



FI083 – Reconocimiento de Patrones

Nombre del programa académico	Ingeniería Electrónica
Nombre completo de la asignatura	Reconocimiento de Patrones
Área académica o categoría	Ingeniería Electrónica
Semestre y año de actualización	2023-2
Semestre y año en que se imparte	2023-2
Tipo de asignatura	[x] Obligatoria [] Electiva
Número de créditos	3
Director o contacto del programa	Arley Bejarano Martínez
Coordinador o contacto de la asignatura	Yurley Tatiana Tovar Martínez

Descripción y contenidos

1. Breve descripción Asignatura teórico-práctica cuyo propósito es el análisis e implementación de sistemas de aprendizaje de máquina para tareas de clasificación y regresión.
2. Objetivos • Comprender los fundamentos del reconocimiento de patrones. • Aplicar técnicas de aprendizaje para tareas de clasificación y regresión. • Evaluar y optimizar modelos de aprendizaje de máquina.
3. Resultados de aprendizaje • RAA1: Implementa sistemas de aprendizaje de máquina para solucionar problemas de clasificación. Se corresponde con los RAP 1, 5. • RAA2: Implementa sistemas de aprendizaje de máquina para solucionar problemas de regresión. Se corresponde con los RAP 1, 5. • RAA3: Utiliza software especializado para desarrollar sistemas de aprendizaje de máquina. Se corresponde con el RAP 2.
4. Contenido • Introducción a Clasificación y Regresión (4 H) ○ Definición de clasificación (etiquetas) y regresión (valores continuos). ○ Ejemplos: detección de rostros (clasificación), predicción de precios (regresión). • Métodos Clásicos de Clasificación y Regresión (16 H) ○ k-NN (k-Vecinos Más Cercanos). ○ Árboles de Decisión. ○ SVM ○ Regresión Lineal. ○ Regresión Polinómica. ○ Regularización (Ridge, Lasso). • Redes Neuronales Básicas (12 H) ○ Redes multicapa (MLP) para clasificación y regresión. ○ Funciones de activación: ReLU y Sigmoide. • Redes Neuronales Convolucionales (CNN) (12 H) ○ Concepto de convoluciones para clasificación de imágenes. ○ Pooling (reducción de dimensionalidad). • Redes Recurrentes (RNN) y LSTM (12 H) ○ RNN para datos secuenciales (texto, series temporales).



○ LSTM para mejorar la memoria a largo plazo en secuencias. • Transformers (12 H) ○ Introducción a la autoatención y manejo de secuencias largas. • Evaluación de Modelos (4 H) ○ Métricas de clasificación: Precisión, F1-score. ○ Métricas de regresión: MSE, MAE.
5. Requisitos: FI953-INTRODUCCIÓN A LA CIENCIA DE DATOS
6. Recursos • Christopher M. Bishop. Pattern Recognition and Machine Learning. Springer. 1ra Edición, 2006. • Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. MIT Press. 1ra Edición, 2012. • Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning. MIT Press. 1ra Edición, 2016. • Software Python • Repositorios de aprendizaje de máquina en internet.
7. Herramientas técnicas de soporte para la enseñanza • Clase magistral. • Estudio y análisis de casos prácticos. • Realización de trabajos individuales. • Talleres para solución en grupo sobre cada contenido de la materia. • Consultas utilizando la bibliografía del curso. • Proyecto integrador de asignaturas desarrollando software para solucionar problemas de ingeniería utilizando aprendizaje de máquina. • Uso de videotutoriales para apoyar los conceptos vistos en clase. • Creación de sitio web o aula virtual para la comunicación con estudiantes y almacenamiento de material de clase.
8. Trabajos en laboratorio y proyectos • Esta materia contempla hasta 2 horas en la semana en la que se desarrollaran programas sistemas de aprendizaje de máquina para desarrollar habilidades según el contenido abordado durante la semana de clase. • Esta materia contempla un proyecto final de curso en el que el estudiante debe utilizar los conceptos aprendido en clase para resolver un problema real.
9. Métodos de aprendizaje • Este curso se desarrollará con la interacción directa estudiante-profesor en la explicación de los conceptos básicos de la teoría para el desarrollo de programas de computador apoyándose con asignación de trabajo extra-clase, talleres de clase. • Como una herramienta de refuerzo de contenidos, se realizará una asignación material complementario, para lograr el entendimiento del contenido, así como el refuerzo de los resultados de aprendizaje. • Se realizará el diseño e implementación de un sistema experto elegido por los estudiantes estimulando de manera activa la participación del estudiante a partir de interrogantes que generan tanto la creatividad, como el pensamiento crítico y reflexivo del educando, así como el estímulo del trabajo en equipo y la división de roles.
10. Métodos de evaluación Teniendo en cuenta el Acuerdo 29 de 2006 del Consejo Académico las evaluaciones se llevarán a cabo en los siguientes momentos: • Evaluación Parcial 1 (Porcentaje de la materia: 35%): Examen con problemas de clasificación y regresión utilizando aprendizaje clásico, se evalúa los RAA1, RAA2y RAA3. Esta evaluación se realiza antes de la semana 8 del periodo académico. • Evaluación Parcial 2 (Porcentaje de la materia: 35%): Examen con problemas de clasificación y regresión utilizando aprendizaje profundo, se evalúa los RAA1, RAA2 y RAA3. Esta evaluación se realiza antes de la semana 8 del periodo académico. • Evaluación proyecto Final (Porcentaje de la materia 30%): Proyecto de implementación de un sistema experto en Python que debe resolver un problema de la sociedad o la industria. Esta



evaluación se realiza en la semana de finales. Se evalúa los resultados de aprendizaje: RAA1, RAA2y RAA3.

• **Métodos de evaluación**

- Evaluación Parcial 1 (Porcentaje de la materia: 20%): Examen escrito con preguntas teóricas sobre la arquitectura de sistemas embebidos, incluyendo microcontroladores y minicomputadoras, pidiendo a los estudiantes que identifiquen y describan componentes específicos como la CPU, memoria, registros y buses. RAA-1.
- Practicas (Porcentaje de la materia: 20%): Prácticas, talleres, exposiciones y trabajos escritos puesto en clase. RAA-1, RAA-2 y RAA-3.
- Evaluación Parcial 2 (Porcentaje de la materia: 25%): Examen con problemas prácticos donde los estudiantes deben desarrollar firmware utilizando lenguajes como C y C++, optimizando los recursos del hardware para dar solución a problemas reales.: RAA-2.
- Evaluación Final (Porcentaje de la materia 35%): Presentación de un proyecto final desarrollado utilizando firmware en lenguajes como C, C++ o Python, donde se debe resolver un problema real integrando diversos periféricos del sistema embebido, como GPIO, ADC, y módulos de comunicación serie (UART, I2C, SPI, RS485). Se evalúa el resultado de aprendizaje: RAA-3.

