



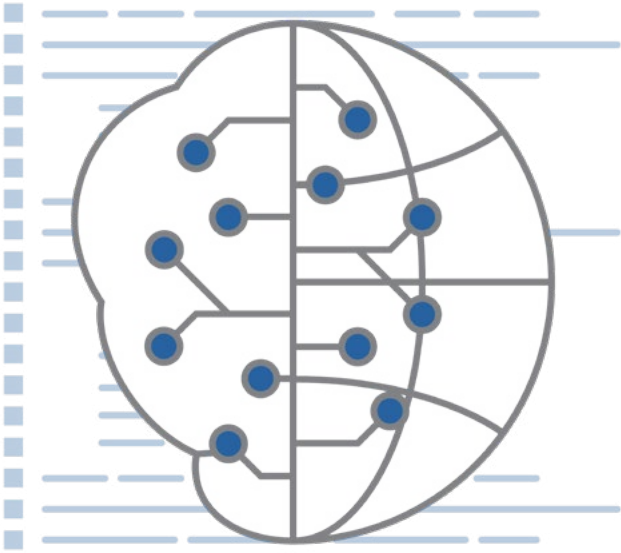
PROMiDAT
IBEROAMERICANO

Programa Iberoamericano de
Formación en Minería de Datos

PROGRAMA
**MÁSTER EJECUTIVO EN
CIENCIA DE DATOS E
INTELIGENCIA ARTIFICIAL¹**



¹Este programa no constituye programa de maestría académica universitaria, es más bien una certificación profesional ejecutiva.



MÁSTER EJECUTIVO EN CIENCIA DE DATOS E INTELIGENCIA ARTIFICIAL

1. Introducción

El Máster Ejecutivo en Ciencia de Datos e Inteligencia Artificial está diseñado para capacitar a profesionales en técnicas avanzadas de ciencia de datos y “deep learning”. Los participantes aprenderán a aplicar estas habilidades en el análisis de datos, desarrollo de modelos predictivos y optimización de procesos en sus organizaciones. Este programa incluye desde fundamentos matemático-estadísticos de métodos supervisados y no supervisados hasta técnicas avanzadas en deep learning y un proyecto final de tesina.

¹Este programa no constituye programa de maestría académica universitaria, es más bien una certificación profesional ejecutiva.



2. Perfil de Entrada de los Estudiantes

Este programa está dirigido a profesionales de cualquier área que desean especializarse en ciencia de datos e inteligencia artificial, que tengan conocimientos previos en programación y analítica de datos.

Este programa es ideal para:

- Graduados de los Programas Experto en Ciencia de Datos y Experto en Inteligencia Artificial de PROMiDAT o personas que ya tienen al menos conocimientos intermedios en Programación R, Programación Python y en técnicas de la Ciencia de Datos e Inteligencia Artificial.
- Científicos de datos, analistas y profesionales de tecnología interesados en profundizar en inteligencia artificial y técnicas avanzadas de modelado.
- Profesionales en roles de análisis, gestión de datos y toma de decisiones, que deseen aplicar modelos predictivos y de optimización.
- Cualquier persona interesada en desarrollar proyectos en ciencia de datos e inteligencia artificial con un enfoque práctico y aplicado con conocimiento real de lo que está haciendo.



3. Objetivos Generales del Programa

El Máster Ejecutivo en Ciencia de Datos e Inteligencia Artificial tiene como objetivo principal formar a profesionales capaces de:

- Conocer los fundamentos matemático estadístico de la Ciencia de Datos y la Inteligencia Artificial.
- Desarrollar los conocimientos avanzados en matemática y en programación necesarios para la comprensión a fondo de la Ciencia de Datos, “machine learning” y “deep learning”.
- Implementar y optimizar modelos avanzados de “machine learning” y “deep learning”.
- Aplicar técnicas de análisis en datos reales y simulados, usando R y Python.
- Desarrollar y defender un proyecto de investigación (tesina) que demuestre su dominio en ciencia de datos e inteligencia artificial.



4. Posibilidades de Empleo

Los egresados podrán desempeñarse en roles como:

- Jefe de unidades especializadas en analítica de datos.
- Científico de Datos: Aplicar métodos avanzados de machine learning y deep learning en problemas complejos.
- Especialista en Deep Learning: Optimizar y desarrollar modelos avanzados para análisis predictivo.
- Consultor en Ciencia de Datos e IA: Asesorar a empresas en el uso efectivo de datos y algoritmos de IA para resolver problemas específicos.



5. Programa General

El Máster Ejecutivo en Ciencia de Datos e Inteligencia Artificial tiene una duración total de 10 meses y se compone de los siguientes cursos especializados:

Cursos Especializados:

1. ME-1000 Programación Paralela en R y Python.
2. ME-2000 Desarrollo Modelos Analíticos apoyados por herramientas de IA.
3. ME-3000 Álgebra Lineal Aplicada en Ciencia de Datos e Inteligencia Artificial.
4. ME-3002 Cálculo y Optimización Aplicado en Ciencia de Datos e Inteligencia Artificial.
5. ME-3004 Probabilidad y Estadística Aplicados a la Ciencia de Datos e Inteligencia Artificial.
6. ME-3006 Programación Avanzada de Dashboards.
7. ME-4000 Métodos No Supervisados Avanzados en R y Python.
8. ME-4002 Métodos Supervisados Avanzados en R y Python.
9. ME-9000 Proyecto Final de Graduación (Tesina).



6. Metodología

El máster se desarrollará en modalidad 100% online, con clases en vivo (presencial-remoto) a través de Zoom, las cuales se grabarán para que los estudiantes puedan acceder a ellas en cualquier momento. Los estudiantes contarán con:



Videoconferencias semanales:

Sesiones en tiempo real para interactuar con el tutor y compañeros.



Trabajos prácticos:

Ejercicios semanales aplicados a situaciones reales, enviados a través del campus virtual.



Foros de debate y consultas:

Espacio para resolver dudas y discutir temas del curso.



Campus virtual en Moodle:

Acceso 24/7 a materiales del curso, grabaciones de las clases, tareas y herramientas de comunicación.



7. Inversión

Duración:
10 meses

Precio por curso:
\$300 + 2% de IVA.

Precio Proyecto Final:
\$400 +2% de IVA.

Inversión total:
\$2800 + 2% de IVA

8. Información adicional:



Correo electrónico : info@promidat.com



Teléfono: +506 4030-1205



Web: www.promidat.com



WhatsApp: +506 8712-6978



Directo: +506 2271-0464



ME-1000

Programación Paralela en R y Python

Duración

Cuatro semanas

Descripción del Curso

La programación paralela es esencial en la ciencia de datos y el aprendizaje automático para el manejo eficiente de grandes volúmenes de datos y la ejecución de modelos computacionalmente intensivos. Este curso explora el uso de herramientas en R y Python para implementar programación paralela y optimizar el rendimiento de algoritmos. Se abordarán paquetes específicos como Snow en R y Joblib en Python, y se explorarán ejemplos prácticos de paralelización de tareas usando scikit-learn y Dask.

Objetivos

1. Introducir los conceptos y técnicas básicas de programación paralela en R y Python.
2. Capacitar a los estudiantes en el uso de herramientas como Snow en R y Joblib en Python para mejorar el rendimiento de tareas computacionales.
3. Explorar la paralelización en el entrenamiento y evaluación de modelos de machine learning utilizando scikit-learn.
4. Enseñar a utilizar documentación técnica para profundizar en el uso de paquetes de programación paralela.

Metodología

El curso combina teoría y práctica para asegurar una comprensión profunda de los conceptos y habilidades prácticas en programación paralela. Se emplearán las siguientes herramientas y recursos:

1. Videoconferencias semanales:

Se ofrecerán sesiones en vivo en Zoom que serán grabadas para que los estudiantes puedan revisarlas cuando deseen.

2. Trabajos prácticos semanales:

Ejercicios y casos de estudio aplicados para implementar programación paralela.

3. Foros de discusión:

Un espacio en Moodle donde los estudiantes podrán plantear dudas y discutir con el tutor y compañeros.

4. Aula virtual en Moodle:

Acceso a materiales, enlaces a documentación, ejemplos de código y otros recursos complementarios.

Luego de este curso el estudiante será capaz de:

1. Implementar programación paralela en R usando el paquete Snow y en Python usando Joblib.
2. Aprovechar la paralelización en modelos de aprendizaje automático en scikit-learn.
3. Comprender y navegar la documentación técnica para aplicar programación paralela en proyectos de ciencia de datos.
4. Optimizar procesos computacionales para reducir tiempos de ejecución en tareas de ciencia de datos.

Contenido

- **Semana 1:**
Fundamentos de Programación Paralela en Ciencia de Datos
 - a. Introducción a la programación paralela y su importancia en ciencia de datos.
 - b. Conceptos básicos de concurrencia y paralelización.
 - c. Herramientas y paquetes principales para programación paralela en R y Python.
- **Semana 2:**
Programación Paralela en R con el Paquete Snow
 - a. Instalación y configuración de Snow en R.
 - b. Creación de clústeres y ejecución de tareas en paralelo.

- c. Ejemplos prácticos de paralelización en análisis de datos y simulaciones.
- d. Documentación: Snow Package Documentation.
- **Semana 3:**
Programación Paralela en Python con Joblib
 - a. Introducción a Joblib en Python.
 - b. Uso de joblib.Parallel para la ejecución de funciones en paralelo.
 - c. Casos prácticos de paralelización de tareas computacionales.
 - d. Documentación: Joblib.Parallel.
- **Semana 4:**
Paralelización en Modelos de Machine Learning con scikit-learn y Dask
 - a. Implementación de programación paralela en scikit-learn usando Joblib.
 - b. Uso de Dask para la escalabilidad en machine learning.
 - c. Ejemplo práctico: Paralelización de la Validación Cruzada en R y Python.
 - d. Ejemplo práctico: Paralelización de calibración y selección de modelos en R y Python.
 - e. Documentación: Scikit-learn con Joblib en Dask.

Evaluación

La evaluación consistirá en 4 tareas prácticas con un valor igual. Cada tarea estará enfocada en implementar y optimizar programación paralela en diferentes contextos de análisis de datos y aprendizaje automático.

Bibliografía

- Vahid, F. (2020). Parallel Programming in Science and Engineering. Springer. Libro.
- Joblib Documentation. (2023). Joblib.Parallel. <https://joblib.readthedocs.io/en/latest/generated/joblib.Parallel.html>
- Dask Developers. (2023). Using Joblib with Dask. <https://ml.dask.org/joblib.html>
- Scikit-Learn Developers. (2023). Parallelism in Scikit-Learn 1.5. <https://scikit-learn.org/1.5/computing/parallelism.html>
- Scikit-Learn Developers. (2023). Parallelism in Scikit-Learn 1.0. <https://scikit-learn.org/1.0/computing/parallelism.html>
- Downey, A. (2016). Think Python: How to Think Like a Computer Scientist. Green Tea Press.
- Eddelbuettel, D. (2013). Seamless R and C++ Integration with Rcpp. Springer.
- Wilkinson, B. & Allen, M. (2005). Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers. Pearson.



ME-2000

Desarrollo Modelos Analíticos apoyados por herramientas de IA

Duración

Cuatro semanas

Descripción del Curso

Este curso avanzado explora el desarrollo e implementación de modelos supervisados y métodos no supervisados, apoyándose en herramientas de inteligencia artificial como ChatGPT para optimizar el rendimiento y la precisión de los modelos. Los estudiantes aprenderán a integrar estas herramientas de IA para mejorar la eficiencia en la construcción y evaluación de modelos, desde la optimización de hiperparámetros hasta la generación de insights para preprocesamiento de datos y selección de características. Con proyectos prácticos en R y Python, el curso abarca aplicaciones de clustering, reducción de dimensionalidad, y modelos de clasificación y regresión.

Objetivos

1. Comprender y aplicar modelos supervisados avanzados para tareas de clasificación y regresión.
2. Utilizar métodos no supervisados como clustering y reducción de dimensionalidad en problemas reales.
3. Integrar herramientas de inteligencia artificial (IA) como ChatGPT para apoyar la construcción y optimización de modelos.
4. Implementar modelos en R y Python, aprovechando el soporte de IA para mejorar la eficiencia y precisión.
5. Desarrollar habilidades para seleccionar y evaluar modelos de acuerdo con el contexto específico del problema.

Metodología

El curso combina teoría avanzada y aplicaciones prácticas, utilizando R y Python para implementar y evaluar modelos. Se incentiva a los estudiantes a utilizar herramientas de IA para tareas como la optimización de hiperparámetros, generación de código, y asesoría en la elección de algoritmos. Las herramientas y recursos incluyen:

- 1. Videoconferencias semanales:**
Clases en vivo mediante Zoom, grabadas para revisión posterior.
- 2. Trabajos prácticos semanales:**
Ejercicios en R y Python para aplicar los conceptos en proyectos reales.
- 3. Foros de discusión:**
Espacio en Moodle para plantear dudas y compartir ideas.
- 4. Aula virtual en Moodle:**
Acceso a materiales, enlaces a bibliografía, ejemplos de código y recursos adicionales.

Luego de este curso el estudiante será capaz de:

- 1.** Implementar modelos avanzados de clasificación y regresión con el apoyo de herramientas de IA.
- 2.** Aplicar métodos no supervisados como clustering, PCA, UMAP y t-SNE para análisis de datos.
- 3.** Utilizar IA como ChatGPT para asistir en la optimización de modelos y generación de insights.
- 4.** Evaluar y ajustar modelos en R y Python, mejorando la precisión y el rendimiento.
- 5.** Resolver problemas de ciencia de datos utilizando técnicas de modelado avanzadas y asistencia de IA.

Contenido

- **Semana 1:**
Integración de Métodos No Supervisados en Proyectos con IA
 - a. Introducción a casos de estudio aplicados en ciencia de datos con IA.
 - b. Integración de herramientas de IA para asistir en la selección de modelos no supervisados.
 - c. Ejemplo práctico en Python: Uso de clustering y clasificación para segmentación con soporte de ChatGPT.
 - d. Ejemplo práctico en R: Implementación de modelos híbridos en un proyecto de marketing, con optimización asistida por IA.
- **Semana 2:**
Modelos Supervisados Avanzados y Optimización con IA
 - a. Revisión de algoritmos avanzados: Árboles de decisión, Random Forest, y Gradient Boosting.

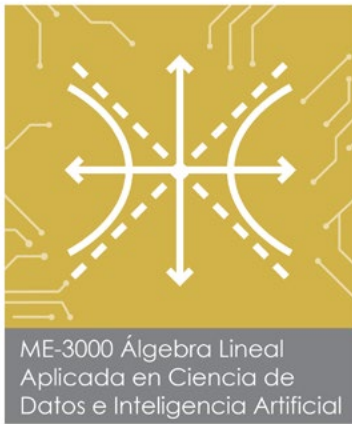
- b. Uso de IA para optimización de hiperparámetros mediante búsqueda en cuadrícula y optimización bayesiana.
 - c. Validación cruzada y ajuste de modelos con soporte de herramientas IA.
 - d. Ejemplo práctico en Python: Optimización de modelos supervisados en Scikit-learn con asesoría de IA.
 - e. Ejemplo práctico en R: Optimización de hiperparámetros con Caret y ChatGPT.
- **Semana 3:**
Métodos No Supervisados Avanzados y Reducción de Dimensionalidad
 - a. Técnicas de clustering: K-means, clustering jerárquico, y DBSCAN.
 - b. Aplicación de PCA, UMAP y t-SNE con recomendaciones y asesoría de IA.
 - c. Ejemplo práctico en Python: Implementación de K-means, PCA y UMAP con apoyo de IA.
 - d. Ejemplo práctico en R: Clustering jerárquico, t-SNE y UMAP con ajuste de parámetros asistido por IA.
- **Semana 4:**
Implementación Completa de Modelos Supervisados y No Supervisados con IA
 - a. Integración completa de modelos de clasificación y regresión en Python y R.
 - b. Uso de ChatGPT para generación de código y sugerencias en preprocesamiento de datos.
 - c. Ejemplo práctico en Python: Implementación de Random Forest para clasificación, con soporte de IA en selección de características.
 - d. Ejemplo práctico en R: Gradient Boosting para regresión con ajustes optimizados mediante IA.

Evaluación

La evaluación consistirá en 4 tareas prácticas con igual peso, cada una enfocada en aplicar y optimizar métodos supervisados y no supervisados con el apoyo de herramientas de inteligencia artificial.

Bibliografía

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.
- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.
- Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R*. Springer.
- Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.
- VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media.
- Lantz, B. (2015). *Machine Learning with R*. Packt Publishing.
- Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*.
- Chollet, F. (2018). *Deep Learning with Python*. Manning Publications.
- Wickham, H., & Grolemund, G. (2016). *R for Data Science*. O'Reilly Media.



ME-3000

Álgebra Lineal Aplicada en Ciencia de Datos e Inteligencia Artificial

Duración

Cuatro semanas

Descripción del Curso

El álgebra lineal es una herramienta fundamental en la ciencia de datos, la inteligencia artificial y el deep learning. Este curso introduce los conceptos esenciales de álgebra lineal necesarios para comprender y aplicar técnicas avanzadas de análisis de datos, reducción de dimensionalidad y modelos de aprendizaje automático. A través de ejercicios prácticos, los estudiantes aprenderán a utilizar vectores, matrices, y transformaciones lineales en el desarrollo de modelos predictivos y algoritmos de aprendizaje profundo. Los temas incluyen operaciones con vectores y matrices, descomposiciones en valores singulares, y sus aplicaciones en el Análisis en Componentes Principales, Máquinas de Soporte Vectorial y Redes Neuronales.

Objetivos

1. Familiarizar a los estudiantes con los conceptos básicos y avanzados de álgebra lineal aplicados a la ciencia de datos y el aprendizaje automático.
2. Dotar a los estudiantes de las habilidades para realizar operaciones con vectores y matrices, y aplicarlas en la construcción de modelos de aprendizaje supervisado y no supervisado.
3. Explicar la importancia de los valores y vectores propios, y su uso en técnicas de reducción de dimensionalidad y descomposición.
4. Comprender cómo el álgebra lineal sustenta el funcionamiento de algoritmos avanzados en deep learning, como redes neuronales convolucionales y redes de memoria a largo plazo (LSTM).

5. Desarrollar habilidades para aplicar técnicas de álgebra lineal en el análisis de datos reales y en el diseño de modelos de inteligencia artificial.

Metodología

La metodología del curso combina teoría y práctica, asegurando que los estudiantes puedan aplicar directamente los conceptos en problemas reales. Las herramientas y recursos incluyen:

1. **Videoconferencias semanales:**
Una sesión en Zoom que quedará grabada para que los estudiantes puedan revisarla en cualquier momento.
2. **Trabajos prácticos semanales:**
Ejercicios y casos de estudio enfocados en la implementación de algoritmos en Python.
3. **Foros de discusión:**
Espacio en Moodle donde los estudiantes pueden plantear preguntas, discutir temas del curso y colaborar con sus compañeros.
4. **Aula virtual en Moodle:**
Acceso a materiales de estudio, presentaciones, ejercicios, y recursos adicionales.

Luego de este curso el estudiante será capaz de:

1. Realizar operaciones complejas con vectores y matrices, incluyendo multiplicación, transposición, y cálculo de determinantes e inversas.
2. Aplicar valores y vectores propios en descomposiciones de matrices y reducción de dimensionalidad, como el Análisis en Componentes Principales (PCA).
3. Modelar rectas y planos en el espacio tridimensional y utilizar estas representaciones en visualización de datos y clustering.
4. Desarrollar e implementar algoritmos de Máquinas de Soporte Vectorial (SVM) y técnicas no supervisadas utilizando conceptos de álgebra lineal.
5. Comprender y construir redes neuronales simples aplicando operaciones matriciales, y entender el funcionamiento de capas convolucionales y de retropropagación en redes profundas.

Contenido

- **Semana 1:**
Fundamentos de Álgebra Lineal Aplicada a Ciencia de Datos
 - a. Introducción a vectores y matrices en ciencia de datos.
 - b. Operaciones con vectores (suma, producto escalar, producto cruz) y matrices (suma, multiplicación, determinantes, inversas).
 - c. Aplicaciones en inteligencia artificial: álgebra lineal en transformaciones de datos y normalización.
 - d. Caso práctico: Preprocesamiento de datos utilizando álgebra lineal.

- **Semana 2:**
Valores y Vectores Propios en Análisis de Datos
 - a. Introducción a valores y vectores propios y su importancia en ciencia de datos.
 - b. Descomposición en valores propios (Eigen-decomposition) y aplicaciones en reducción de dimensionalidad.
 - c. Introducción a descomposiciones de matrices (e.g., SVD) y su uso en deep learning.
 - d. Caso práctico: Implementación del Análisis en Componentes Principales (PCA) para reducción de dimensionalidad en un conjunto de datos.

- **Semana 3:**
Representación en Espacios Tridimensionales y Aplicaciones en Clustering
 - a. Modelado de rectas y planos en el espacio R^3 .
 - b. Interpretación geométrica de datos en espacios multidimensionales y su aplicación en clustering.
 - c. Aplicación práctica en visualización y agrupación de datos (clustering).
 - d. Caso práctico: Representación visual de grupos en 3D y su interpretación para segmentación de clientes.

- **Semana 4:**
Aplicaciones en Inteligencia Artificial y Deep Learning
 - a. Máquinas de Soporte Vectorial (SVM) y su relación con el álgebra lineal.
 - b. Introducción a redes neuronales y operaciones matriciales en deep learning.
 - c. Implementación de capas y propagación hacia adelante utilizando álgebra lineal.
 - d. Caso práctico: Implementación de una red neuronal básica para clasificación de imágenes utilizando operaciones matriciales.

Evaluación

La evaluación consistirá en 4 tareas con un valor igual. Cada tarea estará enfocada en aplicar los conceptos de álgebra lineal a problemas reales de ciencia de datos e inteligencia artificial.

Bibliografía

- Strang, G. (2016). Introduction to Linear Algebra. Wellesley-Cambridge Press.
- Shlens, J. (2014). A Tutorial on Principal Component Analysis. arXiv.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer.
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
- Abdi, H., & Williams, L. J. (2010). Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics, 2(4), 433-459.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.
- Nielsen, M. (2015). Neural Networks and Deep Learning. Online book.
- Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media.
- Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.



ME-3002

Cálculo y Optimización Aplicado en Ciencia de Datos e Inteligencia Artificial

Duración

Cuatro semanas

Descripción del Curso

Este curso explora los conceptos de cálculo y optimización aplicados en ciencia de datos e inteligencia artificial. Los temas abordan desde los fundamentos del cálculo diferencial hasta el uso de métodos de optimización como el gradiente para mejorar el rendimiento de modelos y algoritmos de Machine Learning. A lo largo del curso, los estudiantes aprenderán a calcular derivadas, aplicar la regla de la cadena, usar el vector gradiente, y encontrar máximos y mínimos en funciones, todo enfocado en aplicaciones prácticas en ciencia de datos.

Objetivos

1. Comprender los fundamentos del cálculo diferencial y su aplicación en la optimización de modelos de machine learning.
2. Aprender a calcular derivadas y aplicar la regla de la cadena en el contexto de funciones utilizadas en inteligencia artificial.
3. Utilizar el vector gradiente para identificar direcciones de crecimiento o decrecimiento de funciones.
4. Aplicar técnicas de optimización, como el método del gradiente, para encontrar mínimos en funciones de una o varias variables.
5. Desarrollar habilidades para optimizar modelos predictivos y funciones de costo en ciencia de datos.

Metodología

Este curso combina teoría y práctica en cálculo y optimización. Los estudiantes trabajarán con Python y R para aplicar los conceptos en problemas de ciencia de datos e inteligencia artificial. Las herramientas y recursos incluyen:

- 1. Videoconferencias semanales:**
Sesiones en Zoom que serán grabadas para su revisión posterior.
- 2. Trabajos prácticos semanales:**
Ejercicios aplicados en R y Python para reforzar los conceptos de cálculo y optimización.
- 3. Foros de discusión:**
Espacio en Moodle para discutir dudas y compartir conocimientos con el tutor y compañeros.
- 4. Aula virtual en Moodle:** Acceso a materiales, ejemplos de código y otros recursos.

Luego de este curso el estudiante será capaz de:

- 1.** Calcular derivadas de funciones y aplicar la regla de la cadena en el contexto de Machine Learning.
- 2.** Utilizar el vector gradiente para analizar el comportamiento de funciones en múltiples variables.
- 3.** Encontrar máximos y mínimos de funciones de una o varias variables.
- 4.** Implementar el método del gradiente en problemas de optimización para modelos de ciencia de datos.
- 5.** Aplicar técnicas de optimización para mejorar el rendimiento de algoritmos de inteligencia artificial.

Contenido

- **Semana 1:**
Fundamentos del Cálculo Diferencial en Ciencia de Datos e Inteligencia Artificial
 - a.** Introducción al cálculo de derivadas y su importancia en ciencia de datos.
 - b.** Cálculo de derivadas de funciones básicas.
 - c.** Aplicación de la regla de la cadena en funciones compuestas.
 - d.** Ejemplo práctico en Python y R: Derivadas en funciones de costo de modelos de Machine Learning.

- **Semana 2:**
Vector Gradiente y sus Aplicaciones
 - a. Introducción al vector gradiente y su interpretación en funciones de varias variables.
 - b. Cálculo del gradiente en funciones multivariables.
 - c. El fenómeno de la desaparición de gradiente (vanishing gradient problem)
 - d. Ejemplos de aplicación del gradiente en la optimización de funciones de costo.
 - e. Ejemplo práctico en Python: Cálculo del gradiente en redes neuronales.

- **Semana 3:**
Máximos y Mínimos en Funciones de una y varias Variables
 - a. Identificación de máximos y mínimos en funciones de una variable.
 - b. Extensión a funciones de varias variables y uso del gradiente para identificar extremos.
 - c. Ejemplo práctico en R y Python: Optimización de funciones de error en modelos predictivos.

- **Semana 4:**
Otros métodos de optimización aplicados en la analítica de datos
 - a. Métodos: Nesterov Accelerated Gradient (NAG), AdaGrad, RMSProp, Adam, AdaMax, Nadam, Método de Newton, Quasi-Newton (BFGS, L-BFGS), Levenberg-Marquardt, Optimización Bayesiana, Dropout (en redes neuronales), Batch Normalization (en redes neuronales), Cyclical Learning Rates (CLR).
 - b. Algoritmos Genéticos, Enjambre de Partículas (PSO).
 - c. Ejemplo práctico en Python y R: Implementación del gradiente descendente en regresión lineal.

Evaluación

La evaluación consistirá en 4 tareas prácticas con igual valor. Cada tarea estará enfocada en aplicar y optimizar conceptos de cálculo y optimización en proyectos de ciencia de datos e Inteligencia Artificial.

Bibliografía

- Strang, G. (2016). Calculus. Wellesley-Cambridge Press.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
- Nielsen, M. (2015). Neural Networks and Deep Learning. Online book.
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
- Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R. Springer.
- Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media.
- Chollet, F. (2018). Deep Learning with Python. Manning Publications.
- Luenberger, D. G., & Ye, Y. (2016). Linear and Nonlinear Programming. Springer.



ME-3004

Probabilidad y Estadística Aplicados a la Ciencia de Datos e Inteligencia Artificial

Duración

Cuatro semanas

Descripción del Curso

Este curso proporciona una introducción a los conceptos de probabilidad y estadística fundamentales en el análisis de datos y su aplicación en inteligencia artificial. Los temas cubren desde la definición de probabilidad, probabilidad condicional y el Teorema de Bayes hasta aplicaciones prácticas en árboles de decisión y métodos bayesianos. Al finalizar, los estudiantes habrán desarrollado una base sólida para aplicar conceptos estadísticos en el desarrollo y optimización de modelos de Machine Learning y análisis de datos.

Objetivos

1. Entender los conceptos básicos de probabilidad y estadística y su importancia en ciencia de datos.
2. Aplicar el Teorema de Bayes para inferencia en modelos de inteligencia artificial.
3. Utilizar probabilidades y estadísticas en la construcción y evaluación de árboles de decisión.
4. Introducir los fundamentos de los métodos bayesianos y su uso en machine learning.
5. Desarrollar habilidades para aplicar métodos probabilísticos en el análisis de datos y modelos predictivos.

Metodología

El curso combina teoría con aplicaciones prácticas en R y Python, asegurando que los estudiantes puedan implementar y analizar modelos probabilísticos. Las herramientas y recursos incluyen:

- 1. Videoconferencias semanales:**
Sesiones en Zoom que serán grabadas para su revisión posterior.
- 2. Trabajos prácticos semanales:**
Ejercicios aplicados en R y Python para reforzar los conceptos de probabilidad y estadística.
- 3. Foros de discusión:**
Espacio en Moodle para discutir temas del curso y resolver dudas.
- 4. Aula virtual en Moodle:**
Acceso a materiales, ejemplos de código y recursos adicionales.

Luego de este curso el estudiante será capaz de:

1. Definir y calcular probabilidades en diferentes contextos.
2. Aplicar el Teorema de Bayes para realizar inferencias y tomar decisiones en modelos de datos.
3. Implementar árboles de decisión y analizar su rendimiento en problemas de clasificación.
4. Comprender y aplicar métodos bayesianos en el desarrollo de modelos predictivos.
5. Evaluar modelos de machine learning usando conceptos estadísticos fundamentales.

Contenido

- **Semana 1:**
Introducción a la Probabilidad y Definición de Conceptos Básicos
 - a. Introducción a la probabilidad y su importancia en ciencia de datos.
 - b. Definición de probabilidad, eventos y espacio muestral.
 - c. Conceptos básicos de probabilidad condicionada.
 - d. Ejemplo práctico en Python y R: Cálculo de probabilidades en conjuntos de datos.
- **Semana 2:**
Teorema de Bayes y sus Aplicaciones en IA
 - a. Explicación y demostración del Teorema de Bayes.
 - b. Aplicación del Teorema de Bayes en modelos de clasificación.

- c. Ejemplo práctico en Python: Aplicación del Teorema de Bayes en un clasificador Naive Bayes.
- d. Ejemplo práctico en R: Implementación de un clasificador bayesiano.
- **Semana 3:**
Probabilidad en Árboles de Decisión y su Uso en Ciencia de Datos
 - a. Introducción a los árboles de decisión y su estructura.
 - b. Algoritmos de construcción de árboles: ID3, C4.5, y CART.
 - c. Evaluación de modelos de árboles de decisión en problemas de clasificación y regresión.
 - d. Ejemplo práctico en Python: Construcción y análisis de un árbol de decisión.
 - e. Ejemplo práctico en R: Implementación de árboles de decisión en problemas de clasificación.
- **Semana 4:**
Métodos Bayesianos y Aplicaciones en Machine Learning
 - a. Introducción a los métodos bayesianos y su importancia en machine learning.
 - b. Aplicaciones en modelos predictivos y en aprendizaje supervisado.
 - c. Ejemplo práctico en Python y R: Implementación de un modelo bayesiano para clasificación.
 - d. Ejemplo de caso práctico: Uso de métodos bayesianos en análisis de datos y toma de decisiones.

Evaluación

La evaluación consistirá en 4 tareas prácticas con igual valor, enfocadas en aplicar conceptos de probabilidad y estadística en proyectos de ciencia de datos.

Bibliografía

- Wasserman, L. (2004). All of Statistics: A Concise Course in Statistical Inference. Springer.
- Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
- Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). Bayesian Data Analysis. CRC Press.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R. Springer.
- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
- Silver, N. (2012). The Signal and the Noise: Why So Many Predictions Fail—but Some Don't. Penguin Press.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.
- Feller, W. (1968). An Introduction to Probability Theory and Its Applications. Wiley.
- McGrayne, S. B. (2011). The Theory That Would Not Die: How Bayes' Rule Cracked the Enigma Code, Hunted Down Russian Submarines, and Emerged Triumphant from Two Centuries of Controversy. Yale University Press.



ME-3006

Programación Avanzada de Dashboards

Duración

Cuatro semanas

Descripción del Curso

Este curso avanzado aborda la creación de dashboards interactivos y altamente funcionales utilizando tres herramientas principales: R, Python, y Power BI. Se enfoca en el desarrollo de habilidades para la implementación de visualizaciones dinámicas, la integración de datos de diversas fuentes y el despliegue de dashboards en entornos corporativos y públicos. Al finalizar, los estudiantes estarán capacitados para construir dashboards avanzados adaptados a necesidades específicas de comunicación de datos.

Objetivos

1. Construir dashboards avanzados utilizando R, Python y Power BI.
2. Integrar múltiples fuentes de datos en dashboards funcionales.
3. Personalizar visualizaciones para cumplir con objetivos específicos de usuarios finales.
4. Desplegar dashboards en plataformas web y sistemas de informes corporativos.

Metodología

El curso combina teoría y práctica, utilizando ejemplos reales y ejercicios prácticos. Las actividades incluyen:

- 1. Sesiones sincrónicas:**
Clases grabadas para su consulta posterior.
- 2. Trabajos prácticos:**
Desarrollo de dashboards con datasets proporcionados.
- 3. Foros de discusión:**
Resolución de dudas en la plataforma virtual.
- 4. Acceso a recursos:**
Documentación, scripts y herramientas específicas.

Luego de este curso el estudiante será capaz de:

- 1.** Diseñar dashboards profesionales con capacidades avanzadas de análisis.
- 2.** Integrar funciones de interactividad y automatización en dashboards.
- 3.** Implementar soluciones visuales en R, Python y Power BI adaptadas a audiencias específicas.
- 4.** Desplegar dashboards en entornos web y sistemas empresariales.

Contenido

- **Semana 1:**
Programación Avanzada de Dashboards en R
 - a. Introducción a Shiny para dashboards interactivos.
 - b. Conexión a bases de datos y visualización avanzada.
 - c. Uso de módulos y layouts dinámicos.
 - d. Ejercicio práctico: Creación de un dashboard interactivo en R con Shiny.
- **Semana 2:**
Programación Avanzada de Dashboards en Python
 - a. Uso de Dash y Plotly para visualizaciones dinámicas.
 - b. Integración de APIs y sistemas de bases de datos.
 - c. Personalización de componentes visuales y callbacks.
 - d. Ejercicio práctico: Desarrollo de un dashboard interactivo en Python utilizando Dash.

- **Semana 3:**
Programación Avanzada de Dashboards en Power BI
 - a. Creación de modelos de datos optimizados.
 - b. Personalización de visualizaciones con DAX.
 - c. Publicación y compartición de dashboards en Power BI Service.
 - d. Ejercicio práctico: Diseño de un dashboard avanzado en Power BI con integración de datos en tiempo real.

- **Semana 4:**
Despliegue de Dashboards y Buenas Prácticas de Presentación
 - a. Comparación de herramientas y selección según necesidades del proyecto.
 - b. Estrategias para asegurar la accesibilidad y el rendimiento de dashboards.
 - c. Despliegue de dashboards en plataformas web y aplicaciones empresariales.
 - d. Ejercicio práctico: Publicación y ajuste final de un dashboard avanzado.

Evaluación

La evaluación consistirá en cuatro entregas prácticas:

1. Dashboard interactivo en R.
2. Dashboard funcional en Python.
3. Dashboard optimizado en Power BI.
4. Proyecto final integrando conceptos avanzados de las tres herramientas.

Bibliografía

- Chang, W., et al. (2020). Mastering Shiny. Chapman and Hall/CRC.
- McKinney, W. (2022). Python for Data Analysis. O'Reilly Media.
- Kroenke, D. (2019). Experiencing MIS with Power BI. Pearson.
- Dash Documentation: <https://dash.plotly.com>
- Microsoft Power BI Documentation: <https://docs.microsoft.com/en-us/power-bi/>



ME-4000 Métodos NO Supervisados Avanzados en R y Python

Duración

Cuatro semanas

Descripción del Curso

Este curso avanzado cubre técnicas de métodos no supervisados en el análisis de datos, aplicadas a través de R y Python. Los estudiantes explorarán temas como clustering avanzado, reducción de dimensionalidad y análisis factorial de correspondencias, además de técnicas especializadas como el Análisis Factorial de Correspondencias Simples y Múltiples y el Método STATIS para tablas múltiples. Este conocimiento permitirá a los estudiantes aplicar métodos no supervisados en proyectos complejos de ciencia de datos y resolver problemas de agrupamiento y segmentación.

Objetivos

1. Profundizar en técnicas avanzadas de métodos no supervisados en análisis de datos.
2. Aprender a implementar y evaluar técnicas de clustering avanzado y reducción de dimensionalidad.
3. Aplicar el Análisis Factorial de Correspondencias Simples y Múltiples en R y Python.
4. Explorar y aplicar el Método STATIS para el análisis de tablas múltiples.
5. Desarrollar habilidades para aplicar estos métodos en ciencia de datos, análisis de mercado y estudios multidimensionales.

Metodología

El curso combinará teoría y práctica aplicada en R y Python, permitiendo a los estudiantes implementar y analizar métodos no supervisados en problemas reales de ciencia de datos. Las herramientas y recursos incluyen:

- 1. Videoconferencias semanales:**
Sesiones en Zoom que serán grabadas para su revisión posterior.
- 2. Trabajos prácticos semanales:**
Ejercicios aplicados en R y Python para fortalecer la implementación de métodos no supervisados avanzados.
- 3. Foros de discusión:**
Espacio en Moodle para la resolución de dudas y discusión de temas del curso.
- 4. Aula virtual en Moodle:**
Acceso a materiales, ejemplos de código y recursos adicionales.

Luego de este curso el estudiante será capaz de:

- 5.** Aplicar técnicas avanzadas de clustering y reducción de dimensionalidad en R y Python.
- 6.** Realizar Análisis Factorial de Correspondencias Simples y Múltiples.
- 7.** Implementar el Método STATIS para el análisis de tablas múltiples.
- 8.** Utilizar métodos no supervisados avanzados para resolver problemas de segmentación, agrupación y análisis exploratorio de datos.
- 9.** Evaluar y comparar métodos no supervisados en el contexto de aplicaciones en ciencia de datos.

Contenido

- **Semana 1:**
Clustering Avanzado y Técnicas de Agrupación
 - a. Introducción a métodos de clustering avanzados: DBSCAN, clustering jerárquico, y clustering basado en densidad.
 - b. Evaluación de modelos de clustering y métricas de validación.
 - c. Ejemplo práctico en Python: Implementación de DBSCAN y clustering jerárquico.
 - d. Ejemplo práctico en R: Agrupación de datos utilizando clustering basado en densidad.
- **Semana 2:**
Técnicas de Reducción de Dimensionalidad y Visualización
 - a. Análisis en Componentes Principales (PCA), t-SNE, y UMAP.

- b. Selección de técnicas de reducción de dimensionalidad para visualización y agrupación.
- c. Ejemplo práctico en Python: Implementación de t-SNE y UMAP para visualización de datos.
- d. Ejemplo práctico en R: PCA aplicado a datos multidimensionales.
- **Semana 3:**
Análisis Factorial de Correspondencias Simples y Múltiples
 - a. Introducción al Análisis Factorial de Correspondencias (AFC) para tablas simples.
 - b. Extensión al Análisis Factorial de Correspondencias Múltiples (AFCM) para tablas múltiples.
 - c. Interpretación de los resultados del AFC y AFCM en contextos de ciencia de datos y segmentación de mercado.
 - d. Ejemplo práctico en Python: Implementación de AFC y AFCM.
 - e. Ejemplo práctico en R: Análisis factorial de correspondencias en datos categóricos.
- **Semana 4:**
Método STATIS para el Análisis de Tablas Múltiples
 - a. Introducción al Método STATIS y su aplicación en ciencia de datos.
 - b. Análisis de estructuras comunes entre tablas múltiples y su interpretación.
 - c. Aplicaciones de STATIS en estudios multidimensionales y análisis de tendencias.
 - d. Ejemplo práctico en Python y R: Implementación del Método STATIS para tablas múltiples.

Evaluación

La evaluación consistirá en 4 tareas prácticas con igual valor, enfocadas en aplicar métodos no supervisados avanzados en análisis de datos.

Bibliografía

- Murtagh, F., & Legendre, P. (2014). Ward's Hierarchical Clustering Method: Clustering Criterion and Agglomerative Algorithm. *Journal of Classification*.
- Greenacre, M. (2007). *Correspondence Analysis in Practice*. CRC Press.
- Abdi, H., & Williams, L. J. (2010). *Principal Component Analysis*. Wiley Interdisciplinary Reviews: Computational Statistics.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R*. Springer.
- Lebart, L., Morineau, A., & Warwick, K. M. (1984). *Multivariate Descriptive Statistical Analysis*. Wiley.
- Pagès, J. (2015). *Multiple Factor Analysis by Example Using R*. CRC Press.
- Husson, F., Josse, J., & Pagès, J. (2010). Principal Component Methods – Hierarchical Clustering – Partitional Clustering: Why Would We Need to Choose for Visualizing Data?. *Journal of Statistical Software*.
- Gower, J. C., & Dijksterhuis, G. B. (2004). *Procrustes Problems*. Oxford University Press.
- Lê, S., Josse, J., & Husson, F. (2008). FactoMineR: An R Package for Multivariate Analysis. *Journal of Statistical Software*.
- VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media.



ME-4002

Métodos Supervisados Avanzados en R y Python

Duración

Cuatro semanas

Descripción del Curso

Este curso avanzado se centra en el estudio y la implementación de métodos supervisados avanzados en el análisis de datos, utilizando R y Python. Los estudiantes explorarán algoritmos de clasificación y regresión de última generación, optimización de hiperparámetros, técnicas de ensamblado y métodos de validación para mejorar la precisión y eficiencia de los modelos. Al final del curso, los estudiantes serán capaces de implementar y ajustar modelos supervisados avanzados para resolver problemas complejos en ciencia de datos y machine learning.

Objetivos

1. Comprender y aplicar algoritmos avanzados de clasificación y regresión.
2. Aprender técnicas de ensamblado, como bagging, boosting y stacking, para mejorar la precisión de los modelos.
3. Optimizar los hiperparámetros de modelos supervisados en R y Python.
4. Aplicar métodos de validación cruzada y evaluación de rendimiento en modelos de machine learning.
5. Desarrollar habilidades para seleccionar y ajustar modelos supervisados en función de diferentes aplicaciones en ciencia de datos.

Metodología

El curso combina teoría y práctica aplicada en R y Python, permitiendo a los estudiantes implementar y evaluar modelos supervisados en problemas reales de ciencia de datos. Las herramientas y recursos incluyen:

- 1. Videoconferencias semanales:**
Sesiones en Zoom que serán grabadas para su revisión posterior.
- 2. Trabajos prácticos semanales:**
Ejercicios aplicados en R y Python para fortalecer la implementación de métodos supervisados avanzados.
- 3. Foros de discusión:**
Espacio en Moodle para resolver dudas y discutir temas del curso.
- 4. Aula virtual en Moodle:**
Acceso a materiales, ejemplos de código y otros recursos.

Luego de este curso el estudiante será capaz de:

1. Implementar y evaluar algoritmos avanzados de clasificación y regresión en R y Python.
2. Optimizar modelos supervisados mediante técnicas de ajuste de hiperparámetros.
3. Aplicar técnicas de ensamblado, como bagging y boosting, para mejorar el rendimiento de los modelos.
4. Utilizar métodos de validación y métricas de evaluación para medir la eficacia de los modelos.
5. Seleccionar e implementar el modelo supervisado más adecuado según el contexto y los datos disponibles.

Contenido

- **Semana 1:**
Algoritmos Avanzados de Clasificación y Regresión
 - a. Introducción a algoritmos avanzados como Support Vector Machines (SVM), Gradient Boosting, y Random Forest.
 - b. Comparación de modelos de clasificación y regresión avanzados.
 - c. Ejemplo práctico en Python: Implementación de Gradient Boosting para clasificación.
 - d. Ejemplo práctico en R: Uso de Random Forest para problemas de regresión.

- **Semana 2:**
Optimización de Hiperparámetros en Modelos Supervisados
 - a. Métodos de optimización de hiperparámetros: Grid Search, Random Search y optimización bayesiana.
 - b. Técnicas de regularización para mejorar la generalización de los modelos.
 - c. Ejemplo práctico en Python: Optimización de hiperparámetros usando Scikit-learn.
 - d. Ejemplo práctico en R: Optimización de modelos supervisados con el paquete Caret.

- **Semana 3:**
Técnicas de Ensamblado (Bagging, Boosting y Stacking)
 - a. Introducción a bagging y boosting, incluyendo algoritmos como AdaBoost, XGBoost y stacking.
 - b. Comparación de métodos de ensamblado para mejorar la precisión de los modelos.
 - c. Ejemplo práctico en Python: Implementación de XGBoost en un problema de clasificación.
 - d. Ejemplo práctico en R: Implementación de técnicas de bagging y boosting.

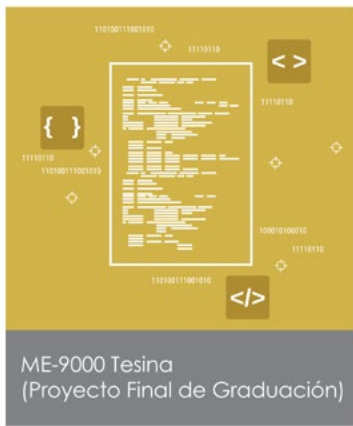
- **Semana 4:**
Validación y Evaluación de Modelos Supervisados
 - a. Técnicas de validación cruzada y métodos de evaluación.
 - b. Métricas de rendimiento para modelos de clasificación y regresión (ROC-AUC, precisión, F1-score).
 - c. Ejemplo práctico en Python: Validación cruzada y evaluación de un modelo de clasificación.
 - d. Ejemplo práctico en R: Evaluación de rendimiento en modelos de regresión.

Evaluación

La evaluación consistirá en 4 tareas prácticas con igual valor, enfocadas en aplicar y optimizar métodos supervisados avanzados en proyectos de ciencia de datos.

Bibliografía

- Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media.
- Kuhn, M., & Johnson, K. (2013). Applied Predictive Modeling. Springer.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.
- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R. Springer.
- Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
- Friedman, J., Hastie, T., & Tibshirani, R. (2001). Additive Logistic Regression: A Statistical View of Boosting. The Annals of Statistics.
- Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. arXiv preprint arXiv:1603.02754.
- Chollet, F. (2018). Deep Learning with Python. Manning Publications.
- Brownlee, J. (2016). Master Machine Learning Algorithms. Machine Learning Mastery.



ME-9000 Tesina (Proyecto Final de Graduación)

Duración

Ocho semanas

Descripción del Curso

Este curso de tesina constituye el proyecto final de graduación, en el cual el estudiante aplicará todos los conocimientos adquiridos en los cursos previos de métodos avanzados en ciencia de datos, deep learning y optimización. El curso está diseñado para guiar al estudiante en el desarrollo de una investigación original y su aplicación en datos reales, desde la revisión de técnicas de investigación hasta la formulación de conclusiones y propuestas de trabajo futuro. Las primeras dos semanas estarán dedicadas a revisar técnicas de investigación y los elementos clave de una tesis. Las siguientes seis semanas se dedicarán a sesiones individualizadas de asesoría y trabajo para orientar al estudiante en el desarrollo de su proyecto.

Objetivos

1. Capacitar al estudiante en técnicas de investigación y en la estructura de una tesis en ciencia de datos.
2. Guiar al estudiante en la construcción de un marco teórico, revisión del estado del arte, y desarrollo de una propuesta de investigación.
3. Orientar el análisis comparativo entre datos reales y simulados y su presentación en tablas y gráficos.
4. Facilitar la redacción de conclusiones y la identificación de áreas de trabajo futuro.
5. Preparar al estudiante para la defensa y presentación final de su proyecto de tesina.

Metodología

El curso combina sesiones teóricas iniciales, trabajo individual, y asesorías personalizadas. En las primeras dos semanas, se cubrirán temas fundamentales de técnicas de investigación y los elementos de una tesis. Luego, el estudiante trabajará de forma independiente en su proyecto, recibiendo orientación en sesiones de asesoría individualizadas cada semana para asegurar el progreso adecuado del proyecto.

Las herramientas y recursos incluyen:

1. Sesiones teóricas:

Durante las dos primeras semanas, se presentarán técnicas de investigación y los elementos estructurales de una tesis.

2. Asesoría individual:

Se realizarán sesiones semanales de asesoría personalizada para apoyar el desarrollo de la tesina.

3. Foro de consulta:

Espacio en Moodle para plantear dudas y recibir retroalimentación continua.

4. Aula virtual en Moodle:

Acceso a materiales de apoyo, ejemplos de estructura de tesis y bibliografía complementaria.

Luego de este curso el estudiante será capaz de:

1. Aplicar técnicas de investigación y los elementos clave de una tesis en su proyecto final de graduación.
2. Realizar una revisión exhaustiva del estado del arte y desarrollar un marco teórico sólido.
3. Comparar y analizar datos reales y simulados, y presentar los resultados en tablas y gráficos.
4. Redactar conclusiones claras y proponer áreas de trabajo futuro en su investigación.
5. Defender y presentar de manera efectiva los resultados de su proyecto de graduación.

Contenido

• **Semana 1:**

Técnicas de Investigación y Metodología

- a. Revisión de metodologías de investigación en Ciencia de Datos e Inteligencia Artificial.
- b. Estrategias para la búsqueda y revisión de literatura científica.
- c. Definición y formulación de preguntas de investigación.
- d. Ejercicio práctico: Formulación de preguntas de investigación y objetivos para la tesina.

- **Semana 2:**
Elementos de una Tesis
 - a. Estructura de la tesis: marco teórico, estado del arte, desarrollo de la propuesta, análisis de resultados, conclusiones y trabajo futuro.
 - b. Construcción del marco teórico y revisión del estado del arte.
 - c. Comparación en tablas de datos reales y simulados.
 - d. Redacción preliminar de conclusiones y propuesta de trabajo futuro.
 - e. Ejercicio práctico: Desarrollo del esquema de la tesis y redacción preliminar de las secciones.

- **Semanas 3 a 8:**
Asesoría Individualizada y Trabajo de Desarrollo

Semana 3: Desarrollo de la propuesta de investigación y asesoría en la estructura de la tesis.

Semana 4: Revisión del marco teórico y orientación en la metodología y análisis de datos.

Semana 5: Implementación del análisis comparativo entre datos reales y simulados.

Semana 6: Presentación preliminar de resultados en tablas y gráficos; retroalimentación y ajustes.

Semana 7: Redacción de conclusiones y discusión del trabajo futuro.

Semana 8: Revisión final, preparación para la defensa y presentación del proyecto de tesina.

Evaluación

La evaluación consistirá en la entrega y defensa del proyecto final de graduación, que incluirá:

1. Propuesta de Investigación: Planteamiento claro de la problemática, objetivos y justificación del proyecto.
2. Marco Teórico y Estado del Arte: Revisión exhaustiva de la literatura y contextualización del problema de investigación.
3. Análisis de Datos: Implementación de métodos avanzados y comparación de resultados entre datos reales y simulados.
4. Conclusiones y Trabajo Futuro: Redacción de conclusiones fundamentadas y propuestas de continuidad en la investigación.
5. Defensa y Presentación: Defensa oral por Zoom del proyecto ante el comité evaluador.

Bibliografía

- Creswell, J. W., & Creswell, J. D. (2018). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. SAGE Publications.
- Booth, W. C., Colomb, G. G., & Williams, J. M. (2008). *The Craft of Research*. University of Chicago Press.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer.
- Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.
- Lencioni, P. (2002). *The Five Dysfunctions of a Team: A Leadership Fable*. Jossey-Bass.
- Silver, N. (2012). *The Signal and the Noise: Why So Many Predictions Fail—but Some Don't*. Penguin Press.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv.
- Material adicional proporcionado por el asesor, específico al proyecto del estudiante.



Correo electrónico : info@promidat.com



Web: www.promidat.com



Teléfono: +506 4030-1205



WhatsApp: +506 8712-6978



Directo: +506 2271-0464



Programa Iberoamericano de
Formación en Minería de Datos