

Titulación	Tipo	Curso
Ingeniería Informática	OB	3
Ingeniería Informática	OP	4

Profesor/a de contacto

Nombre : Jordi Gonzalez Sabate

Correo electrónico : jordi.gonzalez@uab.cat

Equipo docente

Silvana Silva Pereira

Idiomas de los grupos

Puede consultar esta información al [final](#) del documento.

Prerrequisitos

Se recomienda que, para cursar esta asignatura, se hayan adquirido las competencias mínimas de las asignaturas de Álgebra, Cálculo, Matemática Discreta, Fundamentos de los Computadores, Programación 1 y Programación 2 (de primer curso), así como de Inteligencia Artificial, Estadística y Programación 3 (de segundo curso).

Objetivos

La asignatura de Aprendizaje Computacional, se enmarca dentro de la mención de "Computación", junto con asignaturas como "Conocimiento, Razonamiento e Incertidumbre", "Visión por Computador" y "Robótica, Lenguaje y Planificación". Por su temática, esta materia no sólo es para los alumnos que cursan la mención de "Computación", sino para cualquier Mención de la carrera de Ingeniería Informática ya que está estrechamente relacionada con la asignatura de "Inteligencia Artificial" de segundo curso. También es muy recomendable haber cursado y sentirse cómodo con los conceptos vistos en las asignaturas de "Cálculo", "Álgebra" y "Matemática discreta" de primer curso, y de "Estadística" de segundo curso, debido al fuerte contenido matemático de la asignatura.

La asignatura pretende tanto ampliar algunos de los temas desarrollados durante "Inteligencia Artificial", como introducir nuevos problemas asociados a la inteligencia artificial, principalmente el aprendizaje de conceptos y tendencias a partir de datos. Se trata de formar al alumno para ser un "ingeniero o científico de datos", y es una de las profesiones con más futuro y más demandadas en la actualidad por empresas como Facebook, Google, Microsoft y Amazon. De hecho, se prevé que el crecimiento de la demanda de estos profesionales en ingeniería de datos sea exponencial a nivel europeo, sobre todo debido al crecimiento en la generación de datos masivos. Así, el principal objetivo de la asignatura es que el alumno sepa encontrar una buena solución (a veces la mejor es imposible) a problemas en contextos distintos de los tratados en clase, a partir de

identificar las necesidades de representación del conocimiento en cada caso y, según sea éste, aplicar la/s técnica/ s más adecuada/s para generar automáticamente buenos modelos matemáticos que expliquen los datos con un error aceptable.

Los contenidos elegidos para esta asignatura también se dan en las Universidades con más prestigio a nivel mundial, como Stanford, Toronto, Imperial College London, MIT, Carnegie Mellon y Berkeley, por poner los centros más representativos. Por lo tanto, por un lado se ofrece al alumno una oportunidad de alcanzar unos conocimientos y habilidades comparables a los impartidos en las mejores universidades. Por otra parte, el alumno debe ser consciente de que este conocimiento que es punta de lanza del estado del arte tiene una dificultad inherente, implicando un estudio y una dedicación considerables, cuantificada en horas en la sección de Actividades formativas de esta guía. Esto es porque en esta asignatura no sólo se enseñan los contenidos más importantes para convertirse en ingeniero de datos, sino además se trabaja una línea de currículo que permita ampliar el abanico de puestos de trabajo a los que podrá acceder tras la carrera, así como poner las bases metodológicas necesarias para realizar un Master en ingeniería de datos o en inteligencia artificial.

Si se busca una asignatura (i) para aprender seriamente y no sólo para aprobar a la ligera, (ii) para abrir un mercado laboral también a nivel internacional, y (iii) para descubrir los propios límites de programación aprendiendo los algoritmos de aprendizaje computacional más utilizados no sólo para las grandes empresas tecnológicas mencionadas anteriormente, sino también en muchas spin-offs de ingeniería de datos en nuestro país, esta asignatura no defrauda si se pone actitud y aptitud.

Los objetivos de la asignatura se pueden resumir en:

Conocimientos:

- Describir las técnicas básicas de aprendizaje computacional.
- Enumerar los pasos esenciales de los diferentes algoritmos de aprendizaje
- Identificar las ventajas e inconvenientes de los algoritmos de aprendizaje que se explican.
- Resolver problemas computacionales aplicando diferentes técnicas de aprendizaje para encontrar la solución óptima.
- Entender el resultado y las limitaciones de las técnicas de aprendizaje en diferentes casos de estudio.
- Saber escoger el algoritmo de aprendizaje más adecuado para solucionar problemas contextualizados.

Habilidades:

- Reconocer las situaciones en las que la aplicación de algoritmos de aprendizaje computacional puede ser adecuado para solucionar un problema
- Analizar el problema a resolver y diseñar la solución óptima aplicando las técnicas aprendidas
- Redactar documentos técnicos relacionados con el análisis y la solución de un problema
- Programar los algoritmos básicos para solucionar los problemas propuestos
- Evaluar los resultados de la solución implementada y valorar las posibles mejoras
- Defender y argumentar las decisiones tomadas en la solución de los problemas propuestos

Resultados de aprendizaje

1. Desarrollar la capacidad de análisis, síntesis y prospectiva.
2. Identificar, gestionar y resolver conflictos.
3. Entender y evaluar el resultado y limitaciones de las técnicas de aprendizaje más comunes.
4. Conocer y comprender las técnicas de representación del conocimiento humano.
5. Resolver problemas computacionales aplicando diferentes mecanismos de aprendizaje necesarios para encontrar la solución óptima.
6. Conocer y aplicar las técnicas de aprendizaje más adecuadas en diferentes casos de estudio.
7. Generar propuestas innovadoras y competitivas en la actividad profesional.
8. Asumir y respetar el rol de los diversos miembros del equipo, así como los distintos niveles de dependencia del mismo.

Contenidos

TEMA 1: INTRODUCCIÓN AL APRENDIZAJE COMPUTACIONAL

- 1.1 Conceptos básicos del aprendizaje computacional
- 1.2 Datos, modelos, predicción y decisión
- 1.3 Tipos de aprendizaje: supervisado, no supervisado y detección de anomalías
- 1.4 El pipeline de aprendizaje automático: datos, representación, modelo, validación y uso

TEMA 2: ¿QUÉ SIGNIFICA APRENDER DE LOS DATOS? REPRESENTACIÓN Y DISTANCIAS

- 2.1 Objetos, atributos, vectores, matrices y etiquetas
- 2.2 Representación de datos: variables, codificaciones, valores ausentes y calidad de los datos
- 2.3 Métricas de distancia, escalado y normalización
- 2.4 Aprendizaje basado en memoria: K-Nearest Neighbors y variantes ponderadas

TEMA 3: REGRESIÓN DE DATOS

- 3.1 Modelos de regresión, hipótesis, parámetros y predicción
- 3.2 Residuos, funciones de coste y criterios de error
- 3.3 Descenso del gradiente y optimización de parámetros
- 3.4 Regularización, complejidad y generalización

TEMA 4: VALIDACIÓN Y SELECCIÓN DE MODELOS

- 4.1 Separación entre entrenamiento, validación y prueba
- 4.2 Sobreajuste, infraajuste y capacidad de generalización
- 4.3 Validación cruzada y selección de hiperparámetros
- 4.4 Fuga de datos y protocolos de evaluación honesta

TEMA 5: CLASIFICACIÓN DE DATOS

- 5.1 Clasificación probabilística y regresión logística regularizada
- 5.2 Umbrales de decisión, calibración y coste de los errores
- 5.3 Métricas de clasificación: exactitud, precisión, exhaustividad, F1, ROC y PR
- 5.4 Máquinas de vectores de soporte: margen, regularización y kernels

TEMA 6: CLUSTERING Y APRENDIZAJE NO SUPERVISADO

- 6.1 Aprendizaje sin etiquetas y criterios de agrupación

6.2 El algoritmo K-Means y selección del número de grupos

6.3 Modelización mediante Mezcla de Gaussianas

6.4 Expectation-Maximization y detección de anomalías

TEMA 7: SESGOS, ÉTICA Y USO RESPONSABLE DE LA IA

7.1 Sesgos en el pipeline de aprendizaje automático: datos, representación, modelo y uso

7.2 Desequilibrio de clases, subgrupos, proxies e impacto desigual

7.3 Uso responsable de la IA generativa en proyectos de aprendizaje computacional

7.4 Data cards, model cards, trazabilidad y condiciones de no uso

TEMA 8: EVALUACIÓN, PROYECTO Y DEFENSA DE DECISIONES

8.1 Evaluación de resultados: métricas, errores, evidencias y limitaciones

8.2 Comparación de modelos y justificación de decisiones técnicas

8.3 Documentación del proyecto: repositorio, informe, model card y declaración de IA

8.4 Defensa oral: evidencias, riesgos, mitigaciones y decisión final

Actividades formativas y Metodología

Título	Horas	ECTS	Resultados de aprendizaje
MD1: Resolución de problemas	10	0,4	1, 3, 5, 6
MD3: Explicación en python de un caso práctico de aprendizaje computacional	32	1,28	1, 3, 5, 6
MD0: Contenido teórico y seminarios	12	0,48	3, 4, 5, 6, 7
MD0: Estudio individual	12	0,48	1, 3, 4, 5, 6, 7
MD1: Resolución de problemas (individual)	24	0,96	1, 3, 4, 5, 6, 7
MD3: Caso Kaggle	12	0,48	1, 2, 3, 4, 5, 6, 7, 8
MD2: Resolución de proyectos prácticos	6	0,24	1, 2, 3, 4, 5, 6, 8
MD3: Caso Kaggle	12	0,48	1, 3, 5, 6, 7
MD2: Resolución de casos prácticos (en grupo)	16	0,64	1, 2, 3, 4, 5, 6, 8
MD2: Programación de proyectos	6	0,24	1, 2, 3, 4, 6, 8

Toda la información de la asignatura y los documentos relacionados que los estudiantes necesiten se encontrarán en la página de Caronte (<https://caronte.uab.cat/course/view.php?id=95>), el menú de la asignatura Aprendizaje Computacional (102787). Servirá para poder ver los materiales, gestionar los grupos

de prácticas, hacer las entregas correspondientes, ver las notas, comunicarse con los profesores, etc. Para poder utilizarlo es necesario hacer los siguientes pasos:

- Darse de alta como usuario dando el nombre, NIU, y una foto carnet en formato JPG. Si ya se ha dado de alta por alguna otra asignatura, no es necesario volver a hacer, puede ir al siguiente paso.
- Inscribirse en el tipo de docencia "Aprendizaje Computacional (102787)", dando como código de asignatura "apc2026" (sin las comillas).

En el desarrollo de la asignatura se podrán diferenciar siete tipos de actividades docentes:

MD0 Exposición de contenidos de teoría: Presentación de los contenidos teóricos a trabajar en la asignatura (disponible en <https://ddd.uab.cat/record/328857>). Estos contenidos tendrán que haberse preparado antes de la clase a partir de la lectura de textos, búsqueda de información, etc. Los contenidos presentados estarán directamente relacionados con los problemas, proyectos y seminarios propuestos en otras actividades docentes, de modo que serán la base sobre la que se desarrollarán otras actividades del curso. Los contenidos se encontrarán en la página de Caronte (presentaciones y videos) y constarán de dos partes: una presentación en la que se exponen los principales conceptos teóricos y matemáticos relacionados con tareas concretas de aprendizaje computacional (este temario será la base del examen teórico de la asignatura, ver apartado evaluación de esta guía docente), y una segunda parte de código en python sobre Jupyter notebooks que ejemplifican los detalles de codificación y de librerías para implementar en un caso práctico los principales conceptos vistos en la hora anterior. El alumnado podrá ver los vídeos de las clases, descargar las presentaciones y los notebooks de python y probar todos los códigos en su ordenador, para hacer las pruebas que hagan falta y poder jugar con los diversos parámetros para acabar de entender las razones de los distintos rendimientos y precisiones que se alcanzan en una base de datos específica con configuraciones concretas de los algoritmos explicados en la asignatura.

MD1 Resolución de problemas de computación: Entrega de hasta un máximo de 3 problemas implementados en un Jupyter Notebook. Todos los temas de teoría irán acompañados de una relación de notebooks, a partir de los cuales el estudiante tendrá que trabajar en sesiones de problemas y entregar opcionalmente. Estas actividades deben permitir al estudiante profundizar en la comprensión y personalizar el conocimiento teórico en un caso numérico concreto. Se plantearán pues unos ejemplos de datos que requieran el diseño de una solución en la que se utilicen los métodos vistos en las clases de teoría. Es imposible seguir las clases de problemas si no se siguen los contenidos de las clases de teoría. El resultado de estas sesiones es alcanzar las competencias necesarias para la resolución de problemas que se entregarán según el mecanismo específico para la entrega que se indicará en la página web de la asignatura (espacio Caronte).

MD2 Implementación de proyecto corto guiado grupal: Realización de 1 práctica guiada para profundizar en aspectos aplicados de la teoría. La parte práctica de la asignatura quedará completada con sesiones prácticas, donde las y los estudiantes tendrán que resolver problemas concretos de una cierta complejidad implementados en python. Estos proyectos se resolverán en pequeños grupos de 2-4 personas, donde cada miembro del grupo tendrá que hacer una parte y ponerla en común con el resto para tener la solución final. Estos grupos de trabajo se tendrán que mantener hasta mitad del curso y tendrán que autogestionarse: reparto de roles, planificación del trabajo, asignación de tareas, gestión de los recursos disponibles, conflictos, etc. Aunque el profesor guiará el proceso de aprendizaje, su intervención en la gestión de los grupos será mínima. Para desarrollar el proyecto, los grupos trabajarán de forma autónoma y las sesiones de prácticas deben dedicarse principalmente por el profesor a realizar el seguimiento del estado del proyecto, indicar errores a corregir, proponer mejoras, etc. Las dudas que puedan surgir para la realización de las prácticas se transmitirán a través del foro de Caronte, donde los demás alumnos podrán contestarles.

MD3 Resolución de un caso práctico Kaggle: cada grupo de 1-2 personas realizará un jupyter notebook donde se explicarán los distintos pasos realizados para la resolución de un problema de Aprendizaje Computacional. Los proyectos serán aplicados a bases de datos escogidas de la plataforma Kaggle (<https://www.kaggle.com/search?q=machine+learning>), y constarán de tres partes: una explicación de los atributos más importantes de la base de datos y del atributo a predecir/clasificar; breve descripción del método de aprendizaje computacional aplicado, junto con los parámetros escogidos; y una presentación de los resultados que se han obtenido. Ejemplos de jupyter notebooks se pueden encontrar en el siguiente repositorio: <https://datauab.github.io/>

MD4 Consultas y dudas: Horas de libre disposición para el estudiante para consultas y tutorías sobre aspectos en los que necesite ayuda adicional por parte del profesorado. Todas las consultas se realizarán vía online, mediante el foro de la asignatura, o correos a los profesores, por ejemplo. Se valorará que los estudiantes contesten a las dudas de sus compañeros así como que en estas respuestas aporten información que ayude en la comprensión del contenido de las actividades docentes.

MD5 Actividades de evaluación: para cada una de las actividades descritas anteriormente. Ver apartado de evaluación de esta guía docente.

En el caso de los repetidores, si se pide al profesor responsable, se les convalidarán las notas de las actividades docentes que hayan aprobado el curso anterior, en caso de que haya superado la nota mínima.

Nota: se reservarán 15 minutos de una clase dentro del calendario establecido por el centro o por la titulación para que el alumnado rellene las encuestas de evaluación de la actuación del profesorado y de evaluación de la asignatura o módulo.

Evaluación

Actividades de evaluación continuada

Título	Peso	Horas	ECTS	Resultados de aprendizaje
Documentación escrita, implementación y presentación del caso Kaggle	40%	2	0,08	1, 2, 3, 4, 5, 6, 8
Pruebas teóricas individuales	15%	4	0,16	1, 3, 4, 5, 6, 7
Entrega de problemas	25%	0	0	1, 3, 4, 5, 6, 7
Documentación escrita, implementación y presentación de la práctica	20%	2	0,08	1, 2, 3, 4, 5, 6, 8

Actividades e instrumentos de evaluación:

a) Proceso y actividades de evaluación programadas

La asignatura consta de las siguientes actividades de evaluación:

- MD0: Exámenes teóricos, donde por cada examen el alumno presencialmente deberá contestar individualmente y por escrito 5 preguntas (para desarrollar en una página máxima) sobre conceptos de aprendizaje computacional vistos en las clases de teoría. Representa el 15% sobre la calificación final, es opcional, individual y recuperable (habrá dos parciales y sus respectivas recuperaciones, sin penalización).
- MD1: Entrega de informe con hasta 3 problemas resueltos, donde cada alumno individualmente hará entrega de un informe escrito de hasta 3 problemas vistos en clases (regresión, clasificación, memorización y clustering). Representa el 25% sobre la calificación final, es opcional, individual y recuperable (sin penalización, durante la última semana lectiva anterior a las vacaciones de Navidad).
- MD2: Resolución de una práctica con entrega de informe explicando la resolución y los resultados, donde cada grupo compuesto por dos o cuatro personas, entregarán el código python, así como un informe de hasta 30 páginas donde describirán las bases de datos, la estrategia que han utilizado para analizar los datos, así como las pruebas con distintos valores de los parámetros que han probado y los resultados que han obtenido con la mejor configuración posible. Se realizará una presentación opcional de cada proyecto. Representa el 20% (código 5% + informe 10% + presentación 5%) sobre la calificación final, es opcional, grupal y es recuperable (sin penalización, durante la última semana lectiva del año anterior a las vacaciones de Navidad).

- MD3: Elaboración en grupos de una o dos personas de un proyecto python con la descripción del código y los resultados de un caso concreto de aprendizaje computacional (ya sea regresión, clasificación, clustering o memorización, asignado por el profesor de aquí: <https://www.kaggle.com/search?q=machine+learning>), y realización de una presentación opcional donde se describirá con código los datos, los modelos utilizados (con los parámetros que mejor funcionan para los datos), y los resultados del problema escogido. Ejemplos de jupyter notebooks aplicados a casos concretos se pueden encontrar aquí <https://datauab.github.io/>. Representa un 40% de la calificación final (código 15% + informe 15% + github 5% + presentación 5%), es opcional, grupal y NO es recuperable.

A continuación se describe cómo poder aprobar la asignatura con evaluación continuada:

MD0: Exámenes teóricos individuales

La nota final de teoría se calculará a partir de dos exámenes parciales:

$$\text{Nota Teoría} = (0.5 * \text{Parcial1}) + (0.5 * \text{Parcial2})$$

Parcial1 se realiza a mitad de semestre y sirve para eliminar parte de la materia si es aprobada. Parcial2 se realiza al final del semestre lectivo y sirve para eliminar la parte del temario que viene después de Parcial1.

Estos exámenes pretenden una evaluación individualizada del estudiante con sus capacidades de contestar a 5 preguntas largas (desarrollar hasta ocupar una página de folio máximo) sobre las técnicas explicadas en clase, así como evaluar el nivel de conceptualización que el estudiante ha hecho técnicas vistas.

Para aprobar la parte de teoría de la asignatura haciendo exámenes presenciales, será necesario cumplir dos requisitos:

- será necesario que las notas de los parciales 1 y 2 sean igual o superior a 4.0 (en ambos parciales). En caso de que se saque menos de un 4.0 en alguno de los dos Parciales, deberá volver a realizarse el parcial que corresponda durante el examen de recuperación.
- la nota final de teoría debe ser mayor o igual que 4.0. En caso de que la nota de teoría final no sea igual o superior a 4.0, los estudiantes pueden presentarse al examen de recuperación para ser evaluados de todos los contenidos vistos en la asignatura.

Examen de recuperación (finales de enero o principios de febrero). En este examen se puede recuperar el(los) parcial(es) que no haya(n) superado el 4.0, o recuperar todo el temario en caso de que la nota final de teoría no supere el 4.0.

MD1: Entrega individual de un informe con hasta 3 problemas resueltos

Los problemas tienen como objetivo provocar que el estudiante entre con los contenidos de la asignatura de forma continuada y, a partir de pequeños problemas, que se familiarice directamente en la aplicación de la teoría. Como evidencia de este trabajo se pide la presentación obligatoria de un portfolio en el que habrá ido guardando los problemas que habrá ido realizando.

Nota Problemas = Evaluación del portfolio con 3 problemas resueltos. Pueden recuperarse entregando por Caronte la última semana de clase anterior a las fiestas de Navidad, sin penalización.

MD2: Resolución de 1 práctica en grupo

La evaluación del 2 proyecto de prácticas incluirá:

- Evaluación conjunta de cada proyecto: nota única para todos los miembros del grupo de trabajo que valorará el resultado global del proyecto, la calidad del código, la estructura general de la presentación

final y los documentos entregados a lo largo del proyecto.

- Evaluación individual: se valorará el trabajo individual a partir de las respuestas a las preguntas en las sesiones de control online, de la presentación final del proyecto online y principalmente de la participación activa en los foros de Caronte. En los casos requeridos por cualquier grupo (en casos de incidencias entre compañeros), se evaluará un breve formulario confidencial calificando la contribución de cada compañero de grupo al resultado final.

La nota del proyecto se calculará según la fórmula:

$$\text{Nota Prácticas} = (0.2 * \text{Programa}) + (0.2 * \text{Presentación}) + (0.6 * \text{Documentación})$$

En casos muy justificados (p.ej. PIUNE, por cuestiones laborales, familiares o de salud,...), los grupos pueden ser de 1 persona.

MD3: Realización de un caso concreto de aprendizaje computacional de la plataforma Kaggle

La evaluación se basará en el código python y la explicación del código y los resultados que se encontrará en un repositorio github. La nota del caso kaggle se calculará según la fórmula:

$$\begin{aligned} \text{Nota Caso Kaggle} = & (0.1 * \text{Introducción a la base de datos}) + (0.25 * \text{Análisis de los atributos, correlaciones,...}) \\ & + (0.25 * \text{Descripción del método utilizado, como encontrar los mejores parámetros, comparativa de métodos...}) \\ & + (0.3 * \text{Descripción de los resultados, matrices de confusión, gráficas de los modelos y datos, ejemplos de falsos positivos/negativos, curvas ROC, ...}) + (0.1 * \text{Presentación del repositorio Github}) \end{aligned}$$

Ejemplos de jupyter notebooks aplicados a casos concretos se pueden encontrar aquí <https://datauab.github.io/>.

Evaluación de competencias transversales

Los exámenes parciales permitirán evaluar su adquisición de hábitos de pensamiento y de trabajo personal (T01 Evaluar de forma crítica y con criterios calidad el trabajo realizado, Nota Teoría). Con Nota Proyecto de prácticas, también se evaluará el trabajo en equipo (T02 Trabajar cooperativamente en un contexto multidisciplinar asumiendo respetando el rol de las diferentes miembros del equipo, Nota Grup). Con la realización de los problemas y la realización de un caso Kaggle, se evaluará la adquisición de hábitos para solucionar una tarea predeterminada con unos valores de datos totalmente diferentes a los vistos en clase (T04 Utilizar eficazmente la bibliografía y los recursos electrónicos para obtener información, Nota Problemas y Caso Kaggle).

La nota final de la asignatura se obtiene combinando la evaluación de estas 4 actividades de la siguiente forma:

$$\text{Nota Final} = (0.15 * \text{Teoría}) + (0.20 * \text{Práctica}) + (0.25 * \text{Problemas}) + (0.40 * \text{Caso Kaggle})$$

Condiciones para aprobar:

Todas las actividades evaluativas son opcionales, para aprobar la asignatura es necesario que la suma de la evaluación de cada una de las actividades supere los 5 puntos. En caso de no superar la asignatura, la nota numérica del expediente será el valor menor entre 4.5 y la media ponderada de las notas alcanzadas:

- La nota final de teoría MD0 debe ser mayor o igual que 4.0 para poder sumar la parte de teoría a la nota final de la asignatura.
- La nota del proyecto MD2 debe ser mayor o igual que 5.0 para poder sumar la parte de la práctica a la nota final de la asignatura.

- La nota final de la asignatura debe ser mayor o igual que 5.0 para poder aprobar la asignatura.

Si el alumno se presenta a algún examen o si entrega alguna práctica, ya no podrá ser evaluado como "No Evaluable" en caso de que no se presente a ninguna de las otras evaluaciones, sino que se le calculará la nota final a partir de aquellas evaluaciones continuadas en las que se haya presentado.

b) Programación de actividades de evaluación

Las fechas de evaluación continua y entrega de trabajos se publicarán en el Caronte (<http://caronte.uab.cat/>), en el espacio de esta asignatura y pueden estar sujetos a cambios de programación por motivos de adaptación a posibles incidencias ; siempre se informará a caronte.uab.cat sobre estos cambios ya que esta plataforma se convertirá en el mecanismo habitual de intercambio de información entre profesor y estudiantes.

Además, existe un protocolo de «solicitud de reprogramación de actividades de evaluación» según los casos especificados en [los criterios e instrucciones de evaluación de la Escuela de Ingeniería](#).

c) Proceso de recuperación

El estudiante puede presentarse a la recuperación siempre que se haya presentado a un conjunto de actividades que representen un mínimo de dos terceras partes (4 actividades de evaluación sobre 7 total: 2 exámenes de teoría parciales; 1 evaluación de la práctica; 3 entregas de problemas, y 1 entrega del caso kaggle) de la calificación total de la asignatura.

De éstos, se podrán presentar a la recuperación aquellos estudiantes que tengan como media de todas las actividades de la asignatura una calificación superior a 3.0.

Hay que tener en cuenta que la Resolución del Caso Kaggle (MD3) no es recuperable.

d) Procedimiento de revisión de las calificaciones

Para cada examen teórico individual, se indicará un lugar, fecha y hora de revisión en la que el estudiante podrá revisar la actividad con el profesor. En este contexto, se podrán realizar reclamaciones sobre la nota de la actividad, que serán evaluadas por el profesorado responsable de la asignatura. Si el estudiante no se presenta a esta revisión, no se revisará posteriormente esta actividad.

e) Calificaciones

Matrículas de Honor: Se concederán Matrículas de Honor a decisión del profesorado responsable de la asignatura, hasta llegar al cinco por ciento o fracción de los alumnos matriculados en todos los grupos de docencia de la asignatura. La normativa de la UAB indica que las MH sólo podrán otorgarse a estudiantes que hayan obtenido una calificación final igual o superior a 9.00.

No evaluable: Un estudiante se considerará no evaluable (NA) si no se ha presentado en ninguno de los exámenes parciales ni en ninguna de las 2 evaluaciones de las prácticas.

f) Irregularidades por parte del estudiante, copia y plagio

Sin perjuicio de otras medidas disciplinarias que se estimen oportunas, se calificarán con un cero las irregularidades cometidas por el estudiante que puedan conducir a una variación de la calificación de un acto

de evaluación. Por tanto, la copia, el plagio, el engaño, dejar copiar, etc. en cualquiera de las actividades de evaluación supondrá suspenderla con un cero. Las actividades de evaluación calificadas de esta forma y por este procedimiento no serán recuperables. Si es necesario superar cualquiera de estas actividades de evaluación para aprobar la asignatura, esta asignatura quedará suspendida directamente, sin oportunidad de recuperarla en el mismo curso. Estas irregularidades incluyen, entre otras:

- la copia total o parcial de una práctica, informe, o cualquier otra actividad de evaluación;
- dejar copiar;
- presentar un trabajo de grupo no realizado íntegramente por los miembros del grupo (aplicado a todos los miembros, no sólo a los que no han trabajado);
- presentar como propios materiales elaborados por un tercero, aunque sean traducciones o adaptaciones, y en general trabajos con elementos no originales y exclusivos del estudiante;
- tener dispositivos de comunicación (como teléfonos móviles, smart watches, bolígrafos con cámara, etc.) accesibles durante las pruebas de evaluación teórico-prácticas individuales (exámenes);
- hablar con compañeros durante las pruebas de evaluación teórico-prácticas individuales (exámenes);
- copiar o intentar copiar de otros alumnos durante las pruebas de evaluación teórico-prácticas (exámenes);
- usar o intentar usar escritos relacionados con la materia durante la realización de las pruebas de evaluación teórico-prácticas (exámenes), cuando éstos no hayan sido explícitamente permitidos.

La nota numérica del expediente será el valor menor entre 3.0 y la media ponderada de las notas en caso de que el estudiante haya cometido irregularidades en un acto de evaluación (y por tanto no será posible el aprobado por compensación). En ediciones futuras de esta asignatura, al estudiante que haya cometido irregularidades en un acto de evaluación no se le convalidará ninguna de las actividades de evaluación realizadas.

En resumen: copiar, dejar copiar o plagiar (o el intento de) en cualquiera de las actividades de evaluación equivale a un SUSPENSO, no compensable ni recuperable y sin convalidaciones de partes de la asignatura en cursos posteriores.

g) Uso de la IA

En esta asignatura, se permite el uso de tecnologías de Inteligencia Artificial (IA) como parte integrante del desarrollo del trabajo, siempre que el resultado final refleje una contribución significativa del estudiante en el análisis y la reflexión personal. El estudiante deberá identificar claramente qué partes han sido generadas con esta tecnología, especificar las herramientas utilizadas e incluir una reflexión crítica sobre cómo estas han influido en el proceso y el resultado final de la actividad. La falta de transparencia en el uso de la IA se considerará una falta de honestidad académica y puede conllevar una penalización en la calificación de la actividad o sanciones mayores en casos graves.

h) Evaluación de los estudiantes repetidores

A partir de la segunda matrícula, la evaluación de la asignatura consistirá en el examen teórico individual, añadiendo las notas correspondientes a las MD obtenidas la primera vez que el estudiante se ha matriculado de la asignatura, siempre que las notas de prácticas MD2 sean superiores o iguales a 5.0.

Para poder optar a esta evaluación diferenciada, el estudiante repetidor debe pedirlo al profesor a más tardar hasta la semana 5.

i) Evaluación única

La evaluación única de la asignatura constará de las siguientes actividades de evaluación que será necesario entregar el día del segundo parcial del examen de teoría:

- MD0, evaluación de la teoría, 30% sobre la calificación final, recuperable.
- MD1, entrega de los problemas, 30% sobre la calificación final, recuperable.
- MD3, entrega del caso Kaggle (repositorio github), 40% sobre la calificación final, no recuperable.

Para MD0 y MD1, se aplicará el mismo sistema de recuperación que para la evaluación continua: se podrá realizar el examen de recuperación de teoría, y entregar los problemas a través de Caronte durante el mismo día de la recuperación de la teoría.

Además, la revisión de la calificación final sigue el mismo procedimiento que para la evaluación continua.

j) Alumnado con discapacidad o necesidades específicas de apoyo educativo (NESE)

El alumnado con discapacidad o necesidades específicas de apoyo educativo (NESE) puede ponerse en contacto con el Servicio de Igualdad y Diversidad de la UAB, que ofrece apoyo específico para el desarrollo de una vida académica y social plena y autónoma en la universidad.

Bibliografía

Enlaces web

- Caronte: <http://caronte.uab.cat>
- Libro de la asignatura: <https://ddd.uab.cat/record/328857>

Bibliografía básica

- Artificial Intelligence: A Modern Approach. <http://aima.cs.berkeley.edu/>
- S. Russell, P. Norvig. Artificial Intelligence: A Modern Approach. Ed. Prentice Hall, Second Edition, 2003. (Existeix traducció al castellà: Inteligencia artificial: Un Enfoque Moderno)

Bibliografía complementaria

- L. Igual, S. Seguí. Introduction to Data Science. Ed. Springer, 2017
- Bishop, Pattern Recognition and Machine Learning, 2007.
- Duda, Hart, and Stork, Pattern Classification, 2nd Ed., 2002.
- Marlsand, Machine Learning: an Algorithmic Perspective, 2009
- Mitchell, Machine Learning, 1997
- Ripley, Pattern Recognition and Neural Networks, 1996.

Bibliografía relacionada

- Eberhart, Shi, Computational Intelligence: Concepts to Implementations, 2007
- Friedman, Tibshirani, The Elements of Statistical Learning, 2009.
- Gilder, Kurzweil, Richards, Are we spiritual machines? RayKurzweil vs. the Critics of Strong AI, 2011
- Kurzweil, The Singularity is Near: When Humans transcend Biology, 2006
- Rosen, Life Itself: A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life (Complexity in Ecological Systems), 2005
- Witten, Frank, Hall, Data Mining: Practical Machine Learning Tools and Techniques, 2011

Software

El software necesario será el lenguaje de programación Python, un entorno de programación (como Spyder, Pycharm o Visual Studio Code), la aplicación web Jupyter Notebook, y las librerías necesarias para análisis de datos: scipy (contiene NumPy, matplotlib, Pandas), sklearn y Seaborn.

Grupos e idiomas de la asignatura

La información proporcionada es provisional hasta el 30 de noviembre. A partir de esta fecha, podrá consultar el idioma de cada grupo a través de este [enlace](#). Para acceder a la información, será necesario introducir el CÓDIGO de la asignatura

Tipo de docencia	Grupo	Idioma	Semestre	Turno
(TE) Teoría	440	Catalán	primer cuatrimestre	mañana-mixto
(PAUL) Prácticas de aula	441	Catalán	primer cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	441	Catalán	primer cuatrimestre	mañana-mixto
(PAUL) Prácticas de aula	442	Catalán	primer cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	442	Catalán	primer cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	443	Catalán	primer cuatrimestre	tarde
(PLAB) Prácticas de laboratorio	444	Catalán	primer cuatrimestre	mañana-mixto