

# Clasificación Monotónica mediante poda de Bosques Aleatorios

Sergio González, Francisco Herrera, and Salvador García

Dept. de Ciencias de la Computación e Inteligencia Artificial,  
Universidad de Granada. 18071, Granada, España  
sergio.gvz@gmail.com, {herrera, salvagl}@decsai.ugr.es

**Resumen** En problemas de clasificación con restricciones de orden monotónico, la variable que determina la clase debe aumentar conforme lo hace el resto de variables. Los clasificadores estándares no aseguran producir un modelo que satisfaga estas restricciones. Por ello, se han desarrollado nuevos algoritmos que intentan solucionar dicho problema, tales como árboles de decisión con métodos adaptados de crecimiento y poda. En esta contribución, se ha estudiado la posibilidad del uso de estos mecanismos para la generación de *Random Forests*. Con ello, se propone un mecanismo simple de poda en el comité de clasificadores basado en el grado de monotonicidad. Tras un exhaustivo análisis experimental, se puede deducir que la aplicación de estos métodos al *Random Forest* permite obtener mejores resultados que los algoritmos estándares.

**Palabras clave:** Clasificación Monotónica, Árboles de decisión, *Random Forest*, Poda de *ensemble*.

## 1. Introducción

La clasificación de ejemplos en categorías ordinales es un problema popular que ha ido atrayendo la atención de los profesionales en minería de datos durante estos últimos años. En la literatura, han aparecido multitud de términos en relación a este problema, tales como, clasificación ordinal, regresión ordinal o *ranking labelling*, pero todos comparten una propiedad en común: el atributo de salida o clase es ordinal. La clasificación con restricciones monotónicas, también conocida como clasificación monotónica, [4], es un problema con una restricción clara sobre la monotonía de los datos: un valor mayor en un atributo, mientras los demás se mantienen, no debe asignarse una clase menor [9].

Los árboles de decisión [12] y la inducción de reglas [6] constituyen dos de las técnicas más prometedoras para resolver la clasificación monotónica. Cualquier método de clasificación puede integrarse en un clasificador basado en *ensemble*, con el objetivo de mejorar sus resultados. Sin embargo, para lograr esta mejora, es necesario una selección correcta de los diferentes clasificadores que forma la agrupación. Esto recibe el nombre de poda de *ensemble* o *ensemble pruning* [11]. *Random Forests* (RFs) es uno de los *ensembles* de árboles de decisión más conocidos basado en *bagging*.

En esta contribución, se persigue el objetivo de promover el uso de RF en tareas de clasificación monotónica e introducir un mecanismo de poda basado en un orden monotónico para RFs cuyos árboles hacen uso del índice de no monotonicidad.

Este documento está estructurado de la siguiente manera. La Sección 2 presenta el problema de la clasificación ordinal con restricciones monotónicas. En la Sección 3, se describe la propuesta del RF y su adaptación para que satisfaga dichas restricciones. La Sección 4 proporciona información sobre la experimentación realizada, así como, sobre los resultados obtenidos en el estudio empírico, discutiéndolos y analizándolos. Por último, la Sección 5 concluye el trabajo.

## 2. Clasificación Monotónica

Los problemas de clasificación ordinal son aquellos cuya clase no es ni numérica ni nominal, si no valores categóricos con una relación de orden. Por ejemplo, un trabajador se le puede describir como “excelente”, “bueno” o “malo”, y un bono financiero puede ser evaluado como “AAA”, “AA”, “A”, “A-”, etc. Parecido a una escala numérica, una ordinal también posee un orden concreto, pero este carece de una noción de distancia entre los diferentes valores. Los problemas de clasificación ordinal son de gran importancia, por su frecuente presencia en tareas de la vida diaria.

Un clasificador monotónico es aquel que no viola las restricciones de monotonicidad. Informalmente, la clasificación monotónica implica que los valores predichos de clase no decrezcan monotónicamente (en su orden) en relación a los valores de los atributos ordinales. Más formalmente, sea  $\{\mathbf{x}_i, \text{clase}(\mathbf{x}_i)\}$  un conjunto de ejemplos con el vector de atributos  $\mathbf{x}_i = (\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,m})$  y la clase,  $\text{clase}(\mathbf{x}_i)$ , siendo  $n$  el número de instancias y  $m$  el número de atributos. Sea  $\mathbf{x}_i \succeq \mathbf{x}_h$  si  $\forall_{j=1, \dots, m}, \mathbf{x}_{i,j} \geq \mathbf{x}_{h,j}$ . Un conjunto de datos  $\{\mathbf{x}_i, \text{clase}(\mathbf{x}_i)\}$  es monotónico si y sólo si todas las parejas de ejemplos  $i, h$  son monotónicas entre estas [3].

## 3. Random Forest Monotónico

En esta sección, se explica la propuesta basada en RF para abordar la clasificación monotónica. Las modificaciones introducidas al RF estándar se han centrado principalmente en la forma que se realiza la ramificación de cada árbol, promoviendo la diversidad con un nuevo factor aleatorio y la agregación de los resultado con el mecanismo de poda propuesto, manteniendo el método muestreo intacto.

Primero, se debe comenzar definiendo el índice de no monotonicidad o *non-monotonicity index* (NMI) como el ratio de violaciones de la restricción monotónica dividido por el número total de ejemplos del conjunto. Inicialmente, se introdujo un algoritmo basado en MID para el proceso de construcción del árbol. Con este cambio, se consigue el objetivo inicial de adaptar el conocido *ensemble* a la clasificación monotónica. Más concretamente, se eligió el método

---

**Algorithm 1** Algoritmo RF Monotónico.

---

```

function MONRF( $D$  -conjunto de datos,  $nArboles$  - número de árboles aleatorios
construidos,  $R_{limite}$  - factor de importancia para las restricciones monotónicas,  $T$  -
Umbral usado en el procedimiento de poda,  $S$  - predicción para conjunto  $D$ )
  initialize:  $S = \{\}$ ,  $Trees[1..nArboles]$ ,  $D_{bootstraps}[1..nArboles]$ ,
 $NMIs[1..nArboles]$ 
  for  $i$  in  $[1, nArboles]$  do
     $D_{bootstraps}[i] = Muestrear\_Bootstrap(nArboles, D)$ 
     $rand = Aleatorio(1, R_{limite})$ 
     $Arboles[i] = Construir\_Arbol(D_{bootstraps}[i], rand)$ 
     $NMIs[i] = Calcular\_NMI(Arboles[i])$ 
  end for
   $Arboles = Ordenar(Arboles, NMIs)$ 
  for  $i$  in  $[1, \lceil nArboles * T \rceil]$  do
     $Arb\hat{o}les \leftarrow Arboles[i]$ 
  end for
  for  $d$  in  $D$  do
     $S \leftarrow Predecir\_Mayoría\_Voto(Arb\hat{o}les, d)$ 
  end for
  return  $S$ 
end function

```

---

MID-C4.5 para el desarrollo de cada uno de los árboles aleatorios del bosque. Este selecciona el mejor atributo para la ramificación del árbol usando como criterio la medida *total-ambiguity-score*. Esta medida fue introducida por primera vez por Ben-David en [3] como la suma del *E-score* del algoritmo ID3 y la medida *order-ambiguity-score* ponderada por el parámetro  $R$ . El *order-ambiguity-score* se calcula, tal como se muestra en la Ecuación 1, usando el concepto del índice de no monotonicidad, como el ratio entre el número de parejas de ramas no monotónica en el momento del cálculo y el total de parejas que podría haber sido no monotónicas. En el MID-C4.5, la entropía del ID3 se sustituye por la ganancia de información propia del árbol de decisión C4.5.

$$A = \begin{cases} 0 & \text{if } NMI = 0 \\ -(\log_2 NMI)^{-1} & \text{otherwise} \end{cases} \quad (1)$$

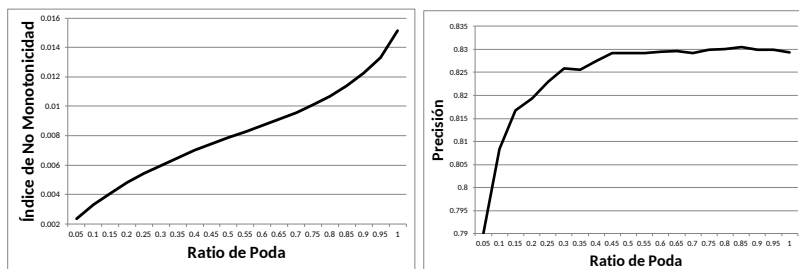
El factor  $R$  fue introducido, también por primera vez, por Ben-David [3] como el factor de importancia de la medida *order-ambiguity-score* en la decisión de la ramificación del árbol con el cálculo de *total-ambiguity-score*. A mayor  $R$ , las restricciones monotónicas se consideran más relevantes. En esta contribución, el parámetro  $R$  de cada árbol se utiliza para introducir, si cabe, más aleatoriedad y diversidad a las diferentes ramificaciones que conforman el RF y al mismo tiempo, se fuerza que el procedimiento de construcción de estos esté dominado por las consideraciones monotónicas. Para conseguir esto, cada árbol se ramifica desde el comiendo con un valor diferente del factor  $R$ , elegido aleatoriamente de 1 a  $R_{limite}$ , elegido como parámetro del Algoritmo mostrado en 1.

Además, no se ha considerado el límite de profundidad máximo impuesto en los árboles del RF estándar, ya que los árboles de decisión monotónico ya reducen drásticamente su complejidad respecto a árboles tradicionales.

Finalmente, se ha diseñado un mecanismo de poda por umbral para el último proceso de combinación de los diferentes resultados y predicción de la clase de cada ejemplo. En lugar de usar todos los árboles de decisión construidos para decidir la clase a través de mayoría de voto de las posibles predicciones, se han tomado los mejores árboles en términos de restricciones monotónicas, regulado por un cierto umbral. Así se muestra en las últimas líneas del Algoritmo 1. Con este objetivo, el RF Monotónico ordena, en orden creciente, los diferentes árboles según su valor de índice de no monotonicidad y el método de poda selecciona los primeros  $n$  árboles, donde  $n$  es el número de árboles calculado con el producto del número total de árboles construidos y el umbral  $T$  dentro del rango de valores  $(0,1]$ . Se recomienda el uso del valor 0,50 para este parámetro, dado los resultados obtenidos en la siguiente sección.

#### 4. Marco experimental, Resultados y Análisis

En esta sección, se presenta el marco experimental seguido para comparar y analizar la aplicación del RF a la clasificación monotónica.



(a) Media de NMI del *Random Forest* respecto a ratio de poda. (b) Media de precisión del *Random Forest* respecto a ratio de poda.

Figura 1: Efecto del ratio de poda en el *Random Forest*

##### 4.1. Metodología experimental

La metodología seguida en la experimentación se describe a continuación con estas directrices básicas:

- Conjuntos de datos: se han usado 50 conjuntos monotónico para este estudio. La mayoría de estos conjuntos son problemas estándares del ámbito de la clasificación, extraídos del repositorio KEEL [1], que han sido re-etiquetados siguiendo el procedimiento usado en el diseño experimental de [5]. Hay que tener en cuenta que este proceso de cambio aplicado a los conjuntos de datos

originales los transforma en conjuntos monotónicos, modificando, por tanto, la distribución de clases y su complejidad pero manteniendo el número de atributos y ejemplos. Además, se han incluido cuatro conjuntos de datos monotónicos clásicos usados en [2]: *ERA*, *ESL*, *LEV*, *SWD*.

- Algoritmos a comparar: la propuesta RF se ha comparado con tres clasificadores monotónicos basados en árboles de decisión: MID-C4.5 [3], MID-CART [12] y MID-RankTree [13]; y dos algoritmos clásicos dentro de este campo: OLM [2] y OSDL [10].
- Medidas de evaluación: Se han utilizado varias medidas con el objetivo de estimar el rendimiento del algoritmo: Precisión (Acc); Error absoluto medio (MAE) [8]; Índice de No Monotonidad (NMI); Número de hojas (NH).
- Configuración de parámetros: Los parámetros de los algoritmos de referencia son los estándares en el software KEEL [1]. La profundidad máxima para CART y RankTree es 90. El valor de  $R$  del algoritmo MID es 1. El RF también hace uso de parámetros estándares, pero para la clasificación monotónica, es necesario establecer el  $R_{limite} = 100$  y el Umbral= 0,5. El número de árboles construidos es 100 en todos los casos.

## 4.2. Resultados

Esta sección está dedicada a presentar los resultados recogidos de las ejecuciones de los algoritmos usando la configuración descrita en la sección anterior.

Primero, se debe presentar el estudio que ha permitido determinar el mejor valor para el parámetro de poda monotónica del RF propuesto. Los árboles construidos para el *ensemble* han sido ordenados en orden creciente del NMI. El mecanismo de poda selecciona los diferentes árboles usando un umbral de 0.05 a 1. Esto representa el ratio de árboles que se considerarán para la votación. De esta manera, si el ratio es 1, se tomarán en consideración todos los árboles construidos, mientras que si el ratio es 0.3, sólo el 30% de los árboles más monotónicos formarán el *ensemble*.

La Figura 1 muestra los efectos, en el RF, del ratio de poda explicado anteriormente. Los valores representados en ambas gráficas están asociados a la media de los valores de precisión y NMI de los 50 conjuntos de datos. Observando la Figura 1a, se puede apreciar la existencia de un punto de inflexión en el crecimiento de la curva, alrededor de un ratio con valor 0.5. Simultáneamente, en la Figura 1b, existe un límite en el que la mejora registrada en precisión para y comienza a decrecer. Este límite coincide exactamente en el mismo punto indicado anteriormente: con un valor de 0,5. Por ello, parece lógico que este valor sea interesante para ser adoptado como ratio de la poda monotónica usada en RFs.

Por tanto, se ha considerado 0,5 como ratio de poda usado en el RF. Bajo la configuración recomendada de la poda y la elección aleatoria del parámetro  $R$ , se ha comparado el RF con los otros algoritmos contendientes. El Cuadro 1 y Figura 2 exhiben los resultados obtenidos para los algoritmos sobre los conjuntos de datos monotónicos, en términos de medias de los resultados de 3 ejecuciones de una validación cruzada de 10 particiones.

Cuadro 1: Resultados de Precisión y MAE obtenidos.

	Precisión						Error Absoluto Medio					
	MID RF	MID C4.5	MID CART	MID RankTree	OLM	OSDL	MID RF	MID C4.5	MID CART	MID RankTree	OLM	OSDL
appendicitis	0,8667	0,8964	0,8864	<b>0,9064</b>	0,8109	0,6418	0,1333	0,1036	0,1136	<b>0,0936</b>	0,1891	0,3582
australian	0,8261	0,8029	0,8319	<b>0,8362</b>	0,7232	0,8319	0,1739	0,1971	0,1681	<b>0,1638</b>	0,2768	0,1681
auto-mpg	0,6529	0,6247	0,4489	0,6223	<b>0,6812</b>	0,3854	0,4630	0,4851	1,0335	0,5001	<b>0,3879</b>	0,8291
automobile	0,8039	<b>0,8250</b>	0,7304	0,7429	0,2333	0,3758	0,3069	<b>0,2688</b>	0,4696	0,4263	2,1046	0,8958
balance	0,9830	0,9777	0,9777	<b>0,9856</b>	0,9776	0,9777	0,0186	0,0239	0,0271	<b>0,0176</b>	0,0272	0,0271
bostonhousing	<b>0,6483</b>	0,5674	0,4982	0,5237	0,3003	0,2569	<b>0,4958</b>	0,6102	0,7504	0,6856	1,3045	1,0099
breast	0,7597	0,7337	0,6933	0,7440	<b>0,8409</b>	0,8015	0,2403	0,2663	0,3067	0,2560	<b>0,1591</b>	0,1985
bupa	0,7981	0,7508	0,7534	0,7879	<b>0,8375</b>	0,7625	0,2019	0,2492	0,2466	0,2121	<b>0,1625</b>	0,2375
car	0,8731	0,9433	0,8183	0,9386	<b>0,9705</b>	<b>0,9705</b>	0,1609	0,0666	0,2396	0,0735	<b>0,0324</b>	<b>0,0324</b>
cleveland	0,5644	0,4909	0,5284	0,5253	<b>0,5793</b>	0,5421	<b>0,6893</b>	0,8332	0,8014	0,8586	0,8311	0,7848
contraceptive	0,8185	0,7991	0,5601	0,7719	<b>0,8799</b>	0,8398	0,2351	0,2552	0,6449	0,2844	<b>0,1534</b>	0,1602
crx	<b>0,8290</b>	0,7903	0,7933	0,7839	0,6110	0,7058	<b>0,1710</b>	0,2097	0,2067	0,2161	0,3890	0,2942
dermatology	<b>0,8633</b>	0,8437	0,8408	0,7512	0,4499	0,1593	<b>0,2810</b>	0,3325	0,3465	0,5339	1,3821	1,6421
ecoli	<b>0,6441</b>	0,6074	0,5750	0,5779	0,6368	0,0652	1,0802	1,0549	1,5201	1,1250	<b>0,9467</b>	2,1998
ERA	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>
ESL	0,9043	0,9159	0,6162	0,9344	0,9179	<b>0,9364</b>	0,1107	0,1026	0,5788	0,0738	0,0923	<b>0,0656</b>
flare	0,9025	0,9165	0,6380	0,9456	<b>0,9738</b>	0,9606	0,1256	0,1191	0,6479	0,0826	<b>0,0318</b>	0,0572
glass	<b>0,7464</b>	0,6773	0,6258	0,6883	0,3175	0,3223	<b>0,4934</b>	0,6929	0,7747	0,6747	1,7994	1,8000
haberman	0,9291	0,9177	0,9312	0,9537	0,9310	<b>0,9606</b>	0,0709	0,0823	0,0688	0,0463	0,0690	<b>0,0394</b>
hayes-roth	0,9042	0,9438	0,7688	0,8500	<b>0,9500</b>	0,9438	0,1104	<b>0,0563</b>	0,2500	0,1688	0,0750	<b>0,0563</b>
heart	<b>0,8235</b>	0,7926	0,7593	0,8111	0,6704	0,6259	<b>0,1765</b>	0,2074	0,2407	0,1889	0,3296	0,3741
hepatitis	0,8917	0,7750	0,8375	<b>0,9000</b>	0,2375	0,8000	0,1083	0,2250	0,1625	<b>0,1000</b>	0,7625	0,2000
housevotes	0,9528	<b>0,9741</b>	0,8750	0,9266	0,9047	0,9096	0,0472	<b>0,0259</b>	0,1250	0,0734	0,0953	0,0904
ionosphere	<b>0,8832</b>	0,8348	0,7863	0,7810	0,6580	0,7237	<b>0,1168</b>	0,1652	0,2137	0,2190	0,3420	0,2763
iris	0,9711	0,9667	0,9733	<b>0,9867</b>	0,9000	0,3733	0,0289	0,0333	0,0267	<b>0,0133</b>	0,1000	0,9067
led7digit	0,8600	0,9520	0,7880	0,9660	<b>0,9820</b>	0,9740	0,3800	0,1140	0,6780	0,0700	<b>0,0340</b>	<b>0,0340</b>
LEV	0,9993	<b>1,0000</b>	0,6990	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	0,0007	<b>0,0000</b>	0,4450	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>
lymphography	<b>0,7819</b>	0,7705	0,6767	0,6900	0,7100	0,7029	<b>0,2314</b>	0,2567	0,3633	0,3714	0,3571	0,2971
machinacepu	<b>0,6520</b>	0,5638	0,4398	0,6369	0,6267	0,6362	0,4741	0,6086	0,7564	0,4726	0,5031	<b>0,4067</b>
mammographic	0,9763	0,9831	0,9735	<b>0,9904</b>	0,9892	0,9831	0,0237	0,0169	0,0265	<b>0,0096</b>	0,0108	0,0169
monk-2	0,9807	0,9746	0,9792	0,9769	0,9721	<b>0,9908</b>	0,0193	0,0254	0,0208	0,0232	0,0279	<b>0,0092</b>
movement_libras	<b>0,6796</b>	0,5194	0,5583	0,5333	0,3139	0,1194	<b>1,1602</b>	2,0306	1,7861	1,8528	4,3472	5,7000
newthyroid	<b>0,8621</b>	0,8279	0,8511	0,8329	0,6223	0,1818	<b>0,1905</b>	0,2186	0,1909	0,2139	0,5504	0,8413
pima	<b>0,8702</b>	0,8242	0,7837	0,8007	0,8151	0,6224	<b>0,1298</b>	0,1758	0,2163	0,1993	0,1849	0,3776
post-operative	0,6968	0,6333	0,4403	0,7403	<b>0,8292</b>	0,7569	0,3773	0,4806	0,7292	0,3153	<b>0,2042</b>	0,2431
saheart	<b>0,7302</b>	0,6627	0,6645	0,6624	0,6862	0,6839	<b>0,2698</b>	0,3373	0,3355	0,3376	0,3138	0,3161
segment	<b>0,9759</b>	0,9649	0,9632	0,9602	0,3061	0,1684	<b>0,0447</b>	0,0610	0,0671	0,0723	2,4022	2,8597
sonar	<b>0,8042</b>	0,7681	0,7250	0,7648	0,4662	0,5724	<b>0,1958</b>	0,2319	0,2750	0,2352	0,5338	0,4276
spectfheart	<b>0,8028</b>	0,7379	0,7412	0,7339	0,2095	0,8016	<b>0,1972</b>	0,2621	0,2588	0,2661	0,7905	0,1984
SWD	0,9993	<b>1,0000</b>	0,3820	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	0,0007	<b>0,0000</b>	1,0240	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>
tae	0,8278	0,8483	0,5904	0,8417	0,8546	<b>0,8946</b>	0,1921	0,1650	0,5283	0,1913	0,1850	<b>0,1054</b>
titanic	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>1,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>	<b>0,0000</b>
vehicle	<b>0,7409</b>	0,6904	0,6384	0,6644	0,2530	0,2588	<b>0,4929</b>	0,5661	0,6475	0,6334	1,4917	0,9717
vowel	<b>0,9525</b>	0,7758	0,2182	0,7778	0,0909	0,0859	<b>0,1010</b>	0,6242	2,9313	0,5232	5,0000	4,8273
wdbc	<b>0,7183</b>	0,6768	0,6749	0,6713	0,6451	0,5607	<b>0,2817</b>	0,3232	0,3251	0,3287	0,3549	0,4393
windsorhousing	0,8932	0,8939	0,8738	0,8664	<b>0,9174</b>	0,8847	0,1068	0,1061	0,1262	0,1336	<b>0,0826</b>	0,1153
wine	<b>0,7926</b>	0,6794	0,7297	0,7578	0,3484	0,3314	<b>0,2882</b>	0,4219	0,3719	0,3154	0,9660	0,9667
wisconsin	<b>0,9747</b>	0,9591	0,9693	0,9591	0,8815	0,9547	<b>0,0253</b>	0,0409	0,0307	0,0409	0,1185	0,0453
yeast	0,4095	0,3659	0,2811	0,3639	<b>0,4596</b>	0,0836	1,7143	1,8605	3,1107	1,8111	<b>1,6300</b>	3,0862
zoo	0,7427	0,8127	0,4564	0,8127	<b>0,8409</b>	0,7727	0,7000	0,4727	1,8073	0,4336	<b>0,3682</b>	0,3745

Con el propósito de respaldar los resultados, se ha incluido un análisis estadístico basados en test no paramétricos. Los resultados obtenidos de la aplicación del test de Friedman y el procedimiento *post-hoc* de Holm está representados en el Cuadro 2. Así como, el P-Valor ajustado (PVA) [7] y calculado por el procedimiento Holm está indicado en el algoritmo cuyo rango no es el mejor de cada grupo.

Cuadro 2: Tabla resumen de las salidas estadística inferidas: Rangos and PVAs

	Precisión		MAE		NH		NMI	
	Rangos PVAs	Rangos PVAs	Rangos PVAs	Rangos PVAs	Rangos PVAs	Rangos PVAs	Rangos PVAs	Rangos PVAs
MID-RF	<b>2,480</b>	—	<b>2,540</b>	—	<b>1,469</b>	—	2,300	0,423
MID-C4,5	3,360	0,037	3,310	0,079	2,041	0,028	<b>2,000</b>	—
MID-CART	4,310	0,000	4,320	0,000	2,345	0,000	4,130	0,000
MID-RankTree	3,200	0,054	3,230	0,079	3,255	0,000	5,600	0,000
OLM	3,650	0,005	3,800	0,003	-	—	3,780	0,000
OSDL	4,000	0,000	3,800	0,003	-	—	3,190	0,001

### 4.3. Análisis

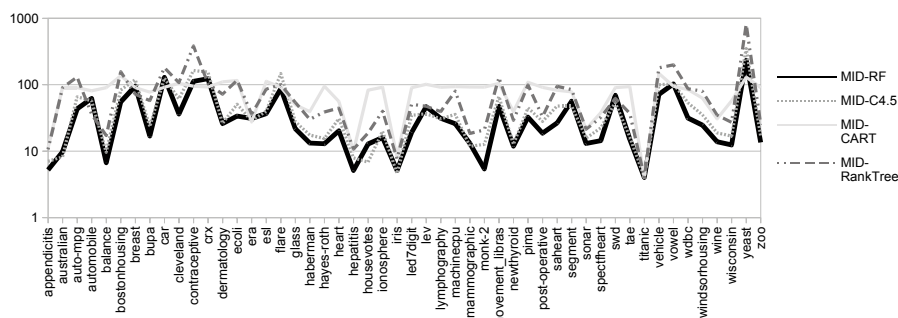
De este estudio, se pueden extraer las siguientes conclusiones:

- En términos de precisión, la calidad del RF Monotónico con el mecanismo de poda por umbral es claro. En todos los casos, el RF sobrepasa a los otros 5 algoritmos con una diferencia significativa, hecho que puede observarse en los cuadros 1 y 2, donde el  $P$ -valor es menor a 0,10.
- Con resultados parecidos, la superioridad del RF en relación al MAE sobre los otros algoritmos es abrumadora. Estos resultados ya eran esperado, por la gran diferencia existente en términos de precisión.
- Además, el RF ha tenido éxito, también, a la hora de obtener arboles menos complejos, como se puede ver con menor valor en media de hojas, en la Figura 2. Un hecho significativo teniendo en mente que el límite de profundidad máxima típico del RF estándar no se usa en este método. Esto ocurre gracias a la variabilidad aportada por el proceso de poda, que permite elegir los árboles más simples y monotónicos.
- Finalmente, a lo que en NMI se refiere, la Figura 2 y el Cuadro 2 reflejan mejores resultados para el MID-C4.5. Sin embargo, la diferencia entre MID-RF y MID-C4.5 no parece significativa.

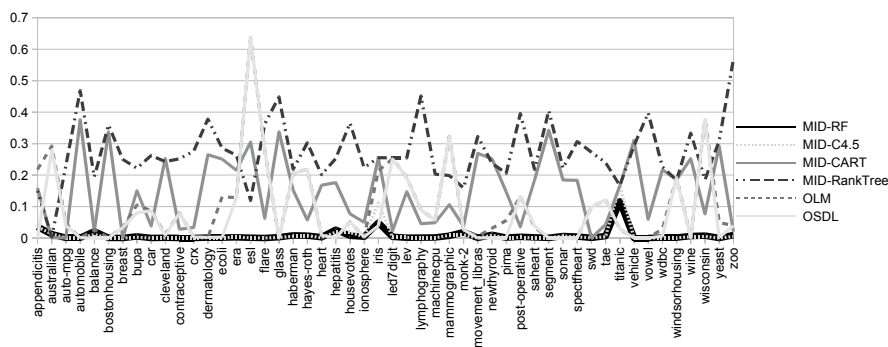
Para concluir, esta propuesta mejora el rendimiento de todos los algoritmos comparados en casi todas las medidas consideradas, exceptuando de NMI cuando se compara con MID-C4.5.

## 5. Concluding Remarks

El propósito de esta contribución es presentar y analizar una propuesta basada en *Random Forest* para la clasificación con restricciones monotónicas. Para adaptarlo a este problema, se ha incluido el ratio de monotonicidad como un parámetro más para ofrecer aleatoriedad y diversidad a la hora de hacer crecer los diferentes árboles. Tras ramificar cada uno de los árboles de decisión, se ha hecho uso de un mecanismo de poda del *ensemble* basado en el índice de monotonicidad de cada árbol, para seleccionar un subconjunto de los árboles de decisión más monotónicos que contribuirán en la votación. Los resultados



(a) Media de NL.



(b) Media de NMI.

Figura 2: Resultados obtenidos del número de hojas y Índice de no monotonicidad.

muestran que Random Forest es un método prometedor para atacar este tipo de problemas, obteniendo resultados muy precisos e involucrando, a su vez, árboles con un índice de no monotonicidad bajo.

**Agradecimientos** Este trabajo se ha sustentado por el Proyecto de Investigación TIN2014-57251-P. S. González disfruta de una beca de iniciación a la investigación de la Universidad de Granada asociada a esta contribución.



## Referencias

1. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera, F.: KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing* 17(2-3), 255–287 (2011)
2. Ben-David, A.: Automatic generation of symbolic multiattribute ordinal knowledge-based dsss: methodology and applications. *Decision Sciences* 23, 1357–1372 (1992)
3. Ben-David, A.: Monotonicity maintenance in information-theoretic machine learning algorithms. *Machine Learning* 19(1), 29–43 (1995)
4. Ben-David, A., Sterling, L., Pao, Y.H.: Learning, classification of monotonic ordinal concepts. *Computational Intelligence* 5, 45–49 (1989)
5. Duivesteijn, W., Feelders, A.: Nearest neighbour classification with monotonicity constraints. In: *ECML/PKDD* (1). pp. 301–316 (2008)
6. Furnkranz, J., Gamberger, D., Lavrac, N.: *Foundations of Rule Learning*. Springer (2012)
7. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences* 180(10), 2044–2064 (2010)
8. Japkowicz, N., Shah, M. (eds.): *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press (2011)
9. Kotłowski, W., Słowiński, R.: On nonparametric ordinal classification with monotonicity constraints. *IEEE Transactions on Knowledge Data Engineering* 25(11), 2576–2589 (2013)
10. Lievens, S., Baets, B.D., Cao-Van, K.: A probabilistic framework for the design of instance-based supervised ranking algorithms in an ordinal setting. *Annals of Operational Research* 163(1), 115–142 (2008)
11. Martínez-Muñoz, G., Hernández-Lobato, D., Suárez, A.: An analysis of ensemble pruning techniques based on ordered aggregation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31(2), 245–259 (2009)
12. Rokach, L., Maimon, O.: *Data Mining with Decision Trees. Theory and Applications*. 2nd Edition. World Scientific (2014)
13. Xia, F., Zhang, W., Li, F., Yang, Y.: Ranking with decision tree. *Knowledge and Information Systems* 17(3), 381–395 (2008)