Instituto de Ingeniería Eléctrica (IIE)
Facultad de Ingeniería - Universidad de la República

Clasificación de Bosques por Información Cartográfica

Autor: Santiago Castro y Matías Valdés
Tutor: Alicia Fernández y Guillermo Carbajal

Trabajo final del curso "Reconocimiento de Patrones"

7 de diciembre de 2015

Índice

1. Introducción

En este trabajo se propone clasificar el tipo mayoritario de árboles en bosques nativos. Se cuenta con datos provenientes de cuatro áreas de bosques naturales, situados en el norte del estado de Colorado, EEUU, como se muestra en la Figura 1. Cada observación corresponde a un área de 30x30 metros. Se debe predecir el tipo de árbol de cobertura mayoritario entre todos los que están presentes. Los 7 tipos de árboles son: Spruce/Fir, Lodgepole Pine, Ponderosa Pine, Cottonwood/Willow, Aspen, Douglasfir, Krummholz. Los datos fueron extraídos de la base pública Cover Type [3] de la Universidad de California en Irvine (UCI).

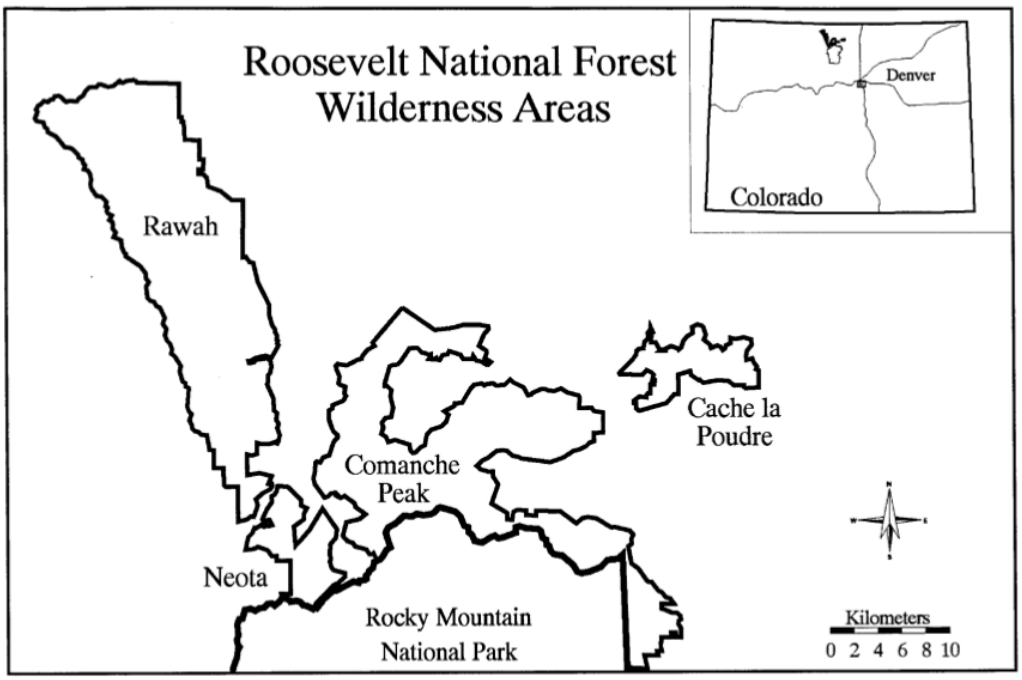


Figura 1: Mapa del área donde se obtuvieron los datos [4].

Para esto se propone diseñar un sistema de reconocimiento de patrones que permita clasificar el tipo de árboles, maximizando el porcentaje de acierto. Para validar el desempeño final del clasificador implementado, se realiza una posterior evaluación con un conjunto de test, el cual tiene una cantidad igual de elementos por clase. Los objetivos específicos son:

- obtener un porcentaje de acierto de al menos el 82 %, evaluado con el conjunto de datos de entrenamiento.
- Explorar el impacto que tiene aplicar alguna técnica de edición sobre los datos.
- En particular, intentar mejorar la información del tipo de suelo, agregando o sustituyendo las características existentes por otras, para explorar su impacto en el desempeño del clasificador.
- Estimar el desempeño esperado para el conjunto de test.

El conjunto de datos de entrenamiento, obtenido desde el sitio EVA, consiste en un subconjunto de 7612 observaciones de la base de datos Cover Type mencionada anteriormente. En la Tabla 1 se muestra la cantidad de patrones por clase para los datos disponibles.

Clase	Patrones
1	1106
2	910
3	1141
4	1130
5	1087
6	1153
7	1085

Cuadro 1: Cantidad de patrones por clase en datos disponibles.

Se dispone de 54 características para la clasificación, las cuales se describen a continuación:

- Elevation - Elevación en metros.
- Aspect - azimut del terreno en grados.
- Slope - Pendiente del terreno en grados.
- Horizontal_Distance_To_Hydrology - Distancia horizontal a la fuente de agua superficial más cercana.
- Vertical_Distance_To_Hydrology - Distancia vertical a la fuente de agua superficial más cercana.
- Horizontal_Distance_To_Roadways - Distancia horizontal a la ruta más cercana.
- Hillshade_9am (valor de 0 a 255) - Índice de sombra a las 9am, en el solsticio de verano.
- Hillshade_Noon (valor de 0 to 255) - Índice de sombra al mediodía, en el solsticio de verano.
- Hillshade_3pm (valor de 0 to 255) - Índice de sombra a las 3 pm, en el solsticio de verano.
- Horizontal_Distance_To_Fire_Points - Distancia horizontal a puntos donde hubo comienzo de incendios.
- Wilderness_Area (4 valores binarios, uno por área) - Nombre del área en la que está situado.
- Soil_Type - Tipo de suelo (hay 40 disponibles, una característica con valor binario por tipo de suelo)
- Cover_Type - Tipo de bosque, que es la clase a estimar. Hay 7 posibles tipos.

Como se observa en la Tabla 1 anterior, las clases de los datos de entrenamiento no están balanceadas. Esto puede llevar a que los clasificadores opten por clasificar más frecuentemente a las clases de mayor presencia, como forma de aumentar su desempeño. Para tener en cuenta este desbalanceo, se suele realizar submuestreo o sobremuestreo, como forma de obtener un conjunto de datos balanceado. En este trabajo, dado que el desbalanceo de las clases no es considerable, en principio no se realiza un balanceo.

2. Primeros resultados

En una primera instancia se utilizaron todas las características y los datos sin procesar. En particular no se normalizan los datos. Se estudió el desempeño de los clasificadores más importantes vistos hasta el momento en el curso. Para estimar el desempeño de cada clasificador, se utiliza validación cruzada con 10 particiones. Este proceso fracciona el conjunto de los datos disponibles en un 90 % (6804) para entrenamiento y un 10 % (756) para prueba. Esto lo repite 10 veces cambiando los subconjuntos de prueba y entrenamiento y luego realiza un promedio del desempeño obtenido. En la Figura 2 se ilustra el procedimiento para el caso de $k = 4$ repeticiones.

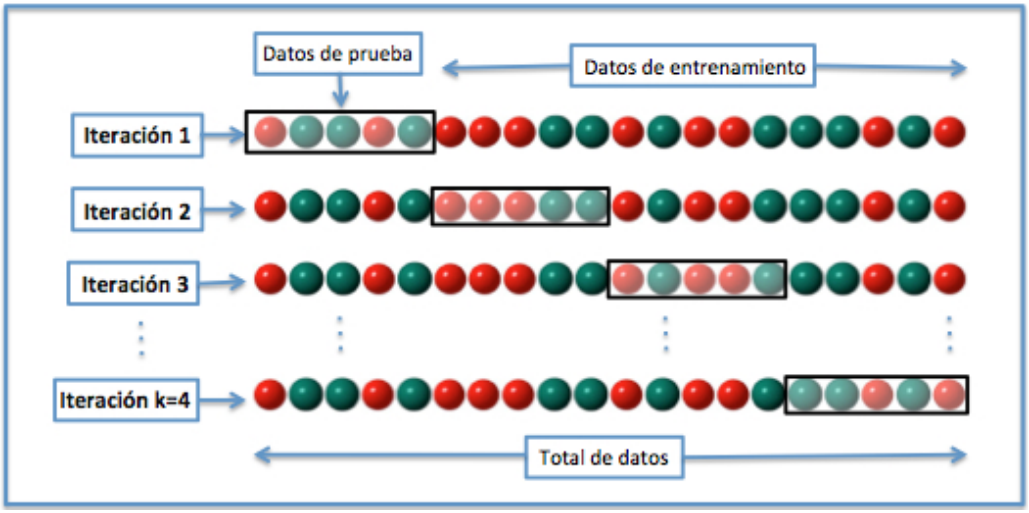


Figura 2: Validación cruzada para $k = 4$ repeticiones (Wikipedia en español - Joan.domenech91)

2.1. Parámetros utilizados

A continuación de describen los parámetros utilizados para cada clasificador y la forma en que se seleccionaron.

2.1.1. Árbol con C4.5 (J48 de Weka)

El factor de confianza se seleccionó de forma óptima mediante CVParamSelection, variándolo de 0.05 a 0.60, con pasos de 0.05 (C 0.05 0.6 12). Se obtuvo un valor óptimo de $\alpha = 0{,}15$. La cantidad mínima de patrones por nodo se tomó igual al valor por defecto 2. El árbol obtenido tiene un tamaño de 1047 nodos, siendo 524 hojas.

2.1.2. Naive Bayes

Se realizaron pruebas con y sin estimación del núcleo (useKernelEstimation = True/False respectivamente).

2.1.3. k vecinos (IBk)

Mediante CVParamSelection se obtuvo un valor óptimo de $k = 1$ vecino.

2.1.4. Perceptron Multicapa (MLP)

Weka permite implementar un Perceptron Multicapa (weka.classifiers.functions.MultilayerPerceptron) con 1 capa oculta de H nodos. Para utilizar este clasificador se normalizaron los datos en $[0, 1]$. Mediante CVParamSelection, se variaron la cantidad de nodos en la capa oculta de 2 a 137, con pasos de 15 y luego en una grilla más fina de 108 a 136 con pasos de 4. Se obtuvo un valor óptimo $H = 132$. Se utlizó este valor y los demás parámetros por defecto: tasa de aprendizaje = 0.3 y momentum = 0.2. Una posible forma de mejorar el desmpeño es optimizar la tasa de aprendizaje (LR) y el Momentum, además de los nodos en la capa oculta, tal como se hace en [4].

2.1.5. Máquinas de Sporte Vectorial (SVM)

Utiilzamos el algoritmo SMO para implementar el clasificador de tipo SVM. Para este caso en que se tienen más de dos clases, el algoritmo utiliza clasificación por pares. Es decir que para cada par de clases se diseña un clasificador por SVM. Para clasificar un nuevo patrón, este se pasa por todos los clasificadores, y se asigna a la clase más votada.

Se normalizaron los datos según lo recomienda [5] y se utilizó un núcleo RBF de parámetro γ. El coeficiente de costo C y el parámetro γ óptimos se seleccionaron mediante GridSearch, variando los mismos de 2^0 a 2^7 y de 2^{-3} a 2^3 respectivamente. Los valores óptimos obtenidos fueron $C = 64$ y $\gamma = 1$.

2.2. Resultados obtenidos

A continuación se informan los resultados obtenidos para los clasificadores con los parámetros mencionados anteriormente. En particular se analizan los porcentajes de acierto global y por clase y los tipos de confusión entre las clases.

2.2.1. Porcentajes de acierto

En la Tabla 2 se resume el desempeño global de los clasificadores utilizados. Los clasificadores con mayor porcentaje de acierto son SVM, 1-NN y C4.5, seguidos de MLP y NaiveBayes, siendo este último el de peor desempeño.

Clasificador	Porcentaje de acierto (%)	Parámetros
SVM	80.7	óptimos
1-NN	78.8	óptimos
C4.5	77.5	óptimos
MLP	76.9	óptimos
NaiveBayes	69.3	con kernel estimation
NaiveBayes	66.0	sin kernel estimation

Cuadro 2: Porcentaje de acierto de los distintos clasificadores.

Ninguno de los clasificadores permite cumplir con el objetivo de obtener un porcentaje de acierto de al menos 82 %. La Tabla 3 muestra el porcentaje de acierto por clase de cada clasificador.

	Porcentaje de acierto (%)				
Clase	SVM	1-NN	C4.5	MLP	NaiveBayes (Sin KE)
Todas	80.7	78.8	77.5	76.9	66.0
1	71.3	67.8	67.6	67.5	64.7
2	57.4	57.5	50.2	51.4	41.6
3	73.8	68.3	71.3	69.4	46.9
4	93.6	91.8	94.0	92.1	90.4
5	91.4	91.7	87.9	88.3	72.2
6	78.9	76.8	75.4	72.3	59.5
7	95.0	94.9	91.9	93.5	82.9

Cuadro 3: Porcentaje de patrones correctamente clasificados por clase.

Para todos los clasificadores:

- las clases con mayor error de clasificación son la 1, 2 y 3.
- las de menor error de clasificación son la 4, 5 y 7.

2.2.2. Confusión de las clases

A partir de la matriz de confusión de cada clasificador, se construyen las Tablas 4 y 5. Estas muestran, para cada una de las clases con mayor error de clasificación, las dos primeras clases (1a y 2a) con las cuales estas se confunden más frecuentemente y el porcentaje de confusión.

Clase	Árbol C4.5		Naive Bayes		1NN	
	1a	2a	1a	2a	1a	2a
1	2 (19.3)	7 (9.31)	2 (15.3)	7 (13.0)	2 (18.0)	7 (8.40)
2	1 (27.5)	5 (12.9)	1 (29.7)	5 (19.0)	1 (23.2)	5 (11.0)
3	6 (24.0)	4 (19.5)	6 (24.0)	4 (19.6)	6 (19.1)	4 (8.76)
6	3 (16.3)	4 (3.64)	3 (20.3)	4 (10.6)	3 (14.8)	4 (4.51)

Cuadro 4: Confusiones más importantes de cada clase en los clasificadores.

Clase	MLP		SVM	
	1a	2a	1a	2a
1	2 (16.2)	7 (10.6)	2 (17.2)	7 (8.05)
2	1 (24.7)	5 (13.8)	1 (23.2)	5 (11.4)
3	6 (20.5)	4 (7.89)	6 (16.8)	4 (7.19)
6	3 (20.5)	4 (4.34)	3 (15.8)	4 (3.38)

Cuadro 5: Confusiones más importantes de cada clase en los clasificadores.

Para todos los clasificadores, se observa que: las clases 1 y 2 se confunden mutuamente y lo mismo ocurre entre las clases 3 y 6. Esto puede estar indicando que las mencionadas clases tienen distirbuciones que se solapan. Para las confusiones anteriores, la Tabla 6 muestra los clasificadores con los cuales se obtiene menor confusión y el respectivo porcentaje de confusión.

Clase	1a	2a
1	2 (15.3 NB)	7 (8.05 SVM)
2	1 (23.2 1NN y SVM)	5 (11.0 1NN)
3	6 (16.8 SVM)	4 (7.19 SVM)
6	3 (14.8 1NN)	4 (3.38 SVM)

Cuadro 6: Clasificadores con menor confusión para las confusiones más importantes de cada clase.

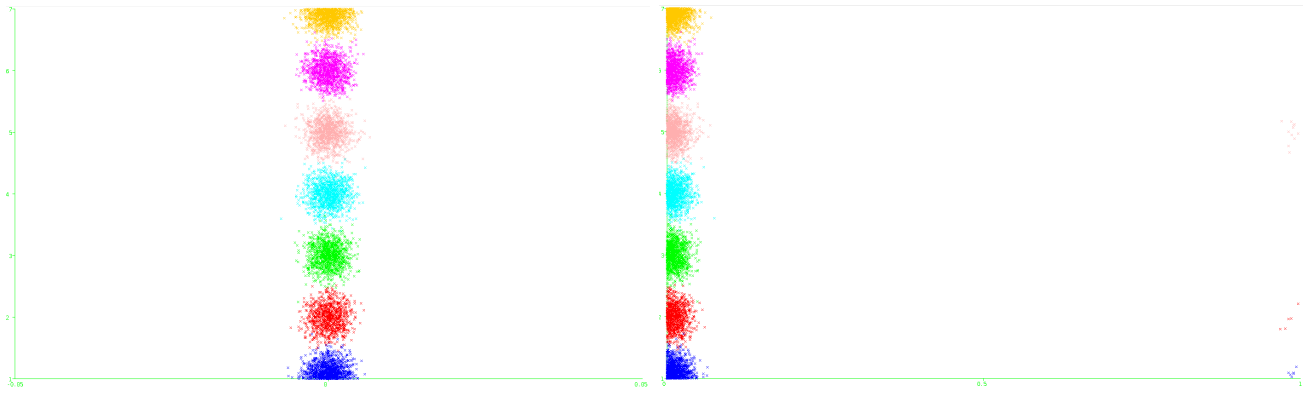
2.3. Datos normalizados

Los resultados anteriores se obtuvieron con datos sin normalizar, excepto para SVM y MLP, para los cuales Weka los normaliza por defecto. En una segunda instancia se normalizaron los datos en el rango $[0, 1]$ mediante: $\frac{x-x_{min}}{x_{max}-x_{min}}$ (weka.filters.unsupervised.attribute.Normalize) y se evaluó el desempeño de J48, 1NN y Bayes. Se obtuvieron resultados muy similares al caso sin normalizar.

3. Selección y combinación de características

3.0.1. Eliminación de tipos de suelo

Inicialmente, mediante el gráfico clase vs. característica, se identificaron aquellos tipos de suelo que no aportan información significativa porque presentan los mismos valores para todas las clases. Estos fueron los tipos: 7 a 9, 15, 18, 19, 21, 25, 27, 28, 34 y 36. A modo de ejemplo, en la Figura 3 se muestra el gráfico para los tipos de suelo 15 y 19.



(a) Tipo de suelo 15. (b) Tipo de suelo 19.

Figura 3: Gráfico clase (eje vertical) vs. característica (eje horizontal) para tipos de suelo 15 y 19.

Esto permitió eliminar 12 características que a priori no resultan relevantes. Para explorar el impacto en el desempeño de los clasificadores, se entrenaron los 3 mejores clasificadores SVM, 1-NN y C4.5 con los nuevos datos de 42 características. Tal como se muestra en la Tabla 7, eliminando estas 12 características, se obtuvieron porcentajes de acierto global muy similares a los obtenidos con los datos originales.

	Acierto (%)			
	C4.5	1-NN	SVM	# características
Original	77.5	78.8	80.7	54
Eliminando (E)	77.2	78.8	80.6	42

Cuadro 7: Porcentaje de acierto al eliminar características.

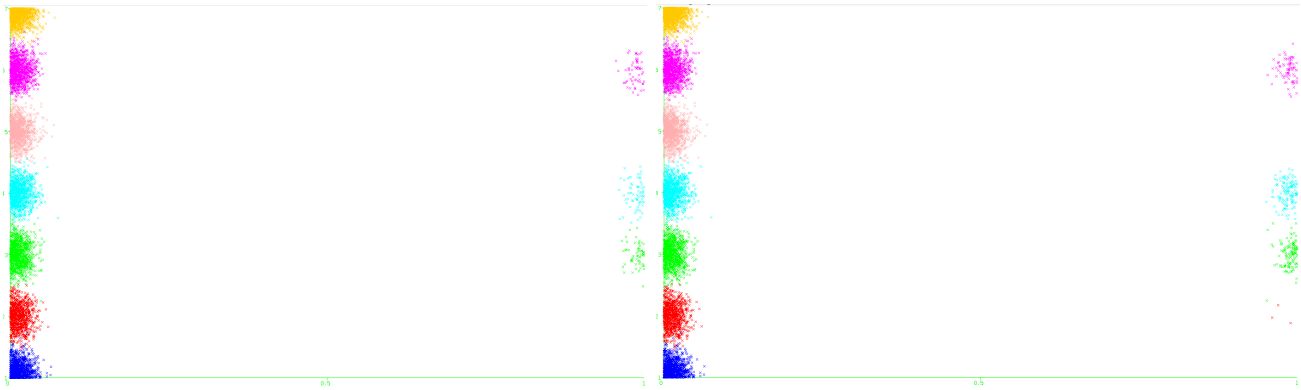
3.0.2. Combinación de tipos de suelo

En una segunda instancia se combinaron algunos tipos de suelo que presentaban un gráfico clase vs. característica similar. Esto se hizo mediante el filtro AddExpression (unsupervised.attribute.AddExpression). Los conjuntos considerados para su combinación fueron:

- 1 - Solo presencia de clases 7 (amarillo) y 1 (azul)
  - tipos de suelo 38, 39 y 40
- 2 - Sin presencia de clases 3 (verde) y 4 (celeste) y 6 (violeta)
  - tipos de suelo 23, 29, 30 y 31
- 3 - Solo presencia de clases 6 (violeta), 4 (celeste) y 3 (verde)
  - tipos de suelo 1, 5 y 6
- 4 - Solo presencia de clase 7 (amarillo)
  - tipos de suelo 35 y 37

- 5 - Sin presencia de clases 3 (verde) y 4 (celeste)
    - tipos de suelo 32 y 33

A modo de ejemplo, en la Figura 4 se muestra el gráfico para los tipos de suelo 1 y 6, pertenecientes al conjunto 3 descripto anteriormente.



(a) Tipo de suelo 1.

(b) Tipo de suelo 6.

Figura 4: Gráfico clase (eje vertical) vs. característica (eje horizontal) para tipos de suelo 1 y 6.

En la Tabla 8, se muestran los porcentajes de acierto para los clasificadores C4.5 y 1-NN, con los datos originales, los datos sin las 12 características que no aportaban información (Eliminando) y al ir agrupando los conjuntos de características descriptos anteriormente.

	Acierto (%)		
	C4.5	1-NN	# características
Original	77.5	78.8	54
Eliminando (E)	77.2	78.8	42
E+1	77.4	78.7	40
E+1+2	77.4	78.5	37
E+1+2+3	77.6	78.5	35
E+1+2+3+4	77.5	78.5	34
E+1+2+3+4+5	76.9	78.5	33

Cuadro 8: Porcentaje de acierto al ir combinando los conjuntos de características.

Para 1-NN, el porcentaje de acierto se mantiene estable al ir combinando los conjuntos de características. Por otro lado, para el árbol, el porcentaje de acierto se mantiene similar al obtenido con los datos originales; excepto cuando se combina el conjunto 5, donde el acierto baja notoriamente. Debido a esto, optamos por no combinar el conjunto 5. Es decir que se combinaron únicamente los conjutos 1 a 4, logrando de esta forma reducir en 8 la cantidad de características.

3.1. Relación entre características

Se analizaron los gráficos de pares de características en busca de correlación entre las mismas. Se observa que las características de índice de sombra (a las 9am, mediodía y 3pm) están

notoriamente correlacionadas con la pendiente del terreno. A modo de ejemplo, en la Figura 5 se muestra el gráfico índice de sombra a las 9 am según la pendiente del terreno.

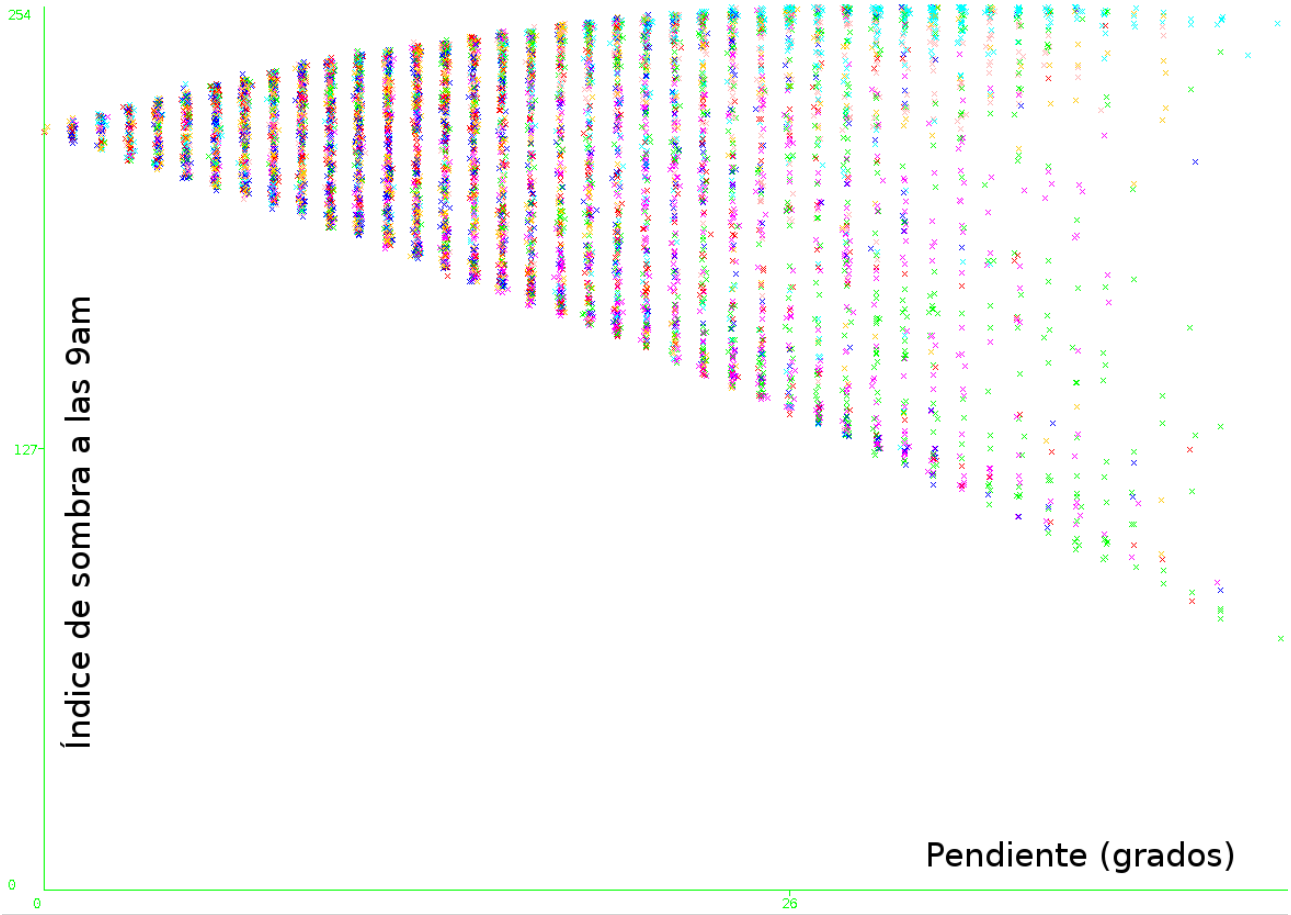


Figura 5: Índice de sombra a las 9 am en función de la pendiente del terreno.

Teniendo en cuenta lo anterior, se eliminaron las 3 características de índice se sombra, manteniendo la pendiente. En la Tabla 9 se muestra el impacto que tuvo esto en el desempeño de los clasificadores SVM, 1-NN y C4.5.

	Acierto (%)			
	C4.5	1-NN	SVM	# características
Original	77.5	78.8	80.7	54
E+1+2+3+4	77.5	78.5	80.6	34
(E+1+2+3+4)+Eind	77.4	80.2	79.6	31

Cuadro 9: Porcentaje de acierto al eliminar los 3 índices de sombra.

El porcentaje de acierto para el árbol C4.5 se mantiene prácticamente igual, mientras que en SVM disminuye un 1 % y en 1-NN aumenta en 1.7 %.

3.2. Selección mediante Ganancia de Información

Utilizano InfoGain (weka.attributeSelection.InfoGainAttributeEval), se realiza un ranking de las características según la Ganancia de Información de la característica x con respecto a la

clase w (información mutua): $I(w,x) = H(w) - H(w|x)$, siendo $H(\cdot)$ la entropía.

Para cada uno de los tres mejores clasificadores, se evaluó su desempeño seleccionando las primeras 20, 25 y 30 características. Los resultados se muestran en la Tabla 10.

		Acierto (%)		
	# características	C4.5	1-NN	SVM
Original	54	77.5	78.8	80.7
(E+1+2+3+4)+Eind	31	77.4	80.2	79.6
Info Gain	30	77.2	80.2	79.5
Info Gain	25	77.6	80.1	79.5
Info Gain	20	77.1	79.9	78.8

Cuadro 10: Porcentaje de acierto al seleccionar con Ganancia de Información.

Manteniendo 20 o más de las características seleccionadas mediante Ganancia de Información, en los tres clasificadores el porcentaje de acierto se mantiene similar al obtenido con las 31 características de la sección anterior. El mayor descenso en porcentaje de acierto se da cuando se utilizan 20 características.

3.3. Conclusiones

Con las técnicas descriptas anteriormente, se eliminaron y combinaron características; reduciendo el número de éstas de 54 al orden de 25. Al realizar esto, el porcentaje de acierto del árbol C4.5 se mantuvo similar al original, el de 1-NN aumentó en 1.3 % y el de SVM se redujo en 1.7 %. De esta forma se logra reducir el costo computacional, sin afectar considerablemente el desemepeño. De todas formas, dado que la variación en el desempeño no es uniforme en los clasificadores, y el objetivo de este trabajo es maximizar el porcentaje de acierto, en lo que sigue se trabaja con el conjunto de características original.

4. Voto por mayoría

Con el objetivo de lograr un clasificador de mayor desempeño, se implementó un clasificador mediante voto por mayoría (weka.classifiers.meta.Vote), a partir de los tres clasificadores de mejor desmpeño hasta el momento. Es decir: SVM con $C = 64$ y núcleo RBF de $\gamma = 1$, 1NN y un árbol con $\alpha = 0{,}15$. Se obtuvo un procentaje de acierto global de 82.6 %, el cual es superior al mayor porcentaje de acierto obtenido con los clasificadores anteriores (80.7 %). En la Tabla 11 se muestran los porcentajes de acierto por clase.

Clase	TP rate (%)
Todas	82.6
1	72.8
2	59.9
3	74.5
4	95.6
5	94.0
6	82.1
7	95.4

Cuadro 11: Porcentaje de patrones correctamente clasificados por clase.

Al igual que en los anteriores clasificadores, las clases con mayor porcentaje de acierto son la 4, 7 y 5, mientras que las de menor porcentaje de acierto son la 2, 1 y 3. En la Tabla 12 se muestra la matriz de confusión obtenida.

Clase	a	b	c	d	e	f	g
a	805	175	0	0	38	6	82
b	191	545	30	0	97	34	13
c	0	7	850	78	21	185	0
d	0	0	24	1080	0	26	0
e	7	25	17	0	1022	16	0
f	1	11	149	34	11	947	0
g	47	3	0	0	0	0	1035

Cuadro 12: Matriz de confusión.

Al igual que antes, la mayor confusión se da entre las clases 1 y 2 por un lado y las clases 3 y 6 por el otro. La Tabla 13 muestra, para cada una de las clases con mayor error de clasificación, las dos primeras clases (1a y 2a) con las cuales estas se confunden más frecuentemente y el respectivo porcentaje de confusión.

Clase	1a	2a
1	2 (15.8)	7 (7.41)
2	1 (21.0)	5 (10.7)
3	6 (16.2)	4 (6.84)
6	3 (12.9)	4 (2.95)

Cuadro 13: Confusiones más importantes.

En comparación con la Tabla 6, se obtienen menores porcentaje de confusión en todos los casos, excepto en la confusión de la clase 1 con la 2, donde el porcentaje de confusión es apenas 0.5 % superior al obtenido anteriormente.

5. Mejora de los clasificadores

Con el voto por mayoría se logra cumplir el objetivo de obtener un porcentaje de clasificación de al menos 82 % en los datos de entrenamiento. Dado que otro de los objetivos es maximizar

este porcentaje de acierto, se buscó mejorar alguno de los tres clasificadores de mejor desempeño, para de esa forma aumentar el acierto del voto por mayoría. En particular se opta por optimizar SVM mediante Bagging y Boosting y el árbol C4.5 mediante Random Forest.

5.1. Random Forest

El clasificador Random Forest consiste en un conjunto de árboles, cada uno generado de forma independiente a partir de un subconjunto aleatorio del total de características. Si se tiene un total de n patrones de entrenamiento, para entrenar cada árbol, se utilizan n patrones, seleccionados de forma aleatoria mediante muestreo con reemplazamiento. Es decir que algunos patrones pueden estar duplicados y otros no estar presentes. Una vez que los árboles están entrenados, la clasificación de un nuevo patrón consiste en clasificar el patrón con cada árbol y luego aplicar un voto por mayoría [6].

Mediante GridSearch, se seleccionaron la cantidad óptima de árboles y de características, variando estos parámetros de 50 a 290, con pasos de 30 y de 5 a 50 con pasos de 5 respectivamente. Los valores óptimos obtenidos fueron 230 y 15. Luego se realizó una búsqueda en una grilla más fina, variando los paràmetros de 210 a 250 con pasos de 10 y de 11 a 19 con pasos de 1. En este caso los valores óptimos obtenidos fueron 240 y 15. Para estos valores se obtiene un porcentaje de acierto global de 84.5 % y los porcentajes de acierto por clase que se muestran en la Tabla 14.

Clase	TP rate (%)
Todas	84.5
1	76.1
2	59.0
3	78.7
4	97.6
5	94.9
6	85.0
7	96.0

Cuadro 14: Porcentaje de patrones correctamente clasificados por cada clase.

Al igual que antes, las clases de menor porcentaje de acierto son la 1, 2 y 3; mientras que las de mayor porcentaje de acierto son la 4, 5 y 7. En la Tabla 15 se muestra la matriz de confusión obtenida.

Clase	a	b	c	d	e	f	g
a	842	141	2	0	32	6	83
b	192	537	27	0	108	36	10
c	0	6	898	69	17	151	0
d	0	0	16	1103	0	11	0
e	4	26	14	0	1032	11	0
f	0	7	126	33	7	980	0
g	39	1	0	0	3	0	1042

Cuadro 15: Matriz de confusión para Random Forest con parámetros óptimos.

La Tabla 16 muestra, para cada una de las clases con mayor error de clasificación, las dos primeras clases (1a y 2a) con las cuales estas se confunden más frecuentemente y el porcentaje de confusión.

Clase	1a	2a
1	2 (12.7)	7 (7.50)
2	1 (21.1)	5 (11.9)
3	6 (13.2)	4 (6.05)
6	3 (10.9)	4 (2.86)

Cuadro 16: Confusiones más importantes en Random Forest.

5.2. Bagging de SVM

Bagging se basa en entrenar una cantidad B de clasificadores, seleccionando los patrones de entrenamiento de cada uno mediante BootStrap. Si se tiene un total de n patrones de entrenamiento, la técnica de BootStrap consiste en seleccionar n patrones mediante muestreo aleatorio con reemplzamiento. Es decir que algunos patrones pueden estar más de una vez y otros no estar presentes. Una vez que los B clasificadores están entrenados, la clasificación de un nuevo patrón consiste en: clasificar el patrón con cada clasificador y luego aplicar un voto por mayoría [6]. Los clasificadores utilizados en Bagging son siempre del mismo tipo, variando únicamente el conjunto de entrenamiento de cada uno. Todos los clasificadores tienen el mismo peso. Esta técnica, aunque no siempre mejora el porcentaje de acierto, permite reducir la varianza en la clasificación [7].

Usando el clasificador Bagging de Weka (weka.classifiers.meta.Bagging), aplicamos Bagging con 10 iteraciones al clasificador SVM con núcleo RBF y los parámetros óptimos obtenidos anteriormente: $C = 64$ y $\gamma = 1$. En la Tabla 17 se resume el porcentaje de acierto global y por clase obtenido.

Clase	SVM sin Bagging	SVM con Bagging
Todas	80.7	80.8
1	71.3	72.4
2	57.4	57.0
3	73.8	72.6
4	93.6	94.4
5	91.4	91.5
6	78.9	80.1
7	95.0	93.8

Cuadro 17: Porcentaje de patrones correctamente clasificados por clase.

El porcentaje de acierto global es casi idéntico al 80.7 % obtenido con SVM sin Bagging. Analizando la matriz de confusión se observa que, al igual que antes, las clases de menor porcentaje de acierto son la 1, 2 y 3 y las de mayor acierto son la 4, 5 y 7.

Este método de combinación no presenta mejoras significativas en los porcentajes de acierto, aunque es posible que permita reducir la varianza de dicho porcentaje. Sin embargo, teniendo en cuenta que la complejidad del clasificador aumenta considerablemente, descartamos su uso en el clasificador final para reemplazar a SVM sin Bagging.

5.3. Boosting de SVM

Al igual que Bagging, Boosting permite combinar M clasificadores de un mismo tipo. En este caso, el conjunto de entrenamiento $\{w_n^i\}$ de cada clasificador $\{y_i\}$ se genera de forma secuencial, a partir del desempeño del anterior clasificador. Para esto se asignan pesos a cada uno de los patrones y dichos pesos se adaptan de forma que los clasificadores se concentren en los patrones más difíciles de clasificar, asignándoles a estos mayor peso [6]. A su vez, durante el entrenamiento, a cada clasificador se le asigna un peso α_m, siendo este mayor cuanto más preciso sea el clasificador. Una vez que los M clasificadores están entrenados, la clasificación de un nuevo patrón consiste en: clasificar el patrón con cada clasificador y luego aplicar un voto por mayoría ponderada. En la Figura 6 se muestra de forma esquemática el proceso de entrenamiento y clasificación de Boosting para el caso de dos clases.

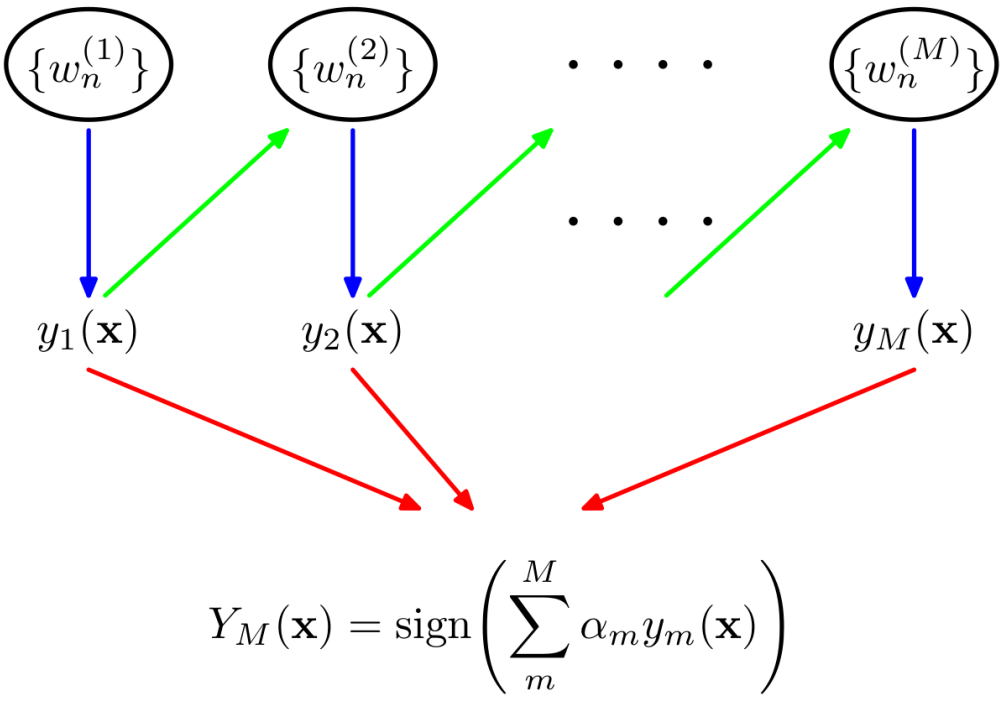


Figura 6: Proceso de entrenamiento y clasificación de Boosting para el caso de dos clases [2].

El algoritmo de Boosting más popular es Adaboost (Adaptive Boosting), propuesto por Freund y Schapire en 1996. Mediante el algoritmo AdaBoostM1 de Weka, y utilizando los datos originales, aplicamos Boosting a SVM con núcleo RBF y los parámetros óptimos obtenidos anteriormente: $C = 64$ y $\gamma = 1$. En la Tabla 18 se resumen los porcentajes de acierto global y por clase obtenidos.

	SVM sin Boosting	SVM con Boosting
Clase	Acierto (%)	
Todas	80.7	80.7
1	71.3	72.0
2	57.4	59.1
3	73.8	74.1
4	93.6	92.0
5	91.4	90.0
6	78.9	79.8
7	95.0	94.3

Cuadro 18: Porcentaje de patrones correctamente clasificados por clase.

El procentaje de acierto global es idéntico al obtenido sin Boosting. Analizando la matriz de confusión, se observa que, al igual que antes, las clases de menor porcentaje de acierto son la 1, 2 y 3 y las de mayor acierto son la 4, 5 y 7.

Al igual que con Bagging, este método de combinación no presenta mejoras significativas en los porcentajes de acierto, mientras que aumenta la complejidad del clasificador. Por lo tanto, también descartamos su uso en el clasificador final para reemplazar a SVM.

6. Combinación mediante Voto ponderado

Teniendo en cuenta que el clasificador Random Forest tiene un desempeño notoriamente superior al de los restantes clasificadores, se implementó un voto por mayoría entre los cuatro mejores clasificadores, pero dándole mayor peso a Random Forest. Esto se hizo con un esquema como el que se muestra en la Figura 7.

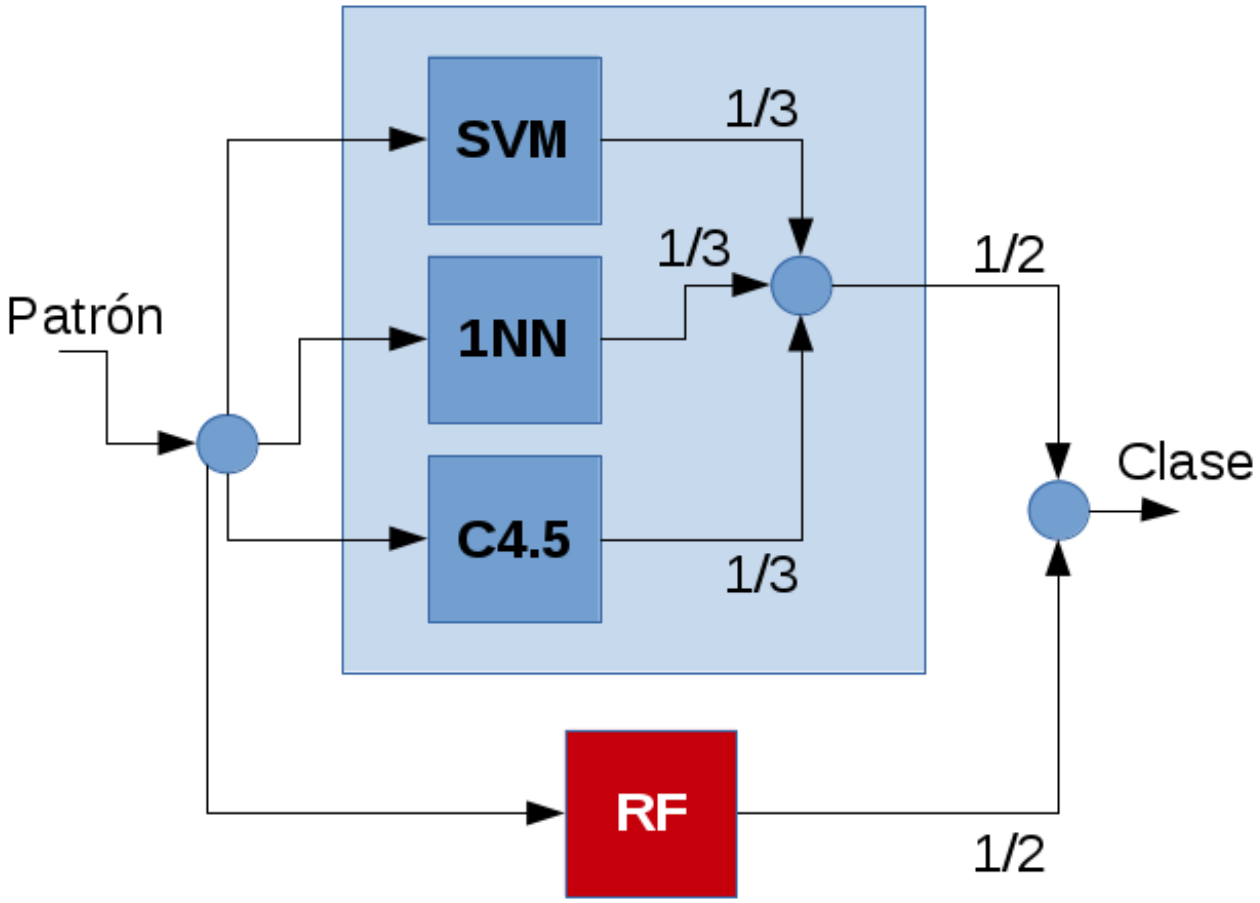


Figura 7: Estructura del voto ponderado.

De esta forma, el voto de Random Forest tiene un peso $w_{RF} = \frac{1}{2}$, mientras que los restantes tienen un peso $w = \frac{1}{3}\frac{1}{2} = \frac{1}{6}$. Mediante validación cruzada de 10 particiones, se obtuvo un porcentaje de acierto global de 83.6 % y los porcentajes de acierto por clase que se muestran en la Tabla 19.

Clase	Acierto (%)
Todas	83.6
1	75.2
2	59.8
3	75.9
4	96.0
5	94.0
6	84.6
7	95.6

Cuadro 19: Porcentaje de patrones correctamente clasificados por cada clase.

Al igual que antes, las clases de menor porcentaje de acierto son la 1, 2 y 3; mientras que las de mayor porcentaje de acierto son la 4, 5 y 7. En la Tabla 20 se muestra la matriz de confusión obtenida.

Clase	a	b	c	d	e	f	g
a	832	155	2	0	34	6	77
b	187	544	25	0	107	36	11
c	0	5	866	73	19	178	0
d	0	0	23	1085	0	22	0
e	5	31	19	0	1022	10	0
f	1	7	125	32	12	976	0
g	44	3	0	0	1	0	1037

Cuadro 20: Matriz de confusión.

La Tabla 21 muestra, para cada una de las clases con mayor error de clasificación, las dos primeras clases (1a y 2a) con las cuales estas se confunden más frecuentemente y el porcentaje de confusión.

Clase	1a	2a
1	2 (14.0)	7 (6.96)
2	1 (20.5)	5 (11.8)
3	6 (15.6)	4 (6.40)
6	3 (10.8)	4 (2.78)

Cuadro 21: Confusiones más importantes.

Si bien el porcentaje de acierto obtenido (83.6 %) es inferior al de Random Forest sin combinar (84.5 %), no descartamos este clasificador ya que podría tener menor varianza en el porcentaje de acierto.

7. Robustez de los clasificadores

En esta sección se estudió la robustez de los tres clasificadores con mejor desmpeño: voto por mayoría de SVM, 1NN y J48, Random Forest y Voto ponderado de SVM, 1NN, J48 y

Random Forest. Para esto se realizó validación cruzada con 10 particiones variando la semilla inicial. Los resultados se muestran en la Tabla 22

Semilla	Voto mayoría	Random Forest	Voto ponderado
1	82.6	84.5	83.6
2	82.4	84.1	83.2
3	82.6	84.1	83.3
4	82.4	84.3	83.4
5	82.5	84.2	83.8
6	82.5	84.5	83.1
7	82.3	84.3	83.7
8	82.4	84.3	83.2
9	82.4	84.3	83.5
10	82.5	84.3	83.5
media	82.5	84.3	83.4
desviación	0.0966	0.137	0.231
mediana	82.5	84.3	83.5
rango intercuartil	0.100	0.100	0.400

Cuadro 22: Robustez de clasificadores.

En la Figura 8 se muestran, para cada clasificador, los porcentajes de acierto de cada semilla junto con la mediana (línea roja) y el rango intercuartil (caja azul).

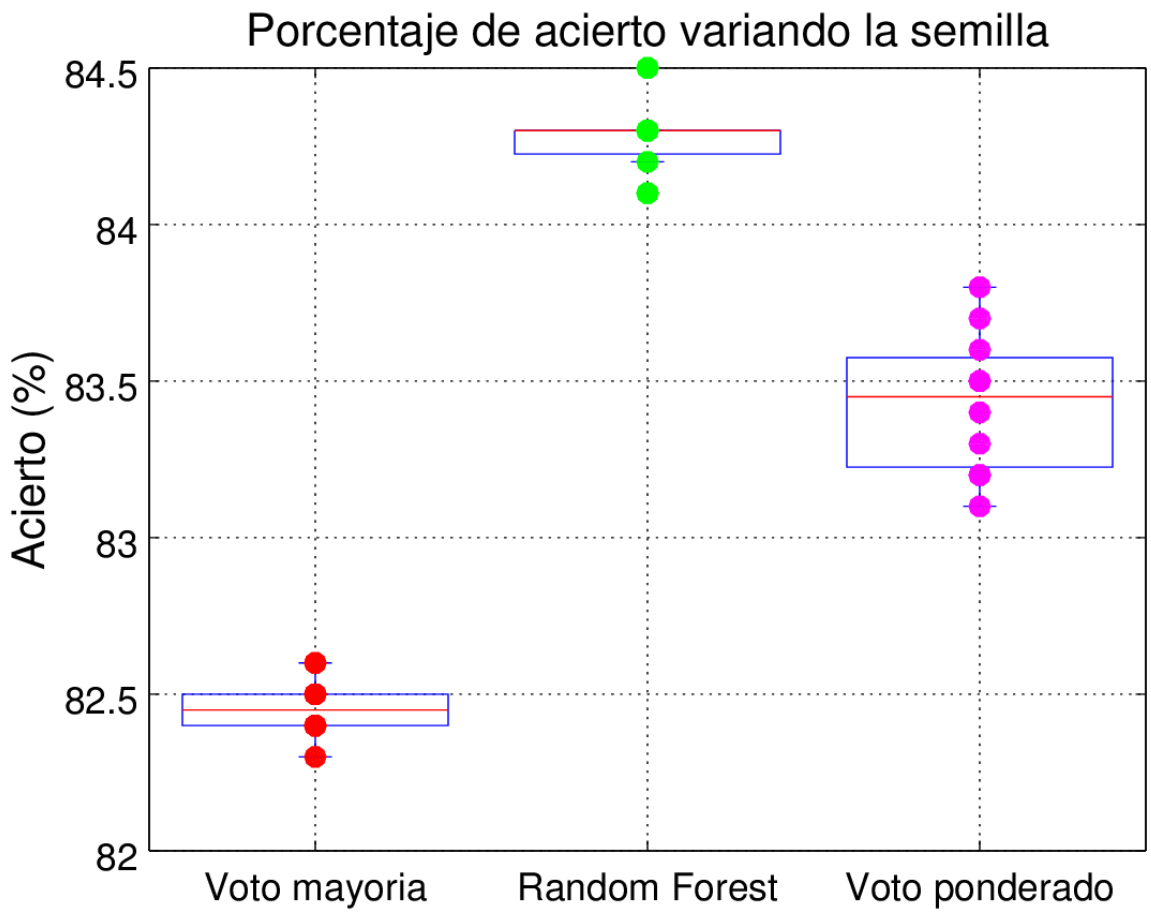


Figura 8: Aciertos con distintas semillas.

El clasificador Random Forest es el de mayor porcentaje de acierto medio con $\mu_{RF} = 84{,}3\,\%$. A su vez, presenta una desviación estándar pequeña $s_{RF} = 0{,}137$, lo cual indica que es robusto. El clasificador de voto ponderado tiene un desempeño medio inferior al de Random Forest aunque superior al voto por mayoría. Sin embargo, presenta una gran dispersión $s_{VP} = 0{,}231$, lo cual indica que es el menos robusto de los clasificadores. Por estos motivos, se opta por utilizar el clasificador Random Forest para probarlo con los datos de test. Realizando un intervalo de confianza al 95 % (factor 1.96), se espera que el porcentaje de acierto en el conjunto test esté en el intervalo $A_{RF} = 84{,}3 \pm 0{,}269\,\% = (84{,}0; 84{,}6)$.

8. Prueba con datos Test

Los datos de test contienen 7049 patrones con las mismas 54 características que los datos originales y pertenecientes a 7 clases como antes. La distribución de las clases es uniforme con 1007 patrones por clase. Utilizando los datos de entrenamiento, se entrenó el clasificador Random Forest de 240 árboles y 15 características obtenido en las secciones anteriores. Con este clasificador se evaluó el conjunto de los datos de test y se obtuvo un porcentaje de acierto global de 84.1 %. Este porcentaje se encuentra dentro del intervalo de confianza obtenido anteriormente.

En la Tabla 23 se muestran los porcentajes de acierto por clase obtenidos al evaluar el clasificador en los datos de test.

Clase	TP rate (%)
Todas	84.1
1	76.5
2	64.8
3	78.9
4	96.3
5	91.0
6	84.5
7	96.9

Cuadro 23: Porcentaje de patrones correctamente clasificados por cada clase.

Las clases de menor porcentaje de acierto son la 1, 2 y 3; mientras que las de mayor porcentaje de acierto son la 4, 5 y 7. En la Tabla 24 se muestra la matriz de confusión obtenida.

Clase	a	b	c	d	e	f	g
a	770	154	0	0	17	1	65
b	199	653	20	0	100	27	8
c	0	4	795	55	17	136	0
d	0	0	20	970	0	17	0
e	3	59	15	0	916	14	0
f	0	6	112	27	11	851	0
g	30	1	0	0	0	0	976

Cuadro 24: Matriz de confusión.

9. Conclusiones

Se implementaron algunos de los clasificadores más importantes vistos en el curso, optimizando sus parámetros para maximizar el porcentaje de acierto.

Se eliminaron, seleccionaron y combinaron características, explorando en cada caso el impacto de esto en el desempeño de los mejores clasificadores. Inicialmente se analizaron los gráficos característica vs. clase y característica vs. característica. Luego se seleccionaron características mediante la función de mérito Ganancia de Información. Se logró reducir el número de características de 54 al orden de 30, disminuyendo así el costo computacional, y sin afectar considerablemente el desemepeño de los clasificadores. Debido a que el objetivo propuesto era maximizar el porcentaje de acierto, en las secciones siguientes se trabajó con el conjunto de características original.

Con los tres mejores clasificadores, se implementó un clasificador mediante voto por mayoría, obteniendo un acierto de 82.6 %, el cual es superior al objetivo mínimo de acierto en el conjunto de entrenamiento (82.0 %).

Más adelante se exploraron formas de mejorar el desempeño de los clasificadores por separado. Para mejorar el árbol C4.5 se utilizó Random Forest con parámetros óptimos, logrando

aumentar el porcentaje de acierto de 77.5 % a 84.5 %. Para mejorar SVM, se utilizaron las técnicas de Bagging y Boosting, logrando porcentajes de acierto similares al de SVM original. Como estas técnicas complejizan el clasificador, se descartó su uso para mejorar SVM.

Mediante un clasificador de voto ponderado, se añadió el clasificador Random Forest al voto por mayoría mencionado anteriormente, pero dándole mayor importancia a su voto. Con este clasificador se obtuvo un porcentaje de acierto de 83.6 %.

Finalmente se analizó la robustez de los tres mejores clasificadores: Voto por mayoría, Random Forest y Voto Ponderado. El clasificador Random Forest resultó ser el de mejor desempeño, con el mayor porcentaje de acierto medio ($\mu_{RF} = 84{,}3$) y baja varianza ($s_{RF} = 0{,}137$). Por este motivo se lo seleccionó como clasificador final a utilizar con los datos de test. Se determinó además el intervalo de confianza al 95 % para el porcentaje de acierto de este clasificador.

El desempeño del clasificador Random Forest se evaluó sobre los datos de test, obteniéndose un porcentaje de acierto de 84.1 %, el cual se encuentra dentro del intervalo de confianza al 95 % obtenido anteriormente.

Referencias

[1] Pattern Classification - Duda, Hart and Stork - John Wiley & Sons, 2001.

[2] Pattern Recognition and Machine Learning - C. M Bishop - Springer, 2006.

[3] UCI Machine Learning Repository, Covertype Data Set - https://archive.ics.uci.edu/ml/datasets/Covertype, octubre 2015.

[4] Blackard, J. A., & Dean, D. J. (1999) - Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables - Computers and electronics in agriculture, 24(3), 131-151.

[5] Hsu, C. W., Chang, C. C. y Lin, C. J. (2003). A practical guide to support vector classification.

[6] Webb, A. R. y Copsey, K. D. (2011). Statistical pattern recognition. 3a edición. John Wiley & Sons.

[7] Witten, I. H., & Frank, E. (2005). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.