

Introducción al Aprendizaje automático

Dra Ana Georgina Flesia

Optativa Ciencias de la Computación
FaMAF-UNC
Oficina 370
georgina.flesia@unc.edu.ar

2020

Aprendizaje supervisado

Objetivo central del aprendizaje supervisado:

Predicción, no causalidad

El foco está puesto en poder realizar predicciones de interés bajo condiciones complejas, **no en estudiar el mecanismo causal que rige bajo esas condiciones**

Aprendizaje supervisado

Suposiciones fundamentales del aprendizaje supervisado:

Las características determinan la clase y el conjunto de ejemplos etiquetados provisto es lo suficientemente grande como para soportar la asociación

Métodos de Evaluación

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
- ▶ Conjunto de validación

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
- ▶ Conjunto de test

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
- ▶ Conjunto de test (cálculo de las métricas de evaluación)

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
- ▶ Conjunto de test (cálculo de las métricas de evaluación)

El **conjunto de test debe** ser completamente independiente de los utilizados en la construcción de los clasificadores.

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
- ▶ Conjunto de test (cálculo de las métricas de evaluación)

El **conjunto de test debe** ser completamente independiente de los utilizados en la construcción de los clasificadores.

Puede ser incluso de distinta naturaleza!

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
85 %
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
- ▶ Conjunto de test (cálculo de las métricas de evaluación)

El **conjunto de test debe** ser completamente independiente de los utilizados en la construcción de los clasificadores.

Puede ser incluso de distinta naturaleza!

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ Conjunto de entrenamiento (construcción del clasificador)
85 %
- ▶ Conjunto de validación (tuning fino o elección del clasificador)
5 %
- ▶ Conjunto de test (cálculo de las métricas de evaluación)

El **conjunto de test debe** ser completamente independiente de los utilizados en la construcción de los clasificadores.

Puede ser incluso de distinta naturaleza!

Métodos de Evaluación

El conjunto de ejemplos etiquetados se divide en tres:

- ▶ **Conjunto de entrenamiento** (construcción del clasificador)
85 %
- ▶ **Conjunto de validación** (tuning fino o elección del clasificador)
5 %
- ▶ **Conjunto de test** (cálculo de las métricas de evaluación)
10 %

El **conjunto de test debe** ser completamente independiente de los utilizados en la construcción de los clasificadores.

Puede ser incluso de distinta naturaleza!



Métodos de Evaluación

Mientras más grande sea el conjunto de entrenamiento mejor resulta el clasificador

Métodos de Evaluación

Mientras más grande sea el conjunto de entrenamiento mejor resulta el clasificador (en la medida que su capacidad de generalización no se deteriore)

Métodos de Evaluación

Mientras más grande sea el conjunto de entrenamiento mejor resulta el clasificador (en la medida que su capacidad de generalización no se deteriore)

Mientras más grande resulta el conjunto de test más robusta (estadísticamente) resulta el cálculo de las métricas

Métodos de Evaluación

Mientras más grande sea el conjunto de entrenamiento mejor resulta el clasificador (en la medida que su capacidad de generalización no se deteriore)

Mientras más grande resulta el conjunto de test más robusta (estadísticamente) resulta el cálculo de las métricas

Concluído el cálculo de las métricas, es lícito fusionar el conjunto de test con el de entrenamiento para re-entrenar el clasificador

Métodos de Evaluación

Método del *holdout*

Métodos de Evaluación

Método del *holdout*

Si no hay restricción en el número de ejemplos disponibles, es posible retener el conjunto de test separado del resto

Métodos de Evaluación

Método del *holdout*

Si no hay restricción en el número de ejemplos disponibles, es posible retener el conjunto de test separado del resto

Ejemplo: Competiciones Klagge

Métodos de Evaluación

Método del *holdout*

Si no hay restricción en el número de ejemplos disponibles, es posible retener el conjunto de test separado del resto

Ejemplo: Competiciones Klagge

La elección del holdout debe realizarse *aleatorizando* los ejemplos para que los conjuntos de entrenamiento y test sean igualmente representativos → **estratificación**



Métodos de Evaluación

Método de validación cruzada (*cross-validation*)

Métodos de Evaluación

Método de validación cruzada (*cross-validation*)

Cuando el número de ejemplos es limitado la solución es repetir el proceso varias veces seleccionando un conjunto de test nuevo y diferente en cada iteración

Métodos de Evaluación

Método de validación cruzada (*cross-validation*)

Cuando el número de ejemplos es limitado la solución es repetir el proceso varias veces seleccionando un conjunto de test nuevo y diferente en cada iteración

En cada iteración se calculan las métricas de evaluación y se promedian al final

Métodos de Evaluación

Método de validación cruzada (*cross-validation*)

Cuando el número de ejemplos es limitado la solución es repetir el proceso varias veces seleccionando un conjunto de test nuevo y diferente en cada iteración

En cada iteración se calculan las métricas de evaluación y se promedian al final

El procedimiento estándar usual consiste en repetir el proceso 10 veces → **tenfold cross-validation**

Métodos de Evaluación

Método de validación cruzada (*cross-validation*)

Cuando el número de ejemplos es limitado la solución es repetir el proceso varias veces seleccionando un conjunto de test nuevo y diferente en cada iteración

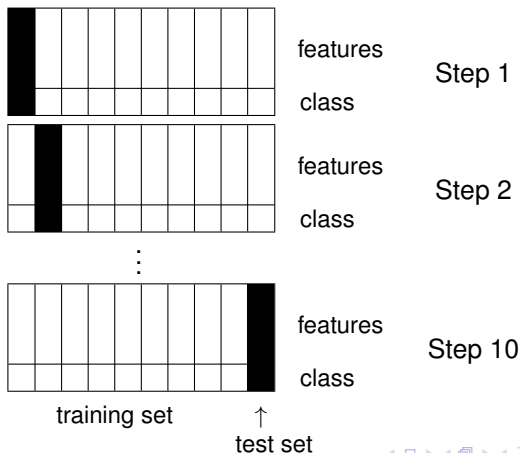
En cada iteración se calculan las métricas de evaluación y se promedian al final

El procedimiento estándar usual consiste en repetir el proceso 10 veces → **tenfold cross-validation**

¿Por qué 10? Lo dicta la experiencia con diferentes técnicas y datasets, pero es una elección arbitraria

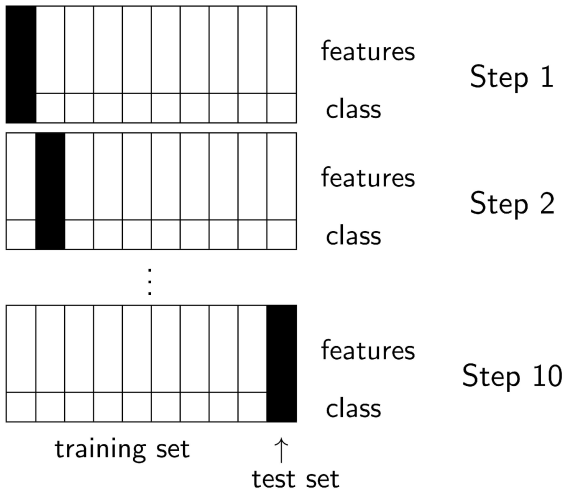
Métodos de Evaluación

Stratified tenfold cross-validation



Métricas para Evaluación

Conceptos de Precisión y Exactitud



Conceptos de Precisión y Exactitud

Métricas para variables numéricas:

- ▶ Parámetro a estimar: μ

Conceptos de Precisión y Exactitud

Métricas para variables numéricas:

- ▶ Parámetro a estimar: μ
- ▶ Promedio: $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
- ▶ Exactitud: $|\bar{x} - \mu|$
- ▶ Precisión: S (medida muestral)

Conceptos de Precisión y Exactitud

Métricas para variables numéricas:

▶ **Parámetro a estimar:** μ

▶ **Promedio:** $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$

▶ **Desviación estándar:** $S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$

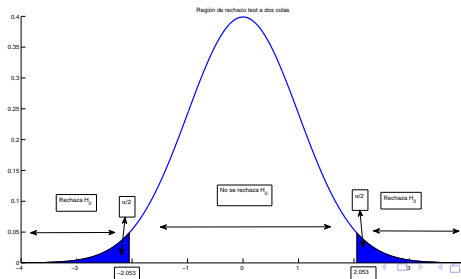
▶ **Exactitud:** $|\bar{x} - \mu|$

▶ **Precisión:** S (medida muestral)

Conceptos de Precisión y Exactitud

- Intervalo de confianza $(1 - \alpha)$:

$$\mu = \bar{x} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$$



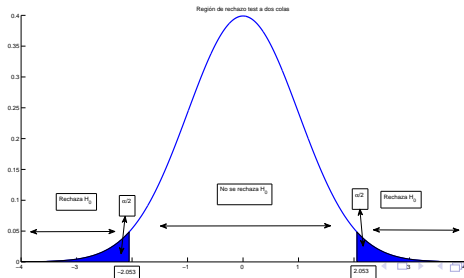
Conceptos de Precisión y Exactitud

- Intervalo de confianza $(1 - \alpha)$:

$$\mu = \bar{x} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$$

- Precisión: semiancho o error del intervalo:

$$z_{\alpha/2} \frac{\sigma}{\sqrt{n}} \approx z_{\alpha/2} \frac{S}{\sqrt{n}} \text{ (muestras grandes)}$$



Métricas para clasificación

Performance de clasificación

La performance de un clasificador *siempre* se mide sobre un conjunto de prueba (test) independiente al de entrenamiento

Métricas para clasificación

Performance de clasificación

La performance de un clasificador *siempre* se mide sobre un conjunto de prueba (test) independiente al de entrenamiento

- ▶ n : número de pruebas realizadas (ejemplos utilizados)
- ▶ s : número de aciertos obtenidos

Métricas para clasificación

Performance de clasificación

La performance de un clasificador *siempre* se mide sobre un conjunto de prueba (test) independiente al de entrenamiento

- ▶ n : número de pruebas realizadas (ejemplos utilizados)
- ▶ s : número de aciertos obtenidos
- ▶ f : número de errores cometidos ($f = n - s$)
- ▶ $\hat{p} = s/n$: **proporción de aciertos**
- ▶ $\epsilon = 1 - \hat{p} = f/n$: **error de clasificación**
- ▶ $\epsilon \% = \epsilon \times 100$: error porcentual

Métricas para clasificación

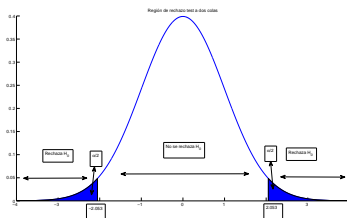
Exactitud: Proporción de aciertos

Métricas para clasificación

Exactitud: Proporción de aciertos

Intervalo de confianza $(1 - \alpha)$:

$$p = \frac{\hat{p} + \frac{z_{\alpha/2}^2}{2n} \pm \sqrt{\hat{p}(1 - \hat{p}) \frac{z_{\alpha/2}^2}{n} + \frac{z_{\alpha/2}^4}{4n^2}}}{1 + \frac{z_{\alpha/2}^2}{n}}$$



Aprox. normal:

$$n \hat{p} > 10$$

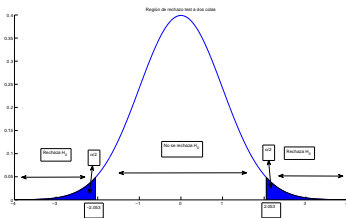
$$n \hat{q} > 10$$

Métricas para clasificación

Exactitud: Proporción de aciertos

Intervalo de confianza $(1 - \alpha)$:

$$p = \frac{\hat{p} + \frac{z_{\alpha/2}^2}{2n} \pm \sqrt{\hat{p}(1 - \hat{p}) \frac{z_{\alpha/2}^2}{n} + \frac{z_{\alpha/2}^4}{4n^2}}}{1 + \frac{z_{\alpha/2}^2}{n}}$$



Aprox. normal:

$$n \hat{p} > 10$$

$$n \hat{q} > 10$$

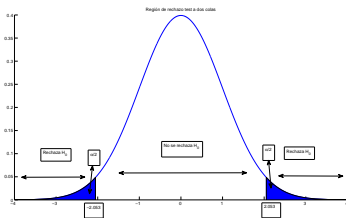
Precisión: semiancho del intervalo

Métricas para clasificación

Exactitud: Proporción de aciertos

Intervalo de confianza $(1 - \alpha)$:

$$p = \frac{\hat{p} + \frac{z_{\alpha/2}^2}{2n} \pm \sqrt{\hat{p}(1 - \hat{p}) \frac{z_{\alpha/2}^2}{n} + \frac{z_{\alpha/2}^4}{4n^2}}}{1 + \frac{z_{\alpha/2}^2}{n}}$$



Aprox. normal:

$$n\hat{p} > 10$$

$$n\hat{q} > 10$$

Precisión: semiancho del intervalo

La precisión depende de la exactitud !

Métricas para clasificación binaria

Métricas para clasificación binaria:

- ▶ TP : Positivos verdaderos

Métricas para clasificación binaria

Métricas para clasificación binaria:

- ▶ TP : Positivos verdaderos
- ▶ FP : Falsos positivos

Matriz de confusión:

		clasificado	
		+	-
etiqueta	+	$\#TP$	$\#FN$
	-	$\#FP$	$\#TN$

Métricas para clasificación binaria

Métricas para clasificación binaria:

- ▶ TP : Positivos verdaderos
- ▶ FP : Falsos positivos
- ▶ TN : Negativos verdaderos

Matriz de confusión:

		clasificado	
		+	-
etiqueta	+	$\#TP$	$\#FN$
	-	$\#FP$	$\#TN$

La matriz de confusión se extiende naturalmente a la clasificación **multiclase**

Métricas para clasificación binaria

Métricas para clasificación binaria:

- ▶ TP : Positivos verdaderos
- ▶ FP : Falsos positivos
- ▶ TN : Negativos verdaderos
- ▶ FN : Falsos negativos

Matriz de confusión:

		clasificado	
		+	-
etiqueta	+	$\#TP$	$\#FN$
	-	$\#FP$	$\#TN$

La matriz de confusión se extiende naturalmente a la clasificación **multiclase**

Métricas para clasificación binaria

- Accuracy:
$$\frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} (= \hat{p})$$

Métricas para clasificación binaria

- Accuracy: $\frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} (= \hat{p})$
- Recall: $\frac{\#TP}{\#TP + \#FN}$

Métricas para clasificación binaria

- Accuracy: $\frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} (= \hat{p})$
- Recall: $\frac{\#TP}{\#TP + \#FN}$
- Precision: $\frac{\#TP}{\#TP + \#FP}$

Métricas para clasificación binaria

- Accuracy: $\frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} (= \hat{p})$
- Recall: $\frac{\#TP}{\#TP + \#FN}$
- Precision: $\frac{\#TP}{\#TP + \#FP}$
- Sensitivity: $\frac{\#TP}{\#TP + \#FP}$

Métricas para clasificación binaria

- Accuracy: $\frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} (= \hat{p})$

- Recall: $\frac{\#TP}{\#TP + \#FN}$

- Precision: $\frac{\#TP}{\#TP + \#FP}$

- Sensitivity: $\frac{\#TP}{\#TP + \#FP}$

- Specificity: $\frac{\#TN}{\#TN + \#FN}$

Métricas para clasificación binaria

Curva Lift

Marketing a una base de 1000000 de clientes donde se sabe que la proporción de respuesta es del 0,1 % (1000).

Métricas para clasificación binaria

Curva Lift

Marketing a una base de 1000000 de clientes donde se sabe que la proporción de respuesta es del 0,1 % (1000).

Un algoritmo de ML identifica un conjunto de 100000 clientes para los cuales la proporción de respuesta es del 0,4 % (400).

Lift factor = 4

Métricas para clasificación binaria

Curva Lift

Marketing a una base de 1000000 de clientes donde se sabe que la proporción de respuesta es del 0,1 % (1000).

Un algoritmo de ML identifica un conjunto de 100000 clientes para los cuales la proporción de respuesta es del 0,4 % (400).

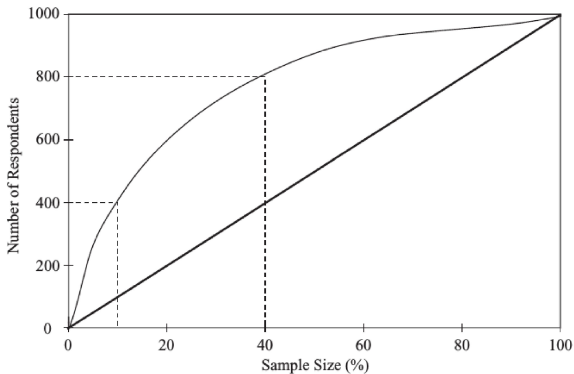
Lift factor = 4

En algoritmo de ML identifica un conjunto de 400000 clientes para los cuales la proporción de respuesta es del 0,2 % (800).

Lift factor = 2

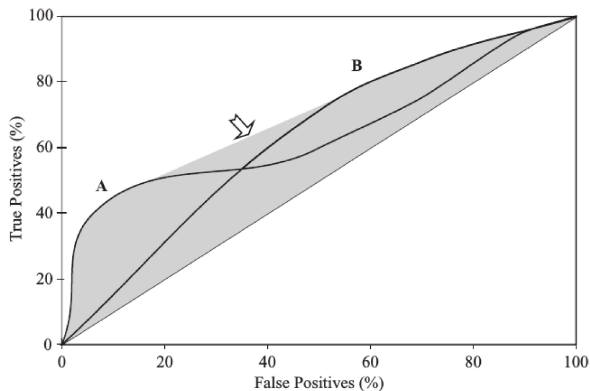
Métricas para clasificación binaria

Curva Lift



Métricas para clasificación binaria

Curva ROC (receiver operating characteristic)



Métricas para clasificación binaria

Curvas de evaluación FP y FN

	Dominio	Ejes (y vs x)
Lift chart	Marketing	$\#TP$ vs proporción de P
ROC curve	Communication	Tasa de TP vs Tasa de FP
Recall-Precision	Information Retrival	Recall vs Precision

- Proporción de P :
$$\frac{\#TP + \#FP}{\#TP + \#TN + \#FP + \#FN}$$
- Tasa de TP :
$$\frac{\#TP}{\#TP + \#FN}$$
- Tasa de FP :
$$\frac{\#FP}{\#FP + \#TN}$$
- Recall:
$$\frac{\#TP}{\#TP + \#FN}$$
- Precision:
$$\frac{\#TP}{\#TP + \#FP}$$

Métricas para clasificación

Coeficiente κ de Cohen

Proporciona una medida de la *reproducibilidad* de la clasificación

Métricas para clasificación

Coeficiente κ de Cohen

Proporciona una medida de la *reproducibilidad* de la clasificación

Dados dos clasificadores el coeficiente κ es la probabilidad condicional de acuerdo bajo la condición de no independencia, es decir que clasifican de forma correlacionada:

$$\kappa = P(\text{acuerdo} \mid \text{no independencia}) = \frac{P(\text{acuerdo} \cap \neg \text{independencia})}{P(\neg \text{independencia})}$$

Métricas para clasificación

Coeficiente κ de Cohen

$$\kappa = \frac{P(\text{acuerdo}) - P(\text{acuerdo} \cap \text{independencia})}{1 - P(\text{acuerdo} \cap \text{independencia})}$$

Métricas para clasificación

Coeficiente κ de Cohen

$$\kappa = \frac{P(\text{acuerdo}) - P(\text{acuerdo} \cap \text{independencia})}{1 - P(\text{acuerdo} \cap \text{independencia})}$$

$$\kappa = \begin{cases} 1 & \text{si hay completo acuerdo} \\ P(\text{acuerdo}) & \text{si } (\text{acuerdo} \cap \text{independencia}) = \emptyset \\ 0 & \text{si hay completo desacuerdo} \\ & \text{o bien independencia perfecta} \end{cases}$$

Métricas para clasificación

Coeficiente κ de Cohen

Implementación muestral:

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

p_o : proporción de acuerdo observado

p_e : proporción de acuerdo independiente

Métricas para clasificación

Coeficiente κ de Cohen

Implementación muestral:

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

p_o : proporción de acuerdo observado
 p_e : proporción de acuerdo independiente

Ejemplo (clasificación binaria):

		B	
		sí	no
A	sí	20	5
	no	10	15

Métricas para clasificación

Coeficiente κ de Cohen

Implementación muestral:

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

p_o : proporción de acuerdo observado
 p_e : proporción de acuerdo independiente

Ejemplo (clasificación binaria):

		<i>B</i>	
		sí	no
<i>A</i>	sí	20	5
	no	10	15

$$P_A(\text{sí}) = 25/50 = 0,5 \quad P_A(\text{no}) = 25/50 = 0,5$$

$$P_B(\text{sí}) = 30/50 = 0,6 \quad P_B(\text{no}) = 20/50 = 0,4$$

$$p_o = 35/50 = 0,7$$

$$p_e = 0,5 \times 0,6 + 0,5 \times 0,4 = 0,5$$

$$\kappa = \frac{0,7 - 0,5}{1 - 0,5} = 0,4$$

Métricas para clasificación

Coeficiente κ de Cohen

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

p_o : proporción de acuerdo observado
 p_e : proporción de acuerdo independiente

Interpretación usual:

κ	confiabilidad en el acuerdo
$[0,0, 0,2)$	muy baja
$[0,2, 0,4)$	baja
$[0,4, 0,6)$	moderada
$[0,6, 0,8)$	buena
$[0,8, 1,0]$	muy buena

Métricas para clasificación

Coeficiente κ de Cohen

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

p_o : proporción de acuerdo observado
 p_e : proporción de acuerdo independiente

Interpretación usual:

κ	confiabilidad en el acuerdo
$[0,0, 0,2)$	muy baja
$[0,2, 0,4)$	baja
$[0,4, 0,6)$	moderada
$[0,6, 0,8)$	buena
$[0,8, 1,0]$	muy buena

Numéricamente para $p_e \approx p_o$ puede resultar $\kappa < 0$
y se corresponde con un clasificador nada confiable.

Métricas para clasificación

La navaja de Occam

“Las entidades no deben multiplicarse más allá de lo necesario”



William of Ockham
(~ 1287 – 1347)

Métricas para clasificación

La navaja de Occam

“Las entidades no deben multiplicarse más allá de lo necesario”

Principio:

Entre todas las hipótesis que predicen igualmente bien un fenómeno debe elegirse aquella que tiene el menor número de suposiciones; es decir, la más simple



William of Ockham
(~ 1287 – 1347)

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T
- ▶ $L[T] + L[E|T]$: longitud de descripción (DL)

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T
- ▶ $L[T] + L[E|T]$: longitud de descripción (DL)
- ▶ (tamaño del modelo más espacio para codificar los errores)

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T
- ▶ $L[T] + L[E|T]$: longitud de descripción (DL)
- ▶ (tamaño del modelo más espacio para codificar los errores)
- ▶ MDL establece minimizar la suma

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T
- ▶ $L[T] + L[E|T]$: longitud de descripción (DL)
- ▶ (tamaño del modelo más espacio para codificar los errores)
- ▶ MDL establece minimizar la suma
- ▶ MDL es un **método de selección de modelo**

Métricas para clasificación

Mínima Longitud de Descripción (MDL)

- ▶ Un esquema de aprendizaje implica una teoría sobre los datos
- ▶ $L[T]$: longitud de codificación de la teoría
- ▶ E set de ejemplos de entrenamiento
- ▶ $L[E|T]$: longitud de codificación de E respecto de T
- ▶ $L[T] + L[E|T]$: longitud de descripción (DL)
- ▶ (tamaño del modelo más espacio para codificar los errores)
- ▶ MDL establece minimizar la suma
- ▶ MDL es un **método de selección de modelo**
- ▶ se implementa con un término de costo por complejidad