
PROGRAMA DE ESTUDIOS: CURSO DE INTRODUCCIÓN AL APRENDIZAJE AUTOMÁTICO

KOICA

HGU HANDONG GLOBAL
UNIVERSITY



 **UNA**

Área: Inteligencia Artificial

Introducción al Aprendizaje Automático

□ Información básica

[Información del curso]

1	Título	<i>Introducción al aprendizaje automático</i>
2	Código	-
3	Año lectivo	2025
4	Semestre propuesto	<i>Segundo (Marzo a Junio)</i>
5	Departamento	<i>Centro de Innovación TIC (FPUNA)</i>
6	Nivel	<i>Básico</i>
7	Formato de clase (tipo)	<i>Clases teóricas, demostrativas, interactivas, sesiones prácticas.</i>

[Horario y Lugar]

1	Días	<i>Martes y Jueves</i>
2	Horario	<i>de 18:00 a 20:30</i>
3	Ubicación	<i>EDUCA</i>

[Información del instructor/a]

1	Nombre	<i>Prof. Dr. Pastor E Pérez Estigarribia</i>
2	Oficina (si aplica)	<i>ExCIA, Último piso Bloque G, Departamento de Informática</i>
3	Contacto (correo)	<i>peperez.estigarribia@pol.una.py</i>
4	Contacto (teléfono)	<i>(0984) 439-677</i>

[Horario de oficina]

-

[Notas adicionales]

-

□ Prerrequisitos

Conocimientos básicos de estadística aplicada. Conocimientos básicos de cálculo y álgebra. Habilidades básicas de programación, de preferencia en *Python*.

□ Descripción del curso

Este curso intensivo de 40 horas proporciona una introducción integral a los conceptos fundamentales y aplicaciones prácticas del aprendizaje automático, haciendo énfasis en mayor medida a un enfoque práctico. Los estudiantes aprenderán sobre preprocesamiento de datos, técnicas de aprendizaje supervisado y no supervisado, redes neuronales y aprendizaje profundo, así como la evaluación y mejora de modelos. El curso culmina con la defensa de un proyecto final, permitiendo a los estudiantes aplicar lo aprendido a un problema real.

□ Objetivos del curso

Al finalizar con éxito este curso los estudiantes serán capaces de:

- Comprender los conceptos básicos y las aplicaciones del aprendizaje automático, incluyendo tipos de aprendizaje, modelos, fundamentos matemáticos y estadísticos.
- Aplicar técnicas de preprocesamiento de datos esenciales para mejorar la calidad y rendimiento de los modelos de aprendizaje automático.
- Conocer los diferentes tipos de modelos de regresión y clasificación utilizados en el aprendizaje supervisado y aplicar técnicas para resolver problemas prácticos.
- Conocer y aplicar diferentes técnicas de aprendizaje no supervisado para identificar patrones en conjuntos de datos y reducir la dimensionalidad de los mismos.
- Entender los fundamentos de las redes neuronales y aplicar técnicas de aprendizaje profundo en problemas prácticos.
- Desarrollar una comprensión más profunda de temas avanzados en redes neuronales y aprendizaje profundo, y mejorar las habilidades de investigación y presentación.
- Aplicar técnicas para evaluar el rendimiento de los modelos y mejorar su precisión y generalización.

Método de Evaluación

Calificación absoluta Calificación relativa

- La calificación del curso está estructurada para evaluar la comprensión, el compromiso y la aplicación práctica de los materiales del curso por parte de los estudiantes.
- La política de calificaciones incluye, entre otros: asistencia, tareas/cuestionarios, examen parcial, examen final, crédito(s) extra(s), etc.
- La política de calificaciones está diseñada para evaluar de forma justa el rendimiento de los estudiantes en los diferentes aspectos del curso, fomentando el esfuerzo constante, la participación activa y una comprensión profunda de la materia.

Libros de texto y otros materiales necesarios

- "*Introduction to Machine Learning*" por Ethem Alpaydin.
- "*Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*" por Aurélien Géron.
- "*Deep Learning*" por Ian Goodfellow, Yoshua Bengio y Aaron Courville.
- Otros materiales: computadora personal y navegador web compatible con Google Colab.

Tarea(s) y examen(es)

Las tareas están diseñadas para reforzar los conceptos teóricos tratados en las clases y proporcionar experiencia práctica. Los exámenes están estructurados para evaluar la comprensión general de los conceptos, las teorías y las aplicaciones relacionadas.

Actividades del curso

Las actividades del curso están diseñadas para que los estudiantes se involucren activamente con los materiales, refuercen su comprensión y desarrollen habilidades prácticas. Estas actividades van desde conferencias y debates interactivos hasta laboratorios prácticos y proyectos en grupo.

- Las clases interactivas incluyen, entre otras cosas, sesiones de preguntas y respuestas y encuestas en tiempo real.

- Las sesiones de debate incluyen, entre otras cosas, estudios de casos y debates.
- Sesiones prácticas: ejercicios de codificación, desafíos de datos, etc.
- Los proyectos en grupo incluyen, entre otras cosas, talleres sobre proyectos y revisiones por pares (peer review).
- También se recomiendan los seminarios y las ponencias de invitados con charlas sobre la industria y preguntas y respuestas con expertos.
- Los talleres sobre temas especializados pueden incluir tutoriales sobre herramientas específicas, etc.
- Las actividades de aprendizaje entre iguales incluyen, entre otras, grupos de estudio, clubes de codificación, etc.

Cada una de estas actividades está diseñada para complementar los conocimientos teóricos adquiridos en las clases, profundizar en la comprensión mediante el debate y la aplicación, y preparar a los estudiantes para tareas relacionadas con el mundo real. Mediante la participación en diversas actividades del curso, los estudiantes desarrollan un conjunto completo de habilidades que incluyen la competencia técnica, el razonamiento ético y las experiencias de trabajo en colaboración.

□ Cronograma del curso

Semana	Tema	Tipo de clases	Materiales
1	Conceptos fundamentales de aprendizaje automático	Teórica	Diapositivas, Colab notebook.
2	Preprocesamiento de datos	Teórica, demostrativa, práctica	Colab notebook, implementación de ejemplos y ejercicios de desafíos.
3	Aprendizaje supervisado: Regresión y Clasificación	Teórica, demostrativa, práctica	Colab notebook, implementación de ejemplos y ejercicios de desafíos.
4	Aprendizaje no supervisado	Teórica, demostrativa, práctica	Colab notebook, implementación de ejemplos y ejercicios de desafíos.
5	Redes neuronales y aprendizaje profundo	Teórica, demostrativa, práctica	Colab notebook, implementación de ejemplos y ejercicios de desafíos.

6	Redes neuronales y aprendizaje profundo	Seminarios	Colab notebook, implementación de ejemplos y ejercicios de desafíos.
7	Evaluación y mejora de modelos	Teórica, demostrativa práctica, debate	Colab notebook, implementación de ejemplos y ejercicios de desafíos.
8	Defensa o predefensa de proyecto final	Examen final: defensa de proyecto	Diapositivas.

□ Contenidos del curso

Semana 1: Tema, Conceptos fundamentales de aprendizaje automático

Este contenido introduce el aprendizaje automático en el ámbito de la inteligencia artificial, describe etapas de un proyecto de ciencia de datos, y los conceptos de modelos, fundamentos matemáticos y estadísticos. Métodos de estimación de parámetros, métricas de rendimiento, optimización, regularización, sobreajuste y subajuste de modelos. Explica algoritmos principales y conceptos de aprendizaje supervisado, no supervisado y por refuerzo, donde los modelos se entrenan con datos etiquetados, sin etiquetar o mediante recompensas y castigos. Menciona algoritmos comunes como regresión lineal, árboles de decisión, máquinas de soporte vectorial y redes neuronales.

Objetivo: Conocer y entender los diferentes tipos de modelos y algoritmos utilizados en el aprendizaje automático.

Tipo de Clase: Teórica con discusión interactiva.

Actividades:

- Introducción al aprendizaje automático: definiciones y tipos (supervisado, no supervisado, reforzado).
- Discusión interactiva sobre las expectativas del curso y los conocimientos previos de los estudiantes.
- Introducción a las etapas en un proyecto de ciencia de datos.
- Ejemplos de aplicaciones del aprendizaje automático en diferentes campos haciendo énfasis en los conceptos fundamentales.

Materiales Requeridos:

- Libro de Texto: "Introduction to Machine Learning" por Ethem Alpaydin. Capítulo 1.
- Fuentes de Referencia: Artículos científicos sobre aplicaciones recientes del aprendizaje

automático.

Instrucción de las Actividades:

1. Presentación multimedia para introducir los conceptos básicos.
2. Análisis de casos de estudio de aplicaciones del aprendizaje automático.
3. Discusión grupal para compartir conocimientos previos y expectativas.

Semana 2: Tema, Preprocesamiento de Datos

Este contenido introduce las técnicas y metodologías fundamentales para preparar los datos antes del análisis y modelado en aprendizaje automático. Incluye la limpieza de datos, manejo de valores faltantes, normalización y estandarización, conversión de variables categóricas, y técnicas de reducción de dimensionalidad. El contenido hace énfasis en enfoques de ingeniería de atributos.

Objetivo: Aprender y aplicar técnicas de preprocesamiento de datos esenciales para mejorar la calidad y rendimiento de los modelos de aprendizaje automático.

Tipo de Clase: Teórica y práctica.

Actividades:

- **Teoría del Preprocesamiento de Datos:**
 - Conceptos de limpieza de datos: identificación y corrección de valores anómalos.
 - Manejo de valores faltantes: métodos de imputación y eliminación.
 - Normalización y estandarización de datos.
 - Conversión de variables categóricas: codificación one-hot y ordinal.
 - Reducción de dimensionalidad: PCA y selección de características.
- **Taller Práctico:**
 - Implementación de técnicas de preprocesamiento usando Python y librerías como Pandas y Scikit-learn.
 - Ejercicio práctico: preprocesamiento de un conjunto de datos de ejemplo, aplicando las técnicas aprendidas.
 - Evaluación y comparación del rendimiento de modelos antes y después del preprocesamiento.

Materiales Requeridos:

- Libro de Texto: "Introduction to Machine Learning" por Ethem Alpaydin. Capítulo 2.
- Fuentes de referencia: Tutoriales online sobre Pandas y Scikit-learn, y artículos sobre mejores prácticas en preprocesamiento de datos.

Instrucción de las Actividades:

1. Presentación Multimedia:

- Introducción a los conceptos clave y técnicas de preprocesamiento.
- Ejemplos visuales y casos de estudio para ilustrar la importancia del preprocesamiento.

2. Ejercicio Práctico:

- Los estudiantes trabajarán en pequeños grupos para limpiar, normalizar y convertir variables en un conjunto de datos de ejemplo.
- Aplicarán técnicas de reducción de dimensionalidad y discutirán cómo estas afectan el rendimiento del modelo.

3. Discusión Grupal:

- Compartir resultados y reflexionar sobre los desafíos encontrados durante el preprocesamiento.
- Discutir estrategias y mejores prácticas para mejorar la calidad de los datos.

Semana 3: Tema, Aprendizaje Supervisado: Regresión y Clasificación

Conceptos y técnicas de aprendizaje supervisado, específicamente en regresión y clasificación. Este contenido introduce el aprendizaje supervisado, donde los modelos se entrenan con datos etiquetados para hacer predicciones. Se describen los fundamentos de la regresión y la clasificación, incluyendo la regresión lineal, la regresión logística, árboles de decisión, máquinas de soporte vectorial etc.

Objetivo: Conocer y entender los diferentes tipos de modelos de regresión y clasificación utilizados en el aprendizaje supervisado y aplicar técnicas para resolver problemas prácticos.

Tipo de Clase: Teórica y práctica

Actividades:

1. Teoría del Aprendizaje Supervisado:

- Introducción al aprendizaje supervisado: definiciones y aplicaciones.
- Explicación de la regresión lineal y regresión logística: conceptos, ecuaciones y aplicaciones.
- Introducción a los algoritmos de clasificación: árboles de decisión, máquinas de soporte vectorial y otros.

2. Taller Práctico:

- Implementación de modelos de regresión y clasificación usando Python y librerías como Scikit-learn.
- Ejercicio práctico: entrenamiento y evaluación de un modelo de regresión y un modelo de clasificación en un conjunto de datos de ejemplo.

- Evaluación y comparación de los resultados obtenidos con diferentes algoritmos.

Materiales Requeridos:

- Libro de Texto: "Introduction to Machine Learning" por Ethem Alpaydin. Capítulos 4 y 5.
- Fuentes de referencia: Tutoriales online sobre Scikit-learn y artículos sobre aplicaciones de regresión y clasificación en la industria.

Instrucción de las Actividades:

1. Presentación Multimedia:

- Introducción a los conceptos clave de regresión y clasificación.
- Ejemplos visuales y casos de estudio para ilustrar las aplicaciones de cada tipo de modelo.

2. Ejercicio Práctico:

- Los estudiantes trabajarán con ejemplos demostrativos para entrenar y evaluar modelos de regresión y clasificación.
- Aplicación de técnicas de ajuste y evaluación de modelos, como la validación cruzada y el cálculo de métricas de rendimiento (MSE, precisión, recall, etc.).

3. Discusión Grupal:

- Compartir resultados y reflexionar sobre los desafíos encontrados durante el entrenamiento y la evaluación de modelos.
- Discutir estrategias para mejorar la precisión y la generalización de los modelos.

Semana 4: Tema, Aprendizaje No Supervisado

Conceptos y técnicas de aprendizaje no supervisado. Este contenido introduce el aprendizaje no supervisado, donde los modelos se entrenan con datos sin etiquetar para identificar patrones y estructuras ocultas. Se describen los fundamentos de las técnicas de agrupamiento y reducción de dimensionalidad, incluyendo K-means, clustering jerárquico y análisis de componentes principales (PCA).

Objetivo: Conocer y aplicar diferentes técnicas de aprendizaje no supervisado para identificar patrones en conjuntos de datos y reducir la dimensionalidad de los mismos.

Tipo de Clase: Teórica, demostrativa y práctica.

Actividades:

1. Teoría del Aprendizaje No Supervisado:

- Introducción al aprendizaje no supervisado: definiciones y aplicaciones.
- Explicación de técnicas de agrupamiento: K-means y clustering jerárquico.

- Introducción a la reducción de dimensionalidad: análisis de componentes principales (PCA).

2. Demostración:

- Uso de Google Colab para implementar técnicas de aprendizaje no supervisado.
- Demostración en vivo de ejemplos de agrupamiento y PCA aplicados a un conjunto de datos de ejemplo.

3. Taller Práctico:

- Implementación de técnicas de agrupamiento y PCA utilizando Python y librerías como Scikit-learn y Pandas.
- Ejercicios prácticos: los estudiantes trabajarán en pequeños grupos para aplicar las técnicas a diferentes conjuntos de datos y resolver desafíos específicos.

Materiales Requeridos:

- Colab Notebook: Plantillas y ejemplos proporcionados por el profesor.
- Implementación de Ejemplos: Código de ejemplo para K-means, clustering jerárquico y PCA.
- Ejercicios de Desafíos: Conjuntos de datos y problemas prácticos para resolver en clase o fuera de clases.

Instrucción de las Actividades:

1. Presentación Multimedia:

- Introducción a los conceptos clave y técnicas de aprendizaje no supervisado.
- Ejemplos visuales y casos de estudio para ilustrar la aplicación de cada técnica.

2. Demostración en Colab:

- Implementación en vivo de técnicas de agrupamiento y PCA.
- Explicación paso a paso del código y los resultados obtenidos.

3. Ejercicio Práctico:

- Los estudiantes trabajarán en pequeños grupos para implementar y probar técnicas de agrupamiento y PCA en conjuntos de datos de ejemplo.
- Resolverán desafíos específicos aplicando los conceptos aprendidos y discutirán los resultados.

4. Discusión Grupal:

- Compartir resultados y reflexionar sobre los desafíos encontrados durante la implementación de técnicas de aprendizaje no supervisado.
- Discutir estrategias para mejorar la identificación de patrones y la reducción de

dimensionalidad en los datos.

Acceso a los Materiales:

- Colab Notebook: Acceso proporcionado a través de enlaces compartidos en la plataforma del curso.
- Tutoriales y Recursos Adicionales: Enlaces proporcionados en la plataforma del curso o en el material entregado por el profesor.
- "Introduction to Machine Learning" de Ethem Alpaydin, Capítulo 7

Semana 5: Tema, Redes Neuronales y Aprendizaje Profundo

Introducción a las Redes Neuronales y el Aprendizaje Profundo. Este contenido abarca los conceptos básicos de las redes neuronales y el aprendizaje profundo, incluyendo su arquitectura, tipos de redes neuronales (como feedforward y convolucionales), y aplicaciones prácticas. Se introducen también técnicas de optimización y entrenamiento.

Objetivo: Conocer y entender los fundamentos de las redes neuronales y aplicar técnicas de aprendizaje profundo en problemas prácticos.

Tipo de Clase: Teórica, demostrativa y práctica

Actividades:

1. Teoría de Redes Neuronales y Aprendizaje Profundo:

- Introducción a las redes neuronales: definición y componentes básicos (neuronas, capas, funciones de activación).
- Tipos de redes neuronales: redes feedforward, redes neuronales convolucionales (CNN), y redes neuronales recurrentes (RNN).
- Técnicas de optimización y entrenamiento: backpropagation, descenso de gradiente, técnicas de regularización y manejo del sobreajuste.

2. Demostración:

- Uso de Google Colab para implementar redes neuronales y técnicas de aprendizaje profundo.
- Demostración en vivo de ejemplos de redes neuronales aplicadas a problemas de clasificación.

3. Taller Práctico:

- Implementación de redes neuronales simples usando Python y librerías como TensorFlow y Keras.
- Ejercicios de desafíos: los estudiantes podrán trabajar en pequeños grupos para construir, entrenar y evaluar modelos de redes neuronales en conjuntos de datos de ejemplo.

- Resolución de desafíos específicos y evaluación del rendimiento de los modelos.

Materiales Requeridos:

- Colab Notebook: Plantillas y ejemplos proporcionados por el profesor.
- Implementación de Ejemplos: Código de ejemplo para redes neuronales feedforward, CNN y RNN.
- Ejercicios de Desafíos: Conjuntos de datos y problemas prácticos para resolver.

Instrucción de las Actividades:

1. Presentación Multimedia:

- Introducción a los conceptos clave y arquitecturas de redes neuronales.
- Ejemplos visuales y casos de estudio para ilustrar aplicaciones de redes neuronales en diferentes campos.

2. Demostración en Colab:

- Implementación en vivo de redes neuronales feedforward y CNN.
- Explicación paso a paso del código y los resultados obtenidos.

3. Ejercicio Práctico:

- Los estudiantes trabajarán en pequeños grupos para construir, entrenar y evaluar modelos de redes neuronales en conjuntos de datos de ejemplo.
- Resolverán desafíos específicos aplicando los conceptos aprendidos y discutirán los resultados.

4. Discusión Grupal:

- Compartir resultados y reflexionar sobre los desafíos encontrados durante la implementación de redes neuronales.
- Discutir estrategias para mejorar la precisión y el rendimiento de los modelos de aprendizaje profundo.

Acceso a los Materiales:

- Colab Notebook: Acceso proporcionado a través de enlaces compartidos en la plataforma del curso.
- Tutoriales y Recursos Adicionales: Enlaces proporcionados en la plataforma del curso o en el material entregado por el profesor.

Recomendación de Bibliografía:

- "Deep Learning" por Ian Goodfellow, Yoshua Bengio y Aaron Courville
- "Neural Networks and Deep Learning: A Textbook" por Charu C. Aggarwal

- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" por Aurélien Géron

Semana 6: Tema, Redes Neuronales y Aprendizaje Profundo

Seminarios sobre Redes Neuronales y Aprendizaje Profundo Este contenido permite a los estudiantes explorar temas avanzados y específicos relacionados con redes neuronales y aprendizaje profundo, ya sea a través de seminarios propuestos por el instructor o temas elegidos por los propios estudiantes. Estos seminarios fomentan la investigación independiente y el desarrollo de habilidades de presentación y discusión académica.

Objetivo: Desarrollar una comprensión más profunda de temas avanzados en redes neuronales y aprendizaje profundo, y mejorar las habilidades de investigación y presentación.

Tipo de Clase: Seminarios teóricos y/o demostrativos

Actividades:

1. Selección de Temas:

- Los estudiantes pueden elegir entre una lista de temas propuestos por el instructor o sugerir sus propios temas relacionados con redes neuronales y aprendizaje profundo.
- Ejemplos de temas: Redes Generativas Adversarias (GANs), Transfer Learning, Redes Neuronales Convolucionales Avanzadas, Reinforcement Learning en Deep Learning, Interpretabilidad de Modelos de Aprendizaje Profundo, etc.

2. Preparación de Seminarios:

- Investigación y preparación de materiales: diapositivas de presentación, código y ejemplos en Colab Notebooks.
- Revisión y feedback del instructor antes de la presentación final.

3. Presentación y Discusión:

- Cada estudiante o grupo de estudiantes presenta su tema seleccionado, utilizando diapositivas y demostraciones prácticas.
- Discusión interactiva y preguntas de los compañeros y del instructor.

4. Evaluación y Retroalimentación:

- Evaluación del contenido presentado, la claridad de la exposición y la capacidad para responder preguntas.
- Retroalimentación constructiva para mejorar futuras presentaciones y proyectos.

Materiales Requeridos:

- Diapositivas: Presentaciones en PowerPoint o Google Slides preparadas por los estudiantes.
- Colab Notebooks: Plantillas y ejemplos proporcionados por los estudiantes.

- Recursos Adicionales: Artículos, tutoriales y código relevante según el tema seleccionado.

Instrucción de las Actividades:

1. Guía de Preparación:

- Instrucciones detalladas sobre cómo investigar y preparar un seminario.
- Plantillas de diapositivas y ejemplos de Colab Notebooks para inspirar y guiar a los estudiantes.

2. Sesiones de Presentación:

- Organización de sesiones de presentación donde los estudiantes exponen sus seminarios.
- Facilitar la discusión y el intercambio de ideas durante y después de cada presentación.

3. Evaluación y Feedback:

- El instructor y los compañeros proporcionan retroalimentación detallada sobre las presentaciones.
- Discusión de las fortalezas y áreas de mejora para cada seminario.

Acceso a los Materiales:

- Diapositivas y Colab Notebooks: Acceso proporcionado a través de enlaces compartidos en la plataforma del curso.
- Tutoriales y Recursos Adicionales: Enlaces proporcionados en la plataforma del curso o en el material entregado por el profesor.

Recomendación de Bibliografía:

- "Deep Learning" por Ian Goodfellow, Yoshua Bengio y Aaron Courville: Una referencia esencial para profundizar en diversos temas de aprendizaje profundo.
- "Neural Networks and Deep Learning: A Textbook" por Charu C. Aggarwal: Proporciona una introducción completa a las redes neuronales y el aprendizaje profundo, con aplicaciones prácticas.
- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" por Aurélien Géron: Un enfoque práctico para aprender sobre aprendizaje automático y profundo, con numerosos ejemplos y ejercicios en Python.

Semana 7: Tema, Evaluación y Mejora de Modelos

Técnicas de Evaluación y Mejora de Modelos de Aprendizaje Automático. Este contenido aborda los métodos y técnicas para evaluar y mejorar los modelos de aprendizaje automático, incluyendo la validación cruzada, el ajuste de hiperparámetros, y las estrategias para evitar el sobreajuste y sobreajuste.

Objetivo: Aprender y aplicar técnicas para evaluar el rendimiento de los modelos y mejorar su precisión y generalización.

Tipo de Clase: Teórica, demostrativa, práctica y debate

Actividades:

1. Teoría de Evaluación y Mejora de Modelos:

- Introducción a la validación cruzada y su importancia en la evaluación del rendimiento del modelo.
- Métodos de evaluación: precisión, recall, F1-score, AUC-ROC, etc.
- Ajuste de hiperparámetros y búsqueda de hiperparámetros: grid search & random search.
- Estrategias para evitar el sobreajuste y subajuste: regularización, early stopping, y técnicas de ensemble.

2. Demostración:

- Uso de Google Colab para implementar técnicas de evaluación y mejora de modelos.
- Demostración en vivo de validación cruzada y ajuste de hiperparámetros aplicados a un conjunto de datos de ejemplo.

3. Taller Práctico:

- Implementación de técnicas de evaluación y mejora utilizando Python y librerías como Scikit-learn y TensorFlow.
- Resolución de desafíos específicos y evaluación del rendimiento mejorado de los modelos.

4. Debate:

- Análisis de casos de estudio donde se aplican diferentes técnicas de mejora de modelos.

Materiales Requeridos:

- Colab Notebook: Plantillas y ejemplos proporcionados por el profesor.
- Implementación de Ejemplos: Código de ejemplo para validación cruzada y ajuste de hiperparámetros.
- Ejercicios de Desafíos: Conjuntos de datos y problemas prácticos para resolver en clase.

Instrucción de las Actividades:

1. Presentación Multimedia:

- Introducción a los conceptos clave y técnicas de evaluación y mejora de modelos.
- Ejemplos visuales y casos de estudio para ilustrar la aplicación de cada técnica.

2. Demostración en Colab:

- Implementación en vivo de validación cruzada y ajuste de hiperparámetros.
- Explicación paso a paso del código y los resultados obtenidos.

3. Ejercicio Práctico:

- Los estudiantes trabajarán con ejemplos para evaluar y mejorar modelos de aprendizaje automático en conjuntos de datos de ejemplo.
- Resolverán desafíos específicos aplicando los conceptos aprendidos y discutirán los resultados.

4. Debate:

- Facilitación de un debate sobre las técnicas de mejora de modelos y su aplicación en diferentes contextos.
- Reflexión sobre los desafíos encontrados y las mejores prácticas identificadas durante el proceso.

Acceso a los Materiales:

- Colab Notebook: Acceso proporcionado a través de enlaces compartidos en la plataforma del curso.
- Tutoriales y Recursos Adicionales: Enlaces proporcionados en la plataforma del curso o en el material entregado por el profesor.

Recomendación de Bibliografía:

- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" por Aurélien Géron: Un enfoque práctico para aprender sobre evaluación y mejora de modelos con numerosos ejemplos y ejercicios en Python.
- "Pattern Recognition and Machine Learning" por Christopher Bishop: Cubre técnicas avanzadas de evaluación y mejora de modelos en el contexto del reconocimiento de patrones.
- "Deep Learning" por Ian Goodfellow, Yoshua Bengio y Aaron Courville: Aborda técnicas de mejora de modelos en el contexto del aprendizaje profundo.

Semana 8: Tema, Defensa o predefensa de proyecto final

Como parte de la evaluación final, la última clase se trabajará en la elaboración de un proyecto final, grupal o individual, con roles y responsabilidades claras en cuanto a contribuciones. El proyecto en cuestión debe reflejar que el estudiante ha incorporado los conocimientos y habilidades básicas según los objetivos de este curso.

Objetivo: Evaluar la comprensión y aplicación de los conceptos aprendidos en el curso a través de la defensa de proyectos finales.

Requerimientos del proyecto:

- Implementación de un código que resuelva un problema mediante la aplicación de algoritmos de aprendizaje automático.
- Se evaluará tres entregables: [Código](#) de implementación en *Colab* debidamente comentado, presentación audiovisual serán el cual se evaluará según el baremo ANEXO al documento en la carpeta.
- Los estudiantes tendrán un tiempo de 15 días para la elaboración del proyecto con la posibilidad de entrega en la semana 8.

ANEXO: BAREMO DE EVALUACIÓN DE PROYECTO FINAL

UNIVERSIDAD NACIONAL DE ASUNCIÓN

CENTRO DE INNOVACIÓN TIC

Baremo de evaluación de proyecto

Campus Universitario de San Lorenzo, enero del 2025

**PROGRAMA DE INTELIGENCIA ARTIFICIAL - BÁSICO
INTRODUCCIÓN AL APRENDIZAJE AUTOMÁTICO**

INSTRUCCIONES

1. Elabore un proyecto *Colab* que demuestre dominio de conocimiento teórico práctico según los objetivos del curso. Tenga en cuenta que las secciones obligatorias se indican con un (*) . El formato de entrega del documento será .ipynb con los archivos necesarios e indicaciones que garanticen su reproducibilidad.
2. Elabore un video de defensa del proyecto en cuestión de al menos 10 minutos y no más de 20 minutos de duración. Levante el video en youtube con la configuración [Vídeos Ocultos](#) y proporcione el enlace o código QR del enlace en la versión final de la presentación del proyecto.
3. Elaborar un video explicando el código implementado en el proyecto en cuestión de al menos 10 minutos y no más de 20 minutos de duración. Levante el video en youtube con la configuración [Vídeos Ocultos](#) y proporcione el enlace o código QR del enlace en la versión final de la presentación del proyecto.
4. El proyecto y la defensa del mismo deben demostrar que el estudiante ha adquirido las habilidades necesarias para el procesamiento de datos. El proyecto debe reflejar el manejo y aplicación técnicas básicas de aprendizaje automático para aplicarlos a datos de la vida real y así como las técnicas para evaluar y corregir los modelos empleados.

Baremo de evaluación de proyecto

Programa: de Inteligencia Artificial

Evaluador: Dr. Pastor E. Pérez Estigarribia

Título del proyecto:

Estudiante/s:

Escala:

100 PUNTOS				
0	--	59	=	1
60	--	69	=	2
70	--	80	=	3
81	--	90	=	4
91	--	100	=	5

CRITERIOS DE EVALUACIÓN	Logrado	Medianamente logrado	No logrado
0. PROMEDIO PONDERADO	50 pts	22 pts	21 pts

0.1 Participación en clase y en foros: Asistencia y participación en clases.

1. ANÁLISIS DE LA PROBLEMÁTICA	10 pts	7 pts	0 pts
--------------------------------	--------	-------	-------

1.1 **Antecedentes:** están definidos claramente los antecedentes y su relación con el problema.

1.2 **Amplitud:** la problemática y sus interrogantes se encuentran delimitadas.

2. EVALUACIÓN DE LOS OBJETIVOS

10 pts

8 pts

0 pts

2.1 **Hipótesis u objetivo general:** su formulación es clara y precisa, es coherente con los antecedentes del tema.

2.2 **Los objetivos específicos:** están contenidos en el objetivo general, se encuentran bien formulados, son factibles y se encuentran claramente delimitados.

3. EVALUACIÓN DE LA METODOLOGÍA

10 pts

7 pts

0 pts

3.1 **La metodología:** es apropiada, se encuentra actualizada en el área de trabajo propuesto.

3.3 **Los objetivos específicos:** ¿pueden alcanzarse con las metodologías propuestas? ¿Son identificables los resultados esperados?

4. SEGURIDAD Y SOLVENCIA

10 pts

8 pts

0 pts

4.1. ¿El alumno comprende el alcance y el contexto del estado del arte del proyecto y muestra solvencia?

5. IMPLEMENTACIÓN DEL CÓDIGO DEL PROYECTO

10 pts

8 pts

0 pts

5.1. ¿El alumno implementa un código funcional y debidamente comentado?

5.2 ¿El alumno demuestra seguridad y elocuencia explicando el código implementado?

TOTAL DE PUNTOS

/100

Comentarios

Recomendaciones