Introducción a IA

Evaluación de Aprendizaje Supervisado

Evaluación

Aprendizaje Supervisado

Evaluación de métodos de aprendizaje supervisado:

- Al desarrollar un clasificador/regresor como parte de algún sistema de toma de decisiones, es crítico evaluar el desempeño
- La evaluación dará evidencia para anticipar el funcionamiento del sistema en producción
- La evaluación es también necesaria para determinar el mejor método de aprendizaje y/o optimización de parámetros

Evaluación

Experimentación

Pasos de la evaluación:

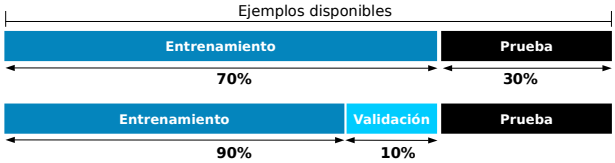
- Obtener un conjunto de datos
- Particionar los datos
- Elegir las métricas de evaluación
- Realizar los experimentos
- Interpretar y analizar los resultados

Evaluación

Particionamiento

Particionamiento: los datos disponibles se dividen en tres subconjuntos

- **Entrenamiento**: se utiliza para aprender el modelo
- **Validación**: se utiliza para calibración y ajuste de parámetros
- **Prueba**: se utilizar para testear el modelo aprendido



Evaluación

Métricas

Matriz de confusión o tabla de contingencia: útil para estimar medidas de evaluación de la clasificación

		Clase Predicha	
	Clase Real	**Positiva**	**Negativa**
Ground Truth	**Positiva**	Verdaderos Positivos (TP)	Falsos Negativos (FN)
	Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)

Evaluación

Métricas

Matriz de confusión o tabla de contingencia: útil para estimar medidas de evaluación de la clasificación

	Clase Predicha	
Clase Real	**Positiva**	**Negativa**
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)

Error tipo II

Error tipo I

Evaluación

Métricas

Matriz de confusión o tabla de contingencia: útil para estimar medidas de evaluación de la clasificación

	Clase Predicha	
Clase Real	**Spam**	**No Spam**
Spam	Verdaderos Positivos (TP)	Falsos Negativos (FN)
No Spam	Falsos Positivos (FP)	Verdaderos Negativos (TN)

Error tipo II

Error tipo I

Evaluación

Métricas

	Clase Predicha		
Clase Real	**Positiva**	**Negativa**	
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	
			Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

	Clase Predicha		
Clase Real	**Positiva**	**Negativa**	
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	
Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$			Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

Accuracy (Exactitud): mide la cantidad de veces que el modelo ha acertado en los ejemplos de prueba

- En conjuntos desbalanceados (como intrusión en redes o detección de fraude) donde hay una clase minoritaria
  → Alto valor de accuracy no implica detectar la clase minoritaria (si hay 1% de intrusión, se puede alcanzar 99% de accuracy)

Evaluación

Métricas

	Clase Predicha		
Clase Real	**Positiva**	**Negativa**	
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	
Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$		Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

	Clase Predicha		
Clase Real	**Positiva**	**Negativa**	
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	Recall $\frac{TP}{TP+FN}$
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	
Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$	NPV $\frac{TN}{FN+TN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

Precision (Precisión): mide del total de predicciones positivas, cuántas fueron correctas

→ número de ejemplos positivos correctamente clasificados del total de ejemplos clasificados como positivos

Recall (Cobertura o Sensitivity): mide del total de ejemplos positivos, cuántos fueron correctamente clasificados

→ número de ejemplos positivos correctamente clasificados del total de ejemplos positivos en el conjunto de prueba

1	99
0	1000

Precision = 100%
Recall = 1%

Porque solo se clasificó 1 ejemplo como positivo correctamente y ninguno negativo incorrectamente

Evaluación

Métricas

Precision y recall evalúan la clasificación sobre la clase positiva, pero es dificil comparar dos clasificadores usando dos medidas separadas

F_1-Score (o F-Measure): combina precision y recall en una única métrica, mejora cuando ambas son altas

$$F_1 - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Evaluación

Métricas

	Clase Predicha: Positiva	Clase Predicha: Negativa	
Clase Real: Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	
Clase Real: Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	
	Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$	NPV $\frac{TN}{FN+TN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Negative Predictive Value (NPV)

Evaluación

Métricas

	Clase Predicha: Positiva	Clase Predicha: Negativa	
Clase Real: Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	Recall $\frac{TP}{TP+FN}$
Clase Real: Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	Specificity $\frac{TN}{FP+TN}$
	Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$	NPV $\frac{TN}{FN+TN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

	Clase Predicha		
Clase Real	**Positiva**	**Negativa**	
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	Recall $\frac{TP}{TP+FN}$
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	Specificity $\frac{TN}{FP+TN}$
Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$	NPV $\frac{TN}{FN+TN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Evaluación

Métricas

	Clase Predicha			
Clase Real	**Positiva**	**Negativa**		
Positiva	Verdaderos Positivos (TP)	Falsos Negativos (FN)	Recall $\frac{TP}{TP+FN}$	True Positive Rate (TPR)
Negativa	Falsos Positivos (FP)	Verdaderos Negativos (TN)	Specificity $\frac{TN}{FP+TN}$	False Positive Rate (FPR) $\frac{FP}{FP+TN}$
Error Rate $\frac{FN+FP}{TP+TN+FP+FN}$	Precisión $\frac{TP}{TP+FP}$	NPV $\frac{TN}{FN+TN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$	

Evaluación

Métricas

En un problema multi-clase se calcula la performance del clasificador como un todo, tomando los promedios por clase

	Clase Predicha				
Clase Real	**Clase 1**	**Clase 2**	**Clase 3**	**...**	**Clase N**
Clase 1					
Clase 2					
Clase 3					
...					
Clase N					

Evaluación

Métricas

Macro-averaging: calcula el valor de la métrica de evaluación para cada clase y los promedia (igual peso a todas las clases)

	Clase Predicha				
Clase Real	**Clase 1**	**Clase 2**	**Clase 3**	**...**	**Clase N**
Clase 1	TP	FN			
Clase 2	FP	TN			
Clase 3					
...					
Clase N					

Evaluación

Métricas

Macro-averaging: calcula el valor de la métrica de evaluación para cada clase y los promedia (igual peso a todas las clases)

$$P_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_i}{TP_i + FP_i} = \frac{\sum_{i=1}^{|C|} P_i}{|C|}$$

Clase 1

10	10
10	970

Clase 2

90	10
10	890

$P_{clase\ 1}$=0.50

$P_{clase\ 2}$=0.90

P_{macro}=0.70

Evaluación

Métricas

Micro-averaging: suma las decisiones de todas las clases, calcula la tabla de contingencia total y calcula la métrica sobre ella

→ está dominada por los resultados de las clases más populares

$$P_{micro} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} TP_i + FP_i}$$

Clase 1

10	10
10	970

Clase 2

90	10
10	890

Micro-average

100	20
20	1860

P_{micro}=100/(100+20) =0.83

Evaluación

Métricas

Medidas comúnmente usadas para evaluación de predicciones cuando la clase es continua:

- MAE (Mean Absolute Error): desviación de las predicciones de los valores verdaderos

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - r_i|$$

- RMSE (Root Mean Square Error): similar a MAE pero pone más énfasis en las desviaciones

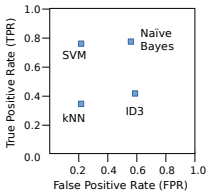
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - r_i)^2}$$

Evaluación

Métricas

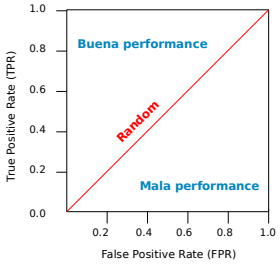
Curvas ROC (Receive Operating Characteristics): enfoque gráfico que muestra el trade-off entre la taza de detección y la de falsa alarma

- True Positive Rates (TPR) y False Positive Rates (FPR)



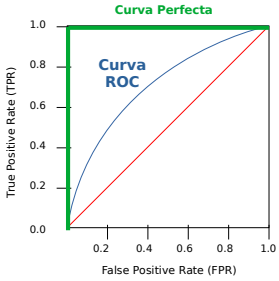
Evaluación

Métricas



Evaluación

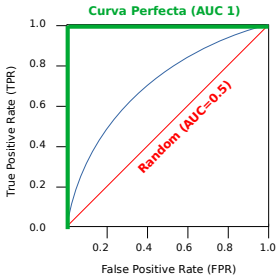
Métricas



Curva ROC: para un umbral dado sobre $\hat{f}(x)$ se obtiene un punto de la curva ROC

Evaluación

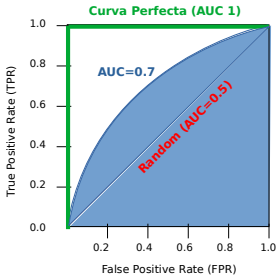
Métricas



Area Under Curve (AUC): es el área bajo la curva y sirve como un único numero de la performance de un clasificador

Evaluación

Métricas



Area Under Curve (AUC): es el área bajo la curva y sirve como un único numero de la performance de un clasificador

Evaluación

Métricas

Consideraciones al optimizar métricas de evaluación:

Evaluación

Métodos de Particionado

Holdout (Método de retención):

- Particiona **aleatoriamente** el conjunto de datos en dos conjuntos (Train/Test), uno para entrenamiento y uno para prueba
- **Importante:** los conjuntos son **disjuntos**, no se usa para testing los ejemplos que se usaron para entrenamiento
  → sobre-estimación del modelo
- Con el conjunto de entrenamiento se aprende el modelo (incluye el conjunto de calibración)
- La utiliza el conjunto de prueba para predicción y cálculo de métricas de performance

Evaluación

Métodos de Particionado

Limitaciones:

- Asume un conjunto de datos lo suficientemente grande para dividir los ejemplos en ambos conjuntos
- En un único experimento las métricas pueden no ser representativas

Evaluación

Métodos de Particionado

Holdout repetitivo: repetir el experimento varias veces pero cambiando la partición entrenamiento/prueba

- En cada experimento se selecciona aleatoriamente una parte de los ejemplos para entrenar el modelo
- Los valores obtenidos para las métricas de los diferentes experimentos se promedian para calcular un valor final

Evaluación

Métodos de Particionado

$$Métrica = \frac{1}{N}\sum_{i=1}^{N} Métrica_i$$

Limitaciones:

- No es óptimo porque los subconjuntos de los experimentos pueden superponerse

Evaluación

Métodos de Particionado

***k*-fold cross validation (Validación cruzada)**: los datos disponibles se particionan en *k* subconjuntos disjuntos de igual tamaño

- Usa cada subconjunto para prueba y junta los *k*-1 para entrenar el modelo
- Se realizan *k* experimentos tomando de a uno los subconjuntos de prueba
- Los valores obtenidos para las métricas de los diferentes experimentos se promedian para calcular un valor final

Evaluación

Métodos de Particionado

- Tiene la ventaja de que todos los ejemplos se usan en algún momento para entrenamiento y para prueba

Evaluación

Métodos de Particionado

Leave-one-out: es un caso especial de *k*-fold cross validation donde *k* es igual al número de ejemplos disponibles

- Se utiliza cuando hay pocos ejemplos disponibles
- Para *N* ejemplos, se realizan *N* experimentos
- En cada experimento se usan *N*-1 ejemplos para entrenamiento y el restante para prueba

Evaluación

Métodos de Particionado

Evaluación

Métodos de Particionado

En la práctica, el valor de *k* depende del tamaño del conjunto de datos

- En conjuntos grandes, 3-fold cross validation puede ser suficiente
- En conjuntos pequeños o dispersos, es preferible usar *leave-one-out* para entrenar sobre la mayor cantidad de ejemplos posibles
- Una elección común es *k*-fold cross validation con *k*=10, es decir 10-fold cross validation

Próxima clase

Algoritmos de Aprendizaje No Supervisado

- k-Means
- Jerárquico
- ...

Evaluación del Aprendizaje

- Metodologías
- Métricas