

# Métricas de WEKA

## Matriz de Confusión

Herramienta que permite visualizar el desempeño del modelo. Empleando como dimensiones los resultados Verdaderos Positivos (TP) y Negativos (TN) en la diagonal principal y los Falsos Positivos (FP) y Negativos (FN).

		PREDICCIÓN	
		POSITIVO	NEGATIVO
CLASE REAL	POSITIVO	True Positives (TP)	False Negatives (FN)
	NEGATIVO	False Positives (FP)	True Negatives (TN)

## Exactitud (Accuracy)

Porcentaje de predicciones correctas. En conjuntos de datos poco equilibrados, no es una métrica útil. Si buscamos predecir una enfermedad rara y nuestro algoritmo clasifica a todos los individuos como sanos, puede ser muy preciso (99%) pero es inútil.

$$\text{Exactitud} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FN} + \text{FP} + \text{TN})$$

## True Positive Rate (TPR), Recall o Sensibility o Cobertura

Capacidad del modelo para predecir casos positivos o relevantes. Predicciones positivas correctas entre el nro total de positivos.

$$\text{TPR} = (\text{TP}) / (\text{TP} + \text{FN})$$

## True Negative Rate (TNR), Specify o Especificidad

Capacidad del modelo de discriminar correctamente los casos negativos.

$$\text{TNR} = (\text{TN}) / (\text{TN} + \text{FP})$$

## False Positive Rate (FPR)

Probabilidad de que sea una falsa alarma. Predicciones positivas incorrectas en el nro total de negativos.

$$\text{FPR} = (\text{FP}) / (\text{FP} + \text{TN})$$

## False Omission Rate (FOR)

*Predicciones negativas incorrectas entre nro total de predicciones negativas.*

$$FOR = (FN)/(FN + TN)$$

## Precisión

*Porcentaje de predicciones positivas correctas.*

$$TP = (TP)/(TP + FP)$$

## F-mesure

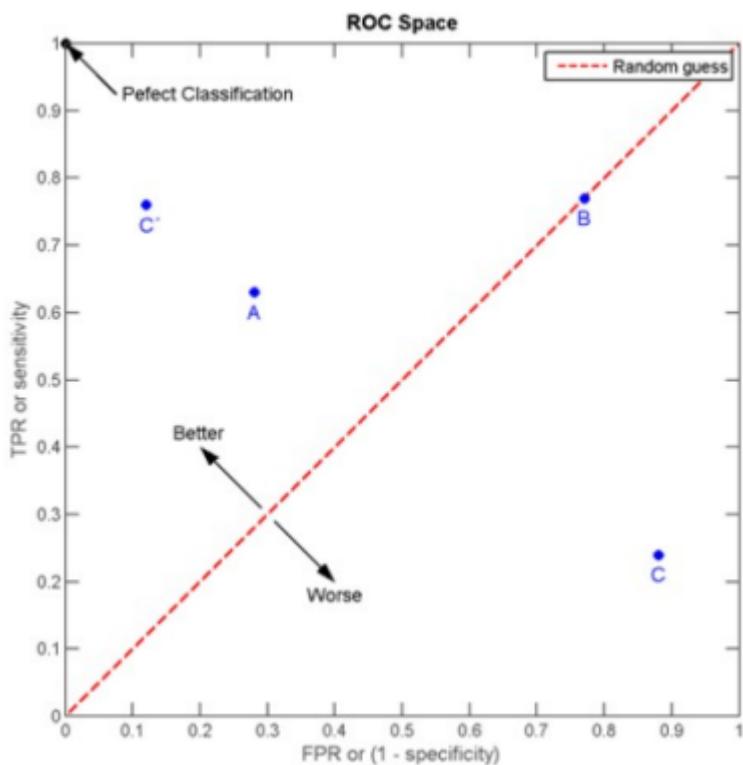
*Permite caracterizar con un único valor la bondad de un clasificador o algoritmo.*

$$F\text{-mesure} = 2 * \text{Precision} * \text{Cobertura} / (\text{Precision} + \text{Cobertura})$$

## Receiver Operating Characteristic Curve (Curva ROC)

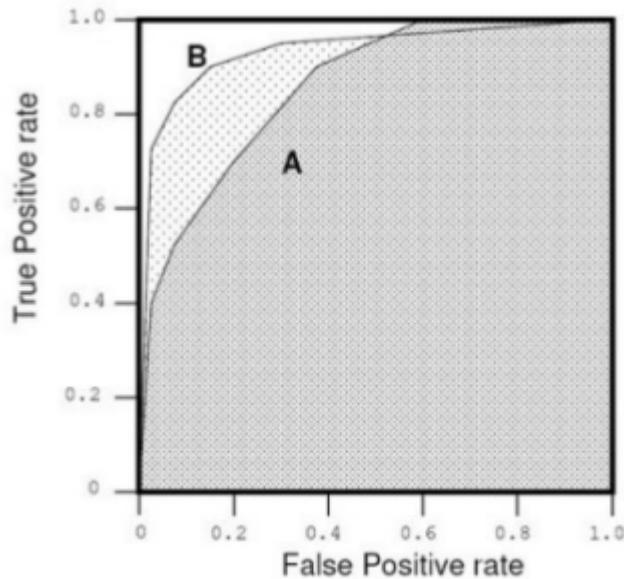
*Mientras más cercano a 1, mejor. En problemas complejos, un clasificador aumentará el nro de TP a costa de incrementar también el FP. Se busca un clasificador que sea capaz de incrementar TP a un ritmo (mucho) mayor que FP.*

*Los gráficos son bidireccionales con X = FPR e Y = TPR. Un gráfico ROC muestra el compromiso entre los beneficio y coste.*



## Area Under the Curve (AUC) o ROC Area

Permite representar en un único valor el rendimiento de un clasificador. Esto resulta útil para comparar clasificadores.



## Coeficiente de Correlación de Matthews (MCC)

Informa la calidad de la clasificación de las clases. Se tienen en cuenta los valores  $TP$ ,  $FP$  y  $FN$ , es considerada una medida equilibrada.

$$MCC = \frac{(TP * TN) + (FP * FN)}{\sqrt{(FP + TP)(TP + FN)(TN + FP)(TN + FN)}}$$

## Área PRC (Precisión vs Recall)

Dice como es el comportamiento de cada clase. De igual manera que la Curva de ROC, mientras más cercano a 1, mejor el comportamiento. Indica en qué porcentaje los clasificadores se comportan de manera correcta.

## Ejemplo de una salida WEKA

--- Detailed Accuracy By Class ---

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	drugA
	1,000	0,005	0,941	1,000	0,970	0,968	1,000	0,996	drugB
	1,000	0,005	0,941	1,000	0,970	0,968	0,998	0,952	drugC
	0,963	0,014	0,963	0,963	0,963	0,949	0,994	0,980	drugX
	0,956	0,018	0,978	0,956	0,967	0,940	0,993	0,987	drugY
Weighted Avg.	0,970	0,013	0,970	0,970	0,970	0,954	0,995	0,985	

--- Confusion Matrix ---

a	b	c	d	e	<-- classified as
23	0	0	0	0	a = drugA
0	16	0	0	0	b = drugB
0	0	16	0	0	c = drugC
0	0	0	52	2	d = drugX
0	1	1	2	87	e = drugY

## Clustering con WEKA

Aprendizaje no supervisado.

## Aplicaciones

- Segmentación de mercado.
- Análisis de redes sociales.
- Análisis de datos astronómicos.

## Algoritmos de clasificación

- Kmeans.
- Coweb.
- EM.

## Kmeans

- Agrupa objetos en  $k$  grupos (clusters) basándose en sus características.
- Agrupamiento minimizando la suma de distancias entre cada objeto y el *centroide* (semilla) del cluster.
- Sólo datos *numéricos*.
- Distancia *cuadrática*.
- Requiere de *normalización*.
- Admite *ruido*.
- Agrupaciones *fijas y disjuntas*.
- Depende mucho de las *secciones*.
- Basado en los *grafos o diagramas de Voronoi*.

nota: No usar con  $k>25$  (recomendación profesor)

## Grafos de Voronoi (Paso a paso)

1. Comenzamos con los datos.
2. Plantamos la/s semilla/s.
3. Asignamos cada dato a la semilla más cercana.
4. Desplazamos las semillas al centro de cada grupo.

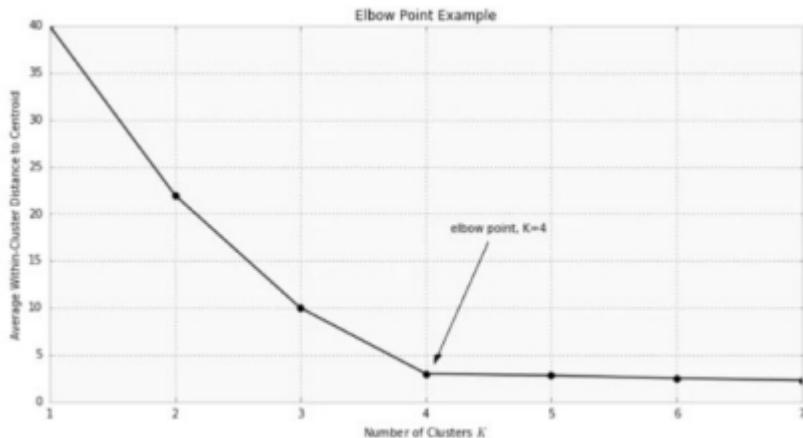
### Formula:

$$\min_{\mathbf{S}} E(\boldsymbol{\mu}_i) = \min_{\mathbf{S}} \sum_{i=1}^k \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

### Clusters

*La cantidad de clusters se determina usando el Método del Codo.*

- Utiliza la distancia media de las observaciones a su centroide.
- Cuanto más grande es el nro. de cluster, la varianza intra-cluster tiende a disminuir.
- Cuanto menor es la distancia intra-cluster mejor, ya que significa clusters más compactos.
- Busca el valor K que satisfaga que un incremento de K, no mejore sustancialmente la distancia media intra-cluster.



### Coweb

- Algoritmo de clustering jerárquico.
- Utiliza *aprendizaje incremental*, es decir, realiza las agrupaciones instancia a instancia.
- Utiliza la media llamada *category utility* para guiar el proceso de aprendizaje y determinar la pertenencia de cada instancia a un grupo. Valores altos = Alta similitud entre elementos y una baja similitud entre elementos de grupos diferentes.

- Durante la ejecución del algoritmo se forma un árbol (de clasificación) donde las hojas representan los clusters y el nodo raíz engloba todos los datos de entrada. Al principio consiste en un único nodo raíz, las instancias se van agregando una a una y el árbol se va actualizando en cada paso.
- La *actualización* consiste en encontrar el mejor sitio donde incluir la nueva instancia, operación que puede requerir una *reconstrucción* de todo el árbol.
- El algoritmo es sensible a dos parámetros: **Acuity** (Medida de error de un nodo, permite controlar el factor de ramificación) y **Cut-off** (Este valor se utiliza para evitar el crecimiento desmesurado del nro. de clusters).

## EM

- Mejora Kmeans con *gaussianas*.
- Permite un nro. de clases *NO* fija.
- Agrupaciones *NO* disjuntas.
- Calcula *semillas con estadística*.
- Sólo valores *numéricos*.
- Requiere *normalización*.

## Recomendaciones

Normalización	Escalado
Va a transformar las características de forma que todas comparten un mismo valor medio y una misma desviación media.	Va a transformar los valores de las características de forma que estén confinados en un rango [a, b], típicamente [0, 1] o [-1, 1].

## Normalización

*Los valores de cada atributo estén en escalas similares.*

- Es una buena idea al usar algoritmos de clustering.
- Ayuda al clustering porque los grupos se forman a partir de distancias.
- Si hay atributos con escalas muy diferentes, los atributos de escala mayor dominarán las distancias.

## Escalado

### Escalado de variables (Feature Scaling o MinMax Scaler)

*Se normaliza entre límites definidos.*

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- No es un método adecuado para datos estables.

## Escalado Estándar (Standard Scaler)

*A cada datos se le resta la media y se lo divide por la desviación típica.*

$$X_{normalized} = \frac{X - X_{mean}}{X_{stddev}}$$

- Método no adecuado para cuando los datos presentan variaciones como picos.

## Campana de gauss

- Desviacion estandar: Se ubica en el punto de inflexion.
- Media: Se encuentra en el centro.
- Intersección de coordenadas: Grado de pertenencia. Mientras mas alta sea Y, mas alto es el grado de pertenencia al conjunto de datos.
- (pregunta final) Matriz de peso (Wk): Representa conocimiento. Se calcula con la probabilidad de pertenencia del dato en el conjunto.

## EM

- Esperanza: Calculo del grado de pertenencia al conjunto.
- Maximización: Se recalcula la media para que se ubique en el centro del conjunto de datos.