

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
COORDINACIÓN GENERAL DE FORMACIÓN BÁSICA
COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA
PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Mexicali; Facultad de Ingeniería, Arquitectura y Diseño, Ensenada; y Facultad de Ciencias Químicas e Ingeniería, Tijuana.
- 2. Programa Educativo:** Ingeniero en Computación
- 3. Plan de Estudios:** 2020-1
- 4. Nombre de la Unidad de Aprendizaje:** Cómputo Suave
- 5. Clave:** 36316
- 6. HC:** 02 **HL:** 02 **HT:** 00 **HPC:** 00 **HCL:** 00 **HE:** 02 **CR:** 06
- 7. Etapa de Formación a la que Pertenece:** Terminal
- 8. Carácter de la Unidad de Aprendizaje:** Optativa
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



Equipo de diseño de PUA

Mauricio Alonso Sánchez Herrera
Olivia Mendoza Duarte

**Vo.Bo. de Subdirectores de
Unidades Académicas**

Alejandro Mungaray Moctezuma
Humberto Cervantes De Ávila
Rocío Alejandra Chávez Santoscoy

Fecha: 17 de octubre de 2019

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La finalidad de esta unidad de aprendizaje es brindar los antecedentes, fundamentos y aplicaciones de los paradigmas de cómputo suave y las herramientas de software requeridas para experimentar con los mismos.

Su utilidad radica en que le permitirá al estudiante resolver problemas de ingeniería en patrones, control, toma de decisiones y optimización a partir de herramientas de software especializado en redes neuronales, sistemas difusos y algoritmos evolutivos.

La asignatura Cómputo Suave se imparte en la etapa terminal con carácter optativo y pertenece al área de conocimiento Ingeniería Aplicada. Es recomendable haber adquirido conocimientos y habilidades referentes a programación estructurada, métodos numéricos, algoritmos y estructura de datos e inteligencia artificial antes de cursar esta asignatura.

III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE

Analizar los paradigmas de cómputo suave, sus fundamentos y su experimentación, para aplicarlos en la resolución de problemas de reconocimiento de patrones, control, toma de decisiones y optimización en el área de ingeniería, utilizando herramientas de software especializadas en redes neuronales, sistemas difusos y algoritmos evolutivos, con objetividad y responsabilidad hacia la sociedad.

IV. EVIDENCIA(S) DE DESEMPEÑO

Elabora reporte técnico donde se describan experimentos donde se apliquen los paradigmas de cómputo suave para la solución a un problema de ingeniería.

V. DESARROLLO POR UNIDADES

UNIDAD I. Cómputo suave

Competencia:

Explicar los conceptos fundamentales relacionados con cómputo suave, su entorno y contexto histórico, mediante su análisis y revisión bibliográfica, para obtener una visión general de la importancia del desarrollo de estos paradigmas, con interés y actitud crítica.

Contenido:

- 1.1 Cómputo duro vs cómputo suave
- 1.2 Contexto histórico
- 1.3 Resolución de problemas dentro de la naturaleza
- 1.4 Cómputo bio-inspirado

Duración: 4 horas

UNIDAD II. Redes Neuronales

Competencia:

Aplicar el paradigma de redes neuronales, mediante su interpretación y experimentación con diferentes arquitecturas, para resolver problemas clásicos de aprendizaje de patrones, de manera organizada y coherente.

Contenido:

Duración: 12 horas

- 2.1 Redes neuronales de una capa
 - 2.1.1 Perceptrón
 - 2.1.2 Funciones de activación
- 2.2 Redes neuronales multi-capas
 - 2.2.1 Entrenamiento en línea a fuera de línea
 - 2.2.2 Validación cruzada y generalización
- 2.3. Aplicaciones
 - 2.3.1 Regresión
 - 2.3.2 Predicción con retardos
 - 2.3.3 Clasificación
- 2.4 Otros tipos de redes neuronales

UNIDAD III. Sistemas difusos

Competencia:

Implementar el paradigma de sistemas difusos, a través de su interpretación y experimentación con diferentes parámetros y heurísticas, para resolver problemas clásicos de control y toma de decisiones, de manera proactiva y organizada.

Contenido:**Duración:** 8 horas

- 3.1 Funciones de membresía
- 3.2 Variables lingüísticas
- 3.3 Inferencia difusa
- 3.4 Reglas difusas
- 3.5 Aplicaciones
 - 3.5.1 Toma de decisiones
 - 3.5.2 Control difuso

UNIDAD IV. Cómputo evolutivo

Competencia:

Aplicar el paradigma de cómputo evolutivo, mediante su interpretación y experimentación con diferentes parámetros, para resolver problemas clásicos de optimización, de manera analítica y proactiva.

Contenido:**Duración:** 8 horas

- 4.1 Fundamentos e ideas básicas
- 4.2 Representación, búsqueda y operadores
- 4.3 Optimización evolutiva
- 4.4 Aplicaciones

VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No. de Práctica	Competencia	Descripción	Material de Apoyo	Duración
UNIDAD I				
1	Identificar los paradigmas de cómputo suave, mediante una investigación bibliográfica, para adquirir una noción general de su desarrollo en el tiempo, de forma proactiva y organizada.	Investiga los avances de los paradigmas de cómputo suave a través del tiempo, y realiza una presentación electrónica por equipo.	Equipo de cómputo con conexión a internet, consulta de base de datos y/o libros electrónicos.	4 horas
UNIDAD II				
2	Desarrollar soluciones a problemas de regresión, predicción con retardos y de clasificación, mediante la experimentación con diferentes arquitecturas de redes neuronales, para comparar su rendimiento, de forma sistemática y eficiente.	Experimenta con casos de estudio tipo Benchmark que se puedan tratar con regresión, usando herramientas de software especializadas en redes neuronales, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en redes neuronales.	4 horas
3		Experimenta con casos de estudio tipo Benchmark que se puedan tratar con predicción de series de tiempo, usando herramientas de software especializadas en redes neuronales, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en redes neuronales.	4 horas
4		Experimenta con casos de estudio tipo Benchmark que se puedan tratar con clasificación, usando herramientas de software especializadas en redes neuronales, de forma individual, y entregando la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en redes neuronales.	4 horas

UNIDAD III				
5	Desarrollar soluciones a problemas de toma de decisiones y de control, mediante la experimentación con diferentes parámetros de diseño de sistemas difusos, para comparar su rendimiento, de forma sistemática y eficiente.	Experimenta con casos de estudio tipo Benchmark para toma de decisiones, usando herramientas de software especializadas en sistemas difusos, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en sistemas difusos.	4 horas
6		Experimenta con casos de estudio tipo Benchmark que se puedan tratar con controladores, usando herramientas de software especializadas en sistemas difusos, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en sistemas difusos.	4 horas
UNIDAD IV				
7	Optimizar soluciones a problemas de toma de decisiones y matemáticos, mediante la experimentación con diferentes parámetros de diseño de algoritmos evolutivos, para comparar su rendimiento, de forma sistemática y eficiente.	Experimenta con casos de estudio tipo Benchmark para toma de decisiones, usando herramientas de software especializadas en algoritmos evolutivos, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en algoritmos evolutivos.	4 horas
8		Experimenta con casos de estudio tipo Benchmark para toma de decisiones, usando herramientas de software especializadas en algoritmos evolutivos, de forma individual, y entrega la solución en un archivo digital.	Equipo de cómputo con conexión a internet y herramientas de software especializadas en algoritmos evolutivos.	4 horas

VII. MÉTODO DE TRABAJO

Encuadre: El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

Estrategia de enseñanza (docente)

Presentaciones para demostración de metodologías, técnicas y plataformas diversas, casos de estudio de referencia (benchmark) para aplicación del conocimiento adquirido y desarrollo de habilidades.

Estrategia de aprendizaje (alumno)

Investigar, presentar, interactuar con plataformas, redactar reportes técnicos y desarrollar actividades de talleres, así como un proyecto final como evidencia de desempeño que integre tanto conocimiento como habilidades adquiridas durante el curso.

VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

Criterios de acreditación

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Exámenes escritos.....	50%
- Prácticas del laboratorio.....	30%
- Evidencia de desempeño (Reporte técnico)	20%
Total.....	100%

IX. REFERENCIAS

Básicas

Ghaboussi, J. (2018). *Soft Computing in Engineering*. Estados Unidos: CRC Press.

Jang, J.-S. R., Sun, C.-T., y Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Estados Unidos: Pearson. Recuperado de http://www.soukalfi.edu.sk/01_NeuroFuzzyApproach.pdf [clásica]

Keller, J. M., Liu, D., y Fogel, D. B. (2019). *Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation*. Wiley-IEEE Press.

Complementarias

Bustine, H. (Ed.) (2019). *Mathware & Soft Computing Magazine*. España: European Society for Fuzzi Logic and Technology. Recuperado de <http://www.eusflat.org/msc/index.php>

Díaz-Cortés, M.A., Cuevas, E., y Rojas, R. (2017). *Engineering Applications of Soft Computing*. Springer

X. PERFIL DEL DOCENTE

El docente que imparta esta asignatura deberá poseer título de licenciatura en Ingeniero en Computación o afín, preferentemente grado de doctor en Computación o afín. Experiencia mínima de dos años en investigación en el área de cómputo suave y preferentemente dos años en docencia. Que desempeñe su labor con profesionalismo y tolerancia.