjunto de datos después de aplicar SMOTE 3 veces.

[2] En matemática, una forma de considerar un número entero más próximo a un número real dado, se pueden considerar: Al entero inferior como piso ($\lfloor$) y al entero superior como techo ($\lceil$).

Estos ejemplos sintéticos ayudan a equilibrar la distribución original de la clase, que generalmente mejora significativamente el aprendizaje. Sin embargo, el algoritmo SMOTE también tiene su desventaja, como por ejemplo generalización del espacio de clase minoritaria [12].

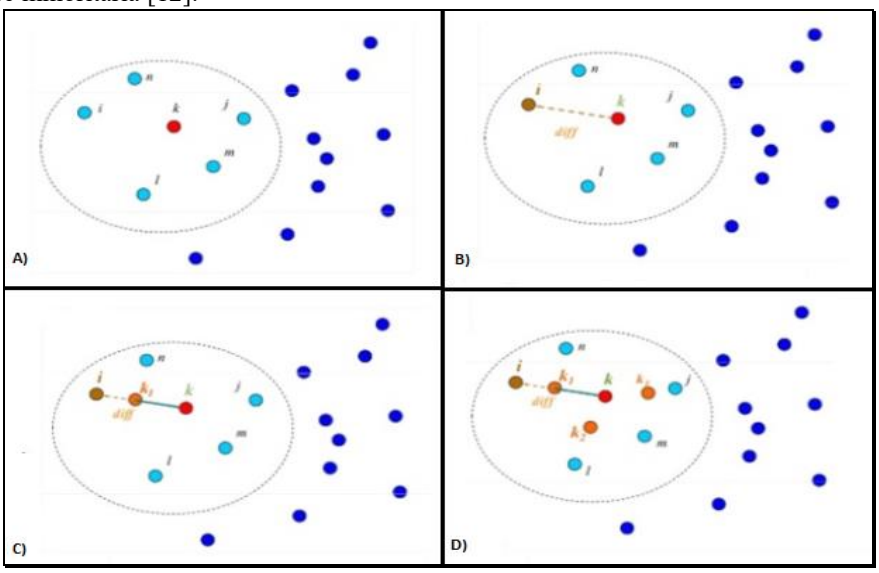

Fig. 1. Ejemplo del algoritmo SMOTE [16].

A partir del algoritmo SMOTE original, se han desarrollado muchos otros algoritmos basados en SMOTE a lo largo de los años y algunos de ellos mejoran efectivamente el rendimiento en el aprendizaje desequilibrado [2][10]. El principal inconveniente del algoritmo es el alto coste computacional que tiene.

3 Método utilizado y resultados

Como ya se mencionó, este trabajo representa una continuación de un trabajo ya realizado. Pero, en esta oportunidad, se realizó una comparación entre los resultados con los datos originales del trabajo, y se los comparó contra los resultados obtenidos, al ejecutar los mismos algoritmos con datos sintéticos. Estos datos creados con el filtro SMOTE, a través del software WEKA. El modelo original, descripto en [3], corresponde a los años 2017 y 2018 y contiene un total de 360 ejemplares hembras. Estas cuales fueron inseminadas, por inseminación artificial, por dos reproductores de la cabaña Las Lilas. La nómina de variables utilizas se muestra en la Tabla 1.

Tabla 1. Descripción de las variables del conjunto de datos.

Nomenclatura	Tipo de Dato	Descripción
ID	Numérico	Identificación de la instancia
PAN_Padre	Numérico	Peso al Nacer
PAD_Padre	Numérico	Peso al Destete
PAF_Padre	Numérico	Peso Final
PAdulto_Padre	Numérico	Peso Real
CEsc_Padre	Numérico	Circunferencia Escrotal
Frame_Padre	Numérico	Altura
Certiv_Padre	Numérico	Edad promedio de los vientres primerizos
PNDEP_Padre	Numérico	DEPs del Toro Progenitor (Padre)
PDDEP_Padre	Numérico	
AMDEP_Padre	Numérico	
CMDEP_Padre	Numérico	
PFDEP_Padre	Numérico	
CEDEP_Padre	Numérico	
AOBDEP_Padre	Numérico	
GDDEP_Padre	Numérico	
MARDEP_Padre	Numérico	
EdadMeses_Madre	Numérico	Datos de la Vaca (Madre)
PAN_Madre	Numérico	
PAD_Madre	Numérico	
UltPeso_Madre	Numérico	
CantNac_Madre	Numérico	
CantAbortos_Madre	Numérico	
CantCesareas_Madre	Numérico	
MuertesAntesDestete_Madre	Numérico	
CEsc_AbueloM	Numérico	DEPs del Toro Progenitor de la Vaca (Abuelo Materno)
Frame_ABueloM	Numérico	
Certiv_AbueloM	Numérico	
PNDEP_AbueloM	Numérico	
PDDEP_AbueloM	Numérico	
AMDEP_Abuelo	Numérico	
CMDEP_AbueloM	Numérico	
PFDEP_AbueloM	Numérico	
CEDEP_AbueloM	Numérico	
AOBDEP_AbueloM	Numérico	
GDDEP_AbueloM	Numérico	
MARDEP_AbueloM	Numérico	
Pnacer_Hijo	Texto	Variable Objetivo en [16], una más en este estudio.

Se describen brevemente las técnicas que se compararon. Para más información, sobre estos algoritmos se pueden consultar en: [1][7][13].

- Árboles de Decisión (AD): como su nombre lo indica es una estructura que se forma por las bifurcaciones en cada una de las decisiones, descubriendo reglas. En WEKA se lo conoce como algoritmo J48, que es una implementación libre en java del algoritmo C4.5, que utiliza el concepto de entropía de la información para la selección de variables que mejor clasifiquen a la variable PN (clase) estudiada.
- Red Neuronal Artificial (RNA): Esta implementación imita el funcionamiento interno de las neuronas humanas [1]. En general, aunque pueden usarse muchos tipos de RNA para clasificación, se han usado las redes multicapa feedforward o perceptron multicapa (MLP) que son los clasificadores basados en redes neuronales más ampliamente estudiados y utilizados. Este algoritmo es entrenado para realizar conexiones entre los valores de entrada y salida, aprendiendo de su error de pronóstico.
- Máquinas de Soporte Vectorial, buscan el límite que separa las clases con el mayor margen posible [1][14], Una de las características de esta técnica es que cuando no se pueden separar correctamente las dos clases, el algoritmo buscan el mejor límite posible. Las MVS efectúan esto, sólo con una línea recta (usa un kernel lineal) y gracias a esta aproximación lineal, se puede ejecutar con bastante rapidez.

Como se hizo anteriormente, para la evaluación de los clasificadores se empleó una Matriz de Confusión (MC) [1][13] y el análisis o curvas ROC (acrónimo de Receiver

Operating Characteristic) [1][15], que entrega WEKA, como resultado luego de testear cada uno de los clasificadores. A modo de resumen, una matriz de confusión muestra la clasificación de las instancias. Brinda información muy útil porque no sólo refleja los errores producidos sino también informa del tipo de éstos. Donde:

- VP es la cantidad de positivos que fueron clasificados correctamente como positivos por el modelo.
- VN es la cantidad de negativos que fueron clasificados correctamente como negativos por el modelo.
- FN es la cantidad de positivos que fueron clasificados incorrectamente como negativos.
- FP es la cantidad de negativos que fueron clasificados incorrectamente como positivos.

De estos valores se definen dos métricas asociadas importantes: Sensibilidad y especificidad:

- La sensibilidad nos indica la capacidad de nuestro estimador para dar como casos positivos los casos realmente lo son; proporción de pesos altos correctamente identificados.
- La especificidad nos indica la capacidad de nuestro estimador para dar como casos negativos los casos realmente lo sean; en nuestro caso, proporción de pesos bajos correctamente identificados.

Por último, la curva ROC, es una herramienta estadística utilizada en el análisis de los clasificadores, determinando la capacidad discriminante de una prueba.

Para realizar la comparación se tomaron los 360 datos originales y luego de cargarlos en WEKA, se le aplicó el filtro SMOTE. Weka permite configurar el valor del vecino más cercano (α) y por el porcentaje de instancias que se necesita crear para que las clases se balanceen. En este caso no se usó el valor por defecto del programa (100%), sino que se ajustó al 40%, para igualar las clases. En la figura 2 se puede observar la distribución de la variable PN, en los datos originales y en la figura 3, la distribución de valores luego de aplicar el filtro.

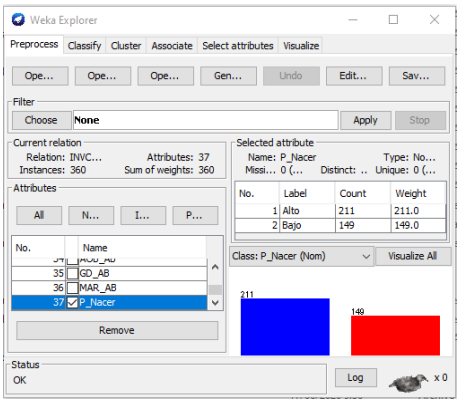

Fig. 2. Distribución con datos originales.

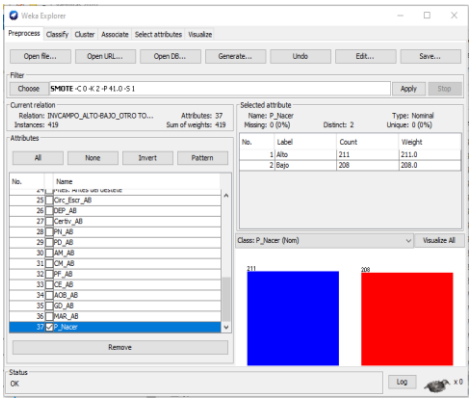

Fig. 3. Distribución con datos sintético.

Como se observa los 360 datos originales, se distribuían en: 211 instancias de PN Altas y 149 del tipo Bajas, luego de aplicar el filtro, las instancias se convirtieron en 419, es decir, 59 instancias más en el peso bajo. Dejando al PN Alta, con la misma cantidad.

Luego de ejecutar cada uno de los algoritmos, con el nuevo set de datos, en la siguiente Tabla es posible comparar los resultados obtenidos. En la solapa Clasificar (Classify), se realiza el entrenamiento de cada algoritmo en WEKA, cada uno de ellos se entrenó usando la opción ***Use training set*** [7]**.**

Tabla 2. Comparación de los resultados obtenidos de las pruebas de clasificación con datos originales y con datos sintéticos.

Porcentaje de Instancias Clasificadas Datos Originales			**Porcentaje de Instancias Clasificadas Datos con instancias artificiales**		
RNA	**MVS**	**Árbol de Decisión**	**RNA**	**MVS**	**Árbol de Decisión**
Correctas: 63.9 % Incorrectas: 36.1 %	**Correctas: 60.3 % Incorrectas: 39.7 %**	**Correctas: 72.5 % Incorrectas: 27.5 %**	**Correctas: 66.8 % Incorrectas: 33.2 %**	**Correctas: 52.9 % Incorrectas: 47.1 %**	**Correctas: 65.9 % Incorrectas: 34.1 %**
Matriz de Confusión a-Alto b-Bajo			**Matriz de Confusión a-Alto b-Bajo**		
a b **146 65** **65 84**	**a b** **173 38** **105 44**	**a b** **184 27** **72 77**	**a b** **186 25** **114 94**	**a b** **194 17** **180 28**	**a b** **150 61** **82 126**
Curva Roc			**Curva Roc**		
0.68	**0.558**	**0.746**	**0.749**	**0.527**	**0.688**

Se observa que solo la RNA mejoró su precisión, respecto a la prueba anterior, pero sin llegar a tener. el porcentaje obtenido por el mejor clasificador (AD). En cambio, los otros clasificadores (AR y MVS) mermaron sus porcentajes, respecto a la prueba anterior. En relación con el área bajo la curva (AUC) en las figuras 4 y 5, está la gráfica de ambas curvas. AUC puede interpretarse como la probabilidad de que, ante una instancia nueva de datos, la prueba los clasifique correctamente. Su rango de valores va desde 0, siendo este valor el correspondiente a una prueba sin capacidad discriminante, hasta 1, que es cuando los dos grupos están perfectamente diferenciados por la prueba. Por tanto, podemos decir que cuanto mayor sea el AUC, mejor será la prueba. También la técnica RNA, tuvo el mejor valor AUC, sin llegar al nivel de la técnica AR, de la prueba anterior.

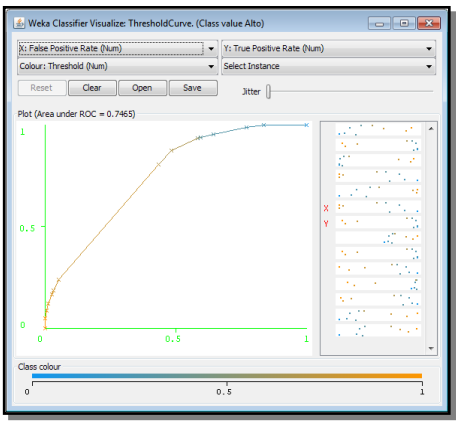

Fig. 4. Curva Roc del AD con los datos originales.

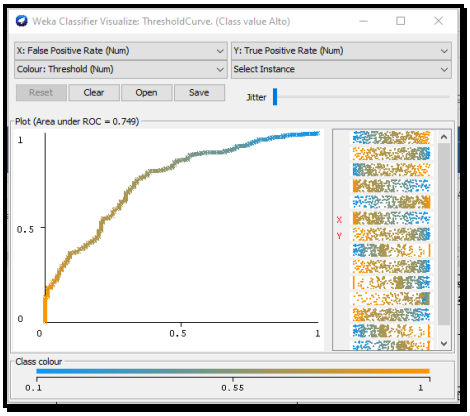

Fig. 5. Curva Roc de la RNA con datos formados por el filtro SMOTE.

4 Conclusiones y Trabajo a Futuro

Este estudio tuvo como objetivo verificar el comportamiento de tres algoritmos supervisados, con un dataset que tiene una porción de los datos construidos en forma artificial, mediante la aplicación del algoritmo de sobre-muestreo SMOTE, provisto por el software WEKA.

Después de haber efectuado todas las pruebas pertinentes sobre los modelos de clasificación propuestos usando estos datos sintéticos, es posible elaborar una serie de conclusiones:

- Es el primer trabajo sobre desbalanceo de clases en el área de la ganadería con estos tipos de datos.
- Dependiendo del tipo de algoritmo, se demuestra que, si bien, con algunas técnicas se puede mejorar los porcentajes de instancias bien clasificadas, en otros casos, los porcentajes disminuyen.
- Si bien, para la comparación se usó el valor alfa propuesto por WEKA (α=5), se probó con distintos valores: 2,3,4 y 10, para realizar las pruebas. Dando distintos porcentajes dependiendo del algoritmo.

Para futuras investigaciones se prevé seguir con las siguientes líneas:

- Realizar una comparación con los algoritmos No supervisados, empleados también en la investigación anterior [3].
- Probar con vistas minables de mayor cantidad de muestras y con un mayor porcentaje de diferencia entre las clases mayoritarias y minoritarias.
- Probar con vistas minables que posean más de dos clases.
- Probar con otras variantes del algoritmo SMOTE [8] y con otros algoritmos que tratan el desbalanceo de clase. Por ejemplo: el método de Sobremuestreo por agrupaciones o Clustered Based Oversampling (CBOS).
- Estudiar la relación de los porcentajes de precisión, dependiendo del valor α propuesto.
- Incursionar en otros programas para el aprendizaje automático y la minería de datos.

Referencias

1 Hernández Orallo, Introducción a la minería de datos, Pearson, 2004.

2 Mera, C. & Arrieta Ramos, J.M., «Estudio Comparativo de Técnicas de Balanceo de Datos en el Aprendizaje de Múltiples Instancias», 2015. [En línea]: https://www.researchgate.net /publication/283642919_Estudio_Comparativo_de_Tecnicas_de_Balanceo_de_Datos_en_el _Aprendizaje_de_Multiples_Instancias/link/5642151a08aec448fa621f60/download. [Último acceso: 01/07/2020].

3 Spositto, O., «Peso al Nacer de Terneros Aberdeen Angus mediante Algoritmos No Supervisados,» 2019. [En línea]: https://www.researchgate.net/profile/Lorena_Matteo/ publication/337445353_Peso_al_Nacer_de_Terneros_Aberdeen_Angus_mediante_Algoritm os_No_Supervisados/links/5dd7f187458515dc2f439029/Peso-al-Nacer-de-Terneros-Aberdeen-Angus-mediante-Algoritmos-No-Supervisad. [Último acceso: 01/07/2020].

4 Quinteros, O. y otros., «Construcción de una vista minable para aplicar minería de datos secuenciales temporales», 2016. [En línea]: http://sedici.unlp.edu.ar/handle/10915/56747. [Último acceso: 01/07/2020].

5 Monti, A, «Interpretación y uso correcto de los DEPs como herramienta de selección», 1998. [En línea]: https://es.scribd.com/document/337947981/20-Interpretacion-Deps. [Último acceso: 01/07/2020].

6 Simeone, O., «A Very Brief Introduction to Machine Learning», 2018. [En línea]: https://arxiv.org/pdf/1808.02342.pdf. [Último acceso: 01/07/2020].

7 Witen I., Data Mining. Practical Machine Learning Tools and Techniques., Morgan Kaufmann. ISBN: 0-12-088407-0., 2005.

8 Castro Pérez, N. «Preprocesamiento de datos termográficos por medio de técnicas de balanceo de clases y análisis de cúmulos (Clustering)», 2013. [En línea]: http://docplayer.es/ 17472207-Preprocesamiento-de-datos-termograficos-por-medio-de-tecnicas-de-balanceo-de-clases-y-analisis-de-cumulos-clustering.html. [Último acceso: 2020 07 01].

9 Chawla, N.V., «SMOTE: Synthetic Minority Oversampling Technique», 2002. [En línea]: https://www.jair.org/index.php/jair/_article/view/10302. [Último acceso: 01/07/2020].

10 Moreno, J. «SMOTE-I: mejora del algoritmo SMOTE para balanceo de clases minoritarias», 2009. [En línea]: https://www.researchgate.net/publication/229045207_SMOTE-I_mejora_ del_algoritmo_SMOTE_para_balanceo_de_clases_minoritarias/_citation/download. [Último acceso: 01/07/2020].

11 Rodríguez Torres, F., «SMOTE-D, una versión determinista de smote», 2017. [En línea]: https://inaoe.repositorioinstitucional.mx/jspui/bitstream/1009/335/1/RodriguezTF.pdf. [Último acceso: 01/07/2020].

12 Huang, P. «Classication of Imbalanced Data Using Synthetic Oversampling Techniques», 2015. [En línea]: https://escholarship.org/content/qt72w743h7/ qt72w743h7.pdf. [Último acceso: 01/07/2020].

13 Jiawei H. y otros, Data Mining: Concepts and Techniques, http://myweb.sabanciuniv.edu/ rdehkharghani/files/2016/02/The-Morgan-Kaufmann-Series-in-Data-Management-Systems-Jiawei-Han-Micheline-Kamber-Jian-Pei-Data-Mining.-Concepts-and-Techniques-3rd-Edition-Morgan-Kaufmann-2011.pdf: 3ra. Edición. (2011). ISBN 978-0-12-381479-1.

14 Farías Concha, M., «Máquinas Vectoriales híbridas para clasificar accidentes de tránsito en la región metropolitana», 2011. [En línea]: http://opac.pucv.cl/pucv_txt/txt-9500/ UCF9980_01.pdf. [Último acceso: 01/07/2020].

15 Benavides, Ana, «Curvas ROC (Receiver-Operating-Characteristic) y sus aplicaciones», 2017. [En línea]: https://idus.us.es/bitstream/handle/11441/63201/Valle%20Benavides %20 Ana%20Roc%C3%ADo%20del%20TFG.pdf?sequence=1. [Último acceso: 01/07/2020].

16 Dal Pozzo, A. y otros. «Racing for unbalanced methods selection», 2013. [En línea]: https://www.slideshare.net/dalpozz/racing-for-unbalanced-methods-selection. [Último acceso: 01/07/2020].