

Guía docente

340658 - INAR - Inteligencia Artificial

Última modificación: 08/01/2026

Unidad responsable:	Escuela Politécnica Superior de Ingeniería de Vilanova i la Geltrú		
Unidad que imparte:	707 - ESAII - Departamento de Ingeniería de Sistemas, Automática e Informática Industrial.		
Titulación:	MÁSTER UNIVERSITARIO EN INGENIERÍA DE SISTEMAS AUTOMÁTICOS Y ELECTRÓNICA INDUSTRIAL (Plan 2012). (Asignatura optativa).		
Curso: 2025	Créditos ECTS: 5.0	Idiomas: Castellano	

PROFESORADO

Profesorado responsable:	Ruiz Vegas, Francisco Javier
Otros:	Catala Mallofre, Andreu Ruiz Vegas, Francisco Javier

CAPACIDADES PREVIAS

Cálculo diferencial de varias variables básico, optimización de funciones de varias variables, cálculo vectorial y matricial.
Estadística básica y probabilidad.
Programación básica (lenguaje Python)

REQUISITOS

No hay requisitos específicos, pero se supone conocimientos suficientes en los campos descritos en las capacidades previas.

METODOLOGÍAS DOCENTES

Se alternarán clases explicativas en las que se presentarán los distintos conceptos con clases prácticas en las que se practicarán dichos conceptos con ayuda de distintas librerías de Python. Durante el curso se propondrán también algunos proyectos que permitan poner en práctica los conocimientos adquiridos.

OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

- * Conocer qué es la Inteligencia Artificial, sus fundamentos como disciplina científico-técnica y su historia.
- * Adquirir las habilidades asociadas a la Ciencia de Datos (programación, cambio entre formatos, limpieza y depuración, análisis estadístico, visualización, obtención de información mediante consultas,...).
- * Conocer las bases de la Inteligencia Artificial clásica (Búsqueda, búsqueda informada, heurística, búsqueda con adversario)
- * Conocer los modelos de Aprendizaje Automático supervisado y no supervisado (árboles de decisión, vecinos más cercanos, regresión logística, regresión, modelos bayesianos, SVM y redes neuronales, clustering,...)
- * Aprender las tareas básicas asociadas al Aprendizaje Automático (separar conjuntos de entrenamiento, validación y test, seleccionar las variables adecuadas, aplicar técnicas de validación, entender las métricas más relevantes,...)
- * Ser capaz de realizar todas las fases de un proyecto de Inteligencia Artificial: captura de los datos, selección, aplicación y valoración del modelo.

HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTE

Tipo	Horas	Porcentaje
Horas grupo grande	30,0	24.00
Horas grupo pequeño	15,0	12.00
Horas aprendizaje autónomo	80,0	64.00

Dedicación total: 125 h

CONTENIDOS

Generalidades sobre Python

Descripción:

- * Elementos del lenguaje (comentarios, asignación de variables, funciones propias, tipos básicos, listas, tuplas, conjuntos, diccionarios, mutabilidad,...)
- * Estructuras de control (if, while, for,...)
- * Definición de funciones.
- * Librerías, ficheros, clases.
- * Notebooks de Jupiter y Colab.

Objetivos específicos:

El alumno tendrá que conocer los elementos básicos del lenguaje Python para poder manipular datos y aplicar con soltura las librerías de aprendizaje automático durante todo el curso.

Actividades vinculadas:

Ejercicios básicos de programación y manipulación de datos.

Dedicación: 9h

Grupo pequeño/Laboratorio: 1h

Aprendizaje autónomo: 8h

Librerías estándar: Numpy, Pandas, Matplotlib.pyplot

Descripción:

- * Arrays de Numpy.
- * Álgebra lineal con Numpy.
- * Estructuras de datos con Pandas (Series y Dataframes)
- * Gráficos con matplotlib.pyplot

Objetivos específicos:

Conocer las librerías estándar de Ciencia de datos.

Actividades vinculadas:

Ejercicios con cálculo con arrays y databases. Ejercicios de visualización de datos y cálculo de métricas.

Dedicación: 9h

Grupo pequeño/Laboratorio: 1h

Aprendizaje autónomo: 8h

Inteligencia Artificial Clásica

Descripción:

- * Búsqueda no informada y búsqueda informada.
- * Heurística. Algoritmo A*.
- * Simulated Annealing.
- * Búsqueda entre adversarios.
- * Algoritmos Genéticos.

Objetivos específicos:

- * Conocer los conceptos básicos de la Inteligencia Artificial clásica.
- * Plantear problemas de decisión y optimización como un problema de búsqueda en un espacio de estados y tener herramientas para resolverlos.

Actividades vinculadas:

Ejercicios sobre el tema (teóricos y prácticos)

Dedicación: 10h

Grupo grande/Teoría: 2h

Aprendizaje autónomo: 8h

Clasificación. Modelo KNN

Descripción:

- * Problemas de aprendizaje automático (Supervisado, no supervisado, por refuerzo)
- * Clasificación: Entrenamiento, validación y test.
- * Métricas de evaluación (matriz de confusión, exactitud, precisión, sensibilidad, especificidad, F1 score, coeficiente kappa, curva ROC)
- * La librería scikit-learn
- * Ejemplo: KNN (K-vecinos más cercanos)

Objetivos específicos:

- * Conocer, saber aplicar y comprender el significado de las métricas de clasificación.
- * Aprender a aplicar un algoritmo KNN con la librería scikit-learn. Entender el significado del parámetro K.
- * Entender el efecto del balanceo y desbalanceo de los datos. Conocer técnicas de aumento de datos.

Actividades vinculadas:

Ejercicios de aplicación de modelos de clasificación con KNN con datos reales. Cálculo e interpretación de métricas.

Dedicación: 12h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 8h

Regresión lineal y no lineal

Descripción:

- * Modelo de regresión lineal (ecuaciones normales)
- * Métricas de regresión
- * Implementación con numpy i amb scikit-learn
- * Descenso del gradiente
- * Efecto de la normalización
- * Implementación con Scikit-learn
- * Regresión polinomial
- * Concepto de sobreajuste. Regresión con regularización

Objetivos específicos:

- * Distingir entre problemes de classificació i regressió
- * Entendre, saber calcular i interpretar mètriques de regressió
- * Entendre la idea del descens del gradient, base d'altres algorismes d'aprenentatge automàtic
- * Saber aplicar models de regressió lineals i no lineals amb numpy i scikit-learn

Actividades vinculadas:

Ejercicios de regresión con numpy y scikit-learn

Dedicación: 8h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 4h

Regresión Logística

Descripción:

- * Problema de clasificación binaria planteado como una regresión
- * La función logística o sigmoidal
- * La función de coste en regresión logística
- * Algoritmo de descenso del gradiente en regresión logística
- * Métodos de optimización numérica avanzados
- * Regresión logística regularizada
- * Clasificación multiclase (one vs all y one vs one)

Objetivos específicos:

- * Aprender a aplicar modelos de regresión logística con scikit-learn.
- * Aprender a transformar un modelo de clasificación binaria en modelos de clasificación multiclase.

Actividades vinculadas:

Ejercicios de regresión logística biclase y multiclase

Dedicación: 8h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 4h

Árboles de decisión y bosques aleatorios

Descripción:

- * Índices de impureza (gine y entropía)
- * Concepto de árbol de decisión y parámetros
- * Implementación con scikit-learn
- * Bosques aleatorios

Objetivos específicos:

- * Saber implementar árboles de decisión y bosques aleatorios con scikit-learn

Actividades vinculadas:

- * Ejercicios de obtención e interpretación de árboles de decisión y bosques aleatorios

Dedicación: 12h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 8h

Perceptrón y Adaline

Descripción:

- * El perceptrón
- * Implementación con sklearn
- * Adaline
- * Bath Gradient Descent
- * Stochastic Gradient Descent
- * Implementación con sklearn

Objetivos específicos:

- * Conocer la historia del perceptron y los primeros modelos neuronales
- * Entender las diferentes versiones del descenso del gradiente
- * Saber implementar un perceptrón y un adaline con sklearn

Actividades vinculadas:

- * Ejercicios de perceptrones y adalinas con sklearn

Dedicación: 8h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 4h

Máquinas de Soporte Vectorial

Descripción:

- * Clasificación de máximo margen
- * Formulación como un problema de programación cuadrática
- * Resolución con MATLAB
- * Resolución con Python
- * Formulación Lagrangiana. Problema dual
- * Implementación con sklearn
- * Kernel trick
- * kernels lineal, polinómico y gaussiano
- * Término de regularización

Objetivos específicos:

- * Conocer y saber aplicar modelos SVM de clasificación.

Actividades vinculadas:

Ejercicios de aplicación de modelos SVM con sklearn

Dedicación: 12h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 8h

Redes Neuronales Artificiales

Descripción:

- * El perceptrón multicapa
- * El algoritmo Backpropagation
- * Funciones de activación
- * Implementación con sklearn

Objetivos específicos:

Entender el funcionamiento de las redes neuronales y saber implementar redes simples con sklearn.

Actividades vinculadas:

ejercicios sobre redes neuronales con sklearn

Dedicación: 12h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 8h

SISTEMA DE CALIFICACIÓN

El curs se evaluará mediante los siguientes elementos de evaluación:

P:Examen parcial (a medio cuatrimestre)

F:Examen final (al final de curso)

Pra: Examen de prácticas (al final de curso)

Pro: Informe y presentación del/de los proyecto/s asignado/s

La calificación final es calculará mediante la siguiente expresión: $\max(F, 0.5 \cdot (P+F)) \cdot 0.5 + Pra \cdot 0.3 + Pro \cdot 0.2$

Si la calificación final no llega al valor 5, se tendrá que recuperar en un examen de reevaluación que tendrá parte teórica y parte práctica con ordenador (R). Para los alumnos que realicen este examen de reevaluación, la nota final será: $\min(7, R \cdot 0.8 + Pro \cdot 0.2)$

NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

En las pruebas escritas (P, F y R) se podrán utilizar apuntes en papel (NO en formato digital) sin restricción.

En la prueba práctica (Pra y R) (con ordenador) se podrá utilizar apuntes en papel o en soporte digital. Incluso podrá consultarse el material colgado en ATENEA, y las páginas de ayuda de Python y sus librerías. No se podrá utilizar ninguna aplicación de consultas (tipo ChatGPT, Copilot, Gemini, DeepSeek o similares) ni programas de comunicación (correo electrónico, whatsapp, chat,...).

En la evaluación del proyecto se tendrá en cuenta el informe escrito y la presentación oral del mismo.

BIBLIOGRAFÍA

Básica:

- Bishop, Christopher M.. Pattern Recognition and Machine Learning. Springer, 2006. ISBN ISBN-10: 0-387-31073-8.
- Russell, Stuart i Norvig, Peter. Inteligencia Artificial Un enfoque moderno. Pearson Prentice Hall, 2008. ISBN ISBN: 978-84-205-4003-0.
- Benítez, Raúl, Escudero, Gerard i Kanaan, Samir. Inteligencia Artificial avanzada. Universitat Oberta de Catalunya,
- Raschka, Sebastian i Mirjalili, Vahid. Python Machine Learning. Packt>, 2019. ISBN ISBN 978-1-78995-575-0.