

C11

AI: Rama de la IA que estudia algoritmos para que las computadoras aprendan a resolver problemas a partir del uso de datos sin ser programadas específicamente.

- Extracción de Información
- Minería de Datos
- Soluciones Aproximadas
- Sistemas de Recomendación
- Detección de Fraudes

"Un programa aprende una tarea si su performance en la tarea mejora con la Experiencia"

T: Reconocer escritura

P: % de palabras bien reconocidas

E: Repo de imágenes de palabras manuscritas y sus transcripciones

Supervisado

Desarrollo modelos basados en datos de entrada y de salida

Regresión
Salida Continua

Clasificación
Salida Discreta

NO SUPERVISADO

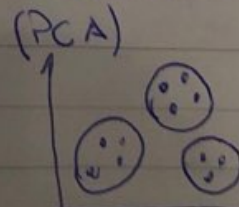
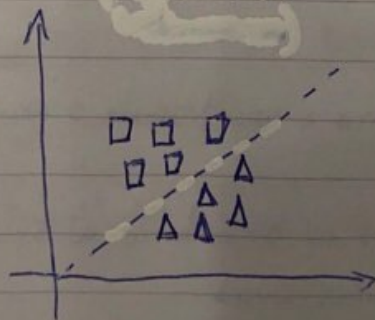
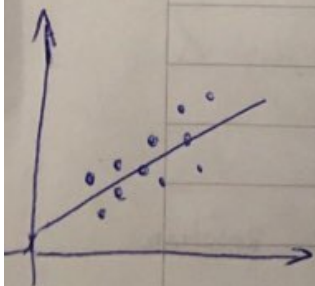
AGrupa e interpreta solo basándose en el Input

Aprendizaje por Refuerzo

1. Aprende un agente autónomo para elegir acciones óptimas

1. Premios y Castigos (Juego de mesa, Robots, etc)

• Clustering
(Segmentación del mercado)
Detección de patrones ocultos
• Reducción de Dimensionalidad (PCA)



C311

Disponibilidad de datos

- Imágenes Satelitales
- Datos meteorológicos
- Señales
- Texto

Big Data

Volume: Scale of Data

Velocity: Analysis of Streaming Data

Variety: Different forms of Data

Veracity: Uncertainty of Data

Digital Footprint

→ Activa: Fotos/Posteos

→ Pasiva: hábitos de Búsqueda/web browsing

Datos - Estructurados → Numéricos / Ordinales / Categóricos
(orden) (sin orden)

- No Estructurados

- Semi Estructurados

Calidad de Datos: Un dato o conjunto de datos "x" es de mejor calidad que un "y" si "x" satisface las necesidades del usuario mejor que "y".

Atributos de Calidad

✓ No hay datos perfectos
es necesario priorizar
las calidades deseadas

- Completitud
- Relevancia
- Vigencia
- Disponibilidad
- Confiabilidad
- Consistencia
- Corrección
- Seguridad/Privacidad

Correctitud/Bias (inclinación por pertenencia a una persona o grupo)

→ Si el conjunto de datos es desbalanceado se optimiza los resultados para ese grupo de individuos.

Open Data: Datos Público-Privados Disponibles. Es la idea que esta data debe estar libre para su publicación sin restricciones

- Disponibilidad & Acceso

- Reuso & Redistribución

- Participación Universal

- La idea no es publicar info de personas en particular, sino de grupos & organizaciones

Problemas Eticos

Privacidad: Quien controla?

Ownership: Quien es el dueño?

Reputación: Como determinan si los datos son confiables?

Identidad: Proveniencia de quien se refieren los datos

C4//

Aprendizaje De Conceptos

Concepto: Subconjunto de objetos o eventos de interés sobre un conjunto mayor.

- función booleana definida sobre el conjunto mayor

APRENDER UN CONCEPTO: INDUCIR AUTOMATICAMENTE UNA FUNCIÓN booleana a partir de gtos de ejemplos o datos.

DADO UN NUEVO CASO, devuelve su clase.

ESQue?: ANIMAL \rightarrow Bool

Se construyen & evalúan hipótesis para aproximar el concepto objetivo

$h = \text{vuela}$

$h' = \text{Dos Patas}$

$h'' = \text{Dos Patas} \wedge \text{tiene plumas} \wedge \text{Pone huevos}$

\vdots

Algoritmo De Aprendizaje: buscar la hipótesis en H que mejor se ajuste a nuestras datos

Los elementos de la conjunción son DATOS que podamos obtener & así Decidir

$\langle ?, ?, ?, ?, ?, ? \rangle$ H General (todos los días son ej: Positivo)
 $\langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$ H específica (ningún valor válido ej: Positivo)
 $\langle ?, \text{cold}, \text{High}, ?, ?, ? \rangle$ H MAS Restrictiva

APRENDIZAJE INDUCTIVO: Construir un modelo general a partir de info específica

↳ Cualquiera H/modelo que aproxime bien una función objetivo sobre un conjunto grande de datos también lo hará sobre datos no observados

Sesgo Inductivo (todos los algoritmos de aprendizaje tienen)

↳ Reducir el espacio de conceptos GRANDES a uno chico.

Algoritmo FIND-S

- busca en el espacio de conjunciones
- Comienza con hipótesis MAS específica
- Si falla en cobertura de ejemplo positivo, generaliza
- Devuelve una H.
- Devuelve hipótesis MAS específica en H consistente con ejemplos positivos.

- Prefiere hipótesis MAS específica, pero podría ser mejor alguna MAS general.
- No es Robusto al Ruido.

Las de atributo objetivo falso no hacen nada

$h_0 = \langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$

$x_1: \langle \text{Sunny}, \text{warm}, \text{Normal}, \text{strong}, \text{warm}, \text{sunny} \rangle +$

$x_2: \langle \text{Sunny}, \text{warm}, \text{High}, \text{strong}, \text{warm}, \text{sunny} \rangle +$

$x_3: \langle \text{Rainy}, \text{cold}, \text{High}, \text{strong}, \text{warm}, \text{change} \rangle -$

$x_4: \langle \text{Sunny}, \text{warm}, \text{High}, \text{strong}, \text{cool}, \text{change} \rangle +$

$h_1 = \langle \text{Sunny}, \text{warm}, \text{Normal}, \text{strong}, \text{warm}, \text{sunny} \rangle$

$h_2 = \langle \text{Sunny}, \text{warm}, ?, \text{strong}, \text{warm}, \text{sunny} \rangle$

Negativa, no hago nada $h_3 = h_2$

$h_4 = \langle \text{Sunny}, \text{warm}, ?, \text{strong}, ?, ? \rangle$

List then Eliminate

- Genera todas las hipótesis H
- Elimina cualquier hipótesis inconsistentes con el training set
- Idealmente queda una " h " que es consistente con todos los datos, si hay más de una Devuelve todas las consistentes.
- Se puede aplicar cuando el conjunto H es finito.
- A H es grande es muy costoso

Eliminación De Candidatos

- Estado de versiones (VS) más compacto
- Conienza con h más general y menos general
- Genera un S y una G y usa esas
 - específico
 - general

Converge a lo h correcto si → Ausencia de error en los ejemplos
→ El concepto objetivo está incluido como h en H

Sesgo Inductivo: Conjunto de afirmaciones que el algoritmo utiliza para construir un modelo.

- Forma de las hipótesis (numero y tipo de parámetros)
- Características de funcionamiento del algoritmo (como recorrer el espacio hasta elegir un único modelo)

C5//

Arbol de decision: - Metodo para Inferencia Inductiva

- Predice valor objetivo en funcion de las Reglas.

- Es una Disyunción de conjunciones sobre valores de atributos

Nodo: Representa test sobre un Atributo de la instancia

Ej: Corresponde a un valor para ese Atributo

Util

- ↳ Instancias Representadas por pares Atributo-valor
- ↳ Función Objetivo tiene valor de Salida Discreto
- ↳ Se pueden Requerir hipótesis disyuntivas.
- ↳ posible conjunto de entrenamiento Ruidoso
- ↳ " " valores de Atributos Faltantes

- 1) Eleccion de mejor atributo para nodo actual "A"
 - 2) Asignar "A" como nodo actual.
 - 3) Para cada valor que toma "A" crear un hijo.
 - 4) Clasificar (Repartir) instancias en los nuevos nodos segun valor de "A".
 - 5) Si las instancias estan bien clasificadas: TERMINAR
- SINO: Iterar sobre nuevos nodos

Information Gain (ENTROPIA)

- Reducción esperada de entropia por partir ejemplos basados en ese atributo

Mide impureza de S

$$\text{Entropia}(S) = \sum_{c \in \text{clases}} -p_c \log_2(p_c)$$

$$\text{Gain}(S, A) = \text{Entropia}(S) - \sum_{v \in \text{valores}(A)} \frac{|S_v|}{|S|} \text{Entropia}(S_v)$$

Partición por Atributo
"A". Reducción de
Entropia

Gain Ratio: Corrige preferencia de
Information Gain sobre
Atributos c/ muchos valores

Gini: Se mide impureza de la muestra (conjunto de entrenamiento)

Impureza
$$Gini(S) = 1 - \sum_{c \in \text{clases}} \left(\frac{|S_c|}{|S|} \right)^2$$

S_c : conjunto de instancias que pertenecen a la clase c

GiniGain
$$GiniGain(S, A) = Gini(S) - \sum_{v \in \text{valores}(A)} \frac{|S_v|}{|S|} Gini(S_v)$$

Posibles valores del atributo A

$$S_v = \{ s \in S \mid A(s) = v \}$$

se elige atributo con MAJOR GiniGain

INFORMATION GAIN

$$\text{Gain}(S, A) = \text{Entropia}(S) - \sum_{\text{valores}(A)} \frac{|S_v|}{|S|} \text{Entropia}(S_v)$$

$$\text{Entropia}(S) = \sum_{C \in \text{clases}} -p_c \log_2 p_c \quad , \quad p_c: \text{proporci3n de instancias de } S \text{ que pertenecen a la clase } C \text{ (¿es a correr o no?)}$$

$$\text{Entropia}([9+, 5-]) = -\left(\frac{9}{14}\right) \log_2 \left(\frac{9}{14}\right) - \left(\frac{5}{14}\right) \log_2 \left(\frac{5}{14}\right) = 0.940$$

Particiones por HUMEDAD: Alta y Normal

$$|S_v| = 7 \text{ Altas y } 7 \text{ Normales}$$

$$|S| = 14$$

$$\text{Entropia}(S_{v=\text{Humedad Alta}}) = -\underbrace{\left(\frac{3}{7}\right) \log_2 \left(\frac{3}{7}\right)}_{-1.222} - \underbrace{\left(\frac{4}{7}\right) \log_2 \left(\frac{4}{7}\right)}_{-0.807} = 0.984$$

+0.523 + 0.467

Cuántas de las
Altas dan Si
¿cuántas dan
no?

$$\text{Entropia}(S_{v=\text{Humedad Normal}}) = -\underbrace{\left(\frac{6}{7}\right) \log_2 \left(\frac{6}{7}\right)}_{-0.222} - \underbrace{\left(\frac{1}{7}\right) \log_2 \left(\frac{1}{7}\right)}_{-2.907} = 0.591$$

+0.190 +0.401

$$\text{Gain}(S, \text{Humedad}) = 0.940 - \left[\frac{7}{14} \times 0.984 + \frac{7}{14} \times 0.591 \right] = \boxed{0.152}$$

$$\frac{|S_v|}{|S|} \rightarrow \begin{matrix} \nearrow \# \text{ Humedad Alta} \\ \searrow \# \text{ total} \end{matrix}$$

Particiones for Viento \rightarrow debil (8/14) \rightarrow Si (6/8)
 \rightarrow fuerte (6/14) \rightarrow Si (3/6)
 \rightarrow NO (3/6)

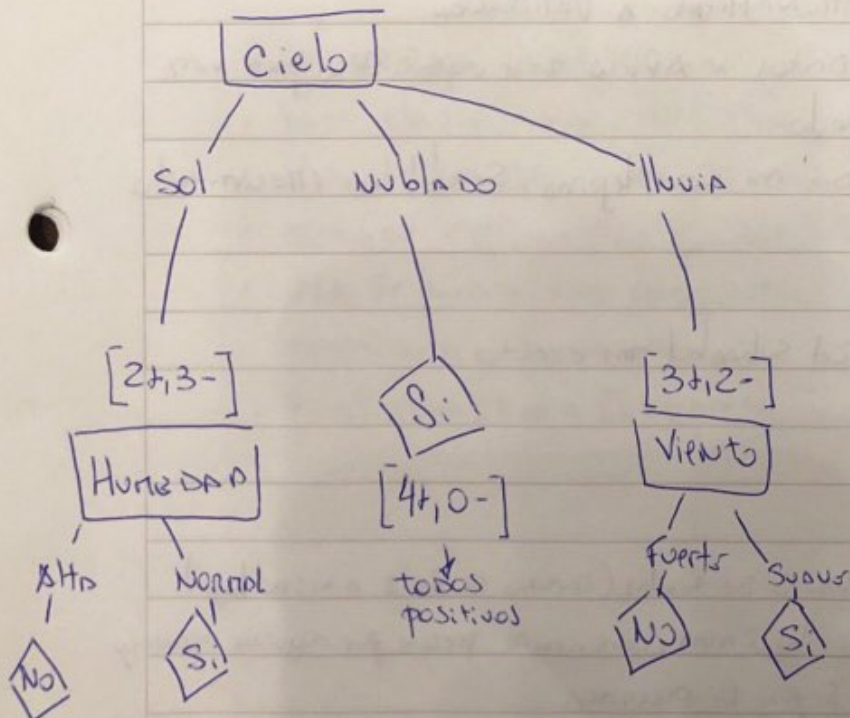
$$\text{Entropia}(S_{V=\text{Viento Fuerte}}) = -\underbrace{(3/6) \log_2(3/6)}_{-1} - \underbrace{(3/6) \log_2(3/6)}_{-1} = 1$$

$$\text{Entropia}(S_{V=\text{Viento Debil}}) = -\underbrace{(6/8) \log_2(6/8)}_{-0.415} - \underbrace{(2/8) \log_2(2/8)}_{-0.5} = 0.811$$

+0.311 +0.5

$$\text{Gain}(S, \text{Viento}) = 0.940 - \left[\frac{6}{14} \times 1 + \frac{8}{14} \times 0.811 \right] = \boxed{0.048}$$

$$\text{Gain}(S, \text{Cielo}) = \boxed{0.246} \checkmark$$



5 casos
 $\text{Gain}(S_{\text{Sol}}, \text{Humedad}) = \frac{\text{H Alta } 3/5}{\text{H Baja } 2/5}$

$$\text{Gain}(S_{\text{Sol}}, \text{Humedad}) = \boxed{0.970}$$

$$\text{Gain}(S_{\text{Sol}}, \text{Temp}) = 0.570$$

$$\text{Gain}(S_{\text{Sol}}, \text{Wind}) = 0.19$$

$$\begin{aligned} \text{Entropia}(S_{\text{Sol}}) &= \underbrace{-\frac{3}{5} \log_2(\frac{3}{5})}_{-0.736} - \underbrace{\frac{2}{5} \log_2(\frac{2}{5})}_{-1.321} = 0.97 \\ &= 0.447 + 0.528 \end{aligned}$$

- Espacio de Hipótesis Completo
 - Una sola hipótesis de salida
 - No hay backtracking
 - Robusto a errores de datos de Entrenamiento
 - Bias Inductivo
 - ↳ Se prefiere el árbol más corto
 - ↳ con atributos con Information Gain Alto cerca de la Raíz
- Navaja de Occam: Se prefiere hipótesis más corta para satisfacer los datos.

Overfitting

- Árbol muy profundo
- Le h es mejor sobre entrenamiento pero h' generaliza mejor
- ↳ Detener crecimiento árbol antes que clasifique perfecto a los datos
- ↳ Podarlo
- ↳ Conjunto de Entrenamiento a Validación
- ↳ uso todos los datos + aplico test estadístico para ver si expando o no
- ↳ MIDL: uso medida de complejidad. Se detiene crecimiento

Reduce error Pruning

- ↳ Producir tamaño menor del subárbol más exacto

Rule post Pruning

- Convertir árbol en conjunto de Reglas (camino de Raíz a nodo hoja)
- Podar cada Regla indep de las demás, removiendo precond que mejoran accuracy
- Ordenar las podams segun su accuracy.

Atributos → valores : Discretizarnos. Busco umbral t a discretivo si el Continuo Atributo es $< t$.

Ordeno instancias de menor a mayor y después parto la lista buscando Reducir impurezas.

valores

Faltantes

- Asignar el valor más común en los datos de entrenamiento
- " " " " " " " " " " fue bueno la misma clasificación - 100%
- Asignar una probabilidad basada en frecuencia observadas de A en modo N.

Costo

Costo

- Referencias con bajo costo, los de Alto sob como es necesario



- / Aprendizaje Supervisado
- / para clasificación y Regresión
- / fácil de usar y entender.
- / Método exploratorio de atributos
- / Se pueden usar categoricos, continuos y binarios
- / fácil visualizar e Interpretar



- / puede haber overfitting
- / Suele necesitar ensambles de árboles para mejor performance

C6//

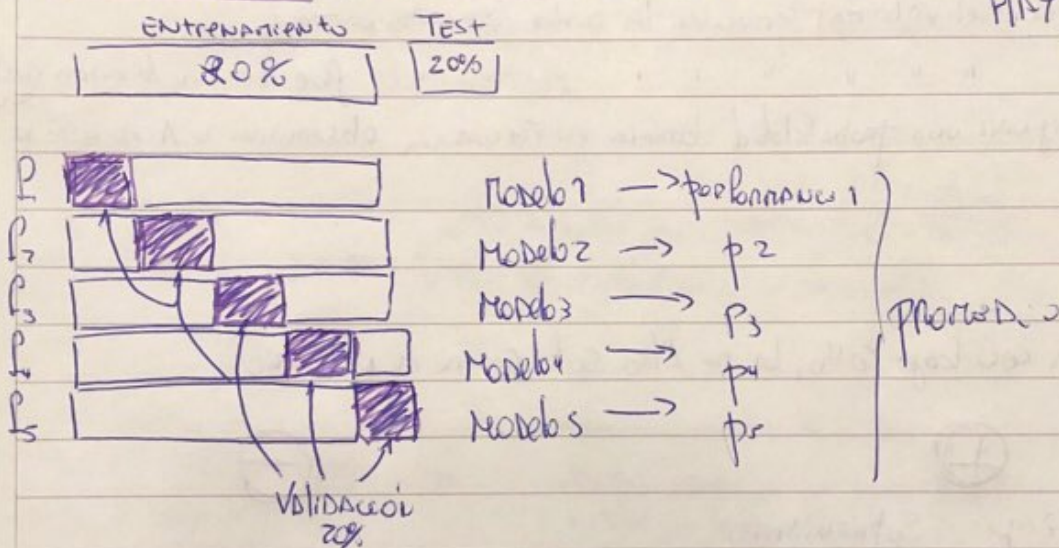
Data Entrenamiento \rightarrow H o modelo que Aproxima f.Accuracy Sobre
Conjunto De Entrenamiento \rightarrow Mala idea; el modelo puede memorizar los datos de entrenamiento y tener eficacia alta

- Medida de performance sobre entrenamiento tiende a sobreestimar los resultados

Se evalúa el modelo contra Test (Hold-Out)

Cross-validation (K-Fold con $k=5$)

Hay que mezclar los datos!



Comparación entre

≠ Atributos

≠ Algoritmos

≠ Hiperparámetros

- \rightarrow Criterio de Selección de Atributos más
- \rightarrow Criterio de Pruning
- \rightarrow Estrategia de poda

- Exploramos espacio de búsqueda usando CV K-fold para medir el desempeño en cada iteración y al final fuoramos con 1 combinación

Random Searchexplora a l
AZORGrid Searchexplora todas
las combinaciones

Matriz De Confusion

		Valores Reales	
		Positivo	Negativo
Predicciones	Positivo	TP	FP
	Negativo	FN	TN

TP: true positives
 FP: false positives
 TN: true negatives
 FN: false negatives

Accuracy: $\frac{TP+TN}{TP+TN+FP+FN}$

No dicen nada sobre los tipos de acierto x de error.

Autentica por voz \rightarrow FP: Autentica a un impostor (mas grave)
 \rightarrow FN: No autentica a un user valido

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

- Cuan utiles son los Resultados de busqueda
- De las clasificadas como positivas, cuantas lo son.
- Test de embarazo.

Recall/Cobrimiento: $\frac{TP}{TP+FN}$

= **TPR**

- De las instancias positivas, cuantas fueron clasificadas como positivas.
- Cuan completos son los Resultados.
- Donde clasificar mal las positivas es lo peor: Clasificar + o - de una enfermedad.
- % de usuarios validos autenticados.

Specificity = **TNR** = $\frac{TN}{TN+FP}$

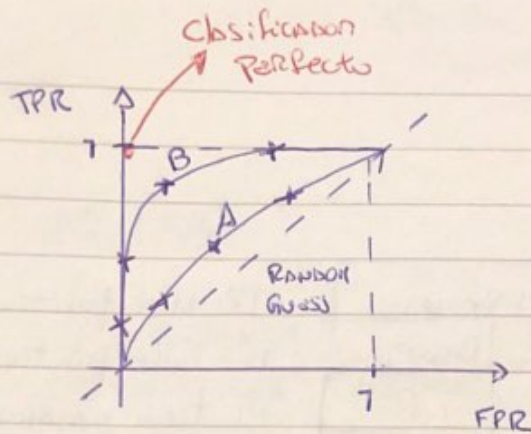
- % de presentes sonos correctamente diagnosticados

FPR = $\frac{FP}{FP+TN}$

- % de impostores fue aceptados erroneamente.

ROC: TPR vs FPR

(% de usuarios validos autenticados vs % de impostores fue aceptados erroneamente)



- Mas X sobre TPR es porque priorizo de las positivas correctas frente a positivas.
- Mas X fuera de TPR, es porque ACEPTO mas FPR (% de impostores que pasan como validos)

Dados dos Roc me fijo con la que mas AUC tenga

$$AUC \in (0, 1)$$

Vamos el umbral de detección entre 0 y 100%. Para cada valor calculo TPR y FPR y obteniendo punto de curva.

Factores para la elección

De modelos

- ↳ Tasa de Error
- ↳ Velocidad de entrenamiento y test
- ↳ Interpretabilidad
- ↳ Fácil desarrollo.

Media Armónica:

$$F\text{-Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

C3//

Clasificación: Develva clas mas probable

$$P(h|D) = \frac{P(D|h) P(h)}{P(D)}$$

Naive Bayes

- Cualquier problema

De clasificación: Clasificar texto

SPAM
CARPETA PARA MAIL
ANALISIS SENTIMENTAL
Tema de un Artículo
etc

- Clasificación probabilística

→ predecir distribución de probabilidades
Sobre un conjunto de clases.

• F que proporcionan probas de ocurrencia de diferentes Resultados posibles de un experimento

Texto: "EL DRAGON Sigue Vivo"

DICCIONARIO	BAG-OF-WORDS
Al	0
DRAGON	1
MUERE	0
VIVO	1
NO	0

$$X_i = \begin{cases} 1^{\#} & , \text{ si palabra esta en DICCIONARIO} \\ 0 & , \text{ si no} \end{cases}$$

Palabras son la # de palabras en vez de si esta o no

• Se asume independencia de las probabilidades de los atributos

• Muy usado para clasificar textos

• Muy Rápido

• Pocos Referenciamientos de Almacenamiento

• busca la h mas probable dado el conjunto de Datos D.

$$C_{\text{map}} = \underset{c \in C}{\text{Argmax}} P(d|c) P(c) = \underset{c \in C}{\text{Argmax}} P(x_1, x_2, \dots, x_3 | c) P(c)$$