FACULTAD DE INGENIERIA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Programa de

TALLER DE APRENDIZAJE AUTOMÁTICO

1. NOMBRE DE LA UNIDAD CURRICULAR

Taller de aprendizaje automático.

2. CRÉDITOS

10 créditos

3. OBJETIVOS DE LA UNIDAD CURRICULAR

Esta unidad curricular está diseñada como un segundo curso en aprendizaje automático, en el que se busca brindar herramientas conceptuales y metodológicas para desarrollar proyectos de aprendizaje automático. A lo largo del curso, los estudiantes abordarán problemas concretos, que deberán resolver de punta a punta, usando técnicas de aprendizaje automático. De este modo, pondrán en práctica los conocimientos adquiridos sobre datos reales y problemas relevantes, e incorporarán habilidades importantes para el desarrollo de su actividad profesional. Además de aplicar varias de las técnicas tradicionales de aprendizaje automático, la mayoría de las cuales se asumen conocidas por el estudiante, se estudiarán técnicas más recientes, en particular técnicas de aprendizaje profundo.

La unidad curricular propone como objetivos que el estudiante sea capaz de enfrentarse a un problema de aprendizaje automático tomando en cuenta los diferentes aspectos claves para su correcto abordaje, como el estudio y preparación de los datos, la selección de características, la selección de modelos y de sus hiper-parámetros, el ajuste fino de los modelos, el sobreajuste y el sub-ajuste a los datos, la correcta estimación de desempeño, así como la presentación de la solución, su puesta en marcha, despliegue y monitoreo.

La modalidad de taller y el abordaje de un proyecto en grupos busca el desarrollo de habilidades blandas como el trabajo en equipo, la planificación y gestión del tiempo, y la comunicación oral y escrita. Además se busca desarrollar capacidades para seguir aprendiendo de forma autónoma, a través de la revisión de artículos del estado del arte, el intercambio comunitario, el abordaje de nuevos desafíos y la elaboración de soluciones propias. Esto es imprescindible para poder acompañar la rápida evolución que tiene el área.

4. METODOLOGÍA DE ENSEÑANZA

El curso está organizado en dos clases semanales de dos horas. La mayoría de las clases serán prácticas, en modalidad de taller. Los estudiantes abordarán problemas específicos utilizando la metodología y las técnicas estudiadas. Desarrollarán habilidades prácticas relativas a la aplicación de técnicas de aprendizaje automático usando paquetes de software específicos (como Scikit-learn y TensorFlow). También existirán clases teóricas donde se discutirán aspectos conceptuales y metodológicos, así como algunas técnicas de aprendizaje automático. En las clases teóricas se presentarán los temas de forma expositiva, conectando los contenidos con ejemplos prácticos y problemas reales. Habrá espacios activos para la reflexión e intercambio, e instancias de presentación oral y escrita de los estudiantes.

Horas de teórico: 25
Horas de práctico: 35
Horas de evaluación: 6
Horas estimadas de dedicación no presencial: 84
Horas de dedicación total: approx. 150

5. TEMARIO

1. Fundamentos de aprendizaje automático: Repaso de conceptos básicos. Los componentes del problema de aprendizaje automático (conjunto de datos, función objetivo, conjunto de hipótesis, algoritmo de aprendizaje, función de costo). Tipos de aprendizaje: supervisado, no supervisado y por refuerzos. Compromisos de aproximación y generalización, sesgo y varianza. Sobreajuste y ajuste al ruido. Regularización. Validación. Selección de modelos.
2. Proyecto de aprendizaje automático: Desarrollo de un proyecto de aprendizaje automático de punta a punta. Definición del problema y requerimientos de la solución. Medidas de desempeño. Manejo, visualización y preparación de datos. Transformación y selección de características. Entrenamiento y selección de modelos. Ajuste fino y evaluación. Despliegue de la solución y monitoreo. Análisis de resultados. Interpretabilidad y explicabilidad. Consideraciones éticas.
3. Técnicas clásicas de aprendizaje automático: Implementación de una solución al problema usando técnicas clásicas, como regresión logística, máquinas de vectores de soporte, random forest, o redes neuronales. El énfasis está puesto en varias cuestiones prácticas como el pre-procesado de datos, detección de datos anómalos, validación cruzada, técnicas de regularización, la selección de hiper-parámetros, el análisis de error, el ajuste fino y la selección de modelos.
4. Técnicas de aprendizaje profundo: Implementación y entrenamiento de redes profundas. Inicialización, función de activación, normalización por lote, redes pre-entrenadas, optimizadores rápidos, regularización. Redes neuronales de convolución (CNN). Capas de convolución, filtros, pooling. Aumentado de datos. Aprendizaje por transferencia. Arquitecturas y aplicaciones en imágenes y audio. Redes neuronales recurrentes (RNN). Aplicaciones en procesamiento de series temporales y procesamiento de lenguaje natural. Mecanismo de atención y Transformers. Grandes modelos de lenguaje (LLM): GPT, Bert. Arquitectura encoder-decoder. Auto-encoders para aprendizaje de representaciones y como

modelos generativos. Redes generativas antagónicas (GANs), auto-encoders variacionales (VAEs), modelos de difusión.

6. BIBLIOGRAFÍA

6.1 Básica

1. Aurélien Géron (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media. 3d Ed.

6.2 Complementaria

2. Bishop, Christopher M. (2011). Pattern Recognition and Machine Learning. USA. Springer.
3. Yaser S. Abu-Mostafa, Malik Magdon-Ismail, Hsuan-Tien Lin (2012). Learning from data. USA. AMLBook.
4. Chollet, François (2017). Deep Learning with Python. Manning Publications, 1st ed.
5. Ian Goodfellow, Yoshua Bengio, Aaron Courville (2016). Deep Learning. MIT Press.

7. CONOCIMIENTOS PREVIOS EXIGIDOS Y RECOMENDADOS

7.1 Conocimientos Previos Exigidos: Aprendizaje supervisado. Sobreajuste. Compromiso aproximación-generalización. Compromiso sesgo-varianza. Regularización. Validación. Selección de modelos. Técnicas de clasificación y regresión. Aprendizaje no supervisado. Técnicas de agrupamiento. Ingeniería de características y reducción de dimensionalidad.

7.2 Conocimientos Previos Recomendados: Programación científica en python.

ANEXO A
Para todas las Carreras

A1) INSTITUTO

Instituto de Ingeniería Eléctrica

A2) CRONOGRAMA TENTATIVO

Consiste en un cronograma de avance semanal con detalle de las horas de clase asignadas a cada tema.

Semana 1	Introducción: Introducción al curso. Componentes del problema de aprendizaje. Tipos de aprendizaje. Repaso de conceptos fundamentales: generalización, sobreajuste, compromiso sesgo-varianza, regularización, validación, selección de modelos. (2 hs de clase) Desarrollo de proyecto de aprendizaje automático: Definición del problema, requerimientos, y medidas de desempeño. Manejo, visualización y preparación de datos. Transformación y selección de características. Entrenamiento y selección de modelos. Ajuste fino y evaluación. Despliegue y monitoreo. (2 hs de clase)
Semana 2	*Proyecto 1 Presentación:* Se presenta el primer proyecto del curso. Se implementan las primeras etapas, con énfasis en la creación de un pipeline de pre-procesado de datos en scikit-learn. (2 hs de clase) Clasificación con modelos clásicos: Se repasan algunas técnicas y conceptos básicos en clasificación multiclase, con énfasis en las medidas de desempeño y el análisis de error. Regresión logística. Descenso por gradiente. Regularización. Clasificación multiclase. Softmax. Matriz de confusión, precision-recall, curva ROC. (2 hs de clase)
Semana 3	*Proyecto 1 Clasificación con modelos clásicos:* Se aplican regresión logística y máquinas de vectores de soporte (SVM). El énfasis está puesto en técnicas de regularización, selección de hiper-parámetros, ajuste fino y análisis de error. (2 hs de clase) Ensambles de clasificadores: Métodos de combinación de clasificadores. Voting y bagging. Árboles de decisión y random forests. Boosting y stacking. (2 hs de clase)
Semana 4	*Proyecto 1 Clasificación con ensambles de clasificadores* Se aplican varios métodos de combinación de clasificadores, como Random forests, AdaBoost y gradient boosting. (2 hs de clase) Aprendizaje no supervisado: Se repasan algunas técnicas de aprendizaje no supervisado con énfasis en el pre-procesado de datos, aprendizaje semi-supervisado y detección de datos anómalos. (2 hs de clase)
Semana 5	*Proyecto 1 Ingeniería de características y limpieza de datos:* Se aplican métodos de reducción de dimensionalidad y de agrupamiento como pre-procesado de los datos y de detección de anómalos. (2 hs de clase) Clasificación con perceptrón multicapa (MLP): Se aplican métodos para seleccionar los hiper-parámetros en redes multicapa poco profundas y los

	parámetros del entrenamiento. (2 hs de clase) - ENTREGA 1 (modelos clásicos o ensambles)
Semana 6	*Proyecto 1 Clasificación con MLP:* Se aplican redes multicapa poco profundas y varios métodos para seleccionar los hiper-parámetros y los parámetros del entrenamiento. (2 hs de clase) Análisis de resultados e interpretabilidad: Análisis de resultados. Interpretabilidad y explicabilidad de la solución. Consideraciones éticas. Guía para la elaboración de un informe técnico. (2 hs de clase)
Semana 7	*Proyecto 1:* Se trabaja en el proyecto. (2 hs de clase) *Proyecto 1:* Se trabaja en el proyecto. (2 hs de clase)
Semana 8	*Proyecto 1:* Se trabaja en el proyecto, el informe y la defensa. (2 hs de clase) *Proyecto 1:* Se trabaja en el proyecto, el informe y la defensa. (2 hs de clase)
Semana 9	Redes neuronales profundas: Implementación de redes multicapa. Introducción al ambiente de trabajo Keras y TensorFlow 2. (2 hs de clase) Entrenando redes profundas: Desvanecimiento y explosión de gradiente. Inicialización, función de activación, normalización, redes pre-entrenadas, optimizadores rápidos, regularización. (2 hs de clase)
Semana 10	*Proyecto 2 Presentación:* Se presenta el segundo proyecto del curso. Se implementa la ingesta y pre-procesado de grandes cantidades de datos de forma eficiente a través de la Data API de TensorFlow. (2 hs de clase) Redes profundas personalizadas: Modelos y algoritmos de entrenamiento personalizados. Implementación de funciones de costo, funciones de activación, regularizadores, medidas de desempeño, capas especializadas, modelos y bucles de entrenamiento. (2 hs de clase)
Semana 11	*Proyecto 2 Redes profundas:* Implementación de redes profundas. Se ponen en práctica técnicas de entrenamiento de redes profundas. (2 hs de clase) Redes de convolución (CNN): Capas de convolución, filtros, pooling. Arquitecturas. Aumentado de datos. Aprendizaje por transferencia. Detección de objetos y segmentación en imágenes. (2 hs de clase)
Semana 12	*Proyecto 2 Redes convolucionales:* Se trabaja con un problema y datos apropiados para aplicar CNN. Puede que coincida con el proyecto o que se trabaje con otro problema para este tema. (2 hs de clase) Redes neuronales recurrentes: Procesamiento de series temporales usando RNN y CNN. Propagación hacia atrás a través del tiempo. Normalización y dropout recurrente. Secuencias largas con LSTM y GRU. CNN 1-D para secuencias. (2 hs de clase)
Semana 13	*Proyecto 2 Redes recurrentes:* Se trabaja con un problema y datos apropiados para aplicar RNN para procesamiento de series temporales. Puede que coincida con el proyecto o que se trabaje con otro problema para este tema. (2 hs de clase) Redes neuronales recurrentes: Procesamiento de lenguaje natural usando RNN y mecanismo de atención. Arquitectura encoder-decoder. Mecanismo de atención. Arquitectura Transformer. Grandes modelos de lenguaje (2 hs de clase)
Semana 14	*Proyecto 2 Redes recurrentes:* Se trabaja con un problema y datos apropiados para aplicar RNN para procesamiento de lenguaje natural.

	Puede que coincida con el proyecto o que se trabaje con otro problema para este tema. (2 hs de clase) Aprendizaje de representaciones y aprendizaje generativo: Auto-encoders para aprendizaje de representaciones y como modelos generativos. Redes generativas antagónicas (GANs). Modelos de difusión (2 hs de clase) - ENTREGA 2 (CNN o RNN)
Semana 15	Despliegue de aplicaciones: Entrenamiento a gran escala y despliegue de aplicaciones. Infraestructura de servidor de modelos. Alternativas en la nube, en móviles y en el navegador. (2 hs de clase) *Proyecto 2:* Se trabaja en el proyecto, el informe y la defensa. (2 hs de clase)

A3) MODALIDAD DEL CURSO Y PROCEDIMIENTO DE EVALUACIÓN

El proceso de evaluación incluye la realización individual de 2 entregas y de dos defensas de proyecto grupales. Las entregas podrán incluir preguntas teóricas y ejercicios prácticos, en muchos casos vinculados con el proyecto que se esté realizando en el curso. En las defensas el grupo de estudiantes deberá presentar su solución al problema planteado, así como el proceso que condujo a ella, haciendo énfasis en el análisis de resultados y su alcance.

Para aprobar la unidad curricular los estudiantes deberán realizar todas las entregas y participar activamente de la elaboración del proyecto y de las defensas. Además, deberán alcanzar el 60% de los puntos en el promedio de las 2 entregas y aprobar satisfactoriamente ambas defensas. Los docentes podrán solicitar a un estudiante que mejore o amplíe una entrega a fin de alcanzar los mínimos requeridos para la aprobación.

A4) CALIDAD DE LIBRE

Los estudiantes no podrán acceder a la Calidad de Libre.

A5) CUPOS DE LA UNIDAD CURRICULAR

No se definen cupos.