

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN


Validación

Comparación

Resp. ejerc.

Bibliografía

Tema 3: Aprendizaje automático (Machine Learning)

Profesor: Juan R. Rico 

Desafíos de programación

Dpto. Lenguajes y Sistemas Informáticos. Universidad de Alicante

Versión 1.0



- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Función de distancia
 - Técnica de los k vecinos más cercanos
 - Edición
 - Condensado
 - Importancia de los prototipos
 - kNNc
- Árboles de decisión (DT)
- Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 **Introducción**
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Introducción I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

En este tema se presentará:

- los **sistemas predictivos** como cambio de paradigma en la solución de problemas;
- el concepto de **ciencia de los datos** (Data Science);
- la definición de **aprendizaje automático** (Machine Learning);
- la definición de **reconocimiento de formas** (Pattern Recognition);
- las familias de **clasificadores** y sus características;
- las redes neuronales avanzadas (**Deep Learning**);

Sistemas predictivos presente y futuro

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Los sistemas predictivos están cambiando la forma en la que se abordan ciertos problemas y que cambiarán nuestro entorno. Algunos ejemplos:
 - La **conducción autónoma** de una programación por casos (if/else) se hace inabordable mientras que con la imitación de un humano aprendiendo de su comportamiento es viable;
 - **Traducción automática** de estar basada en reglas a aprender de textos traducidos por humanos;
 - **Búsqueda de empleo** de emparejar currículum vitae con las demandas de empleo a predecir cuales serán los mejores candidatos para un trabajo aprendiendo de los emparejamientos previos.
- Se puede decir que los datos son la nueva fuente de energía;
- Más información en [Prediction Machines: The Simple Economics of Artificial Intelligence](#) por el profesor Ajay Agrawal (fundador de Creative Destruction Lab and co-fundador de AI/robotics company Kindred).

Aprendizaje automático

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

El **aprendizaje automático** (Machine Learning) [Mitchell, 1997] es una **rama** de la **inteligencia artificial** cuyo objetivo es desarrollar técnicas que permitan **aprender** a los ordenadores. Se trata de crear programas capaces de **generalizar comportamientos** a partir de una información suministrada en forma de **ejemplos**. Es un proceso de **inducción del conocimiento**.

Se centra en el estudio de la **complejidad computacional** de los problemas y puede ser visto como un intento de **automatizar** algunas partes del **método científico** mediante métodos matemáticos.

Para tener una visión general podemos ver la siguiente **infografía**.

Tipos de aprendizajes

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Según el uso del modelo aprendido

- Supervisado
 - Clasificación
 - Regresión
- No supervisado
- Por refuerzo

Se detallan a continuación.

Supervisado

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

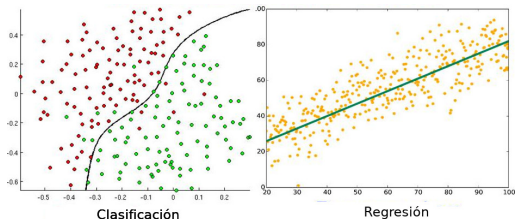
Comparación

Resp. ejerc.

Bibliografía

Se conocen las características de entrada y su **valor de predicción** correspondiente, por lo que se pueden usar para realizar predicciones (es el más utilizado).

- **Clasificación:** La predicción se realiza sobre un conjunto limitado de clases o categorías (opciones o decisiones). Ej.: Estar enfermo o no, identificación de matrículas o números de policía (calle/portal);
- **Regresión:** La predicción es un valor numérico. Ej. Valor de una casa en venta o probabilidad de aprobar una asignatura.



No supervisado

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

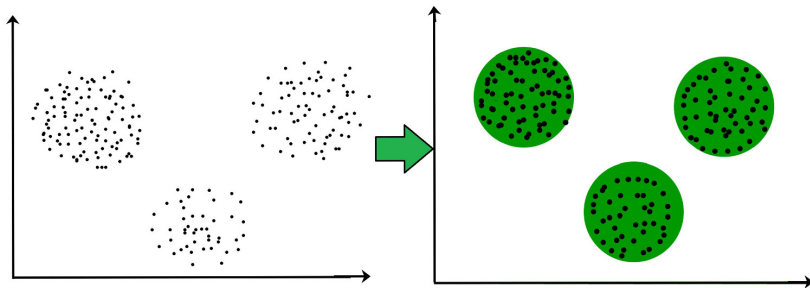
Validación

Comparación

Resp. ejerc.

Bibliografía

Únicamente se conocen las características de los objetos y NO el valor o la clase a la que pertenecen, por lo que solo se pueden usar técnicas de agrupamiento.



Por refuerzo

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

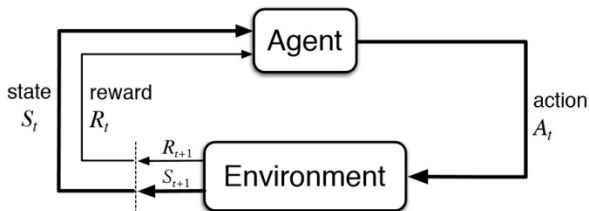
Validación

Comparación

Resp. ejerc.

Bibliografía

Es un sistema que compuesto por un agente, un entorno, modelo de estados, acciones y recompensas/penalizaciones. Se trata de aprender cómo el agente puede interactuar con el entorno realizando una serie de acciones de las cuales obtendrá una recompensa y maximizar su beneficio final. El modelo de estados se encarga de almacenar la información del contexto en cada situación y las posibles acciones asociadas. Se utiliza principalmente en el área de videojuegos.



Algunos tipos más de aprendizaje según la disponibilidad de los ejemplos

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- **Offline learning:** Disponemos de muchos ejemplos para entrenar el modelo.
- **Online learning:** Los ejemplos se suministran poco a poco.
- **Active learning:** Es un caso especial de aprendizaje semi-supervisado, donde hay ejemplos etiquetados y la mayoría no lo están, pero se puede pedir su etiqueta con un coste.
- **Stream learning:** Los ejemplos se suministran de forma rápida, continua e ininterrumpida. Se necesitan modelos rápidos y muy eficientes.

Aprendizaje automático vs. programación tradicional

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Aplicaciones I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Previsión **meteorológica**: permite crear mapas de predicción automática.
- Reconocimiento de **caracteres manuscritos** (OCR): posibilita la transcripción automática de documentos manuscritos.
- Reconocimiento de **voz**: un ejemplo claro son los teleoperadores automáticos, transcripción a texto directamente de audios/vídeos
- Aplicaciones en **medicina**: análisis de biorritmos, detección de irregularidades en imágenes de rayos-x, detección de células infectadas, marcas en la piel...
- Reconocimiento de **huellas dactilares**: identificación de personas.
- Reconocimiento de **caras**: verificación de entrada/salida de un puesto de trabajo, detección de personas en una escena/foto, seguimiento de personas y detección de riesgos para las mismas...

Aplicaciones II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- **Interpretación de fotografías** aéreas y de satélite: gran utilidad para la agricultura, geología, geografía, planificación urbana...
- **Predicción** de magnitudes máximas de **terremotos**.
- Reconocimiento de **objetos**: con importantes aplicaciones para personas con discapacidad visual, robótica,...
- Reconocimiento de **música**: identificar el tipo de música o la canción concreta que suena, transcripción de partituras (modernas, clásicas o antiguas), interpretación de una partitura con estilos aprendidos automáticamente...
- **Automoción**: vehículo de conducción asistida o autónoma.
- **Sector financiero**: sistemas de inversión asistidos, análisis de riesgos (préstamos)...

Ahora podemos pensar en nuevas aplicaciones...

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

?

- Formamos **grupos** de 2 ó 3 personas y **durante 10'** pensamos nuevas **aplicaciones** donde interesara usar técnicas de reconocimiento de formas/aprendizaje automático para **mejorar** un problema, una situación o una carencia actual.
- A continuación exponemos en común la **idea principal**, **cómo** obtener **datos** y **qué** tendría que **aprender** el modelo/aplicación.

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Notas sobre extracción de características

Extracción de características I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Proceso que **genera características** para la posterior clasificación de los datos.
- En ocasiones lo precede un **preprocesado de la señal**, para filtrar errores del sensor, o para **preparar los datos**.
- Las **características elementales** están explícitamente presentes en los **datos adquiridos** mientras que las de un **orden alto** son **derivadas** de las **elementales** mediante transformaciones.

La estructura de las características será un vector, una cadena, un árbol o un grafo según se necesite. Esta elección determinará el tipo de clasificadores que usaremos en la siguiente etapa.

Extracción de características II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

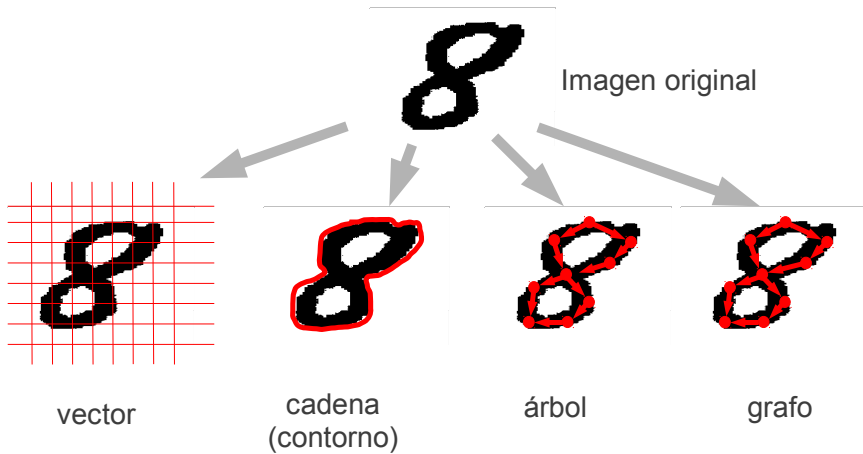
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Selección de variables

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Consiste en **seleccionar** cuál es el **tipo** de **características** o rasgos más adecuados para **describir** los **objetos**.
- Se deben localizar los **rasgos** que **inciden** en el **problema** de manera determinante.

Clasificación

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

La **clasificación** trata de **asignar** las diferentes características a grupos o **clases**, basándose en las **características** extraídas. En esta etapa se usa el **aprendizaje automático**, cuyo objetivo es desarrollar técnicas que permitan **aprender** a los ordenadores.

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

1 Introducción

2 Familias de clasificadores

- Vecinos más cercanos
- Árboles de decisión (DT)
- Máquina de vectores soporte (SVM)

3 Redes Neuronales Artificiales (ANN)

4 Validación un modelo

5 Comparación entre clasificadores

6 Respuestas a los ejercicios

7 Bibliografía

Principales familias de clasificadores

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Según la estrategia usada para la clasificación

Familia	Tipos estructurados	Fácil interpretación de resultados
Los k-vecinos más Cercanos (kNN) / distancias	Sí	Sí
Árboles de Decisión (DT)	No	Sí (1) / No (N)
Máquina de Vectores Soporte (SVM)	No	No
Redes Neuronales Artificiales (ANN)	No	No

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Definición

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

La **función** de **distancia** o de disimilitud devuelve un valor que indica la **diferencia** entre las características de diferentes prototipos.

Ejemplos de distancias entre dos prototipos I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Vectores

Se aplica a vectores de número reales.

- **Manhattan ó L1:** Distancia en bloques rectilíneos y se define como $d_1(\vec{x}, \vec{y}) = \sum_i \|x_i - y_i\|$ [Craw, 2010].
- **Euclídea ó L2:** Es la más utilizada y se define como $d_2(\vec{x}, \vec{y}) = \sqrt{\sum_i (x_i - y_i)^2}$.

Cadenas

Se aplica a cadenas de símbolos.

- **Edición de cadenas o Levenshtein:** Se define como la mínima cantidad de operaciones para convertir una cadena en otra y fue implementada eficientemente por [Wagner and Fischer, 1974].

Ejemplos de distancias entre dos prototipos II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Otras estructuras

- **Edición de árboles:** Distancia entre árboles con nodos y aristas etiquetados y ordenados [Zhang and Shasha, 1989].
- **Edición de grafos:** La distancia entre dos grafos es un problema NP-completo [Kubicka et al., 1990], por lo que solo encontraremos aproximaciones a la solución a este problema en tiempo polinómico.

Ejercicio 1

Implementa en Java un programa para calcular la distancia mínima entre dos cadenas cualquiera.

Para que sea más sencillo se puede acceder a Wikipedia, buscar distancia de edición entre dos cadenas e implementar el programa.

¿Qué distancia hay entre “aab” y “ab”?

¿Y entre “aabcc” y “ccbaa”?

Series temporales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

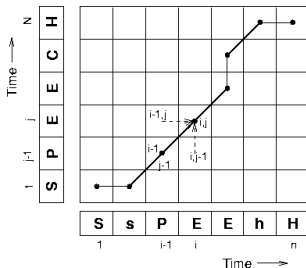
Comparación

Resp. ejerc.

Bibliografía

DTW (Dynamic Time Warping)

Se aplica en análisis de series temporales cuando hay variación de tiempo o velocidad [[Gupta et al., 1996](#)].



ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Técnica de los k vecinos más cercanos

Definición I

ML

Juan R. Rico

Introducción

Familias de clasificadores

k-NN

Función de distancia

k-NN

Edición

Condensado

Importancia prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

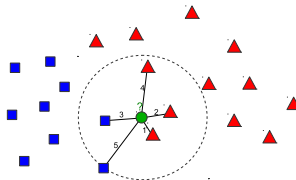
Resp. ejerc.

Bibliografía

El método **k-NN** (k Nearest Neighbors [[Silverman and Jones, 1989](#)]) es un método de clasificación **supervisada** que estima la función de densidad que nos sirve para calcular la **probabilidad a posteriori**, $\hat{P}(\omega_i|x) = k_i/k$, de que un ejemplo x pertenezca a la clase ω_i (k_i es el número de vecinos de la clase i).

Ejemplo

K=5



Características destacables

ML

Juan R. Rico

Introducción

Familias de clasificadores

k-NN

Función de distancia

k-NN

Edición

Condensado

Importancia

prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- **Fácil** de implementar.
- El **error** de la predicción está **acotado** por el doble del mínimo error de Bayes [Cover and Hart, 1967].
- En muchas ocasiones la técnica del **1-NN** es **suficiente** para obtener una **buena** tasa de **clasificación**.
- Sea $\mathcal{D}_n = \{(x_1, \theta_1), (x_2, \theta_2), \dots, (x_n, \theta_n)\}$ un conjunto de prototipos donde $\theta_i = \{\omega_1, \dots, \omega_c\}$ es la clase del elemento x_i y x un nuevo ejemplo. La técnica del kNN se enunciaría como:

$$\text{mode} \left(\theta_k \left(\arg \min_{x_i \in \mathcal{D}_n} \{d(x, x_i)\} \right) \right)$$

Acercar las búsquedas en k-NN

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia

prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- En **principio** para conocer el **vecino más cercano** de un prototipo, x , a un conjunto \mathcal{D}_n se necesitan calcular **todas las distancias**, n .
- Existen varios **algoritmos** (basado en técnicas de ramificación y poda) para **acelerar** estas búsquedas. Un ejemplo son los conocidos como AESA (Approximation-Elimination Search Algorithm [Vidal, 1986]) y sus extensiones [Micó et al., 1994, Micó et al., 1996].

Los prototipos previsiblemente conflictivos

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Existen técnicas llamadas de edición (Editing) que eliminan los prototipos que pueden estar mal etiquetados, o pertenecer a una zona de solapamiento entre clases. Como ejemplo tenemos el algoritmo de Wilson[Wilson, 1972].

```
ArrayList<Example> Editing(int k, ArrayList<Example> T) { //Training set
    ArrayList<Example> S = new ArrayList<Example>(T), //Edited set
        R = new ArrayList<Example>(); //Misclassified set

    for (Example p : S) {
        if (!p.labelClass.equals(p.getClasskNN(k,S))) { //Misclassified example
            R.add(p); //Remove example
        }
    }
} //for
S.removeAll(R); //Remove all misclassified examples

return S;
}
```

Los prototipos más importantes

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Existen técnicas sencillas de condensado (Condensing) de prototipos como CNN (The condensed nearest neighbor rule [Hart, 1968]) y derivadas.

```
ArrayList<Example> CNN(int k, ArrayList<Example> T) { //Traning set
    ArrayList<Example> S = new ArrayList<Example>(); //CNN set
    boolean updated;

    Collections.shuffle(T); //Shuffle array elements
    do {
        updated = false;
        for (Example p : T) {
            if (!p.labelClass.equals(p.getClasskNN(k,S-{p}))) { //Misclassified
                S.add(p); //It's needed
                updated = true;
            }
        }
    } while (S.size() < T.size() && updated);

    return S;
}
```

Consideraciones: edición / condensado

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- La **intención** de las técnicas de **edición** es eliminar los prototipos **mal etiquetados** en base a unos criterios, pero no estaremos seguros si eliminamos un prototipo etiquetado correctamente. Dependemos del conjunto de entrenamiento.
- La **intención** de las técnicas de condensado es eliminar los **prototipos superfluos** pero tampoco podemos verificar si éstos afectan a la tasa de clasificación final que es lo que más nos interesa.
- Si aplicamos repetidamente la técnica de edición hasta que todos los prototipos se clasifiquen correctamente se conoce como Multiedit.
- Si combinamos ambas técnicas se conoce como Multiedit Condensing.

Establecer la importancia de cada prototipo

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

k-NNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Una aproximación actual es la intención es saber cuan importantes son los prototipos para clasificar [[Rico-Juan and Iñesta, 2012](#)].

- 1 1-FN (Farthest Neighbor): Sistema de votación al amigo (misma clase) más lejano previo al enemigo (distinta clase) más cercano.
- 2 1-NE (Nearest Enemy): Sistema de votación al amigo más cercano que esté cerca del enemigo más cercano para una clasificación correcta.

1-FN (Farthest Neighbor)

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

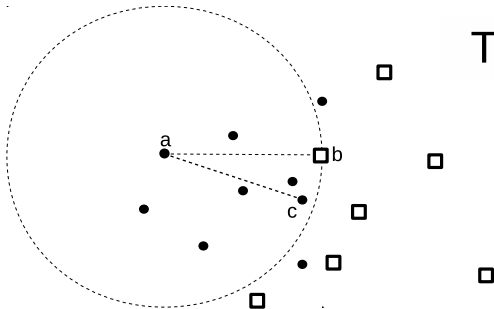
Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Se recorrería todo el conjunto de entrenamiento. En este caso, a es el prototipo que se examina actualmente.
- 2 b es el enemigo más cercano al prototipo a .
- 3 c es el prototipo más alejado de a antes de b y su presencia permite clasificar correctamente al prototipo a .



1-NE (Nearest Enemy)

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

ANN

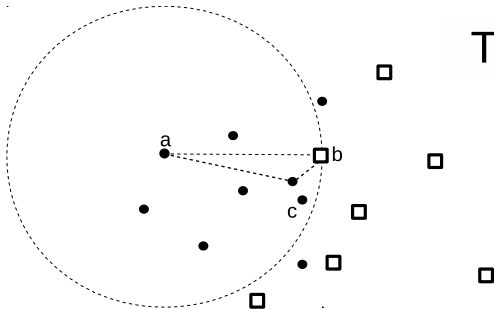
Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Se recorrería todo el conjunto de entrenamiento. a es el prototipo que se examina actualmente.
- 2 b es el enemigo más cercano al prototipo a .
- 3 c es el prototipo de la misma clase que a y más cercano a b . Con su presencia se clasificaría correctamente el prototipo a .



Algoritmo kNNc para acelerar y mejorar la clasificación en presencia de ruido

kNNc acelera la clasificación en dos pasos I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado
Importancia
prototipos

kNNc

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Esta técnica permite **descartar** clases muy distintas al prototipo desconocido x cuando se clasifica. Esta técnica es útil para problemas con **múltiples clases** [Calvo-Zaragoza et al., 2015].

Descripción:

- 1 Seleccionar un conjunto representativo, P , de prototipos a partir del inicial, T .
- 2 Para clasificar un nuevo prototipo desconocido x , solicitar las c clases más cercanas en P .
- 3 Usar k -NN para clasificar x únicamente con las c clases más cercanas.

kNNc acelera la clasificación en dos pasos II

ML

Juan R. Rico

Introducción

Familias de clasificadores

k-NN

Función de distancia

k-NN

Edición

Condensado

Importancia prototipos

kNNc

DT

SVM

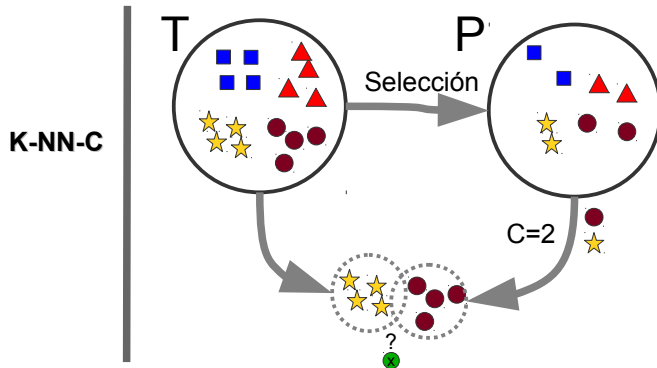
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



kNNc sin ruido

ML

Juan R. Rico

Introducción

Familias de clasificadores

k-NN

Función de distancia

k-NN

Edición

Condensado

Importancia prototipos

kNNc

DT

SVM

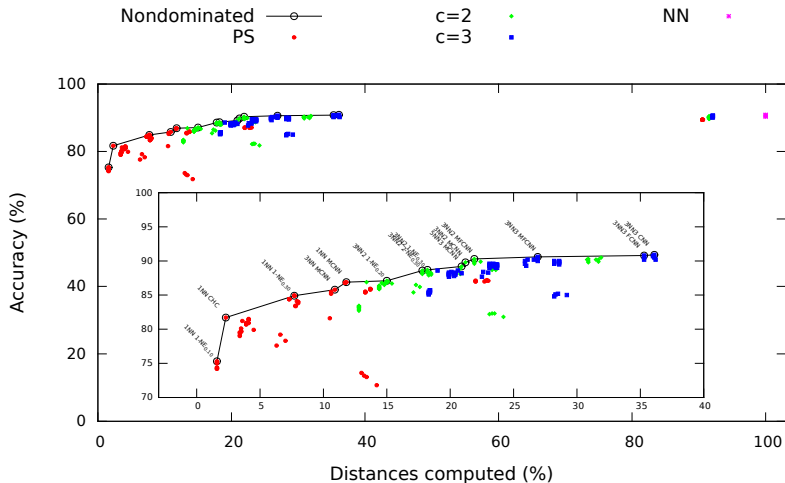
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



kNNc con 40 % ruido

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

Función de
distancia

k-NN

Edición

Condensado

Importancia
prototipos

kNNc

DT

SVM

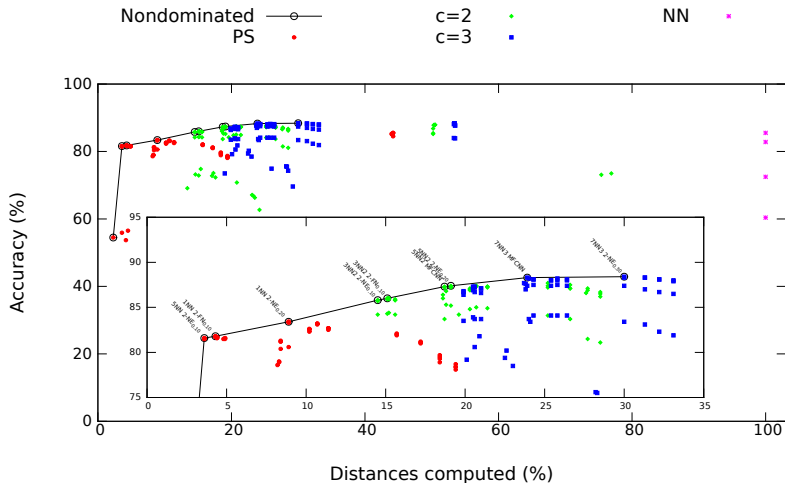
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores
k-NN

DT
SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Árboles de decisión (DT)

Árbol de decisión (DT)

Juan R. Rico

Introducción

Familias de clasificadores

k-NN

DT

SVM

ANN

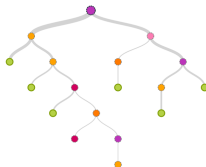
Validación

Comparación

Resp. ejerc.

Bibliografía

Un **árbol de decisión** (Decision Tree) es un diagrama jerárquico cuyos **nodos** representan estados y sus ramas posibles **decisiones**. En general se construye a partir un conjunto de **vectores** de características **etiquetados**. Cada **nodo** del árbol genera **particiones** de sí mismo formando el siguiente **nivel**, para ello se siguen ciertos criterios (decisiones) que minimizan alguna **estimación de error**. Este proceso se aplica recursivamente a cada nodo hasta agotar los elementos o conseguir alcanzar el error deseado.



Algoritmos representativos básicos

- **ID3** y **C4.5** son algoritmos presentados por Quinlan y recopilados en el libro [Quinlan, 2014].
- Podemos consultar más detalles en Wikipedia **ID3** y **C4.5**.
- La intención es tener una idea general y NO aprender los detalles de los algoritmos. Ejemplos en vídeo:
 - Cómo funcionan los árboles de decisión.
 - Ejemplo ID3 paso a paso.

Evolución de los árboles de decisión

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Algoritmos avanzados

- Los árboles de decisión tienen una dependencia elevada de los datos, es decir, si realizamos unos pocos cambios en los datos originales los árboles resultantes son muy distintos.
- La solución es generar muchos árboles y calcular una solución promedio como en Random Forest [Breiman, 2001], Rotation Forest [Rodriguez et al., 2006], etc.

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

1 Introducción

2 Familias de clasificadores

- Vecinos más cercanos
- Árboles de decisión (DT)
- Máquina de vectores soporte (SVM)

3 Redes Neuronales Artificiales (ANN)

4 Validación un modelo

5 Comparación entre clasificadores

6 Respuestas a los ejercicios

7 Bibliografía

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Máquina de vectores soporte (SVM)

Máquina de Vectores Soporte (SVM)

ML

Juan R. Rico

Introducción

Familias de
clasificadores

k-NN

DT

SVM

ANN

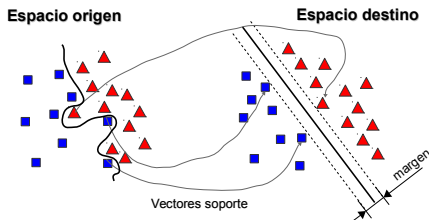
Validación

Comparación

Resp. ejerc.

Bibliografía

Normalmente esta técnica se aplica a dos clases y consta de dos pasos: el primero consiste en **trasladar los datos** originales (n -dimensiones) donde no son separables linealmente, a otra dimensión superior (incluso infinita), donde sí lo son (kernel); el segundo, calcula un **hiperplano** lo más alejado posible de ambas clases. Esta técnica fue desarrollada inicialmente por Vladimir Vapnik [Cortes and Vapnik, 1995].



Kernel más usados:

- Polinomial: $k(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \cdot \vec{x}_j)^d$
- Gaussian Radial Basis Function (RBF): $k(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2)$ donde $\gamma > 0$

Separación óptima:

- Hard-margin: Minimizar $\|w\|$ tal que $y_i(\vec{w} \cdot \vec{x}_i - b) \geq 1$ para $i = 1, \dots, n$
- Soft-margin: $[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(\vec{w} \cdot \vec{x}_i - b))] + \lambda \|\vec{w}\|$

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 **Redes Neuronales Artificiales (ANN)**
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Redes Neuronales Artificiales (ANN)

Redes Neuronales Artificiales (ANN)

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Conocidas por sus siglas en inglés (ANN) se inspiran en las **redes neuronales** de los **seres vivos**.

Se usan generalmente en sistemas interconectados de neuronas (capas). Las conexiones tienen unos pesos numéricos que se ajustan con datos de entrenamiento.

Célula biológica vs artificial

ML

Juan R. Rico

Introducción

Familias de clasificadores

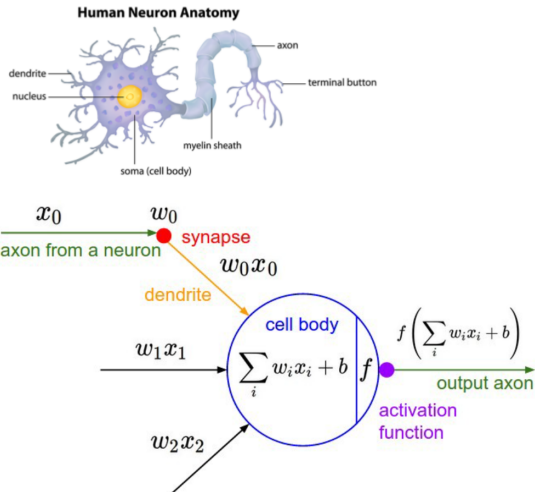
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Función de activación

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Tipos de activación

- Sigmoid: $f(x) = (1 + e^{-x})^{-1}$
- Hyperbolic tangent: $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- Rectified Linear Unit: $f(x) = \max\{0, x\}$

Función de activación

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Tipos de activación

- Sigmoid: $f(x) = (1 + e^{-x})^{-1}$
- Hyperbolic tangent: $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- Rectified Linear Unit: $f(x) = \max\{0, x\}$

?

¿Qué tienen de especial estas funciones? ¿qué papel cumplen?

Perceptron multicapa

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

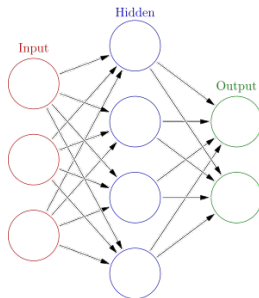
Comparación

Resp. ejerc.

Bibliografía

Modelo clásico

- Capa de entrada
- Capa oculta
- Capa de salida



Redes recurrentes

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

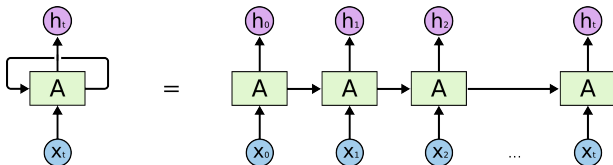
Comparación

Resp. ejerc.

Bibliografía

Recurrencia

- Útil para manejar **secuencias temporales**
- La salida de una o más neuronas en el instante t se utiliza como entrada de otras en el instante $t + 1$



Algoritmo de aprendizaje: retro-propagación

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Descenso por gradiente

- Durante el entrenamiento tenemos **pares (entrada, salida)**
- Se calcula una **función de pérdida** para cada ejemplo
- Se calcula el **gradiente** de dicha función
 - El gradiente nos dice *hacia dónde hay que mover* la red para **minimizar la pérdida**
- El ajuste de pesos se **propaga desde la salida hacia la entrada**, ajustando los pesos en cada caso para minimizar la pérdida en cada capa
- Requiere que las funciones de activación sean, además, **diferenciables**

Aprendizaje profundo

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

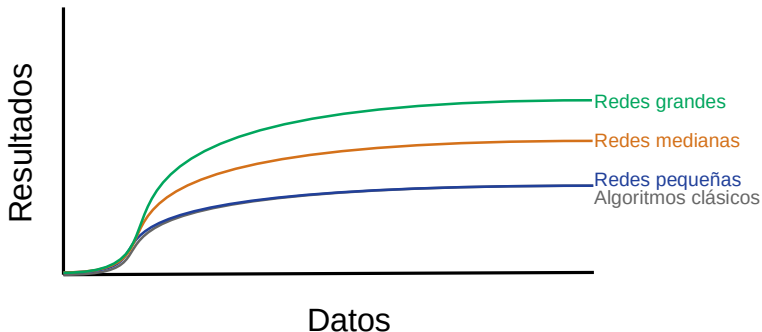
Validación

Comparación

Resp. ejerc.

Bibliografía

En oposición a las redes clásicas, el Aprendizaje Profundo (Deep Learning) propone redes neuronales con una mayor profundidad. Una red se considera **profunda** si contiene **3 ó más capas ocultas**.



Fundamentos del Deep Learning

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Neuronas no-convencionales
- Explosión definitiva en 2012
- Crecimiento apoyado en
 - Inicialización inteligente
 - Mayor capacidad de cómputo
 - Uso eficiente de GPUs
 - Cantidad de datos disponible

Fundamentos del Deep Learning

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Elementos clave

- Redes convolucionales
- Redes recurrentes con LSTM

Redes Convolucionales

- Propuestas (inicialmente) para trabajar con **imágenes**
- Utilizan una o más capas convolucionales
- Las capas convolucionales sustituyen los pesos de entrada por **filtros de convolución**
- El mismo filtro se utiliza a lo largo de toda la entrada
- Vídeo explicativo [Convolutional Neural Networks - The Math of Intelligence](#) (Siraj Raval)

Long Short-Term Memory (LSTM)

- LSTM es una neurona con una pequeña memoria
- Permite a una red recurrente aprender **dependencias a largo plazo**
- Los nodos LSTM son entrenados para saber cuándo deben mantener cierta información, cuándo deben suministrarla para la clasificación y cuándo deben sustituirla por otra

Impacto

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

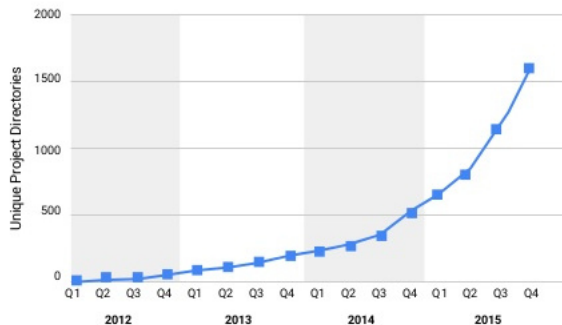
Comparación

Resp. ejerc.

Bibliografía

Google

- Google es uno de los principales promotores del Deep Learning
- Cada vez más usado en sus desarrollos



Impacto

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

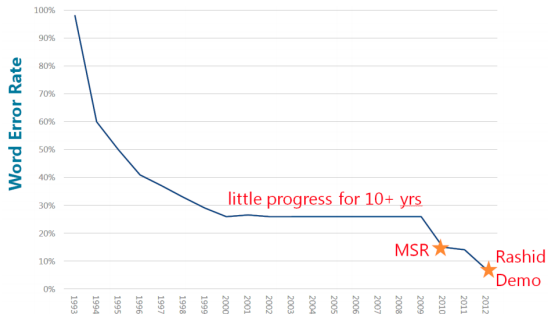
Comparación

Resp. ejerc.

Bibliografía

Reconocimiento del habla

- Permitió superar el estancamiento en el campo



Procesamiento del lenguaje natural

- Redes recurrentes para aprender *word embeddings*
- Cada palabra del vocabulario es representada por un vector en un espacio n -dimensional
- Ejemplo de representaciones 2D

Procesamiento del lenguaje natural

- Redes recurrentes para aprender *word embeddings*
- Cada palabra del vocabulario es representada por un vector en un espacio n -dimensional
- Ejemplo de representaciones 2D

?

¿cómo se pueden codificar palabras como vectores de entrada?

Impacto

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Uso extendido a todo tipo de tareas en inteligencia artificial

- Reconocimiento y detección de objetos en imágenes o vídeos
- Transcripción de documentos
- Biometría
- Traducción automática
- Recomendación musical
- *Question answering / Image captioning*

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Las redes convolucionales (Convolutional Neural Networks, CNN) son redes que incluyen **capas convolucionales**. Las neuronas son sustituidas por filtros convolucionales cuyos pesos se aprenden durante el aprendizaje de la red.

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Fundamentos

- Inicialmente pensadas para recibir **imágenes** como entrada
- Es irrealizable desarrollar capas clásicas para imágenes de entrada de tamaño relativamente grande
- Nodos convolucionales
 - Pesos compartidos
 - Aprovechamiento de correlación espacial
 - Aprender jerarquía de filtros

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Capas convolucionales

Constan de tres pasos

- Convolución
- Activación
- Votación

Convolución

- | Input | | | | | Kernel | | | Intermediate Output | | | Output | | |
|-------|---|---|---|---|--------|---|---|---------------------|----|----|--------|----|----|
| 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 7 | 5 | 3 | | | |
| 1 | 1 | 3 | 2 | 1 | 0 | 0 | 2 | 4 | 7 | 5 | | | |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 7 | 2 | 8 | | | |
| 2 | 3 | 2 | 1 | 3 | | | | | | | | | |
| 0 | 2 | 0 | 1 | 0 | | | | | | | | | |
| 1 | 0 | 0 | 1 | 0 | | | | | | | | | |
| 2 | 0 | 1 | 2 | 0 | 2 | 1 | 0 | 5 | 3 | 10 | 19 | 13 | 15 |
| 3 | 1 | 1 | 3 | 0 | 0 | 0 | 0 | 13 | 1 | 13 | 28 | 16 | 20 |
| 0 | 3 | 0 | 3 | 2 | 0 | 3 | 0 | 7 | 12 | 11 | 23 | 18 | 25 |
| 1 | 0 | 3 | 2 | 1 | | | | | | | | | |
| 2 | 0 | 1 | 2 | 1 | | | | | | | | | |
| 3 | 3 | 1 | 3 | 2 | 1 | 0 | 0 | 7 | 5 | 2 | | | |
| 2 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 11 | 8 | 2 | | | |
| 3 | 1 | 3 | 2 | 0 | 0 | 0 | 2 | 9 | 4 | 6 | | | |
| 1 | 1 | 2 | 1 | 1 | | | | | | | | | |

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Convolución

- Hiper-parámetros
 - Profundidad
 - Tamaño de los filtros
 - Solapamiento

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Votación

- **Combinar activaciones** como única entrada para la siguiente capa
- Diferentes formas de implementar
 - Votación promedio (average pooling)
 - Votación máxima (max pooling)

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

2	2	7	3
9	4	6	1
8	5	2	4
3	1	2	6

Max Pool
→
Filter - (2 x 2)
Stride - (2, 2)

9	7
8	6

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

2	2	7	3
9	4	6	1
8	5	2	4
3	1	2	6

Max Pool
→
Filter - (2 x 2)
Stride - (2, 2)

9	7
8	6

?

¿para qué sirve esta etapa? ¿qué beneficios tiene?

Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de clasificadores

ANN

Validación

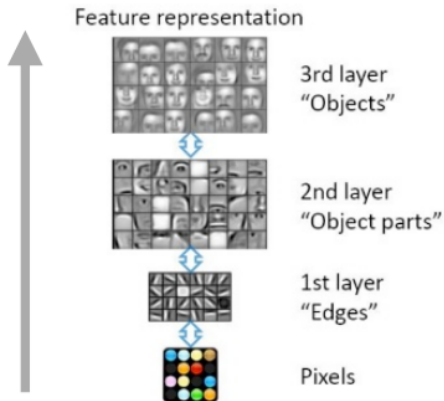
Comparación

Resp. ejerc.

Bibliografía

Jerarquía de características

- Cada capa detecta un **nivel de abstracción** diferente



Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

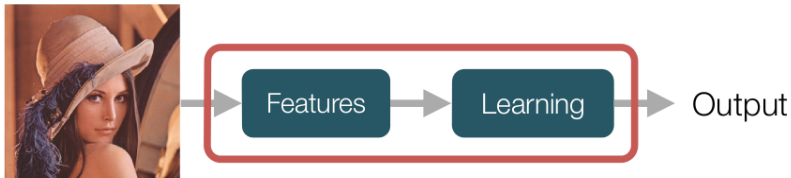
Comparación

Resp. ejerc.

Bibliografía

Jerarquía de características

- La red no sólo aprende a clasificar la imagen sino a **extraer una buena representación** interna de la misma



Redes Convolucionales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Arquitecturas famosas

- LeNet (1989)
 - Primera red convolucional en ser entrenada
- AlexNet (2012)
 - Inicio de la era del Deep Learning

Redes Convolucionales: LeNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

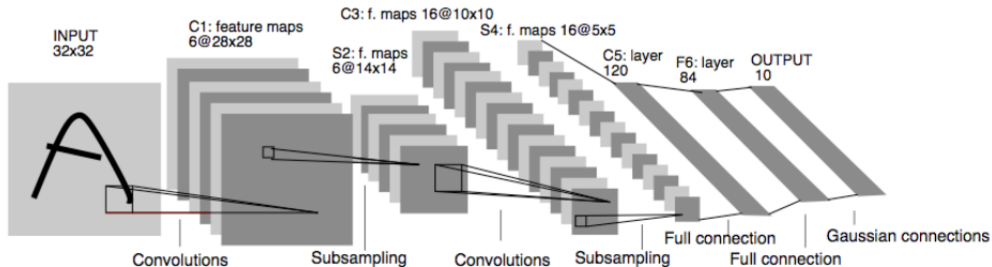
Validación

Comparación

Resp. ejerc.

Bibliografía

- Clasificación de dígitos en imágenes de 32×32



Redes Convolucionales: LeNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

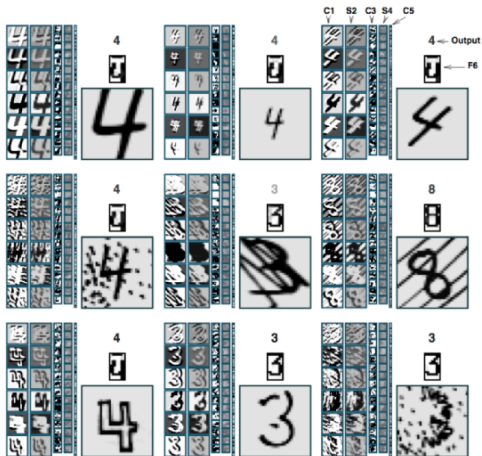
Validación

Comparación

Resp. ejerc.

Bibliografía

- Poder de representación [DEMOS]



Redes Convolucionales: AlexNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

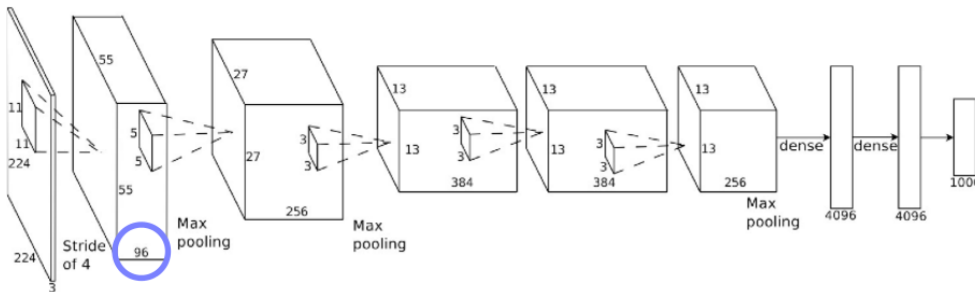
Validación

Comparación

Resp. ejerc.

Bibliografía

- Revolución en la comunidad de visión artificial
- Reducción de un 50 % de error en reconocimiento de objetos
- Introducción de activación ReLU, aumento de datos y *dropout*



Redes Convolucionales: AlexNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Ejemplo de 96 primeros filtros aprendidos por la red



Redes Convolucionales: AlexNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Dropout

- Durante el entrenamiento, algunas neuronas son **desactivadas al azar**
- En AlexNet el *dropout* se aplica al 50 % en las 2 últimas capas

Redes Convolucionales: AlexNet

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Dropout

- Durante el entrenamiento, algunas neuronas son **desactivadas al azar**
- En AlexNet el *dropout* se aplica al 50 % en las 2 últimas capas

?

¿qué beneficios tiene el *dropout*?

Redes Convolucionales: otras arquitecturas

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- GoogleNet: aprendizaje **muy** profundo de objetos [Topología]
- ConvNet para recomendación musical (Spotify) [Topología]
- Redes residuales [Explicación]
- Fully Convolutional Networks [Explicación]
- Deep Convolutional GAN [Explicación]
- NLP: GPT 3 (OpenAI) [Explicación]

Algunos resultados interesantes I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Las redes neuronales actuales no solo clasifican o predicen un valor;
- Los vectores de las últimas capas representan la información de una entrada de forma estructurada y compacta que se puede usar como nuevas características para combinarlas con otros algoritmos;

Ejemplo 1. I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

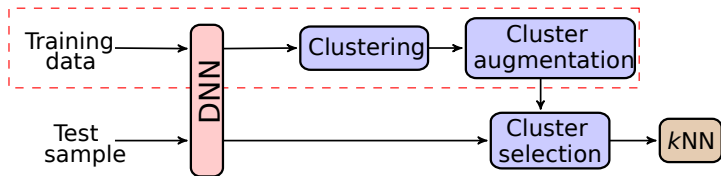
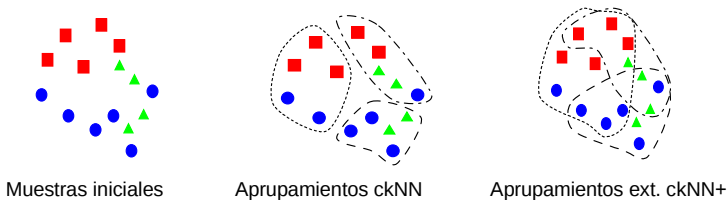
Validación

Comparación

Resp. ejerc.

Bibliografía

Clustering-based k-Nearest Neighbor Classification for Large-Scale Data with Neural Codes Representation [Gallego et al., 2018]



Ejemplo 1. II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

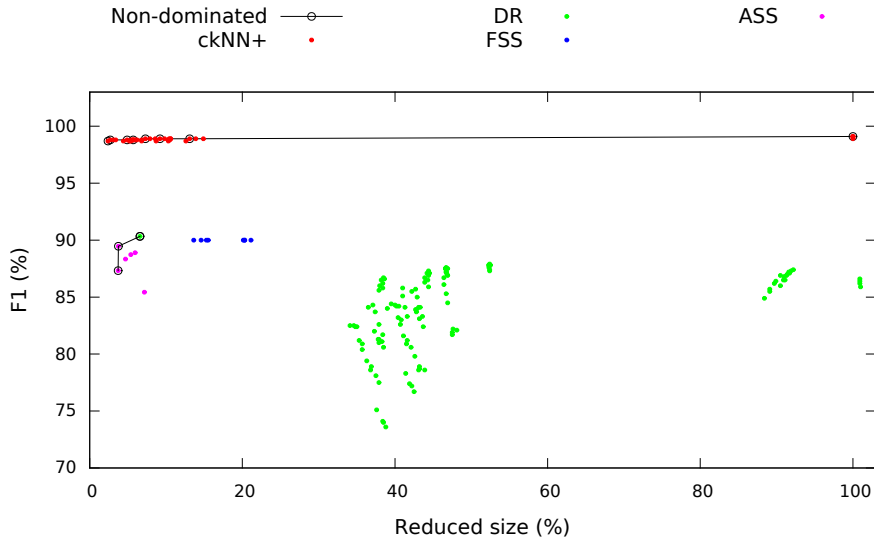
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Ejemplo 2. I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

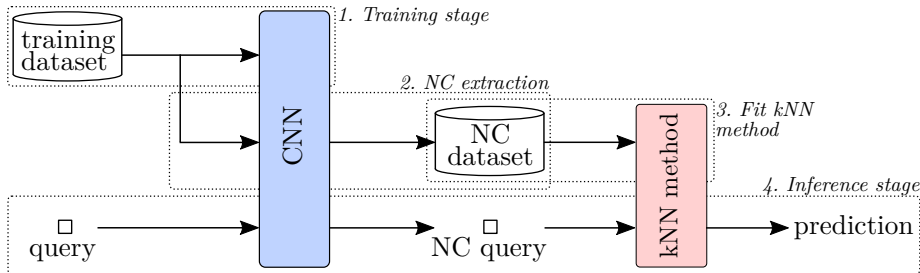
Validación

Comparación

Resp. ejerc.

Bibliografía

Insights into efficient k-Nearest Neighbor classification with Convolutional Neural Codes [Gallego et al., 2020].



Ejemplo 2. II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

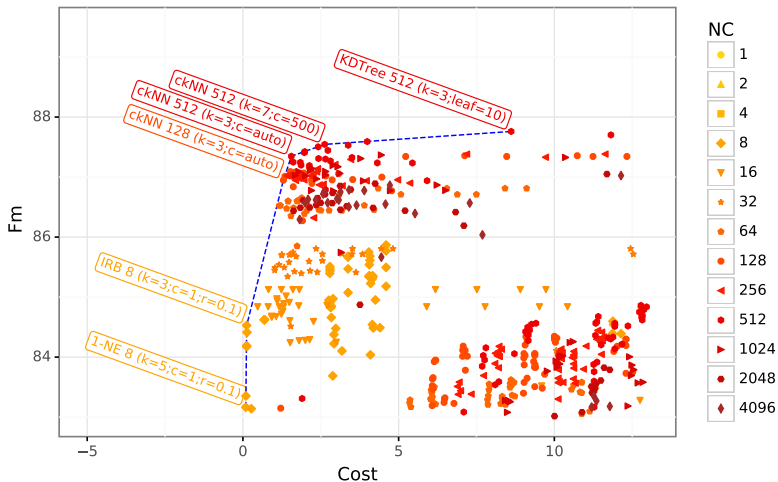
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Ejemplo 3. I

ML

Juan R. Rico

Introducción

Familias de clasificadores

ANN

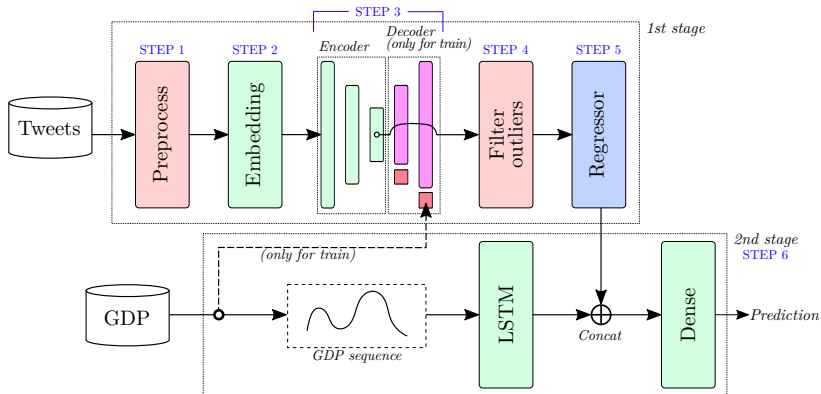
Validación

Comparación

Resp. ejerc.

Bibliografía

A Multimodal Approach for Regional GDP Prediction using Social Media Activity and Historical Information [Ortega-Bastida et al., 2021].



Ejemplo 3. II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

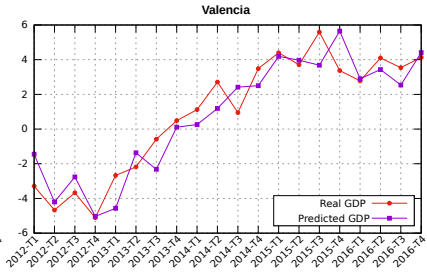
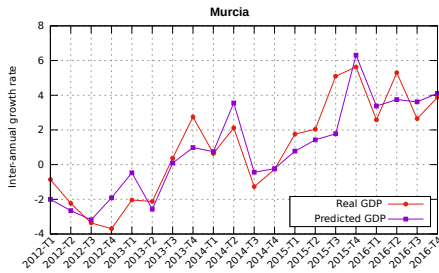
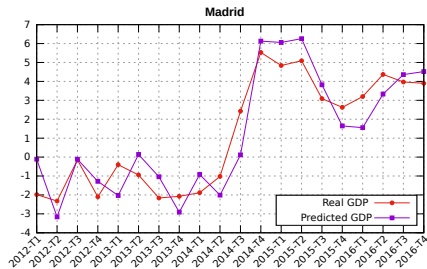
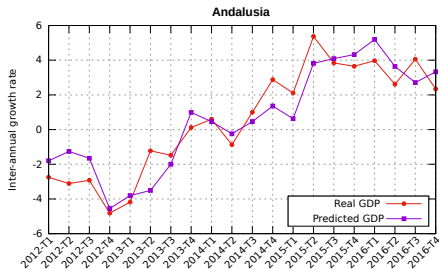
ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Redes neuronales artificiales

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Consideraciones finales:

- La teoría subyacente no está totalmente comprendida;
- Las redes también sirven para el aprendizaje de características;
- Se busca cómo usar el aprendizaje por transferencia

Entornos para su programación

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Podemos distinguir niveles según su facilidad de uso (**Vídeo**):
 - Nivel 0: Programación desde cero;
 - Nivel 1: Bibliotecas para el uso de tensores;
 - **Nivel 2**: Modelado de redes neuronales con capas, conexiones, optimizadores, etc.;
 - Nivel 3: Invocación de redes neuronales con pocos parámetros (habitualmente perceptrones multicapa - MLP).
- Existen muchos entornos para desarrollo con soporte para GPU: TensorFlow (**tf.keras**), PyTorch, MxNet y Caffe, entre otros.

Keras era una biblioteca minimalista, altamente modular, pensada para facilitar la experimentación con redes neuronales. Estaba implementada en **Python** y consistía en una capa de abstracción que se ejecutaba sobre librerías del bajo nivel de **Deep Learning** como TensorFlow (Google), CNTK (Microsoft) o Theano (Université de Montréal). Fue absorbido en 2019 por TensorFlow (Google) y conserva sus características principales. Habitualmente se le conoce como **tf.keras**.

Características

- Desarrollo sencillo y rápido;
- Redes convolucionales y/o recurrentes;
- Transparente ante ejecución en CPU o GPU;
- Bien documentada
- Integrado completamente en el ecosistema de TF 2.x (depuradores, servidores, móviles, API, etc.)
- Tres estilos de programación: **secuencial (fácil)**, funcional (medio) y subclases (avanzado).

Ampliar información sobre Deep Learning

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Referencias clave

- [DeepLearning.net](#)
- Artículo introductorio en **Nature**: Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning” *Nature* 521.7553 (2015): pp. 436-444
- Deep Learning Book (PDF, HTML) [[Goodfellow et al., 2016](#)]

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 **Validación un modelo**
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Validación cruzada

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

La **validación cruzada** (cross-validation) es una técnica utilizada para evaluar los resultados de un **análisis estadístico** y **garantizar** que son **independientes** de la partición elegida (entrenamiento o prueba).

Características

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Consiste calcular la **media aritmética** obtenida de las medidas de evaluación sobre diferentes **particiones**.
- Se utiliza en entornos donde el objetivo principal es la **predicción** y se quiere estimar cómo de **preciso** es un **modelo** que se llevará a cabo a la práctica.
- Es una técnica **muy utilizada** en proyectos de **inteligencia artificial** para validar **modelos**.

Ejemplo: 10 cross-validation

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

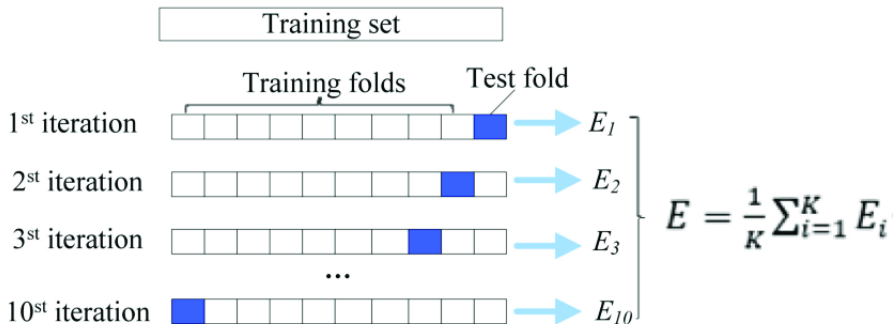
Validación

Comparación

Resp. ejerc.

Bibliografía

- En la actualidad el esquema de 10-CV es el más utilizado.



Caso particular: Dejar uno fuera

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- La validación cruzada, dejar uno fuera o (**leave-one-out**) consiste en separar los datos (N) de forma que tenemos 1 para test y $N - 1$ para entrenamiento.
- Se pueden realizar hasta N iteraciones.
- El **error** cometido en la evaluación es muy **bajo**, a cambio de un **alto coste** computacional.

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Comparación entre varios clasificadores

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

La **comparación** entre **la tasa de aciertos** de varios **clasificadores** obedece a la necesidad de garantizar con una determinada confianza que los resultados de un clasificador son significativamente mejores que las de otro u otros.

Hipótesis nula

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

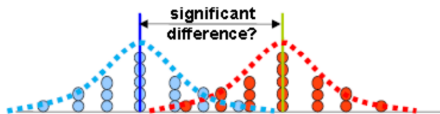
Validación

Comparación

Resp. ejerc.

Bibliografía

La hipótesis nula es una declaración de ausencia de efecto o de ninguna diferencia. Se espera que los resultados sean rechazados por el experimentador.



Ejemplo de dos distribuciones normales

Nivel de confianza α

ML

Juan R. Rico

Introducción

Familias de clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Se trata de un umbral de confianza que nos informa de si debe o no rechazarse la hipótesis nula.
- Debe ser determinado por el experimentador y sus valores usuales son: 90 % (0,1), 95 % (0,05) que es la más utilizado por lo general y también 99 % (0,01).
- Si se decide un nivel de confianza de 0,05 (95 % de certeza), necesitamos un valor de p (dato proporcionado por la prueba) menor que 0,05 para que se pueda rechazar la hipótesis nula. Es decir, sí que hay diferencia significativa entre los resultados.

Tests estadísticos

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Paramétricos vs. no paramétricos: Los primeros necesitan como base la distribución estadística con sus parámetros; los segundos solo contrastan resultados.

nº variables	Paramétrico	No paramétrico
2	t-test	Wilcoxon test Sign test
> 2	ANOVA	Friedman test, Iman-Davenport
	Turkey, Dunnet	Holm , Hochberg, Bonferroni-Dunn..

Más sobre tests estadísticos

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- Diapositivas con información más detallada sobre análisis estadístico de resultados [▶ Statistical Analysis of Experiments](#).
- Una forma de calcular los test estadísticos es usar [▶ el test de Wilcoxon](#) (ej. con Google Colab)

Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Respuesta

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

Ejercicio 1 El algoritmo implementado en Java, entre otros lenguajes, lo encontramos en Wikipedia.

Suponiendo que los costes de las operaciones básicos son:

- inserción = 1 (cadena 1);
- borrado = 1 (cadena 2);
- sustitución(x, y) = $(x == y) ? 0 : 1$;

$$d("ab", "aab") = b(a) + s(a, a) + s(b, b) = 1 + 0 + 0 = 1$$

$$d("aabcc", "ccbba") = s(a, c) + s(a, c) + s(b, b) + s(c, a) + (c, a) = 1 + 1 + 0 + 1 + 1 = 4$$



Índice

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía

- 1 Introducción
- 2 Familias de clasificadores
 - Vecinos más cercanos
 - Árboles de decisión (DT)
 - Máquina de vectores soporte (SVM)
- 3 Redes Neuronales Artificiales (ANN)
- 4 Validación un modelo
- 5 Comparación entre clasificadores
- 6 Respuestas a los ejercicios
- 7 Bibliografía

Bibliografía I

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Breiman, L. (2001).

Random forests.

Machine learning, 45(1):5–32.



Calvo-Zaragoza, J., Valero-Mas, J. J., and Rico-Juan, J. R. (2015).

Improving knn multi-label classification in prototype selection scenarios using class proposals.

Pattern Recognition, 48(5):1608–1622.



Cortes, C. and Vapnik, V. (1995).

Support-vector networks.

Machine learning, 20(3):273–297.



Cover, T. and Hart, P. (1967).

Nearest Neighbor Pattern Classification.

IEEE Transactions on Information Theory, IT-13:21–27.

Bibliografía II

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Craw, S. (2010).

Manhattan Distance.

In Sammut, C. and Webb, G. I., editors, *Encyclopedia of Machine Learning*, page 639. Springer.



Gallego, A. J., Calvo-Zaragoza, J., and Rico-Juan, J. R. (2020).

Insights into efficient k-nearest neighbor classification with convolutional neural codes. *IEEE Access*, 8:99312–99326.



Gallego, A. J., Calvo-Zaragoza, J., Valero-Mas, J. J., and Rico-Juan, J. R. (2018).

Clustering-based k-nearest neighbor classification for large-scale data with neural codes representation.

Pattern Recognition, 74(1):531–543.



Goodfellow, I., Bengio, Y., and Courville, A. (2016).

Deep learning.

Book in preparation for MIT Press.

Bibliografía III

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Gupta, L., Molfese, D., Tammana, R., and Simos, P. (1996).
Nonlinear alignment and averaging for estimating the evoked potential.
IEEE Transactions on Biomedical Engineering, 43(4):348 – 356.



Hart, P. (1968).
The condensed nearest neighbor rule.
IEEE Transactions on Information Theory, 14(3):515–516.



Kubicka, E., Kubicki, G., and Vakalis, I. (1990).
Using graph distance in object recognition.
In *Proceedings of the 1990 ACM annual conference on Cooperation*, CSC '90, pages 43–48,
New York, NY, USA. ACM.



Micó, L., Oncina, J., and Carrasco, R. C. (1996).
A fast branch & bound nearest neighbour classifier in metric spaces.
Pattern Recognition Letters, 17(7):731–739.

Bibliografía IV

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Micó, L., Oncina, J., and Vidal, E. (1994).

A new version of the nearest-neighbour approximating and eliminating search algorithm (AESAs) with linear preprocessing time and memory requirements.

Pattern Recognition Letters, 15(1):9–17.



Mitchell, T. (1997).

Machine Learning.

McGraw-Hill, New York.



Ortega-Bastida, J., Gallego, A. J., Rico-Juan, J. R., and Albarrán, P. (2021).

A multimodal approach for regional gdp prediction using social media activity and historical information.

Applied Soft Computing, 111:107693.



Quinlan, J. R. (2014).

C4. 5: programs for machine learning.

Elsevier.

Bibliografía V

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Rico-Juan, J. R. and Iñesta, J. M. (2012).

New rank methods for reducing the size of the training set using the nearest neighbor rule.
Pattern Recognition Letters, 33(5):654–660.



Rodriguez, J. J., Kuncheva, L. I., and Alonso, C. J. (2006).

Rotation forest: A new classifier ensemble method.

Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28(10):1619–1630.



Silverman, B. W. and Jones, M. C. (1989).

E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951).

International Statistical Review / Revue Internationale de Statistique, 57(3):233–238.



Vidal, E. (1986).

An algorithm for finding nearest neighbours in (approximately) constant average time.
Pattern Recognition Letters, 4:145–157.

Bibliografía VI

ML

Juan R. Rico

Introducción

Familias de
clasificadores

ANN

Validación

Comparación

Resp. ejerc.

Bibliografía



Wagner, R. and Fischer, M. (1974).

The String-to-String Correction Problem.

Journal of the ACM, 21(1):168–173.



Wilson, D. (1972).

Asymptotic properties of nearest neighbor rules using edited data.

IEEE Transactions on Systems, Man and Cybernetics, 2(3):408–421.




Zhang, K. and Shasha, D. (1989).

Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems.

SIAM J. Comput., 18(6):1245–1262.

Tema 3: Aprendizaje automático (Machine Learning)

Profesor: Juan R. Rico 

Desafíos de programación

Dpto. Lenguajes y Sistemas Informáticos. Universidad de Alicante

Versión 1.0

