Aprendizaje Automático

Segundo Cuatrimestre de 2016

Evaluación de Hipótesis

DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Aproximación de Funciones

Marco del problema:

- Conjunto de instancias posibles X
- Función objetivo desconocida $f: X \rightarrow Y$
- Conjunto de hipótesis $H = \{ h \mid h : X \rightarrow Y \}$

Entrada del algoritmo de aprendizaje:

- Ejemplos de entrenamiento $\{<x^{(i)}, y^{(i)}>\}$ de la función f.

Salida del algoritmo de aprendizaje:

- Hipótesis $h \in H$ que mejor aproxima a la función f.

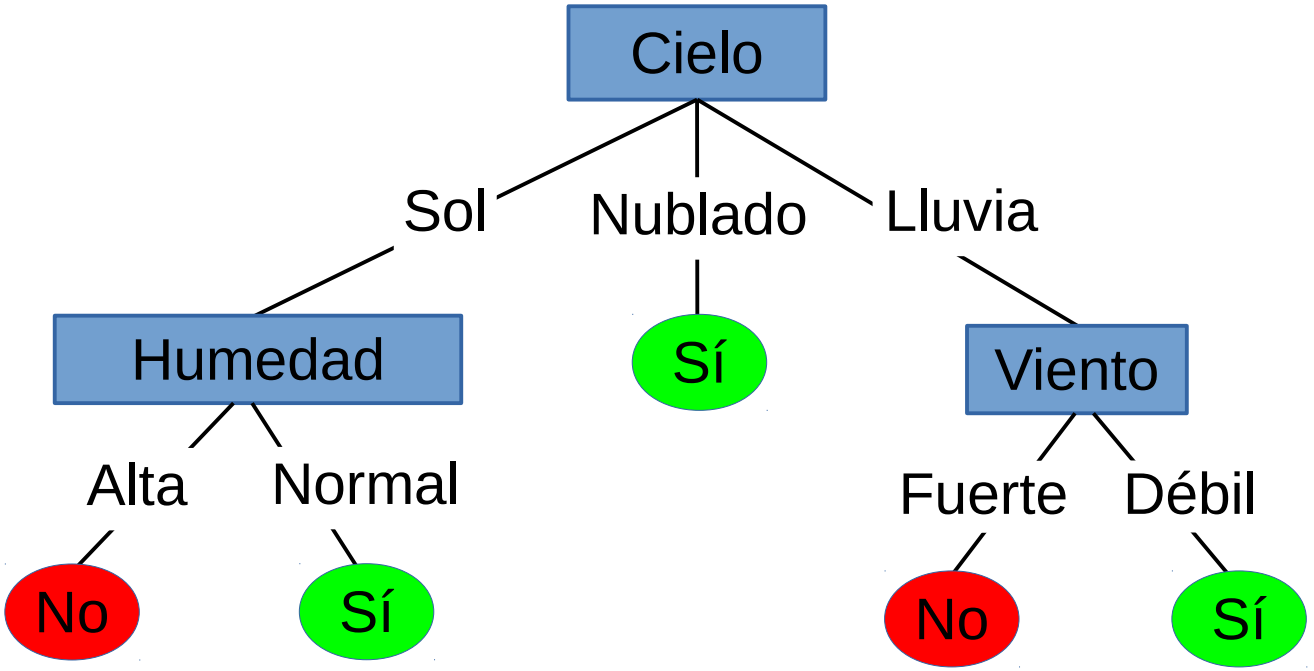
atributos

clase

instancias

Día	Cielo	Temperatura	Humedad	Viento	Tenis?
1	Sol	Calor	Alta	Débil	No
2	Sol	Calor	Alta	Fuerte	No
3	Nublado	Calor	Alta	Débil	Sí
4	Lluvia	Templado	Alta	Débil	Sí
5	Lluvia	Frío	Normal	Débil	Sí
6	Lluvia	Frío	Normal	Fuerte	No
7	Nublado	Frío	Normal	Fuerte	Sí
8	Sol	Templado	Alta	Débil	No
9	Sol	Frío	Normal	Débil	Sí
10	Lluvia	Templado	Normal	Débil	Sí
11	Sol	Templado	Normal	Fuerte	Sí
12	Nublado	Templado	Alta	Fuerte	Sí
13	Nublado	Calor	Normal	Débil	Sí
14	Lluvia	Templado	Alta	Fuerte	No

Árboles de Decisión (ID3, C4.5)

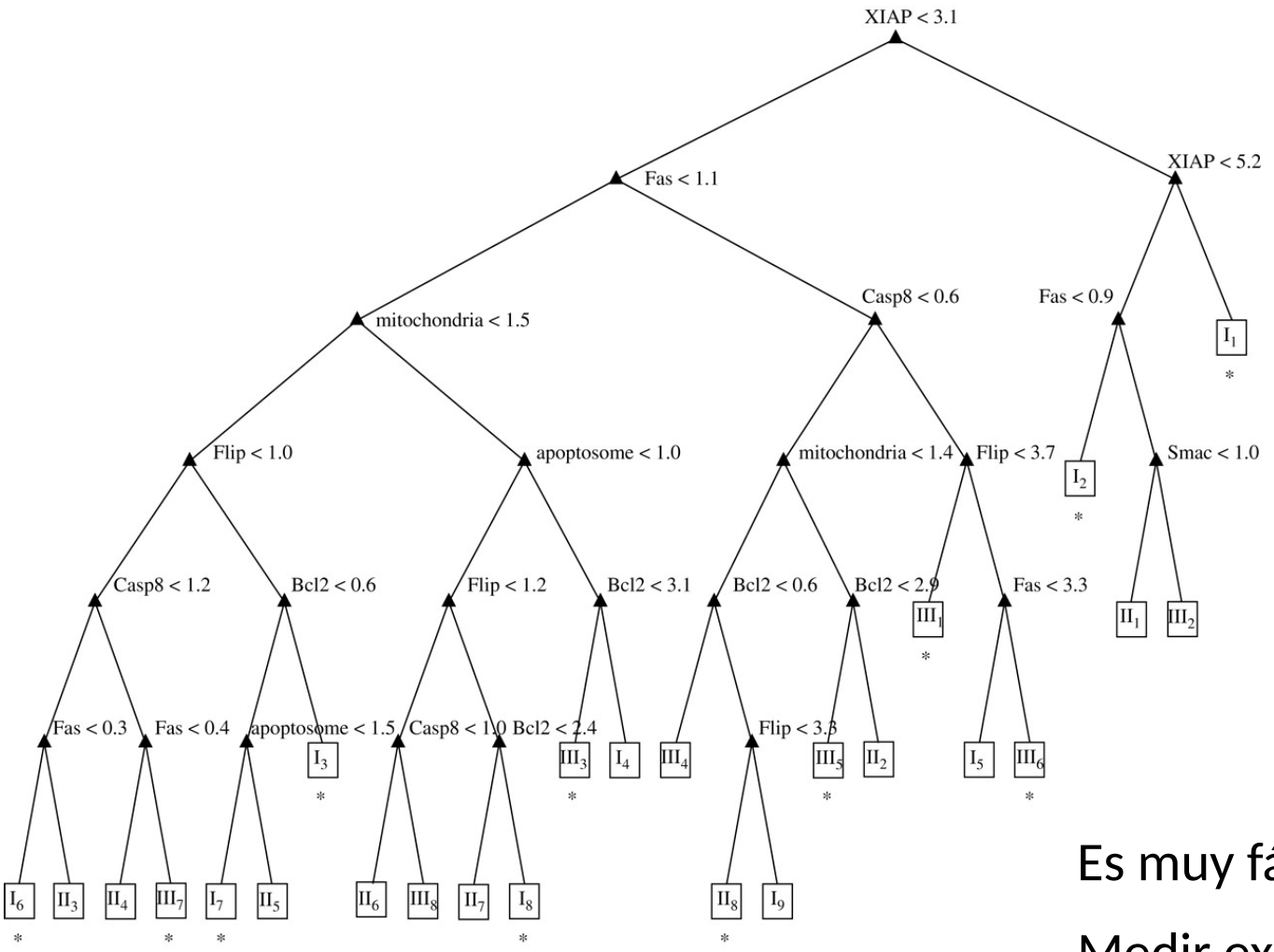


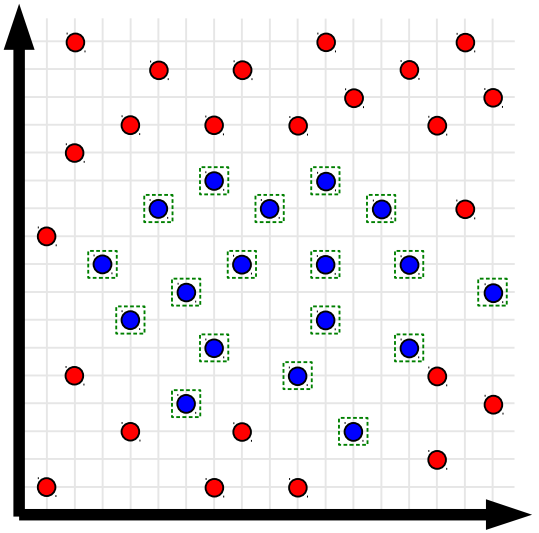
- $f : \langle X_1, \ldots, X_n \rangle \rightarrow Y$
- Cada nodo interno evalúa un atributo discreto X_i
- Cada rama corresponde a un valor para X_i
- Cada hoja predice un valor de Y
- En cada nodo se elige el atributo más *informativo*.

Evaluación de Hipótesis

- Concepto: desconocido.
- ¿Cómo sabemos cuán buena es nuestra hipótesis?
- Primera idea:
  - Exactitud (*accuracy*): Porcentaje de datos de entrenamiento clasificados correctamente.

Sobreajuste (*Overfitting*)





Es muy fácil caer en un sobreajuste.

Medir exactitud sobre datos de entrenamiento → mala idea.

Sobreajuste *(Overfitting)*

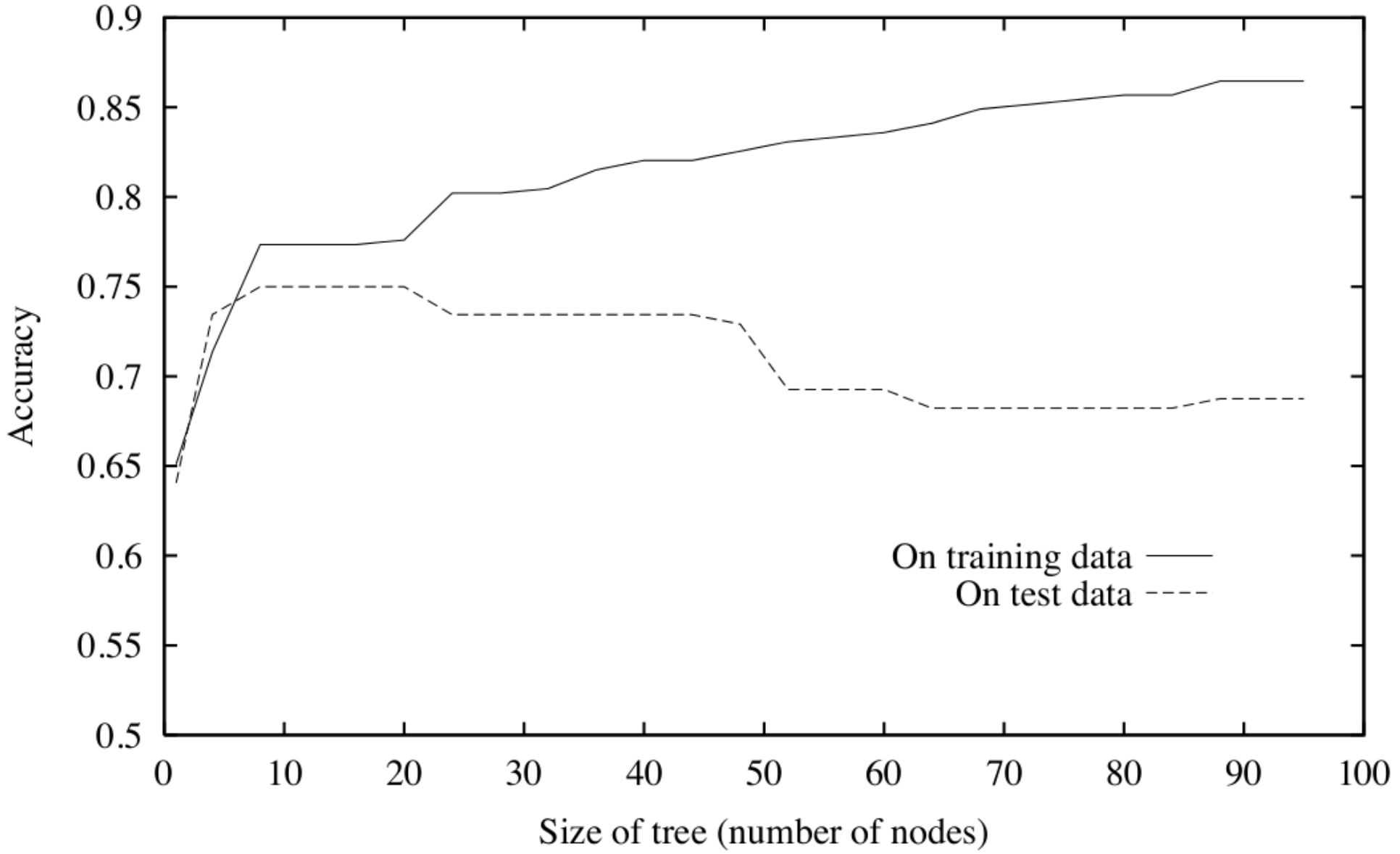
- Considerar el error de una hipótesis *h* sobre:
  - *D* (instancias de entrenamiento): $\text{error}_D(h)$
  - *X* (todas las instancias posibles): $\text{error}_X(h)$
- Definición: *h* se sobreajusta a los datos de entrenamiento si existe *h'* tal que:

$$\text{error}_D(h) < \text{error}_D(h')$$

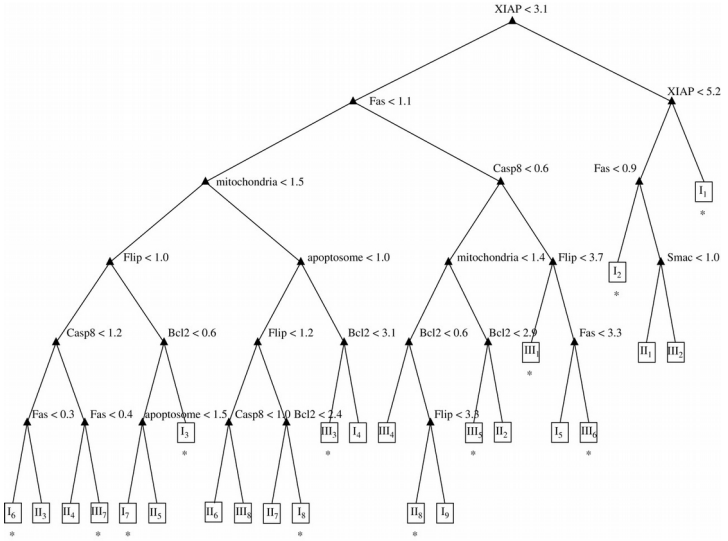
$$\text{error}_X(h) > \text{error}_X(h')$$

- O sea: *h* es mejor sobre *D*, pero *h'* generaliza mejor.

Sobreajuste *(Overfitting)*



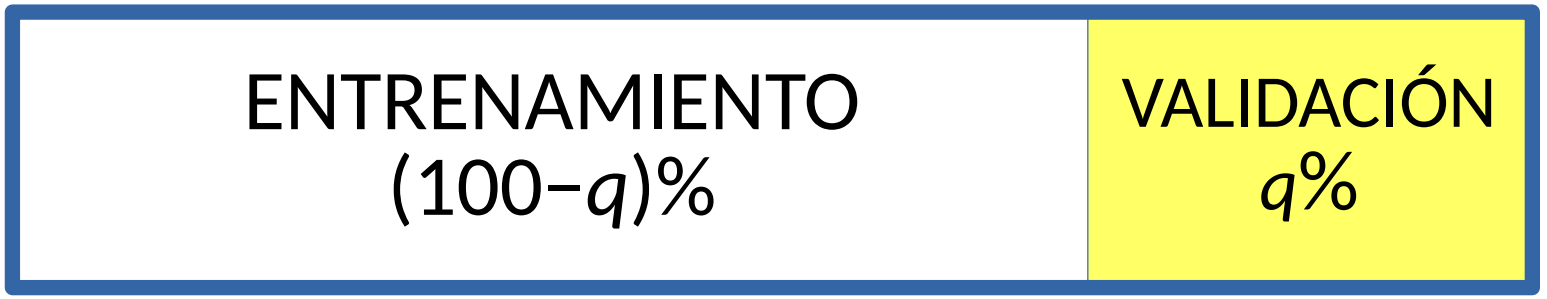
Sobreajuste en Árboles



- Soluciones:
  - Criterio de parada
    - No construir más allá de cierta profundidad.
  - Pruning (poda)
    - Construir el árbol entero; podar las ramas cuando ello mejore la exactitud *sobre datos separados*.
  - Rule post-pruning
    - Construir el árbol entero; convertir árbol a reglas; sacar precondiciones de las reglas cuando ello mejore su exactitud *sobre datos separados*; reordenar las reglas según exactitud.

Entrenamiento y Validación

- Es muy fácil caer en un sobreajuste.
- Medir exactitud sobre datos de entrenamiento → mala idea.
- Surge la necesidad de separar un q% de datos, para validar los modelos: datos de validación (ej.: q=20).



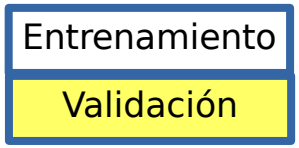
- Los datos se deben separar al azar, para evitar cualquier orden/estructura subyacente en los datos.
- Not.: “validación” / “test” se usan muchas veces en forma intercambiable. En un rato clarificaremos qué es cada uno.

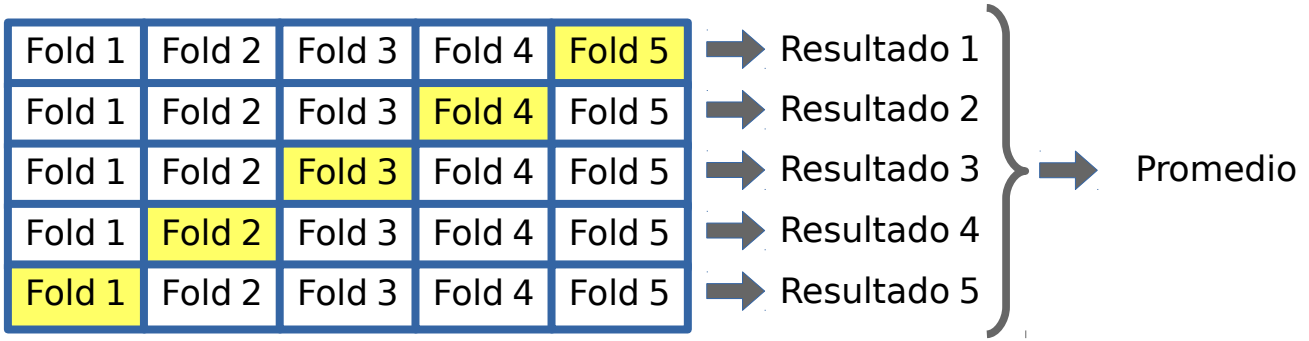
Validación Cruzada

- ¿Qué puede pasar si tenemos mala suerte al separar los datos para entrenamiento/validación?
- *k*-Fold Cross Validation:
  1) Desordenar los datos.
  2) Separar en *k* folds del mismo tamaño.
  3) Para *i* = 1..*k*:

     Entrenar sobre todos los folds menos el *i*.

     Evaluar sobre el fold *i*.
- Ej. para *k*=5:





Comparando Hipótesis

- Queremos comparar 2 hipótesis (modelos): h_1, h_2
- Opción 1: Comparar sólo la exactitud media de c/u.
  - Contra: No podemos saber si diferencias pequeñas son en realidad una consecuencia del azar.
- Opción 2 (mejor): Comparar los vectores de resultados de los *k* folds con un test estadístico:
  1) *k*-fold CV para h_1 $\rightarrow \overrightarrow{Ex}_1 = \langle Ex_{1,1}, Ex_{1,2}, \ldots, Ex_{1,k} \rangle$
  2) *k*-fold CV para h_2 $\rightarrow \overrightarrow{Ex}_2 = \langle Ex_{2,1}, Ex_{2,2}, \ldots, Ex_{2,k} \rangle$
  3) Test apareado entre $\overrightarrow{Ex}_1$ y $\overrightarrow{Ex}_2$.
     - *paired t-test* (paramétrico), o bien *Wilcoxon signed-rank test* (no paramétrico).
     - Output del test: *p*-valor, que nos dice el grado de significancia estadística de la diferencia entre la performance de ambos modelos. (P.ej., $p<0.05 \rightarrow$ diferencia significativa.)

Otra vez sopa...

- Escenario frecuente:
  - Conseguimos un dataset.
  - Experimentamos mucho: extraemos y elegimos atributos, probamos algoritmos, ajustamos parámetros.
  - Llegamos a un modelo que funciona "bien".
  - Lo ponemos a funcionar con datos nuevos, y los resultados son bastante peores.
  - ¿Qué falló?
- Otro nivel de sobreajuste.
  - Sobreajustamos nuestra experimentación a los datos.
- ¿Solución?

Datos de Test

- Lo antes posible, hay que separar un conjunto de datos de test (*test set*), y NO TOCARLOS hasta el final.
- Todas las pruebas y ajustes se hacen sobre el conjunto de datos de desarrollo (*dev set*).
- Cuando termina el desarrollo, se evalúa sobre los datos de test separados. La estimación de performance será más realista.
- ¡No volver atrás!

Medidas de Performance

- Un modelo tiene una exactitud (*accuracy*) del 95%.
  - O sea, de cada 100 instancias, clasifica bien 95.
- ¿Qué significa esto?
- Según la tarea y la distribución de clases en el dominio, 95% puede ser muy bueno o pésimo.
- No dice nada sobre el tipo de aciertos y errores que comete el modelo.
- Ejemplos:
  - **Filtro de spam:** descarta directamente los mails sospechosos.
  - **Detección de fraude:** prepara un listado de casos sospechosos para ser revisados por humanos.
  - Identificación de meteoritos. xkcd.com/1723 :-)
- Veamos otras medidas de performance más útiles…

Matriz de Confusión: (Clasificación binaria)

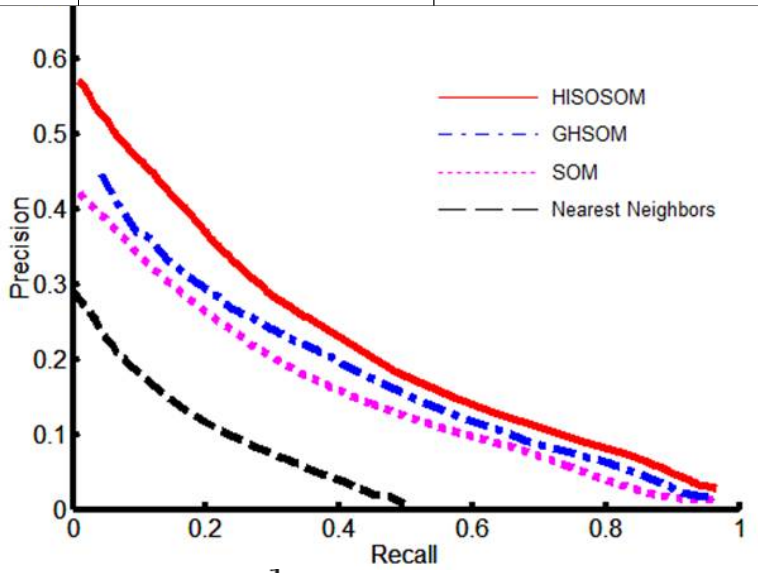
tp: true positives
tn: true negatives
fp: false positives
fn: false negatives

	SPAM (predicho)	**NO SPAM (predicho)**
SPAM (real)	2739 **tp**	56 **fn**
NO SPAM (real)	4 **fp**	1042 **tn**

Precisión y Recall *("exhaustividad")*:

$$\text{Precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}$$

$$\text{Recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}$$



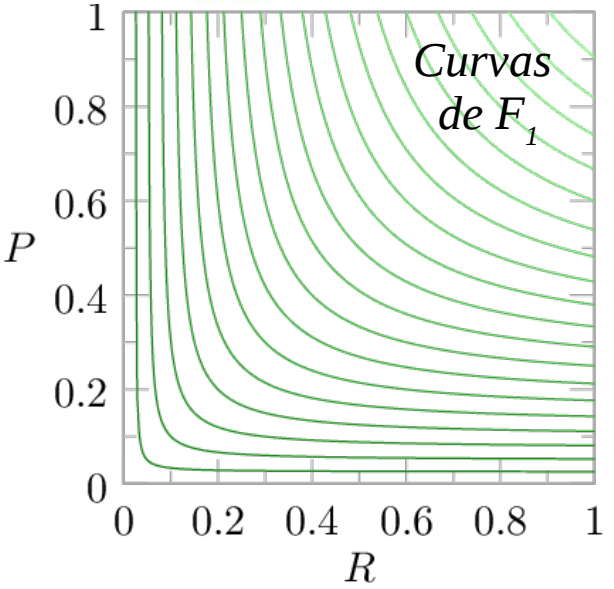
$$F\text{-measure} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Media armónica. También llamada F_1 score.

Fórmula general:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

F_2 enfatiza recall; $F_{0.5}$ enfatiza precision.



Matriz de Confusión: (Clasificación binaria)

tp: true positives
tn: true negatives
fp: false positives
fn: false negatives

	Positivo (predicho)	Negativo (predicho)
Positivo (real)	**tp**	**fn**
Negativo (real)	**fp**	**tn**

Terminología de Recuperación de la Información:

Documento recuperado = Positivo predicho (ej: mail clasificado como spam por el modelo)

Documento relevante = Positivo real (ej: mail anotado como spam por el usuario)

$$\text{Precision} = \frac{tp}{tp + fp}$$ De los documentos recuperados, qué porcentaje son relevantes.

$$\text{Recall} = \frac{tp}{tp + fn}$$ De los documentos relevantes, qué porcentaje fueron recuperados.

Ejemplos de Aprendizaje Automático:

¿Cuál medida (p/r) debería priorizar cada uno de estos sistemas?

- Filtro de spam: descarta directamente los mails sospechosos.
- Detección de fraude: prepara un listado de casos sospechosos para ser revisados por humanos.

Matriz de Confusión: (Clasificación binaria)

tp: true positives
tn: true negatives
fp: false positives
fn: false negatives

	Positivo (predicho)	Negativo (predicho)
Positivo (real)	**tp**	**fn**
Negativo (real)	**fp**	**tn**

Sensibilidad y Especificidad (Medicina, Biología):

$$\text{Precision} = \frac{tp}{tp+fp}$$

$$\text{Recall} = \frac{tp}{tp+fn} = \text{Sensitivity o bien True Positive Rate}$$

$$\frac{tn}{tn+fp} = \text{Specificity o bien True Negative Rate}$$

Sensitivity: Porcentaje de pacientes enfermos correctamente diagnosticados.
Specificity: Porcentaje de pacientes sanos correctamente diagnosticados.

"...the use of repeatedly reactive enzyme immunoassay followed by confirmatory Western blot or immunofluorescent assay remains the standard method for diagnosing HIV-1 infection. A large study of HIV testing in 752 U.S. laboratories reported a sensitivity of 99.7% and specificity of 98.5% for enzyme immunoassay"

Chou R et al., "Screening for HIV: A review of the evidence for the U.S. Preventive Services Task Force", Annals of Internal Medicine, 143 (1): 55-73. 2005.

Matriz de Confusión: (Clasificación binaria)

tp: true positives
tn: true negatives
fp: false positives
fn: false negatives

	Positivo (predicho)	Negativo (predicho)
Positivo (real)	**tp**	**fn**
Negativo (real)	**fp**	**tn**

Curva ROC:

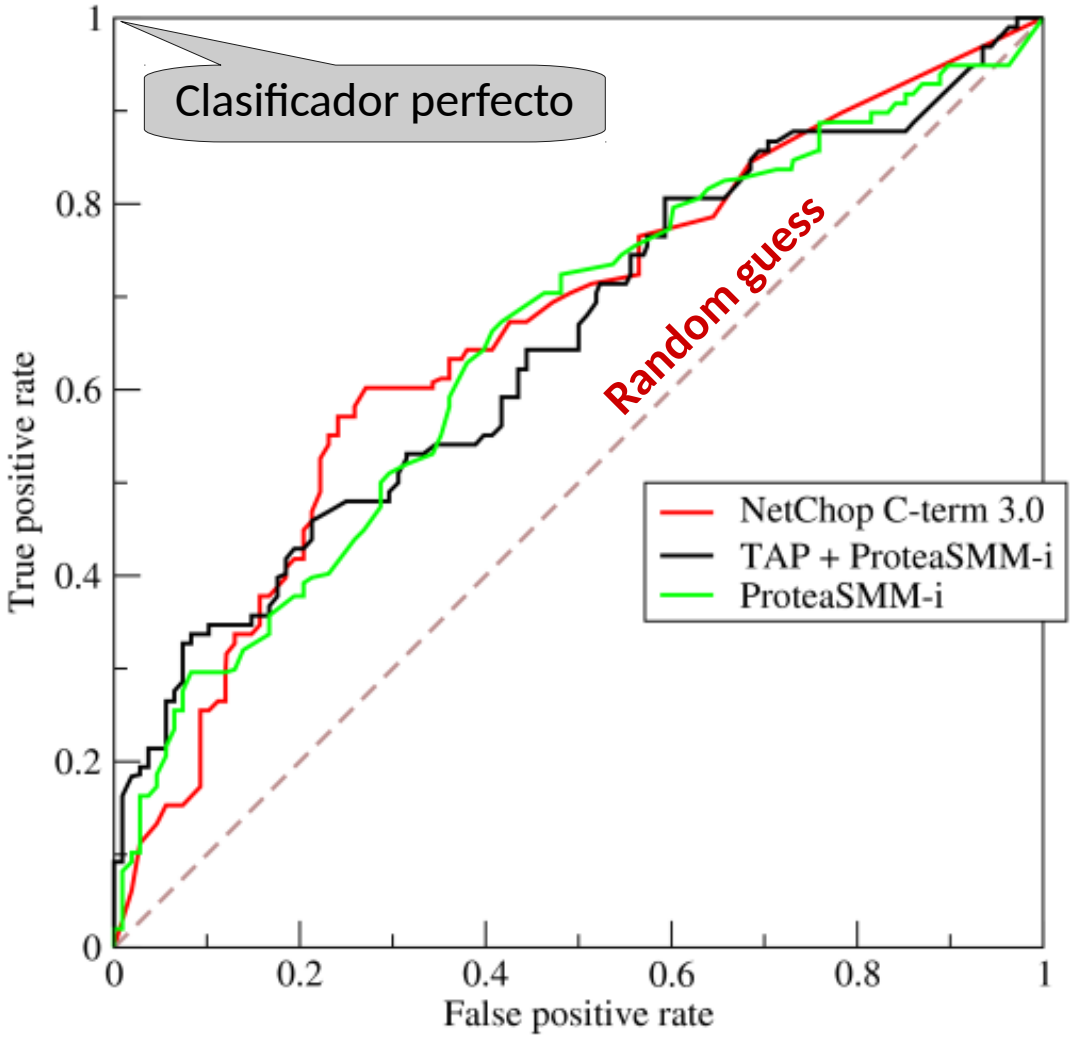
"Receiver operating characteristic"

Gráfico TPR (recall) vs. FPR.

$$\text{Recall} = \text{TPR} = \frac{\text{tp}}{\text{tp}+\text{fn}}$$

$$\text{FPR} = \frac{\text{fp}}{\text{fp}+\text{tn}}$$

Construcción: Variar el umbral de detección entre 0 y 100%. Para cada valor, calcular TPR y FPR (un punto en la curva).



Área bajo la curva (AUC)

Entre 0 y 1. Random = 0.5.

Matriz de Confusión: (Clasificación ***n*-aria**)

	Manzana (predicho)	Naranja (predicho)	Oliva (predicho)	Pera (predicho)
Manzana (real)	**MM**	**MN**	**MO**	**MP**
Naranja (real)	**NM**	**NN**	**NO**	**NP**
Oliva (real)	**OM**	**ON**	**OO**	**OP**
Pera (real)	**PM**	**PN**	**PO**	**PP**

Las medidas precision, recall, etc. solo pueden formularse en forma binaria: cada clase contra el resto.

$$\text{Precision}\left(\text{Manzana}\right)=\frac{\text{MM}}{\text{MM}+\text{NM}+\text{OM}+\text{PM}}$$

$$\text{Recall}\left(\text{Manzana}\right)=\frac{\text{MM}}{\text{MM}+\text{MN}+\text{MO}+\text{MP}}$$

Resumen

- Sobreajuste.
- Validación cruzada.
- Datos de entrenamiento, validación, test.
- Exactitud (*accuracy*).
- Matriz de confusión, precision, recall.
- Sensibilidad y especificidad.
- Curva ROC.