Random Forests

Teoría y ejemplos

Romain Gouron[1]

[1]Departamiento de Ingenería Matemática
Doble titulo Ecole Centrale de Nantes (Francia)

Conferencia 9, GLAM, 2016

Outline

Outline

Árboles de decisión
Definición

Definición

Un árbol de decisión es un clasificador - en forma de árbol - tál que :

- En cada nodo se prueban los *features*
- Hay una rama por cada valor del *feature* probado
- Las hojas simbolizan las categorías (*output*)

Árboles de decisión
Ejemplo

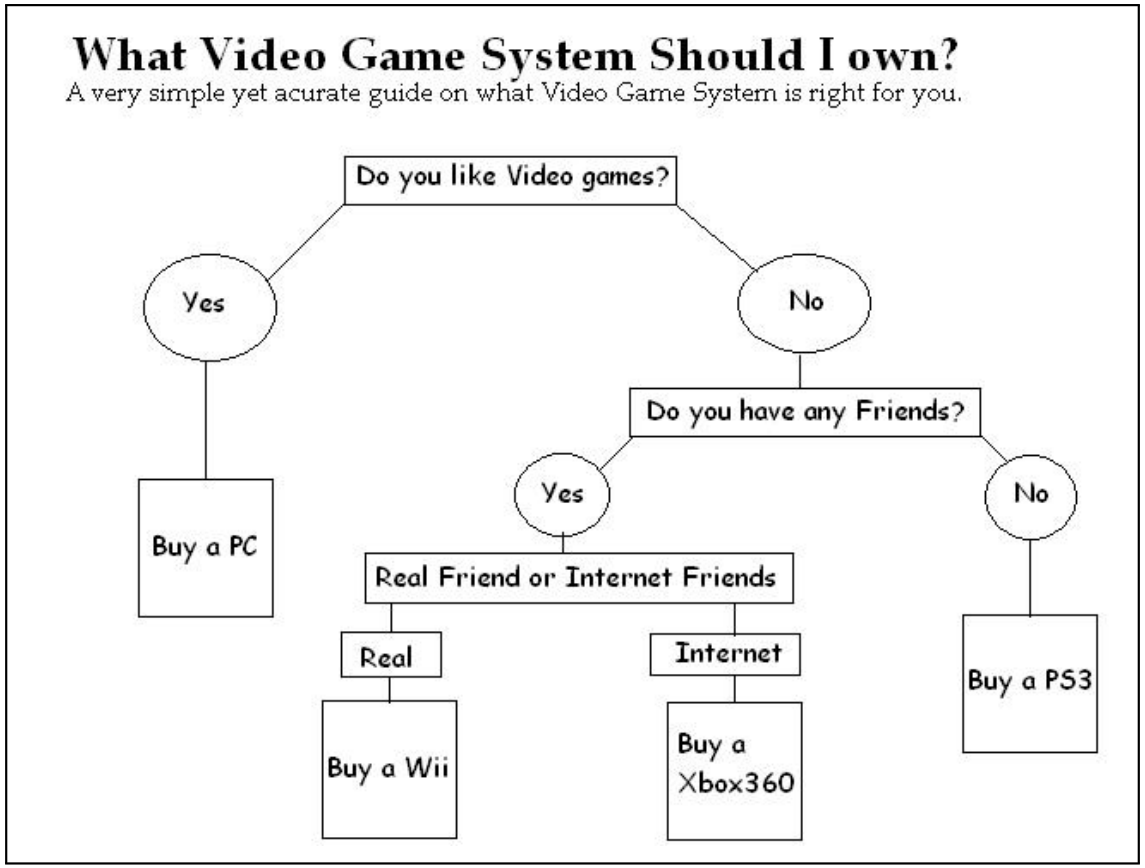

Figura: Ejemplo de árbol de decisión

Árboles de decisión
Caracteristicas

Ventajes

Los árboles de decisión :

- funcionan bien con datos cualitativos (si el numero de *features* es razonable)
- son interpretables

Árboles de decisión
Interpretabilidad

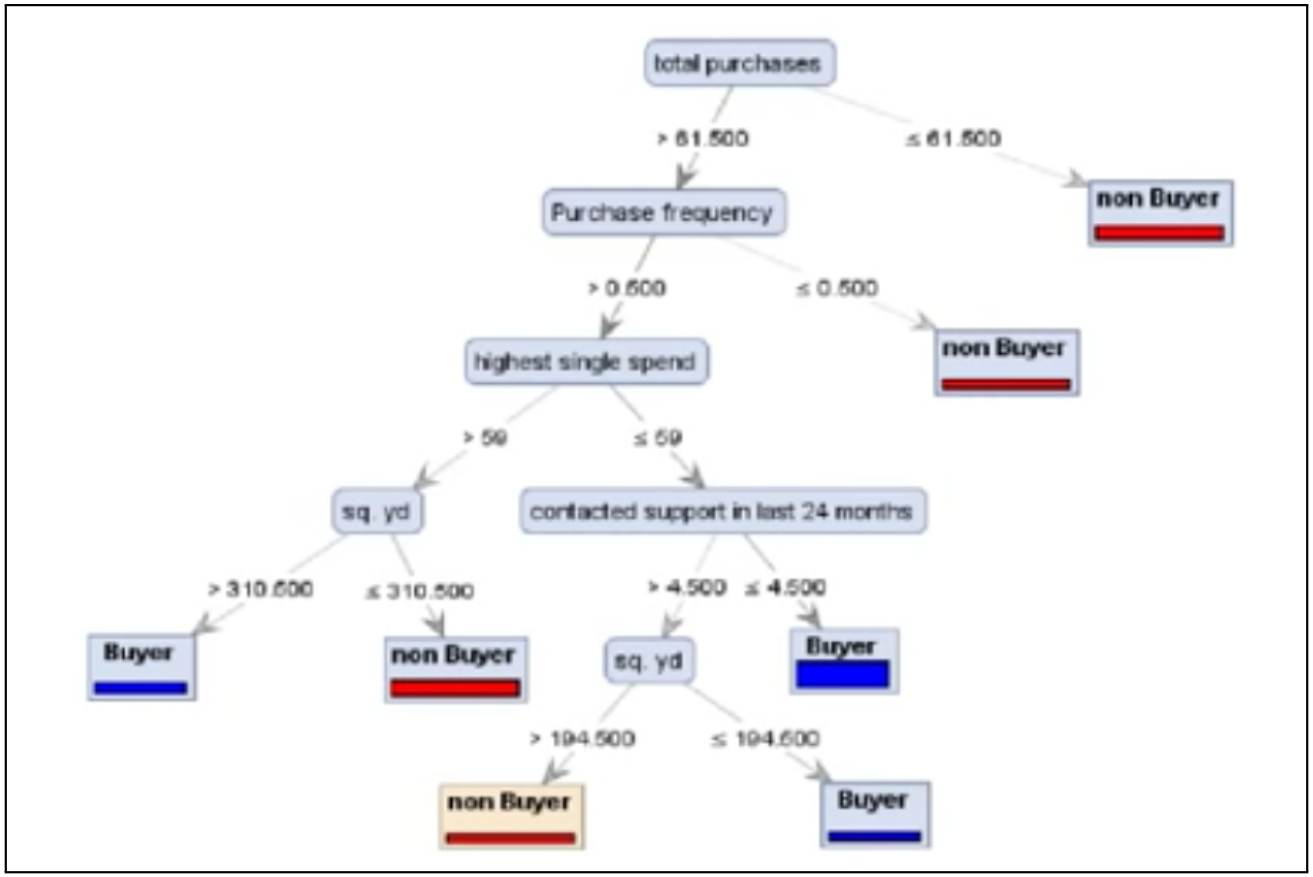

Figura: Árbol interpretable

Árboles de decisión

A qué hace un buen árbol?

Caracterización de un buen árbol

Un buen árbol es un árbol que hace sus preguntas en un orden económico - i.e. que tiene ramas tan cortas como posible.

Árboles de decisión
Ejemplo

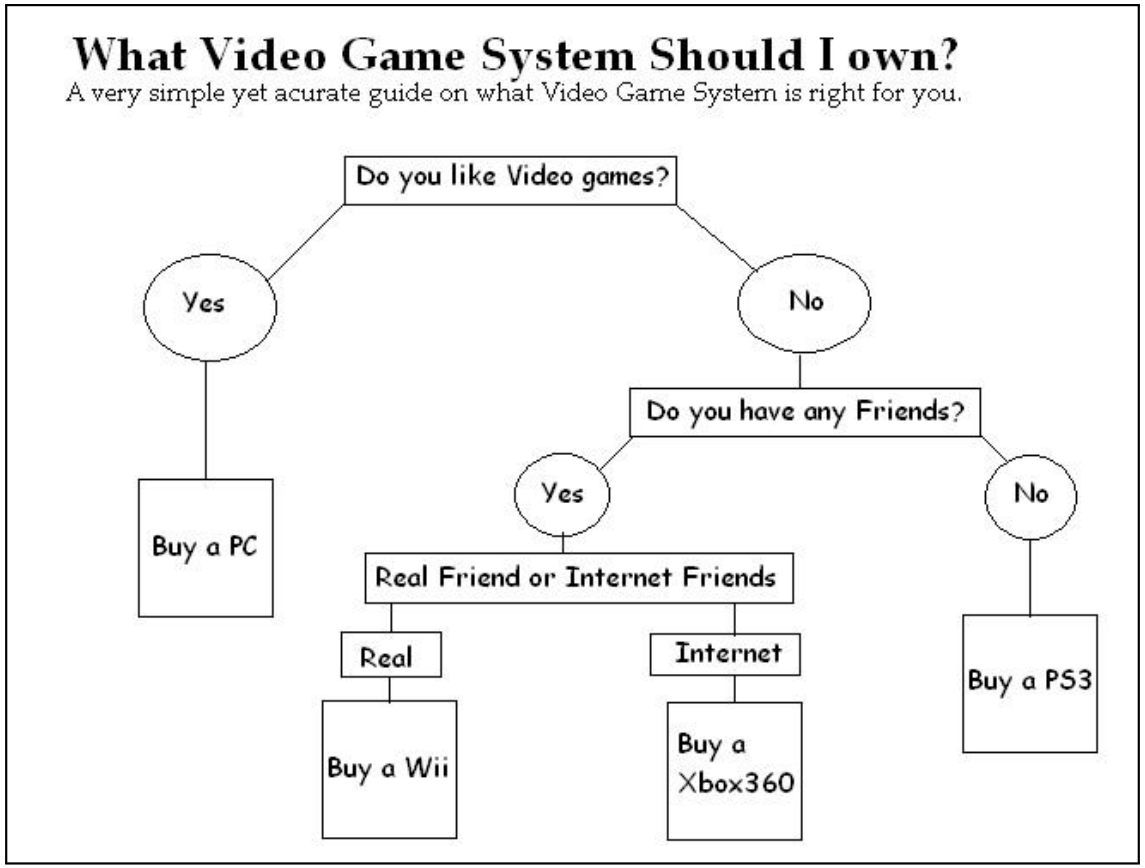

Figura: Ejemplo de árbol de decisión

Árboles de decisión

Ejemplo de mal árbol

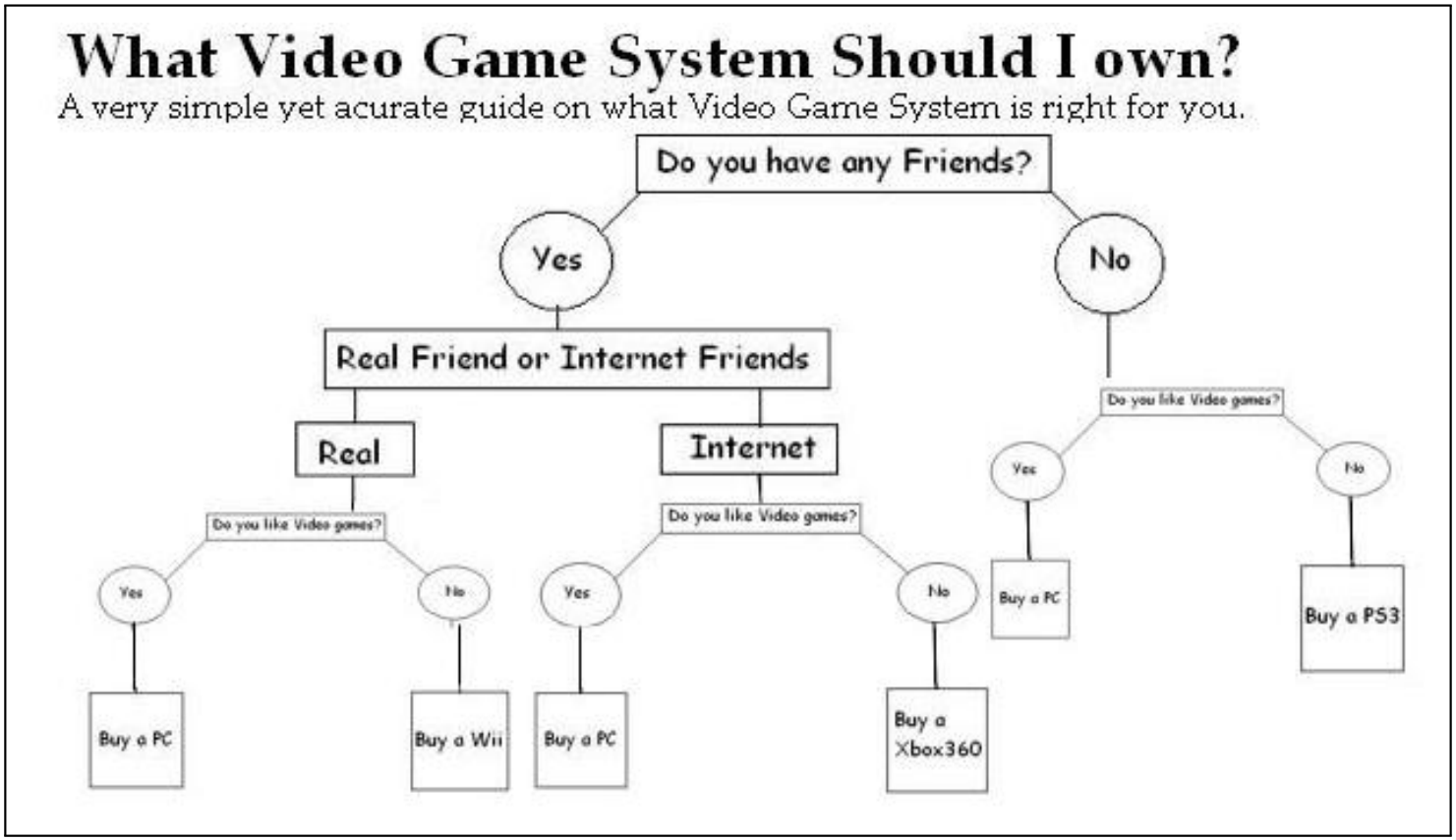

Figura: Árbol malo : podría ser más separador

Outline

Árboles de decisión

Que preguntas hacerse al construir un árbol ?

Como construir un árbol ?

- Como elegir el *feature* de división ?
- Como manejar los *features* continuos ?
- Como definir el tamaño del árbol ?

Eso depende del algoritmo elegido. Los principales son CART y CHAID. Ocupan reglas que veremos más adelante.

Árboles de decisión
Resumen

Lo importante

- Aprendizaje supervisado
- Pro : un árbol permite construir reglas a partir del conjunto de datos para ordenarlo
- Contra : fuerte propensión al *overfitting*

Outline

Random Forests : Contexto

Marco histórico

Random Forests, por L. Breiman, 2001

Idea

Generar un numero importante de árboles, entrenarlos y calcular el promedio de su salida.

Random Forests

Por qué se llaman así? Que tan aleatorio son ?

Construcción de los RF

A cada árbol se le asigna :

- una parte aleatoria de los datos (*tree bagging*)
- una parte aleatoria de los features (*feature sampling*)

Formula de los RF

$RF = tree\ bagging + feature\ sampling$

Random Forest
Construcción de los RF

Tree Bagging

La construcción de B árboles se hace con sigue :

- Se tiran al azar, y con reposición, B muestras del problema (X,Y), que notamos (X_b, Y_b) $(b \in \{1..B\})$
- Se entrena un árbol sobre cada par (X_b, Y_b)

Eso baja el *overfitting*

Random Forest

Tree Bagging - Ejemplo

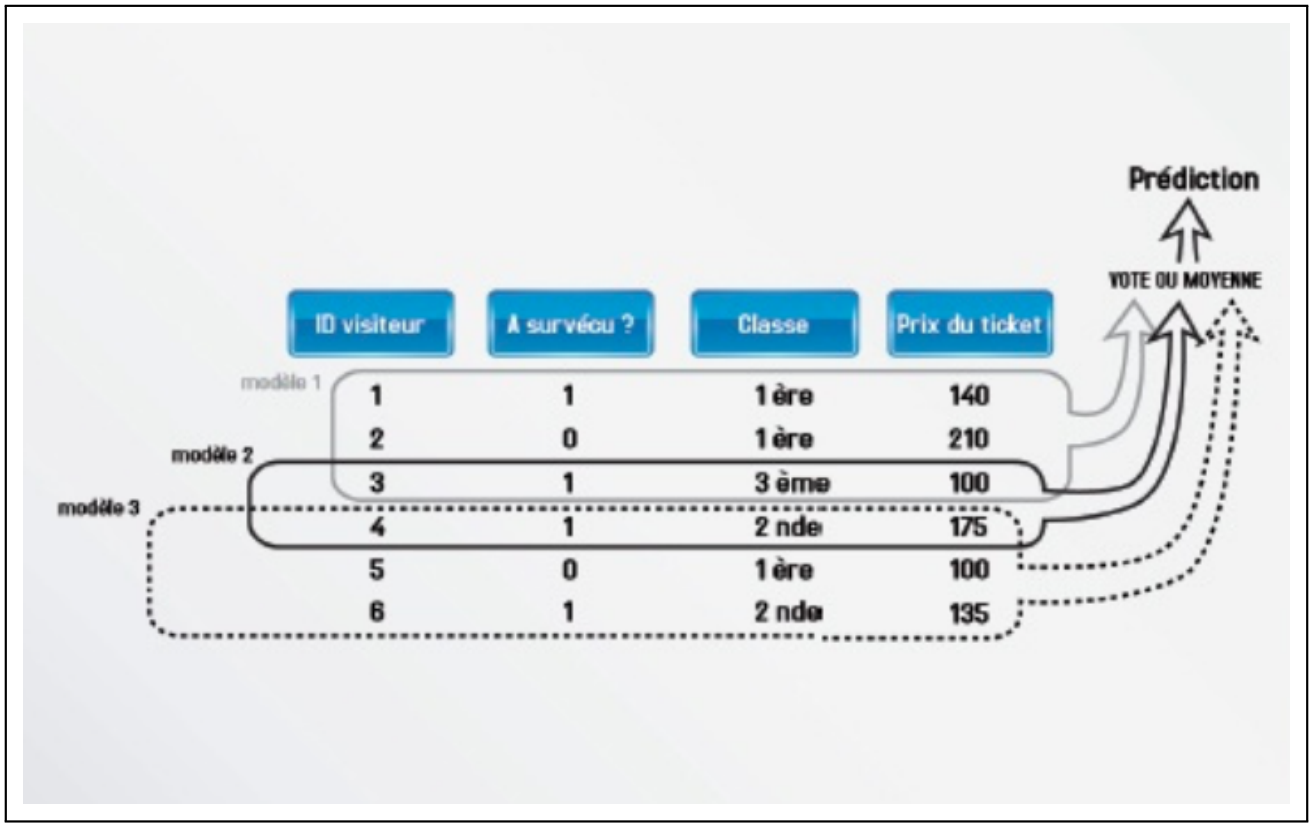

Figura: *Tree Bagging* con datos del Titanic

Random Forest
Construcción de los RF

Feature Sampling

Sobre los *n features*, cada árbol tiene acceso a solamente una parte (típicamente $\sqrt{n}$).
Esto baja la correlación entre los árboles (notado por el coeficiente ρ).

Random Forest

Tree bagging & Feature Sampling : Efecto sobre la varianza

Varianza de N árboles

El promedio de N variables aleatorias iid tiene varianza $\frac{\sigma^2}{N}$
Si los árboles no son independientes (hipótesis clásica) :

$$V_{forest} = \rho\sigma^2 + (1-\rho)\frac{\sigma^2}{N} \quad (1)$$

Random Forest
Varianza del RF

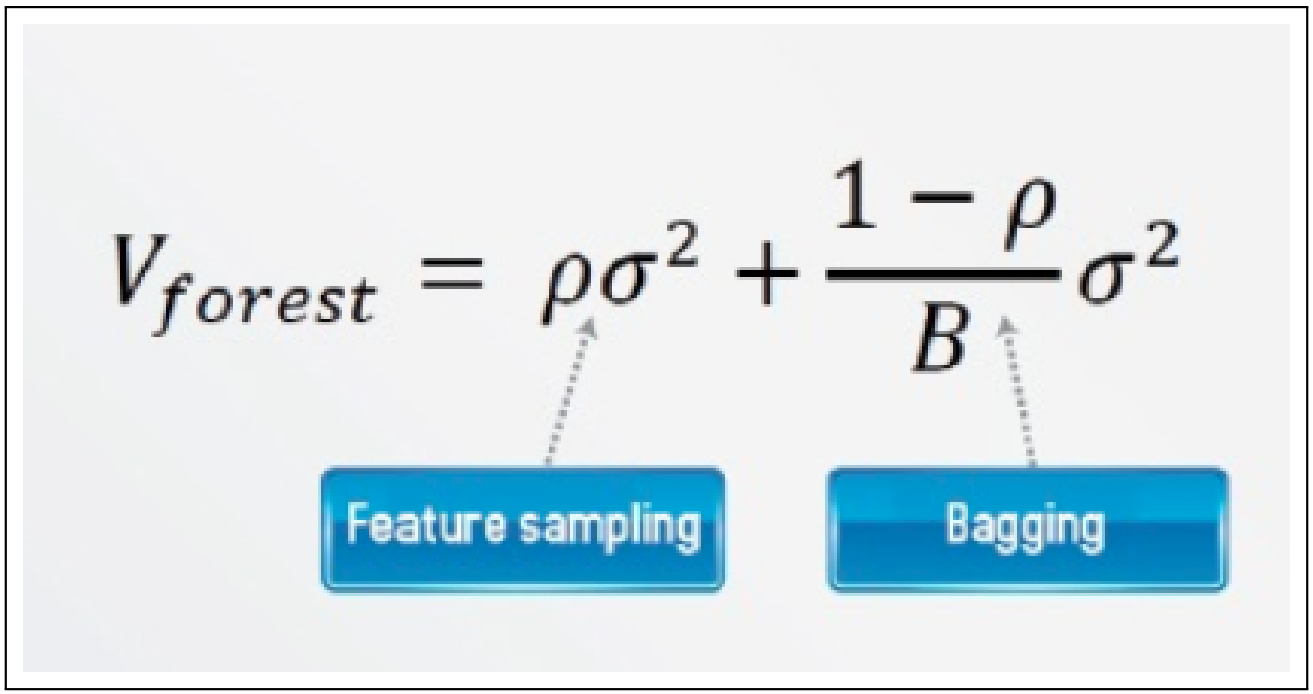

Figura: Factores influyendo sobre la varianza del RF

Outline

Random Forest

Que se minimiza en un RF ?

Criterios de split

Existen dos criterios para hacer una división de ramas (*split*):

- el criterio de Gini
- el criterio de entropía

Random Forest

Que se minimiza en un RF ?

Criterios de Gini

Principio : tomar la clase la más representada, y ver por que *feature* se distingue

Random Forest

Criterio de Gini - Ejemplo

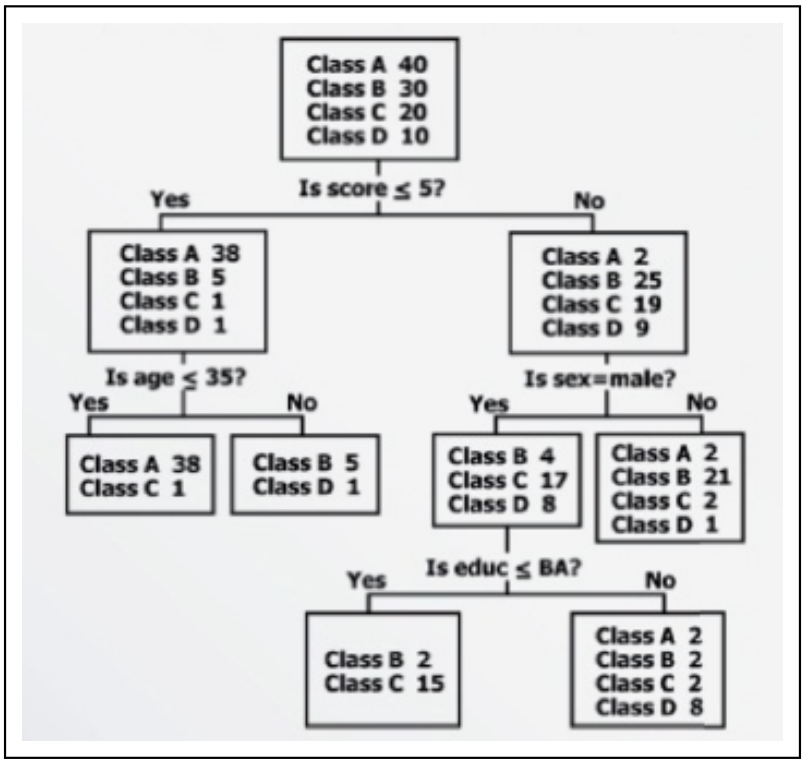

Figura: Ejemplo de aplicación del criterio de Gini

Random Forest

Criterio de entropía

Definición

- Sea S un conjunto de datos labelizados { +, - }
- Sea p_+ la proporción de población positiva en S, y p_- negativa. Entonces :

$$Entropía(S) = -p_+ \log p_+ - p_- \log p_- \tag{2}$$

Ganancia de información

Sea R un nuevo nodo N dando H nuevas hojas

$$Ganancia(S, R) = Entropía(S) - \sum_{h \in H} \frac{|S_h|}{|S|} Entropía(S_h) \tag{3}$$

Random Forest

Criterio de entropía - Ejemplo

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Normal	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Figura: Bajo que condiciones digno jugar tenis?

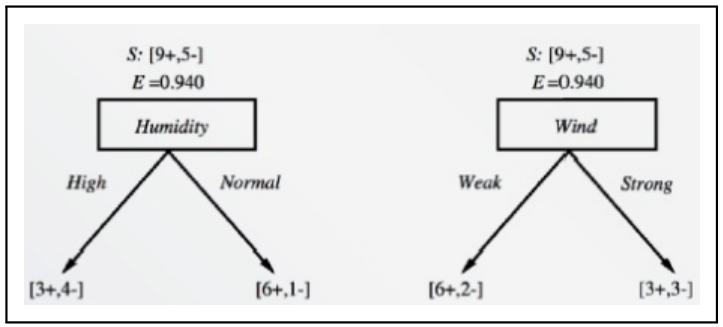

Figura: ¿En cual *feature* hacer el split?

Outline

Ejemplo de clasificación en Python

Spoiler : muy exitoso

Ejemplo de implementación

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=10,
criterion='gini',
max_depth=None,min_samples_split=2, min_samples_leaf=1,
max_features='auto',
max_leaf_nodes=None, bootstrap=True,oob_score=False, n_jobs=1,
random_state=None, verbose=0, min_density=None,
compute_importances=None)
```

Figura: Implementación de RF con *Scikit-learn*

Outline

Resumen

Pros y contras

Pros :

- Fácil de implementar
- Paralelizable

Contra :

- No interpretable
- Muchos parámetros para un reglaje fino

Outline

Extremely randomized trees

Contexto

Marco histórico

Extremely randomized trees, por Geurts P., Ernst D., Wehenkel L., 2005

Extremely randomized trees

Definición

En los RF, en cada árbol :

- cantidad de datos y numero de *features* = aleatorio,
- criterio de *split* = determinista

Con extremely randomized Trees, el *split* incluso es aleatorio. Varios *splits* son generados aleatoriamente, y se elige el que da mejores resultados.

Outline

Rotation Forests

Contexto

Marco histórico

Rotation forest: a new classifier ensemble method, por Rodriguez JJ., Kuncheva LI., Alonso CJ., 2006

Definición

Idea :

1. Tomar K muestras de N_k variables ($k \in \{1, ..., K\}$)
2. En cada muestra, realizar un analisis en componente principal
3. Hacer el ensamblaje de los componentes principales en una matriz
4. Realizar el aprendizaje con esa nueva matriz

Outline

AdaBoost

Contexto

Marco histórico

A Short Introduction to Boosting, por Freund & Schapire, 1999

AdaBoost
Comparación con RF

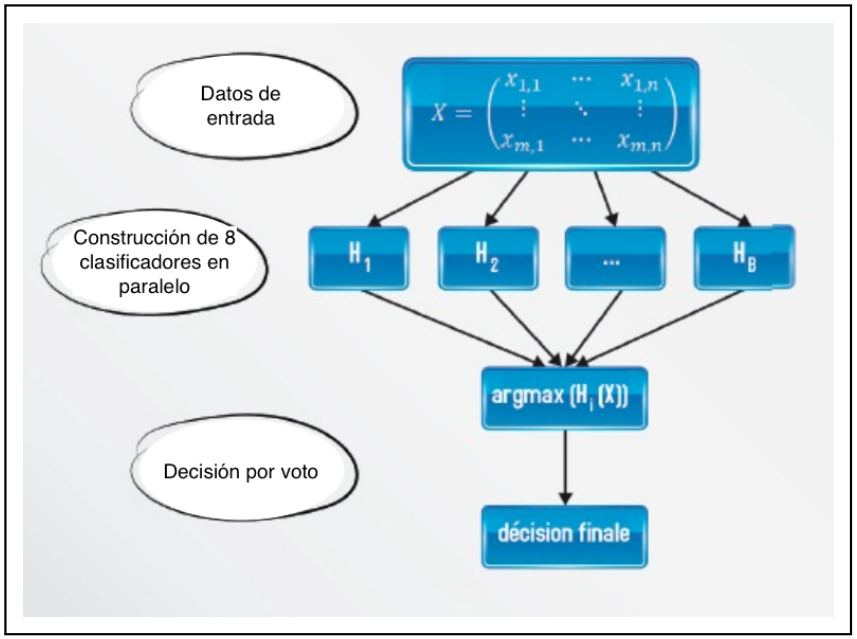

Figura: Esquema de RF

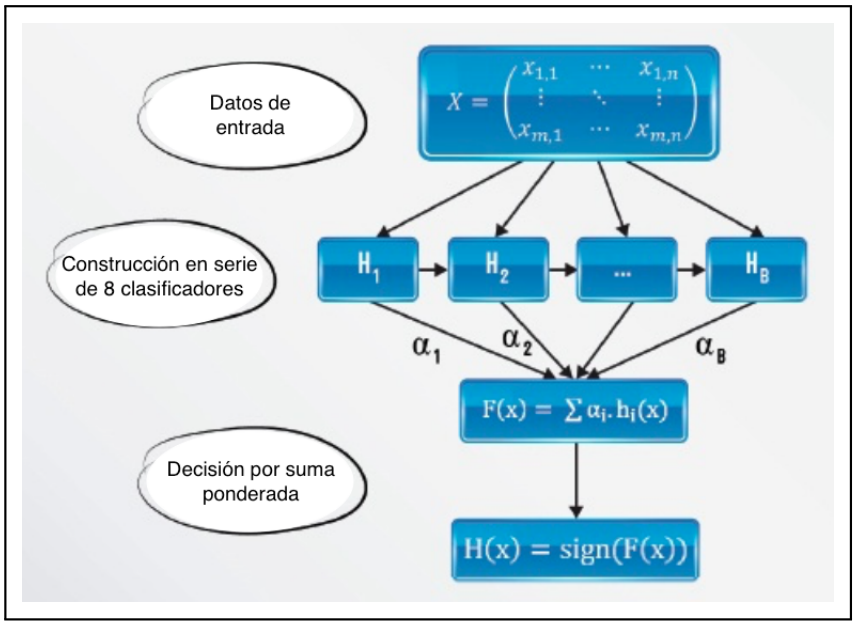

Figura: Esquema de AdaBoost

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

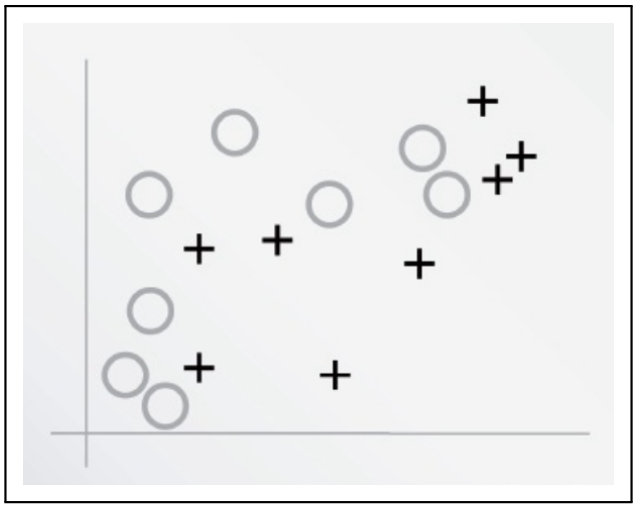

Figura: Puntos a clasificar con el algoritmo AdaBoost

Condición adicional

Solamente se pueden ocupar clasificadores verticales o horizontales

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

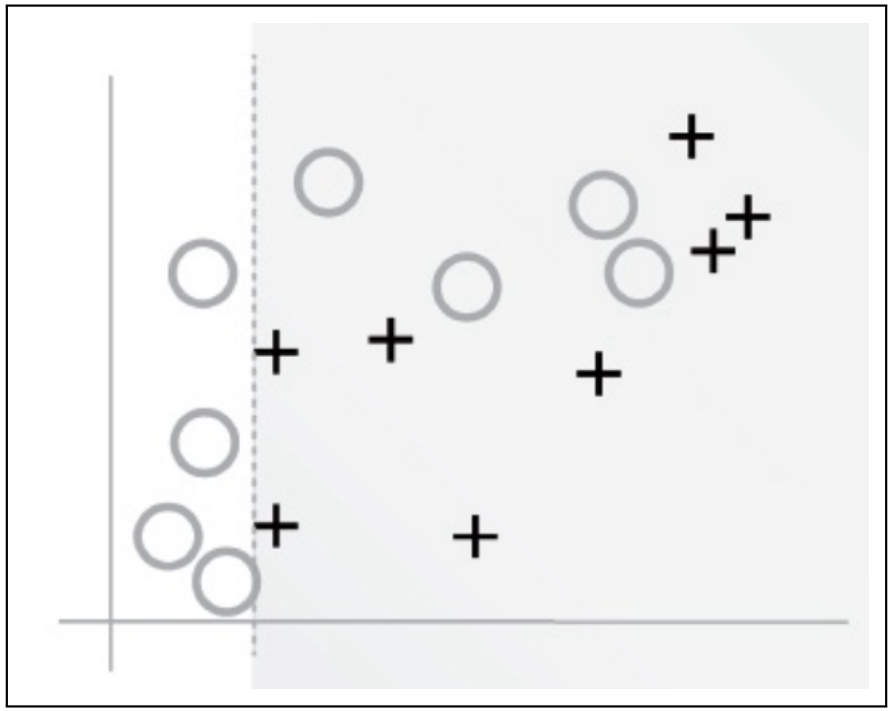

Figura: Clasificación de H1

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

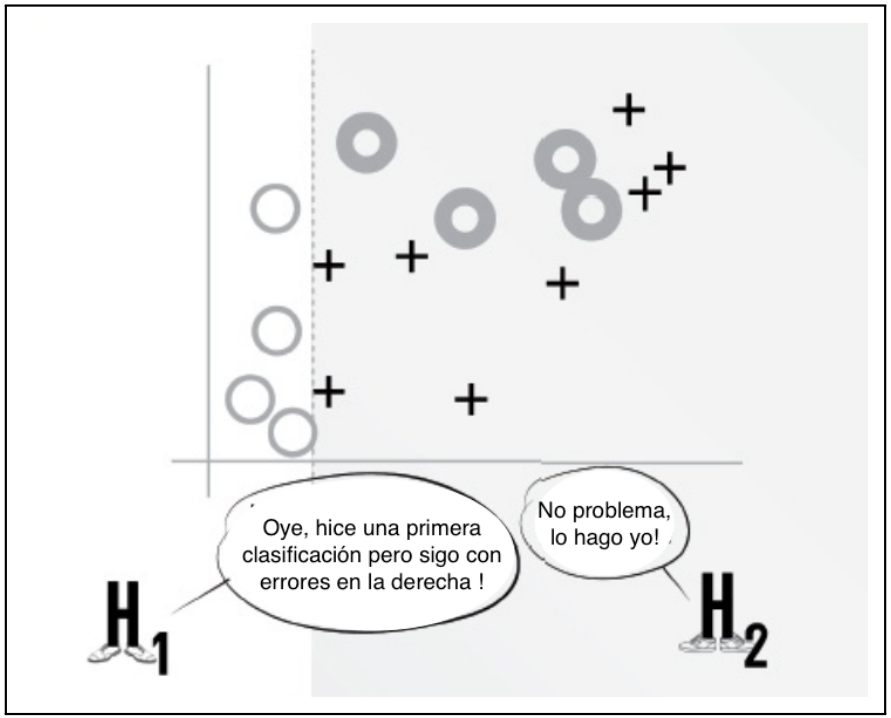

Figura: Informaciones de H1, entregadas a H2

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

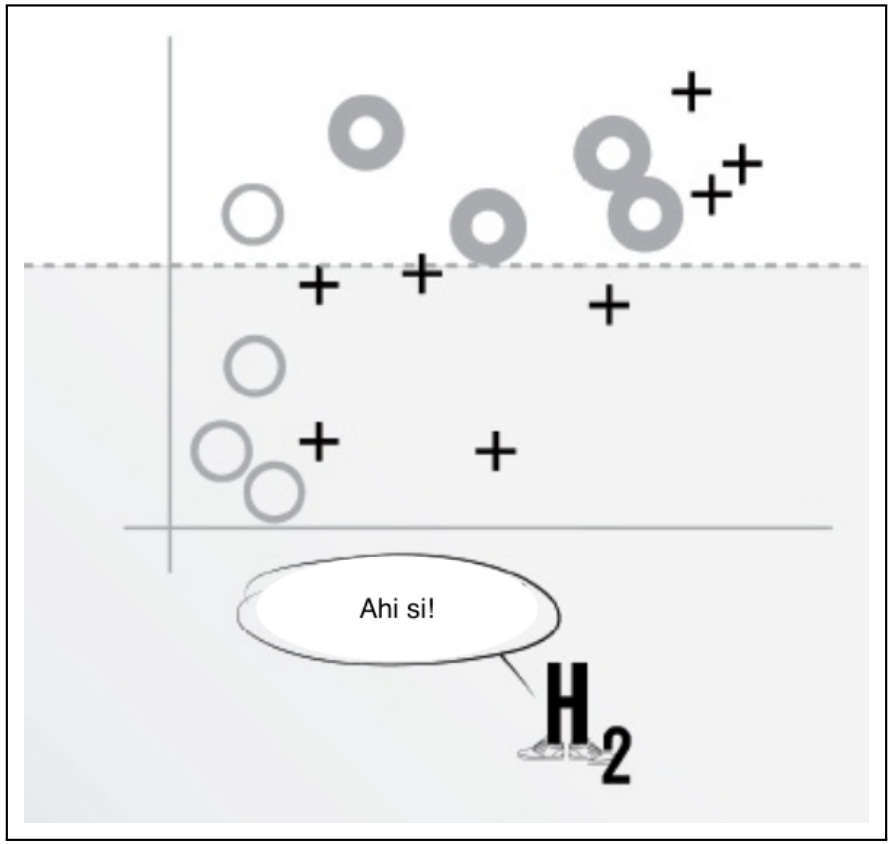

Figura: Clasificación de H2, con la informaciones de H1

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

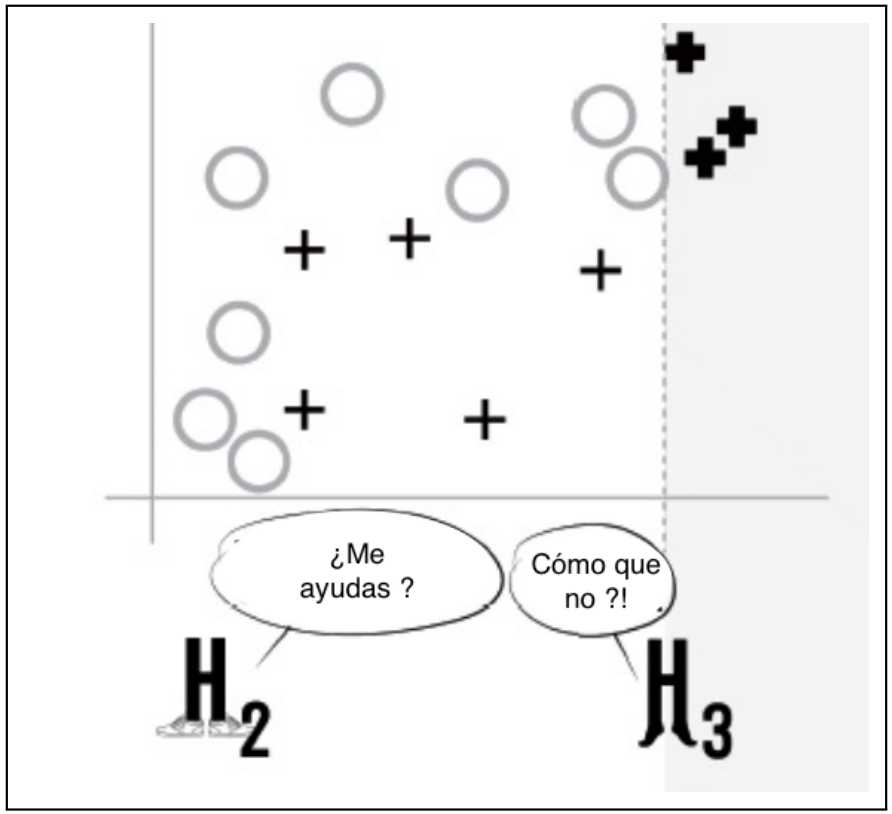

Figura: Informaciones de H1 y H2, entregadas a H3

AdaBoost

Como clasificar esos puntos con clasificador lineales ?

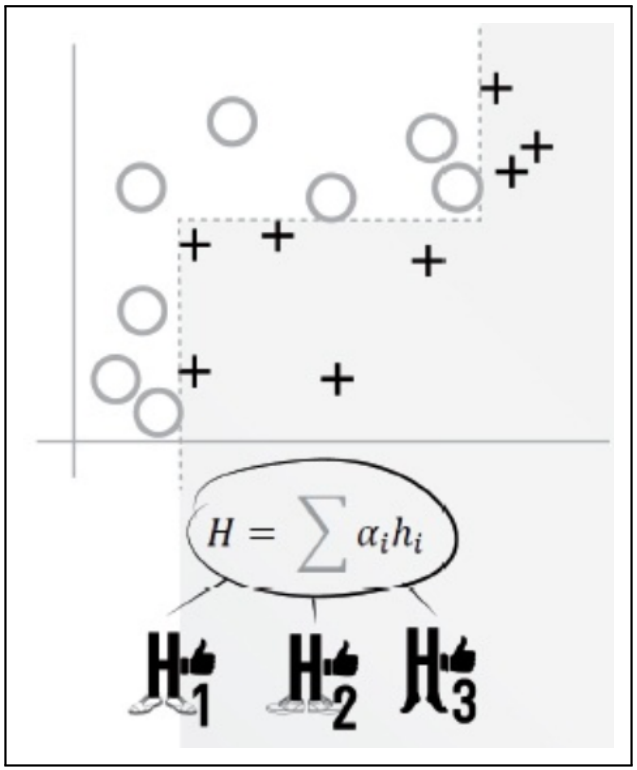

Figura: Clasificación de H1, H2 y H3

Gradient Boosting

Idea

Gradient Boosting = *AdaBoost* + *Gradient descent*

Gradient Boosting

Función de costo

Idea

Se minimiza una función de costo :

$$J(h) = \sum_{i=1}^{n} j(y_i, H(x_i)) \tag{4}$$

Donde :

$$H = H_K = \sum_{k=1}^{K} h_k \tag{5}$$

En cada paso, se busca comparar los resultados de la función h_i con lo "dejado"por h_{i-1}.

Ejemplo

Al principio, H es igual a h_1. Queremos :

$$\forall i \in \{1...m\}\ h_2(x_i) \approx H(x_i) - y_i \tag{6}$$

O sea :

$$h2.fit(X,y\text{-}H) \tag{7}$$

Construcción de H

$$H(x_i) := H(x_i) - \frac{\partial J}{\partial H(x_i)},\ \forall i \in \{1...m\} \tag{8}$$

Outline

Datos

Nombre	Tipo	Anchura	Decimales	Descripción
AGNO	Numérico	4	0	Año del proceso
RBD	Numérico	5	0	Rol Base de Datos del establecimiento
DGV_RBD	Numérico	1	0	Dígito verificador del RBD
NOM_RBD	Texto	100		Nombre del establecimiento
LET_RBD	Texto	1		Letra del establecimiento
NUM_RBD	Numerico	4	0	Número del establecimiento
COD_REG_RBD	Numérico	2	0	Código de región en que se ubica el establecimiento

Figura: Datos del ministerio

Datos

Nombre	Tipo	Anchura	Decimales	Descripción
COD_COM_RBD	Numérico	5	0	Código oficial[1] de comuna en que se ubica el establecimiento
NOM_COM_RBD	Texto	20		Nombre de la comuna
COD_DEPE	Numérico	1	0	Dependencia administrativa: 1: Corporación Municipal 2: Municipal DAEM 3: Particular Subvencionado 4: Particular Pagado 5: Corporación de Administración Delegada
RURAL_RBD	Numérico	1	0	Área geográfica en que se ubica el establecimiento: 0: Urbana 1: Rural
COD_ENSE	Numérico	3	0	Tipo de enseñanza (Ver anexo I)
COD_GRADO	Numérico	2	0	Grado (Ver Anexo II)
LET_CUR	Texto	2		Letra del curso
MRUN[2]	Numérico	8	0	Identificador único del estudiante (máscara del RUN)
GEN_ALU	Numérico	1	0	Sexo 1: Masculino 2: Femenino
FEC_NAC_ALU	Numérico	8	0	Fecha de nacimiento (aaaammdd)
INT_ALU[3]	Numérico	1	0	Indicador de alumno integrado[4]: 0: No 1: Sí
COD_COM_ALU	Numérico	5	0	Código oficial de comuna de residencia del estudiante
NOM_COM_ALU	Texto	20		Nombre de la comuna de residencia del estudiante
COD_SEC[5]	Numérico	3	0	Sector económico[6] (Ver Anexo III)
COD_ESPE	Numérico	5	0	Especialidad (Ver Anexo III)
PROM_GRAL	Numérico	3	1	Promedio general anual[7]
ASISTENCIA	Numérico	3	0	Porcentaje anual de asistencia

Figura: Datos del ministerio

Datos

Nombre	Tipo	Anchura	Decimales	Descripción
SIT_FIN	Texto	1		Situación de promoción al cierre del año escolar: P: Promovido[8] R: Reprobado Y: Retirado
SIT_FIN_R	Texto	1		Situación de promoción al cierre del año escolar, con indicador de traslado: P: Promovido R: Reprobado Y: Retirado T: Trasladado[9]

Figura: Datos del ministerio

Outline

	AGNO	RBD	DGV_RBD	COD_COM_RBD	COD_DEPE	RURAL_RBD	COD_ENSE	COD_GRADO	MRUN
0	2008	4	3	15101	2	0	310	3	39580.0
1	2008	4	3	15101	2	0	310	3	42871.0
2	2008	4	3	15101	2	0	310	3	249292.0
3	2008	4	3	15101	2	0	310	3	277787.0
4	2008	4	3	15101	2	0	310	3	334474.0

Figura: Datos (parte izq.)

GEN_ALU	INT_ALU	COD_COM_ALU	PROM_GRAL	ASISTENCIA	SIT_FIN	DIST	SIT_FIN_y
2	0	15101	6.3	96	1	0.0	1
2	0	15101	6.7	97	1	0.0	1
2	0	15101	6.1	97	1	0.0	1
2	0	15101	6.2	98	1	0.0	1
1	0	15101	6.1	96	1	0.0	1

Figura: Datos (parte der.)

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=10,
criterion='gini',
max_depth=None,min_samples_split=2, min_samples_leaf=1,
max_features='auto',
max_leaf_nodes=None, bootstrap=True,oob_score=False, n_jobs=1,
random_state=None, verbose=0, min_density=None,
compute_importances=None)
```

Figura: Implementación de RF con *Scikit-learn*

Outline

Determinación de parametros - Caso sin asignación de min_{ss} y min_{sl}

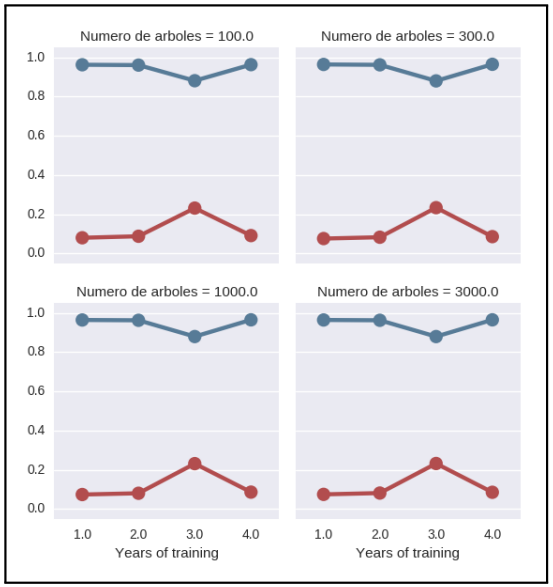

Figura: Predicciones con datos de 1er básico

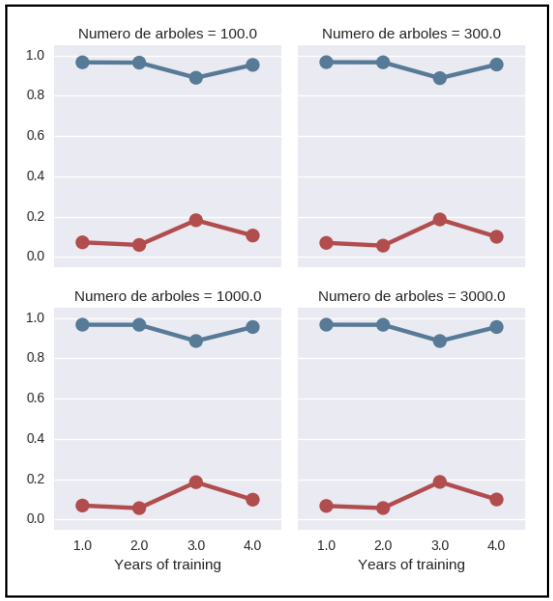

Figura: Predicciones con datos de 5to básico

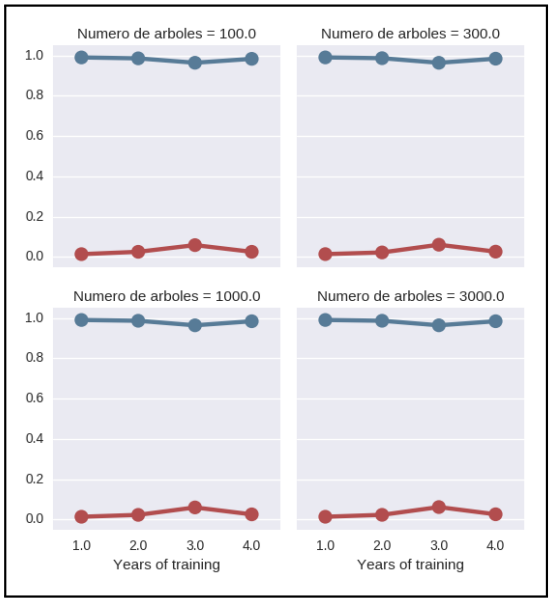

Figura: Predicciones con datos de 3er medio

Determinación de los parámetros

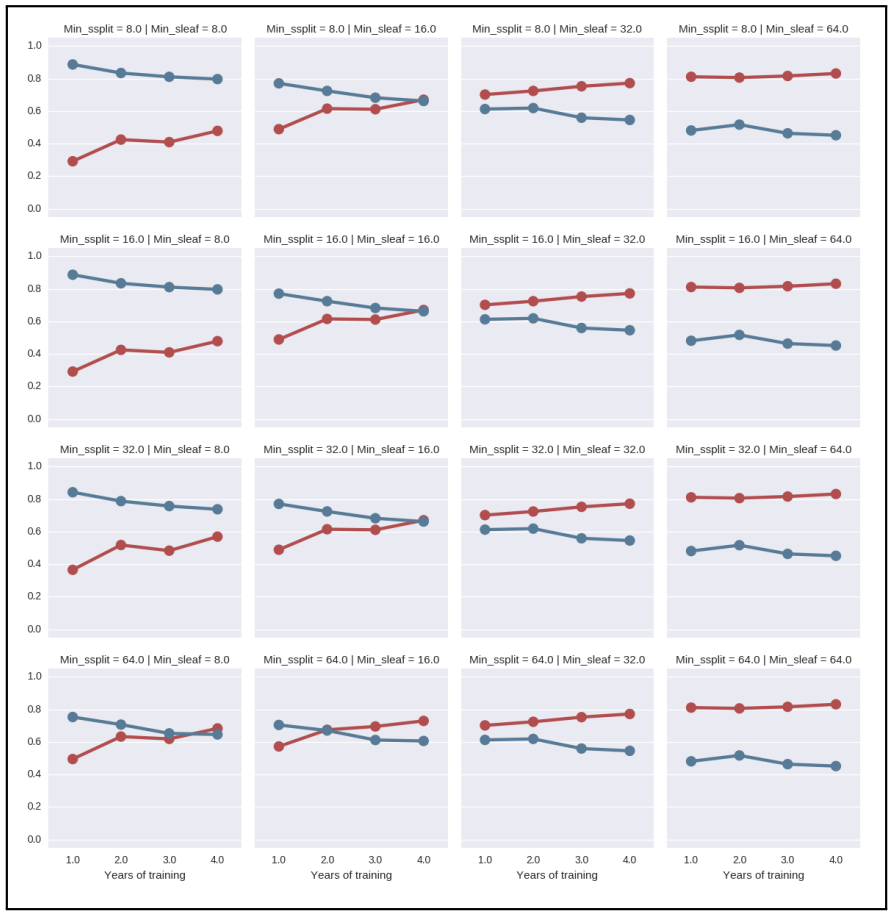

Figura: Determinación de min_{ss} y min_{sl} en 3m

Conclusiones

- Predicciones ciertas a 70 %
- Limitaciones de los datos
- Como mejorarlas ?

For Further Reading I


E. Biernat, M. Lutz, Y. LeCun
Data Science : fondamentaux et études de cas.
Eyrolles, 2015.


L. Breiman.
Random Forests 2001.


J.J. Rodriguez, L.I. Kuncheva.
Rotation Forests : A New Classifier Ensemble Method.
IEEE Transactions on pattern analysis and machine intelligence, VOL. 28, NO. 10, 2001.


P. Geurts, D. Ernst, L .Wehenkel
Extremely randomized trees
Springer Science, 2006