

UNIVERSIDAD
NACIONAL
DE COLOMBIA

APRENDIZAJE CON CLASES DESBALANCEADAS

Carlos Mera, PhD

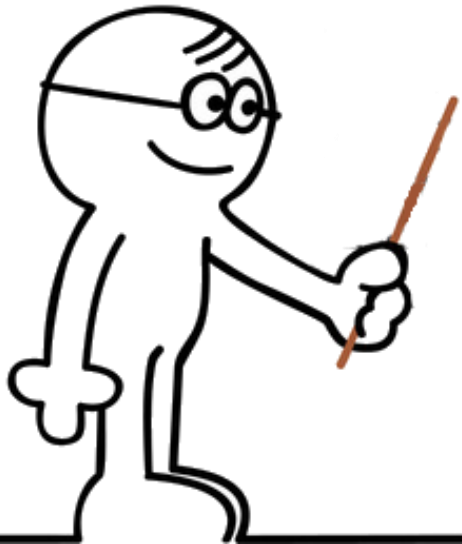
Departamento de Ciencias de la Computación y de la Decisión
Investigador del Grupo de I+D en Inteligencia Artificial – GIDIA

camerab@unal.edu.co

Contenido

- El Problema de las Clases Desbalanceadas
 - Métodos de muestreo
 - Ensamblajes de Clasificadores

MOTIVACIÓN



Motivación

ANALICE EL SIGUIENTE CASO:

- Se ha recolectado la información de 100.000 personas que han estado asociadas a una nueva enfermedad que está azotando al país.
- Se confirmó que 5.000 de esas personas tienen la enfermedad y el resto son personas sanas.
- La información que se reunió de cada persona involucra tanto datos demográficos como datos clínicos.
- Esos datos fueron pre-procesados y con ellos se entrenó un clasificador binario que dados los datos de un paciente determina si el paciente tiene o no la enfermedad.
- El clasificador resultante obtuvo una precisión del 95%, es decir se equivocó solo en el 5% de los datos.

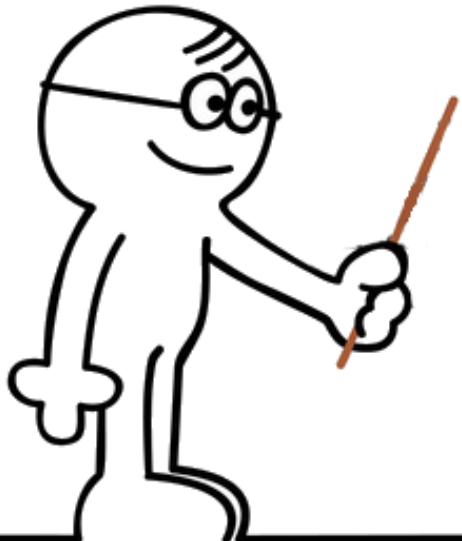
Matriz de Confusión del Clasificador

	Predicción Enfermo	Predicción Sano
Real Enfermo	0	5000
Real Sano	0	95000



¿Qué opinión puede dar usted del clasificador?
¿Es bueno ese clasificador?

¿CUÁL ES EL PROBLEMA DE APRENDER
CON CLASES DESBALANCEADAS?



El Problema

- Se habla del aprendizaje sobre conjuntos de datos desbalanceados cuando las muestras de una clase son relativamente pocas, comparadas con las de la otra clase.

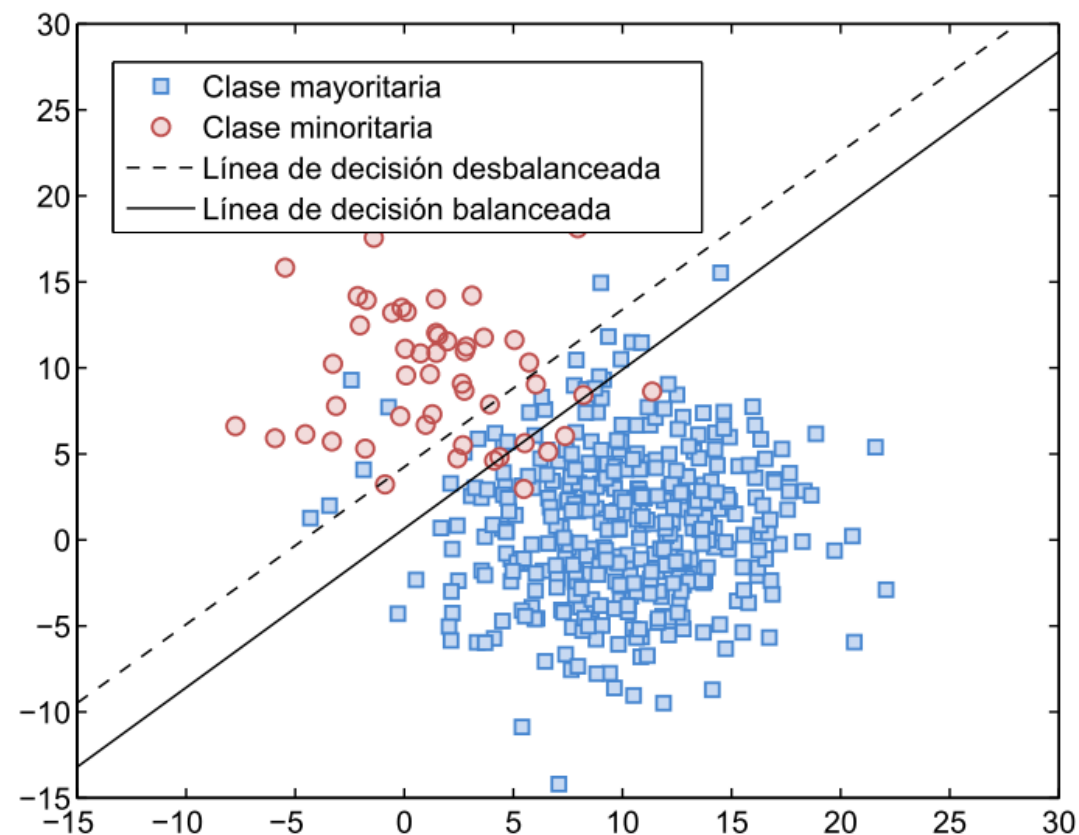


Importante:

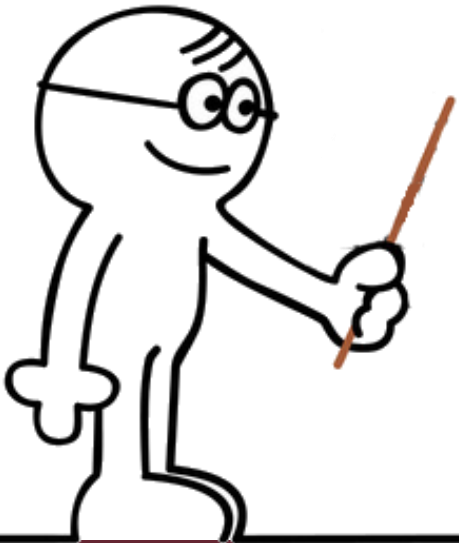
Esto supone un problema puesto que la mayoría de los algoritmos de aprendizaje asumen que el conjunto de datos está relativamente balanceado y cuando esto no ocurre el algoritmo se puede sesgar, afectando su capacidad de generalización y perjudicando las clases minoritarias.

En la práctica hay muchos problemas donde el conocimiento más importante reside en la clase minoritaria

[HG2009] H. He & E.A. Garcia. "Learning from Imbalanced Data"



¿POR QUÉ ES UN PROBLEMA?



El Problema

 POR QUÉ EL SESGO DE LOS ALGORITMOS DE APRENDIZAJE:

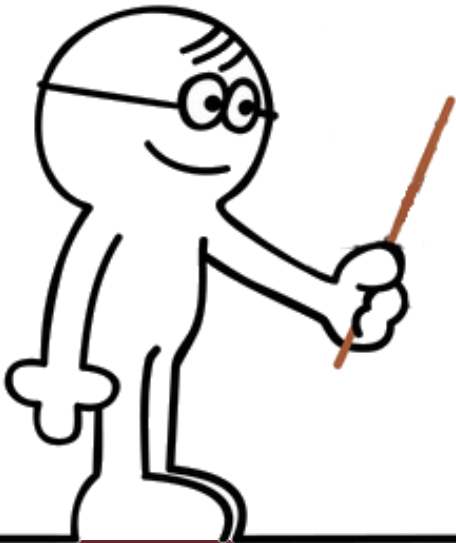
 Una justificación rápida desde la perspectiva de los Clasificadores basados en el Teorema de Bayes:

$$f^*(\mathbf{x}) = \arg \max_{\omega_j \in \Omega} P(\omega_j | \mathbf{x})$$

$$f^*(\mathbf{x}) = \arg \max_{\omega_j \in \Omega} \frac{p(\mathbf{x} | \omega_j) P(\omega_j)}{p(\mathbf{x})}$$




$$f^*(\mathbf{x}) = \begin{cases} \omega_1, & \text{si } \frac{p(\mathbf{x} | \omega_1)}{p(\mathbf{x} | \omega_2)} > \frac{P(\omega_2)}{P(\omega_1)} \\ \omega_2, & \text{en otro caso} \end{cases}$$

¿CÓMO TRATAR DE SOLUCIONAR ESTE
PROBLEMA?







La Solución

¿CÓMO RESOLVER EL PROBLEMA?

-  Balancear la distribución de las clases
-  Eliminar aquellos datos “ruidosos” que quedan en el lado equivocado de la frontera de decisión
-  Considerar un costo para aquellos datos que son mal clasificados y que pertenecen a la clase minoritaria

APROXIMACIONES

-  Métodos de muestreo
-  Métodos de aprendizaje basados en ensambles de clasificadores
-  Métodos de aprendizaje basados en costos
-  Métodos de aprendizaje de una sola clase

TÉCNICAS DE MUESTREO SIMPLE

Consiste en eliminar muestras de la clase mayoritaria (submuestreo) y / o agregar más ejemplos de la clase minoritaria (sobremuestreo).

Si el conjunto de
datos está
desbalanceado...



Modifique la
distribución de los
datos ..



Cree un conjunto
balanceado

Técnicas de Muestreo Simple




Importante:

El muestreo (o re-muestreo) es el proceso de manipular la distribución de los datos de entrenamiento en un esfuerzo por mejorar el rendimiento de los clasificadores.

Así, la idea general de las técnicas de muestreo es agregar o eliminar datos del conjunto de entrenamiento con la esperanza de alcanzar la distribución óptima de los datos.

Técnicas de Muestreo Simple


SOBREMUESTREO ALEATORIO

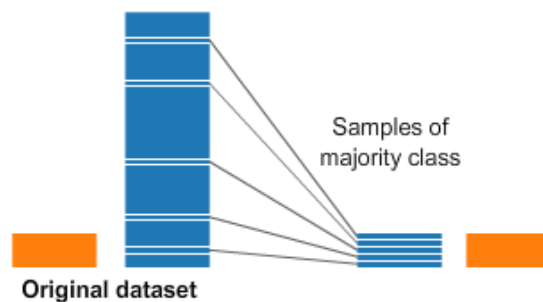
-  Consiste en seleccionar, aleatoriamente, datos en la clase minoritaria y duplicarlos hasta que el número de muestras en ambas clases sea similar.



Desventaja: Genera problemas de sobreajuste, debido a las instancias que se repiten en el conjunto de datos

SUBMUESTREO ALEATORIO

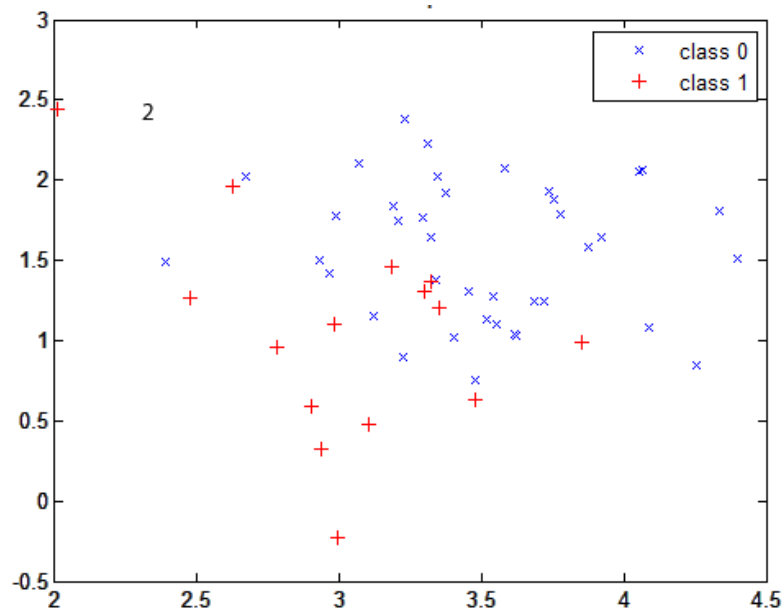
-  Consiste en seleccionar, aleatoriamente, datos en la clase mayoritaria y eliminarlos hasta que el número de muestras en ambas clases sea similar.



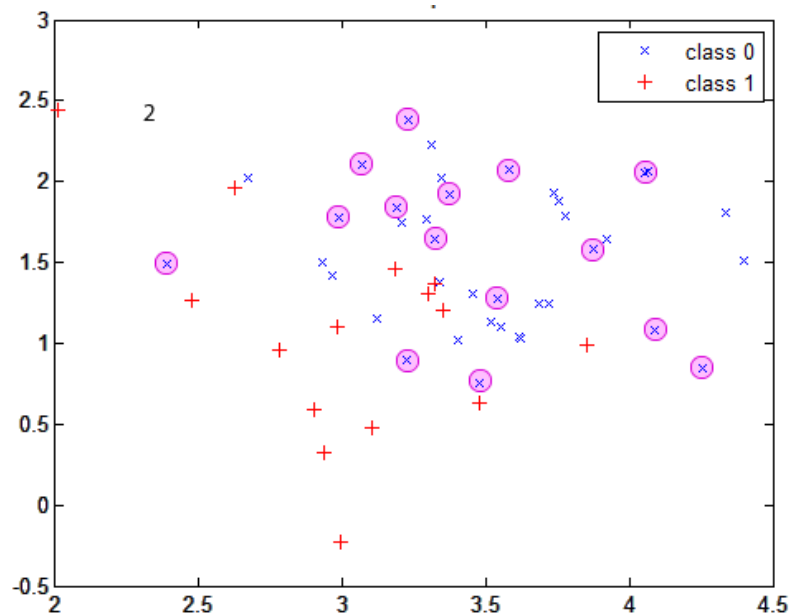
Desventaja: Se pueden perder instancias importantes en el proceso de selección aleatoria

Técnicas de Muestreo Simple

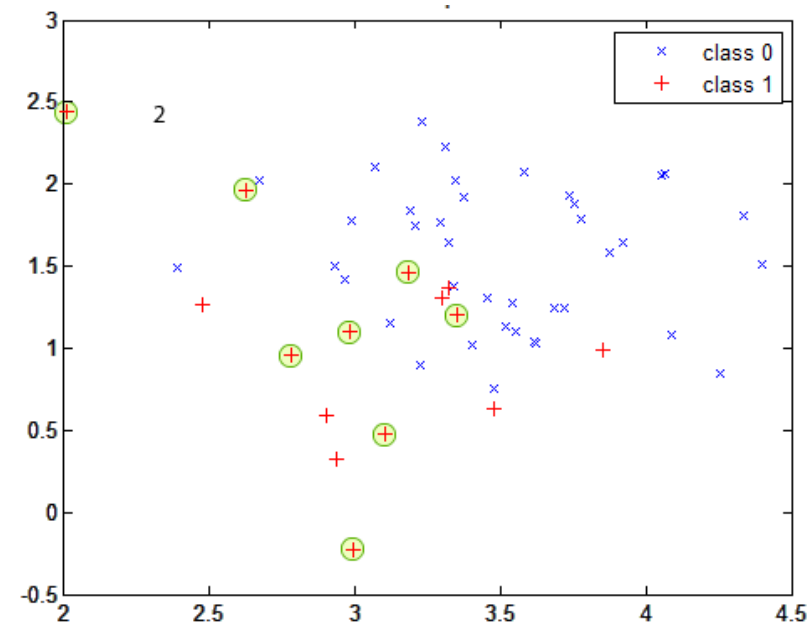
Conjunto Original



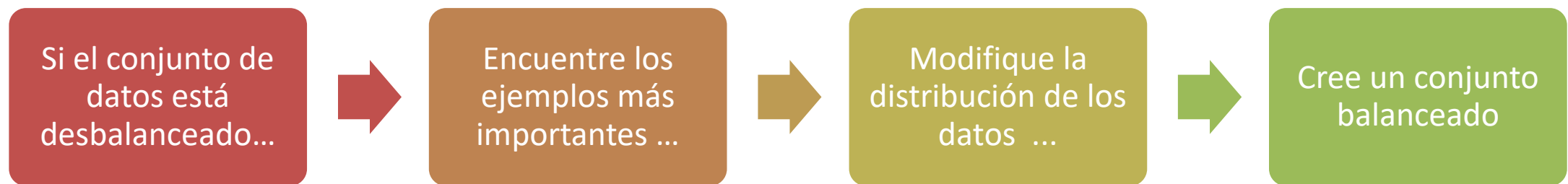
Instancias seleccionadas para eliminar



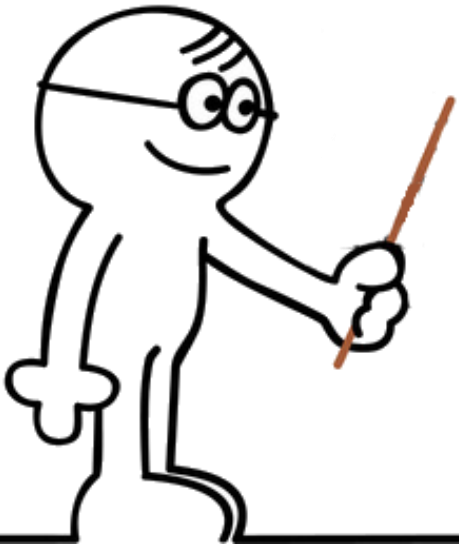
Instancias seleccionadas para duplicar



TÉCNICAS DE MUESTREO INFORMADO



SUB-MUESTREO INFORMADO



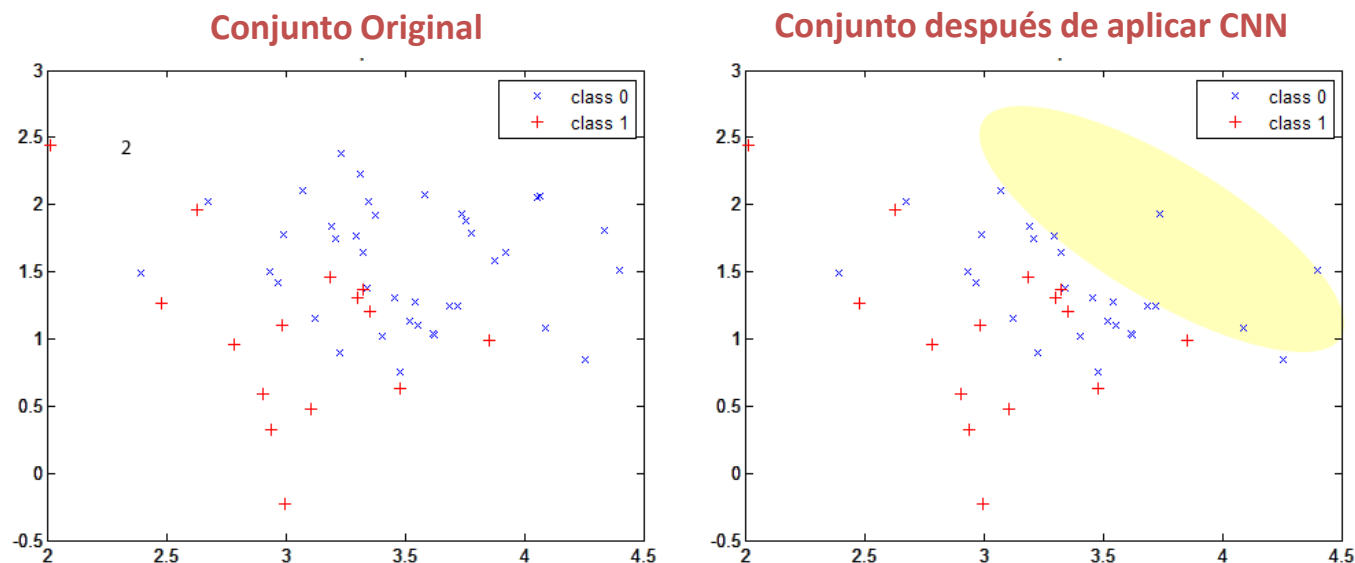
Técnicas de Sub-Muestreo Informado

CONDENSED NEAREST NEIGHBOR RULE (CNN)

🔑 CNN es una técnica cuyo objetivo es encontrar los puntos de la clase mayoritaria que están lejos de los límites de decisión y eliminarlos.

El algoritmo es bastante simple:

1. A partir del conjunto de entrenamiento Z , cree un nuevo conjunto de datos (Z_{new}) que tenga todos los datos de la clase minoritaria y un dato de la clase mayoritaria seleccionado de manera aleatoria.
2. Usando el algoritmo kNN, con $k=1$, clasifique cada uno los puntos de la clase mayoritaria usando el conjunto Z_{new} . Si un punto de la clase mayoritaria es mal clasificado, agréguelo a Z_{new} .



Es una técnica sensible a los datos con ruido, por tanto mantiene los datos ruidosos en el conjunto resultante

[Hart1968] P. E. Hart. "The Condensed Nearest Neighbor Rule".

Técnicas de Sub-Muestreo Informado

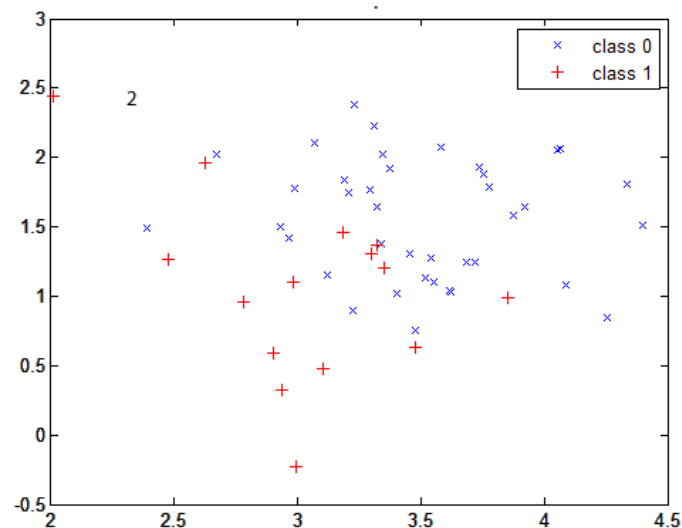
TOMEK LINKS

- 🔗 Tome Links elimina aquellos puntos de la clase mayoritaria que se mezclan con los puntos de la clase minoritaria en la frontera de decisión.

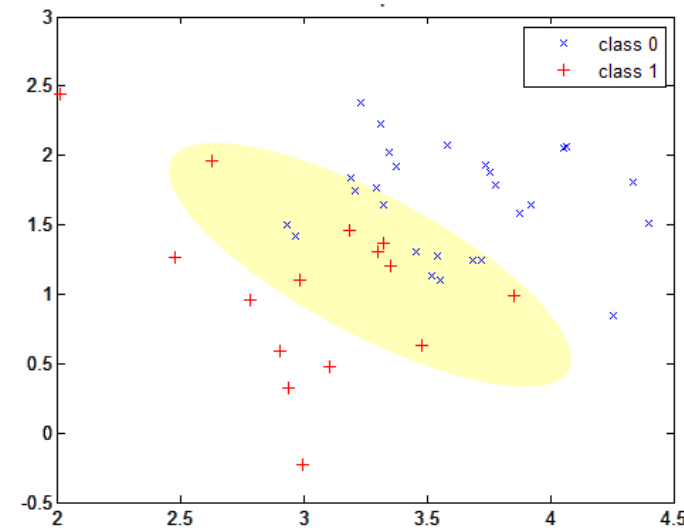
El algoritmo es el siguiente:

1. Dados dos ejemplos x_i en Z_{\min} y x_j en Z_{\max} , $d(x_i, x_j)$ define la distancia entre ellos
2. El par (x_i, x_j) se llama un **Tomek Link** si no hay un dato x_k , de modo que:
 $d(x_i, x_k) < d(x_i, x_j)$ o $d(x_j, x_k) < d(x_i, x_j)$
3. Solo elimine los puntos que pertenecen a la clase mayoritaria en los **Tomek Links**

Conjunto Original



Conjunto después de aplicar Tomek Links



[Tomek1976] I. Tomek, "Two modifications of CNN".

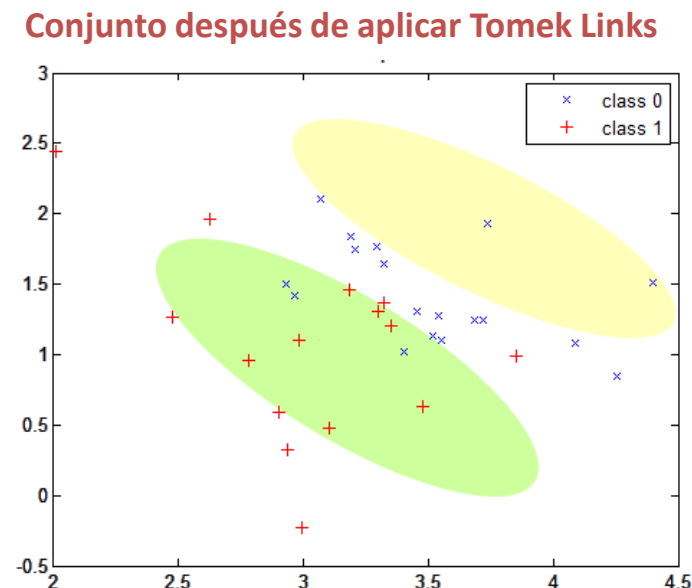
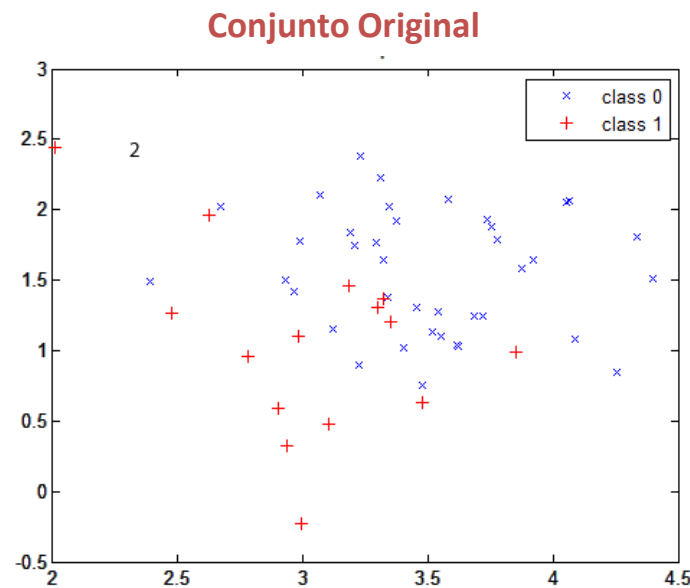
Técnicas de Sub-Muestreo Informado

ONE-SIDED SELECTION (OSS)

Esta técnica es una mezcla entre los dos anteriores.

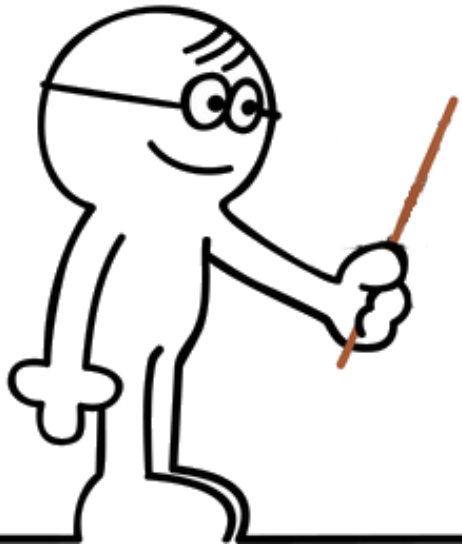
El algoritmo es el siguiente:

1. Sea Z el conjunto de datos original
2. Use el algoritmo CNN para eliminar los datos redundantes de la clase mayoritaria. Al nuevo conjunto de datos llámelo S .
3. Ahora Aplique el algoritmo de Tomek Links sobre S para eliminar los puntos "ruidosos" en la frontera de decisión.



[KM1997] M. Kubat and S. Matwin. "Addressing the curse of imbalanced training sets: One-sided selection".

SOBRE-MUESTREO INFORMADO



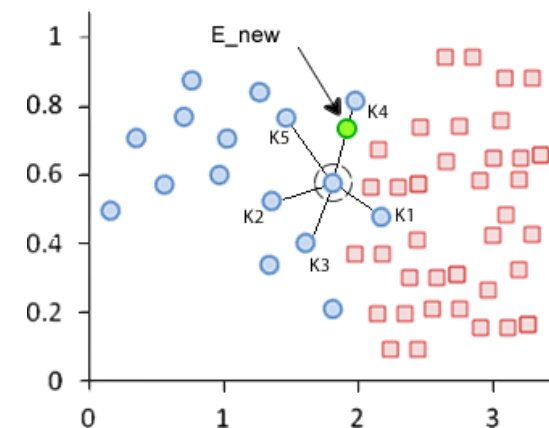
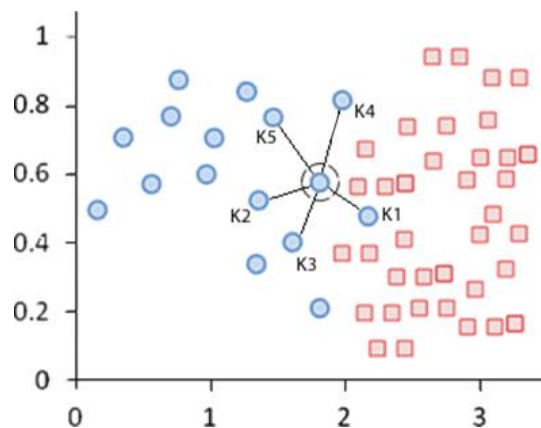
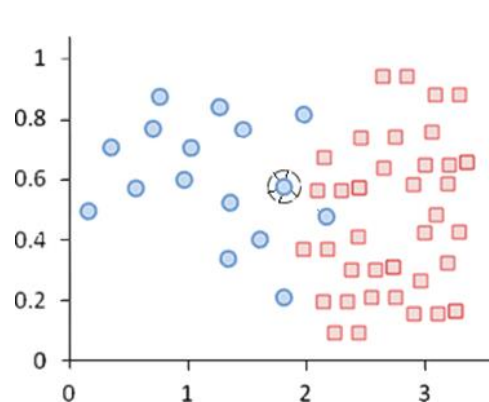
Técnicas de Sobre-Muestreo Informado

SMOTE: SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE

Esta técnica genera datos sintéticos en la clase minoritaria.

El algoritmo es simple:

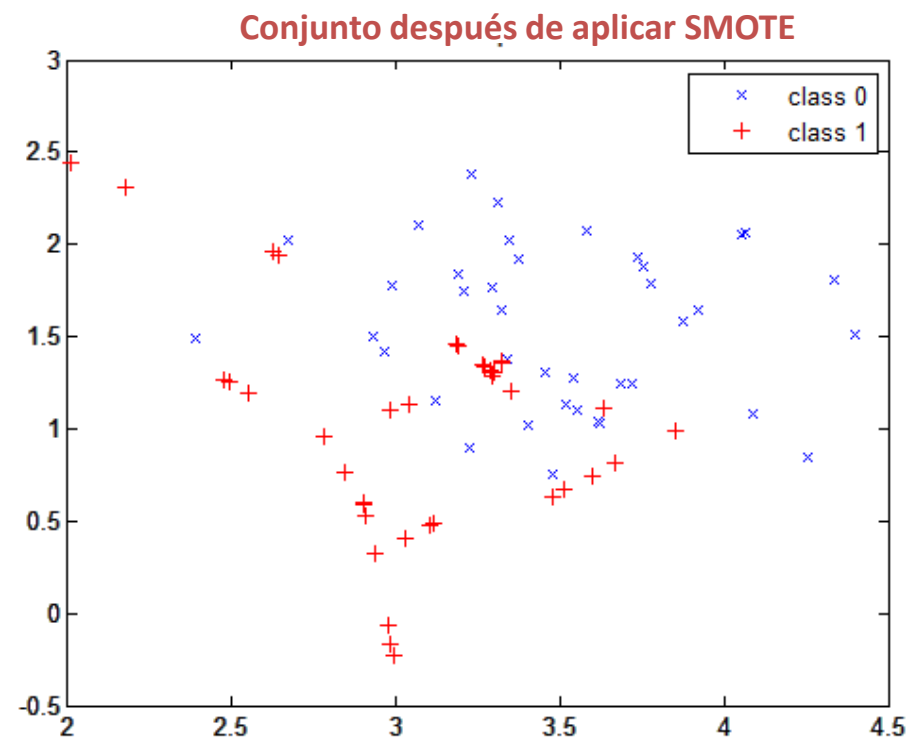
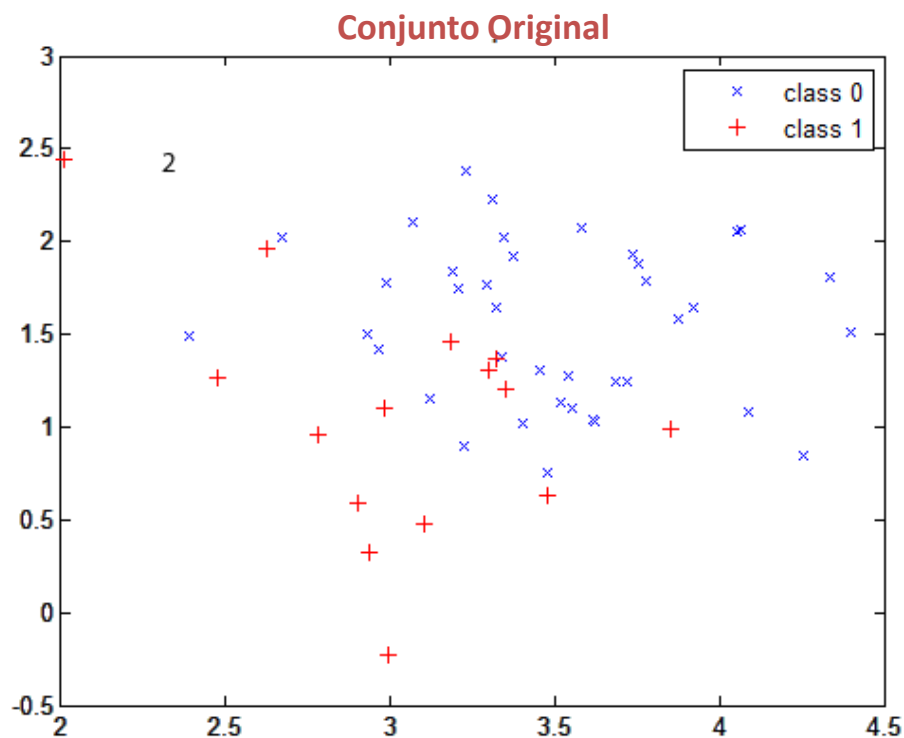
1. Para cada ejemplo E_i de la clase minoritaria
2. Encuentre los k vecinos más cercanos a E_i
3. Seleccione aleatoriamente uno de esos k -vecinos y llámelo E_j
4. Cree un dato sintético E_{new} y agréguelo al conjunto de datos:
$$E_{new} = E_i + (E_j - E_i) * \alpha$$
, donde α es un valor aleatorio entre 0 y 1



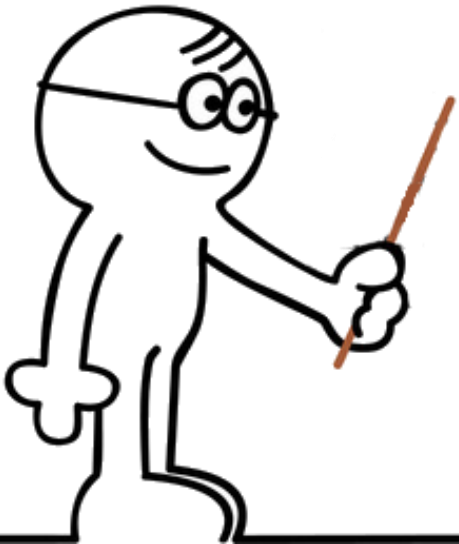
[CBH+2002] N. V. Chawla et al. "SMOTE: synthetic minority over-sampling technique"

Técnicas de Sobre-Muestreo Informado

● SMOTE: SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE



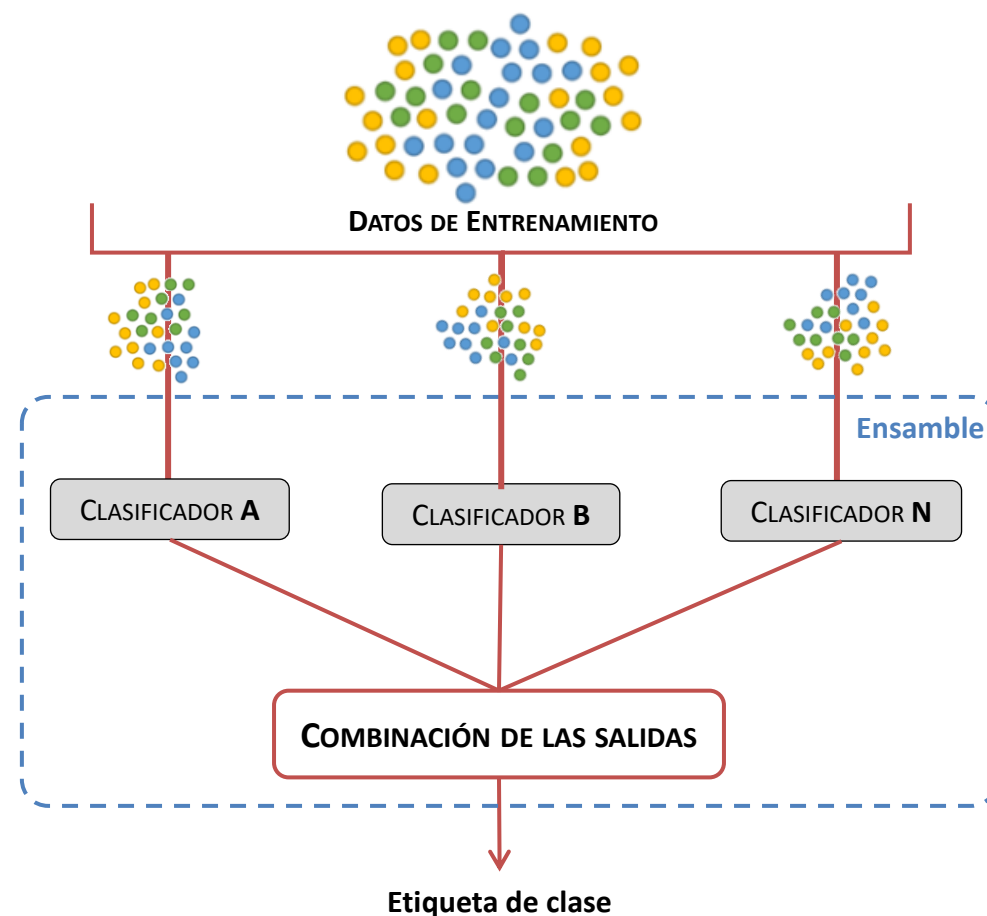
ENSAMBLES DE CLASIFICADORES



Ensamblajes de Clasificadores

MOTIVACIÓN

- Los clasificadores basados en ensambles intentan mejorar el rendimiento de los clasificadores individuales, combinándolos para obtener un nuevo clasificador que supere a cada uno de ellos.
- La idea básica es construir varios clasificadores a partir de los datos originales y luego combinar sus predicciones para clasificar los datos desconocidos.
- Esta idea sigue el comportamiento natural humano que tiende a buscar varias opiniones antes de tomar una decisión importante. Cuando las personas tienen que tomar decisiones difíciles, muchas veces toman en cuenta la opinión de varios expertos, buscando mejorar sus decisiones.



[Kuncheva2004] Ludmila Kuncheva. "Combining Pattern Classifiers, Methods and Algorithms"

[Rokach2009] L. Rokach *et al.* "Taxonomy for Characterizing ensemble methods in classification tasks"

Ensamblas de Clasificadores



Importante:

Puede ser útil explotar la *diversidad entre clasificadores* y combinar sus salidas para mejorar el desempeño de los métodos tradicionales de clasificación

[\[Kuncheva2004\]](#) [\[Rokach2009\]](#).

A partir de esta premisa surge la pregunta: ¿Cómo garantizamos la diversidad de los clasificadores?



BOOSTING

Ensamblajes de Clasificadores

BOOSTING

- Es una estrategia en la que el ensamble se construye de manera secuencial.
- El algoritmo funciona entrenando un clasificador inicial con un subconjunto aleatorio del conjunto de entrenamiento original.
- Los clasificadores posteriores se construyen ajustando los valores del error residual del clasificador anterior.
- Así, la idea general es tratar de centrar la atención de un clasificador en aquellas observaciones que el clasificador anterior estimó pobremente.
- Una vez que se crea la secuencia de los clasificadores, las predicciones hechas por los estos son ponderadas por sus puntuaciones de precisión y los resultados se combinan para crear una estimación final.
- Algunos de los modelos que normalmente se utilizan en la técnica de refuerzo son ADABOOST y XGBOOST

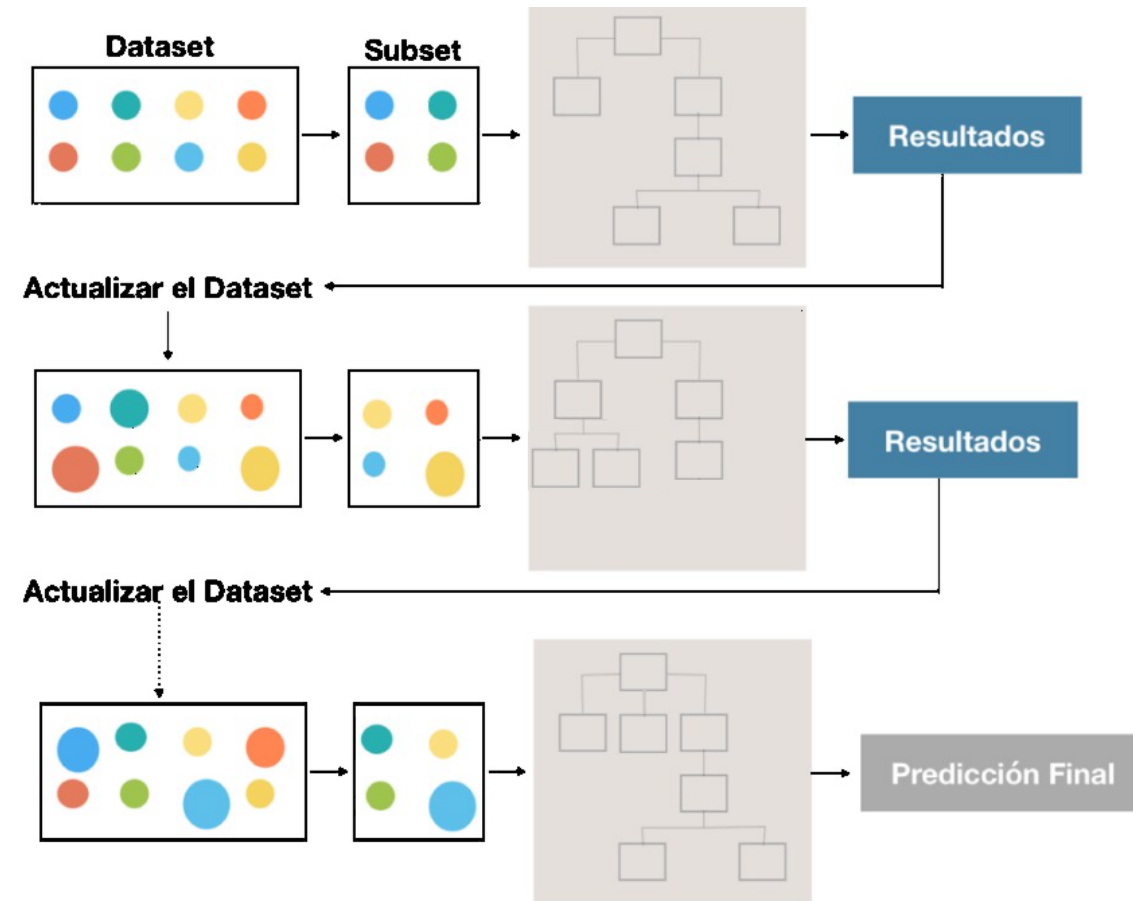


Imagen modificada de: <https://www.flickr.com/photos/ligdieli/48051633738/>

Ensamblas de Clasificadores

BOOSTING: ADABOOST

ADABOOST

1. Assign every observation, x_i , an initial weight value, $w_i = \frac{1}{n}$, where n is the total number of observations.
2. Train a "weak" model. (most often a decision tree)
3. For each observation:
 - 3.1. If predicted incorrectly, w_i is increased
 - 3.2. If predicted correctly, w_i is decreased
4. Train a new weak model where observations with greater weights are given more priority.
5. Repeat steps 3 and 4 until observations perfectly predicted or a preset number of trees are trained.

Chris Albon

https://chrisalbon.com/machine_learning/trees_and_forests/adaboost_classifier/

BAGGING

Ensamblas de Clasificadores

BAGGING

- El **bagging** es utilizado para generar subconjuntos de datos haciendo una **selección aleatoria con reemplazo**, es decir, cada dato con el que se crea un subconjunto no deja de pertenecer al conjunto de datos original.
- Esto permite que cada observación pueda estar varias veces en el subconjunto o estar en diferentes subconjuntos.
- Cada subconjunto creado es utilizado para entrenar un clasificador independiente.
- La predicción final es el voto mayoritario entre todos los clasificadores entrenados.
- Uno de los algoritmos más usados de **Bagging** son los bosques aleatorios (**Random Forest**) basado en árboles de decisión.

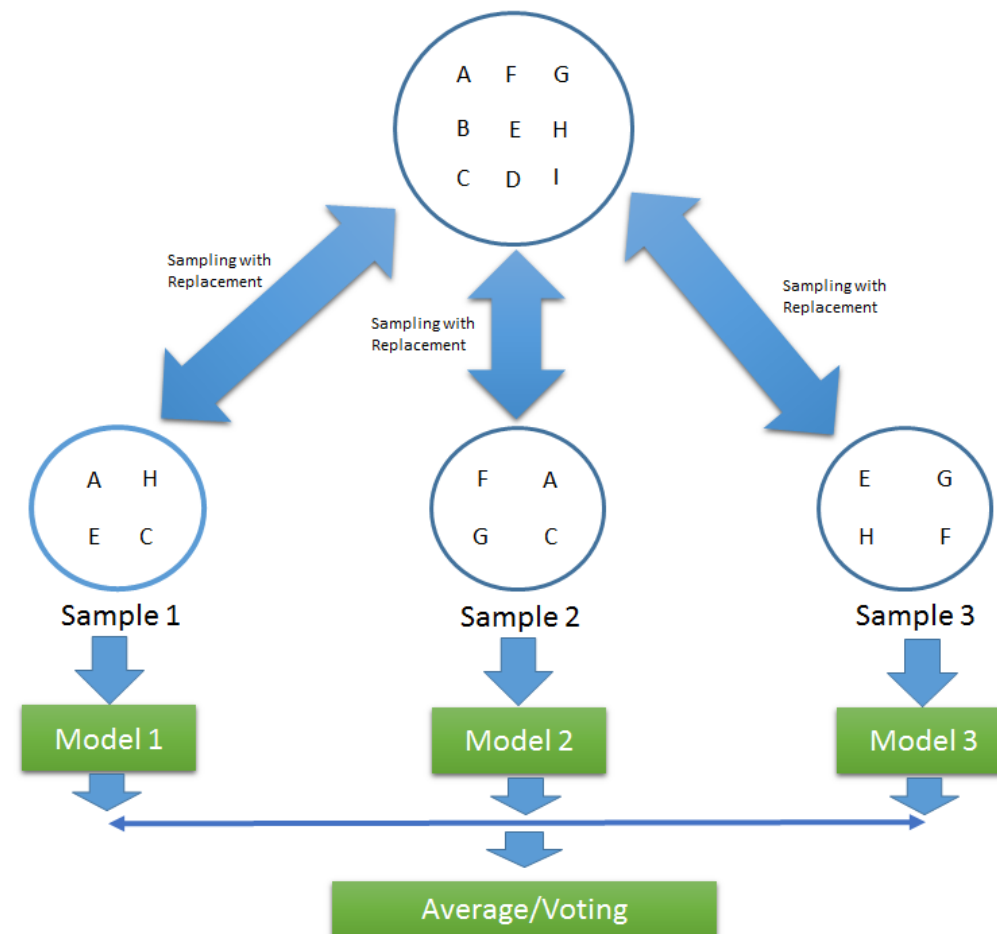


Imagen de: <https://medium.com/ml-research-lab/bagging-ensemble-meta-algorithm-for-reducing-variance-c98ffa5489f>

SUB-ESPACIOS ALEATORIOS

Ensamblas de Clasificadores

SUB-ESPACIOS ALEATORIOS

- 🔑 En este método cada modelo se entrena con todos los ejemplos, pero solo considera un **subconjunto de los atributos**. El tamaño de estos subconjuntos es el parámetro del método, y de nuevo el resultado es el promedio o votación de los resultados individuales de los modelos.

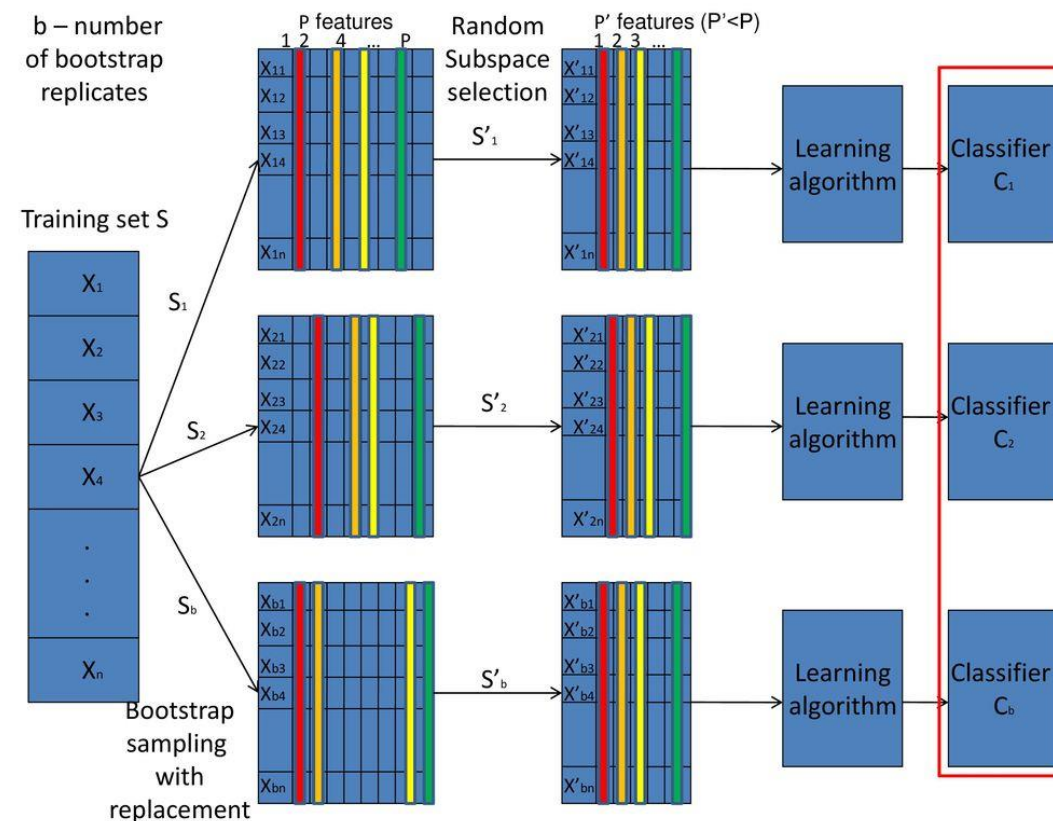


Imagen de: <https://medium.com/ml-research-lab/bagging-ensemble-meta-algorithm-for-reducing-variance-c98fffa5489f>



ESTRATEGIAS DE COMBINACIÓN

Ensamblas de Clasificadores

DISEÑO DEL COMBINADOR:

Para métodos de clasificación:

- ✓ Voto Mayoritario
- ✓ Voto por Pluralidad
- ✓ Voto Ponderado



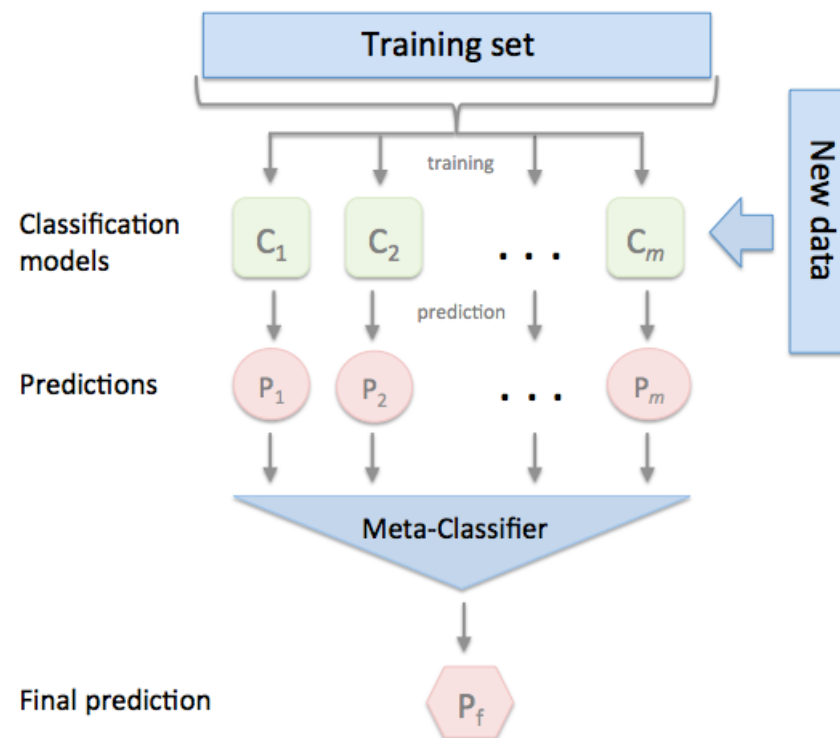
Para métodos de regresión:

- ✓ *Promedio Simple*
- ✓ *Promedio Ponderado*
- ✓ Reglas del Máximo y el Mínimo

$$H(x) = \frac{1}{L} \sum_{i=1}^L h_i(x) \quad H(x) = \frac{1}{L} \sum_{i=1}^L w_i h_i(x)$$

Combinación por Stacking:

- ✓ Con las salidas de los clasificadores individuales se entrena otro clasificador

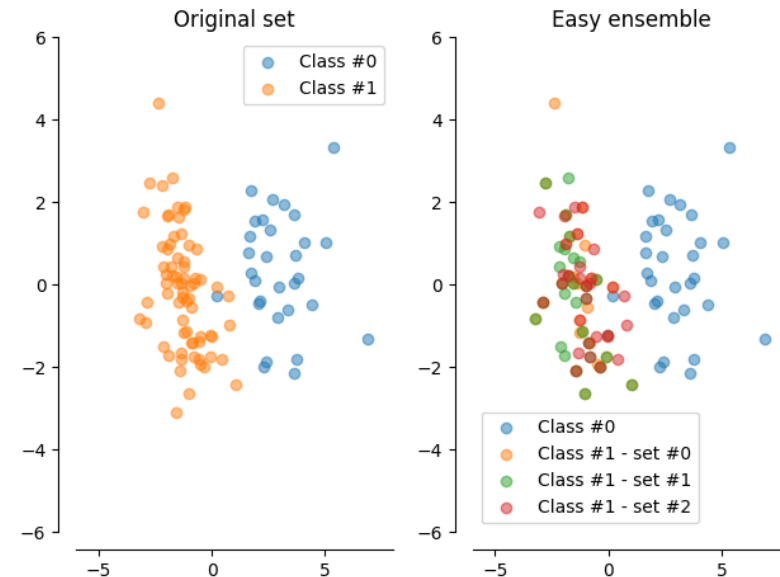


ENSAMBLES PARA CONJUNTOS DE DESBALANCEADOS

Ensamblas para Conjuntos Desbalanceados

EASY ENSEMBLE

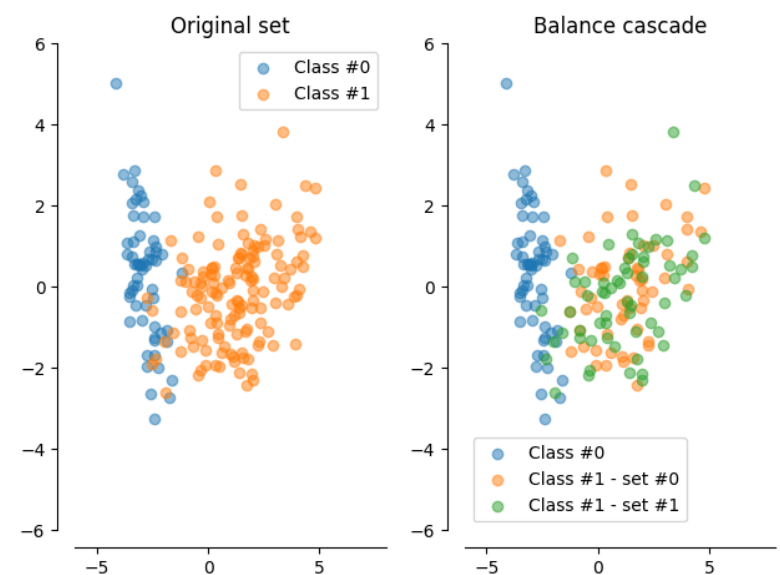
- 📁 Crea subconjuntos de datos aleatorios de la clase mayoritaria que tengan el tamaño de la clase minoritaria. Con cada subconjunto entrena un clasificador y combine sus salidas usando voto mayoritario.



BALANCE CASCADE

- 📁 Crea de forma iterativa, subconjuntos balanceados y va eliminando aquellas muestras redundantes en la clase mayoritaria para formar un clasificador final.

1. Generate $E \subset S_{maj}$ (s. t. $|E| = |S_{min}|$), and $N = \{E \cup S_{min}\}$
2. Induce $H(n)$
3. Identify N_{maj}^* as samples from N that are correctly classified
4. Remove N_{maj}^* from S_{maj}
5. Repeat (1) and induce $H(n + 1)$ until stopping criteria is met



Ensamblas para Conjuntos Desbalanceados

INTEGRATION OF SAMPLING AND BOOSTING

SMOTEBoost:

- SMOTE + AdaBoost.M2
- Introduce datos sintéticos en cada iteración del *boosting*

DataBoost-IM:

- AdaBoost.M1
- Genera datos sintéticos a partir de los datos que son difíciles de aprender, esto tanto para la clase mayoritaria como para la clase minoritaria.

Enlaces

UN PAR DE ENLACES DE INTERÉS:

 **ToolBox de Python para Balanceo de Datos:**

<https://imbalanced-learn.readthedocs.io/en/stable/index.html>

 **Un tutorial del uso del ToolBox:**

<https://www.aprendemachinelearning.com/clasificacion-con-datos-desbalanceados/>

 **Un blog explicativo de los métodos de muestreo:**

<https://machinelearningmastery.com/undersampling-algorithms-for-imbalanced-classification/>

 **Teorema del NO FREE LUNCH:**

El artículo: <http://ti.arc.nasa.gov/m/profile/dhw/papers/78.pdf>

Una explicación: <https://www.kdnuggets.com/2019/09/no-free-lunch-data-science.html>

Preguntas ...

CARLOS ANDRÉS MERA BANGUERO, PH.D.
camerab@unal.edu.co

