



PROMiDAT
IBEROAMERICANO

Programa Iberoamericano de
Formación en Minería de Datos

PROGRAMA EXPERTO EN INTELIGENCIA ARTIFICIAL EN PYTHON



TM



EXPERTO EN MACHINE LEARNING

1. Introducción

El Programa Experto en Inteligencia Artificial en Python está diseñado para formar profesionales en el campo de la inteligencia artificial (IA), capacitándolos en técnicas de machine learning, deep learning y procesamiento de datos utilizando Python. Este programa permite a los participantes desarrollar habilidades prácticas para analizar y crear modelos supervisados, no supervisados, de series de tiempo y de "Deep Learning" que resuelvan problemas complejos y generen impacto real en sus organizaciones. Con un enfoque en Python, el programa aborda desde la programación orientada a objetos hasta técnicas avanzadas de IA.



2. Perfil de Entrada de los Estudiantes

Este programa está dirigido a personas que ya tienen conocimientos previos en programación en algún lenguaje (R, C++, Java, etc.) y desean profundizar en inteligencia artificial y machine learning.

Este programa es ideal para:

- Desarrolladores interesados en aplicar IA en el desarrollo de software y aplicaciones.
- Analistas de datos y científicos de datos con experiencia en otros lenguajes que busquen adquirir experiencia práctica en IA con Python.
- Ingenieros de software que quieran ampliar sus competencias en machine learning y deep learning.
- Profesionales en áreas de negocios, economía, o investigación que deseen aplicar técnicas de IA en su campo.



3. Objetivos Generales del Programa

El programa busca capacitar a los estudiantes en:

- **Programación en Python:**
Aplicando técnicas avanzadas de programación y conceptos de orientación a objetos.
- **Desarrollo y Evaluación de Modelos IA:**
Creación, entrenamiento y evaluación de modelos de machine learning y deep learning.
- **Implementación Práctica:**
Aplicando IA en casos de uso reales mediante proyectos que abarcan desde la manipulación de datos hasta la predicción y automatización.
- **Análisis de Datos y Visualización:**
Transformación de datos en información visual y desarrollo de modelos supervisados y no supervisados en Python.



4. Posibilidades de Empleo

Al concluir el programa, los egresados estarán capacitados para desempeñarse en roles tales como:

- **Ingeniero de Machine Learning:**
Desarrollo e implementación de modelos de aprendizaje automático en entornos de producción.
- **Científico de Datos:**
Diseño de modelos predictivos y análisis de datos utilizando IA.
- **Especialista en Deep Learning:**
Aplicación de redes neuronales profundas para el reconocimiento de patrones en datos no estructurados.
- **Analista de IA:**
Implementación de técnicas de IA en la mejora de procesos empresariales y optimización de estrategias.



5. Programa General

El programa tiene una duración total de 12 meses, con clases en vivo y acceso a un campus virtual disponible 24/7.

El programa está compuesto por los siguientes cursos:

- 1. ML1000:**
Programación Orientada a Objetos en Python
 - Introducción a Python, clases y objetos, polimorfismo, y manejo de paquetes.
- 2. ML2000:**
Métodos No Supervisados en Python
 - Clustering, reducción de dimensionalidad, y análisis exploratorio de datos.
- 3. ML2003:**
Métodos Supervisados en Python
 - Modelos predictivos, como árboles de decisión, máquinas de soporte vectorial y ensambles.
- 4. ML2006:**
Visualización e Interpretación de Datos en Python
 - Principios de visualización, diseño de gráficos estáticos e interactivos, visualización geoespacial.
- 5. ML3000:**
Modelos Analíticos Avanzados en Python
 - Calibración y validación de modelos avanzados, optimización y modelos predictivos para clasificación y regresión.
- 6. ML3003:**
Deep Learning en Python
 - Redes neuronales, CNN, RNN y otros modelos profundos con Keras y TensorFlow.

7. ML3006:
Web Mining en Python

- Extracción y análisis de datos de la web, incluyendo redes sociales y minería de texto.

8. ML3009:
Series de Tiempo en Python

- Análisis y modelado de series de tiempo con métodos como ARIMA y redes neuronales recurrentes.

9. ML9000:
Proyecto Final de Graduación

- Desarrollo completo de un proyecto en IA, desde la obtención y limpieza de datos hasta la implementación de modelos.





6. Metodología

Las clases se desarrollarán en modalidad online en vivo, con sesiones grabadas para acceso posterior. El programa fomenta la interacción en tiempo real entre tutores y estudiantes, permitiendo consultas directas durante las clases. Se empleará un enfoque basado en la práctica, con trabajo semanal y acceso a un campus virtual para material de apoyo, software y foros de discusión.



**Modalidad
online en vivo**



**Consultas directas
durante las clases**



**Campus
virtual**



7. Inversión

Precio por curso:
\$200 + 2% de IVA.

Precio Proyecto Final:
\$300 + 2% de IVA.

Inversión total:
\$1900 + 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



ML1000

Programación Orientada a Objetos en Python

Descripción

Este curso ofrece una introducción práctica a las técnicas modernas de computación de datos utilizando el lenguaje de programación Python. Para esto se enseñarán los fundamentos básicos de Python como lenguaje de programación. En este curso se estudiará en primera instancia a Python como un Lenguaje de Programación Funcional y luego como lenguaje Orientado a Objetos. Se estudiarán los principales aspectos del paradigma de Programación Orientada a Objetos, como son Herencia de Clases, Composición de Objetos, Polimorfismo, Atributos y Métodos Públicos y Privados. Todo esto haciendo mucho énfasis en el verdadero estilo "Pythonico" de Programación. Se estudiará además la instalación y uso de paquetes en Python, estos permiten ampliar las capacidades del lenguaje existiendo actualmente más de 5,000 paquetes disponibles para su instalación.

Objetivos

En este curso el estudiante será capaz de:

1. Comprender los conceptos básicos de la lógica de programación en lenguaje Python.
2. Entender la sintaxis y semántica del lenguaje así como poder identificar errores.
3. Instalar y actualizar paquetes.
4. Importar datos desde diversos orígenes.
5. Comprender y programar en Python como lenguaje Orientado a Objetos.
6. 6. Desarrollar métodos y clases propias en lenguaje Python.
7. Crear documentos autoreproducibles en diversos formatos como html, doc y pdf.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Programar en lenguaje Python, haciendo uso de clases.

Contenido

1. **Instalación.**
 - a. Python como Lenguaje de Programación.
 - b. Python 2 versus Python 3.
 - c. Instalando Anaconda, Spyder, Scikit-Learn, NumPy, SciPy, IPython, Jupiter, Matplotlib, Pandas y Sympy.
2. **Sintaxis.**
 - a. Variables y tipos de datos.
 - b. Listas y Diccionarios en Python.
 - c. Sentencias de control tipo "if".
 - d. Ciclos tipo "While".
 - e. Funciones en Python.
3. **Programación Orientada a Objetos..**
 - a. Definiciones en Programación Orientada a Objetos.
 - b. Diseño de programas Orientados a Objetos.
 - c. Objetos y Clases en Python.
 - d. Estilo Pythónico de Programación.

4. Herencia.

- a. Herencia simple y múltiple.
- b. Clases Abstractas y Polimorfismo.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Anaconda | The World's Most Popular Data Science Platform. Anaconda. url: <https://www.anaconda.com/> (visited on 10/20/2022).
- [2] Steven F. Lott. Mastering Object-oriented Python. Google-Books-ID: qtdnAwAAQBAJ. Packt Publishing Ltd, Apr. 22, 2014. 1146 pp. isbn: 978-1-78328-098-8.
- [3] Eric Matthes. Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming. Google-Books-ID: w1v6DwAAQBAJ. No Starch Press, May 3, 2019. 546 pp. isbn: 978-1-59327-928-8.
- [4] John Paul Mueller and Luca Massaron. Python for Data Science For Dummies. Google-Books-ID: Uh2EDwAAQBAJ. John Wiley & Sons, Feb. 27, 2019. 502 pp. isbn: 978-1-119-54762-4.
- [5] Andreas C. Müller and Sarah Guido. Introduction to Machine Learning with Python: A Guide for Data Scientists. "O'Reilly Media, Inc.", Sept. 26, 2016. 400 pp. isbn: 978-1-4493-6990-3.
- [6] Dusty Phillips. Python 3 Object-oriented Programming. Google-Books-ID: dYloCgAAQBAJ. Packt Publishing Ltd, Aug. 20, 2015. 460 pp. isbn: 978-1-78439-595-7.
- [7] Jake VanderPlas. Python Data Science Handbook: Essential Tools for Working with Data. Google-Books-ID: xYmNDQAAQBAJ. "O'Reilly Media, Inc.", Nov. 21, 2016. 548 pp. isbn: 978-1-4919-1214-0.



ML2000

Métodos No Supervisados en Python

Descripción

En este curso se definirán los principales conceptos en Machine Learning, con énfasis principalmente en los métodos No Supervisados. Para ello, se verán técnicas relacionadas con exploración de los datos, reducción de la dimensión y clustering. Para esto se hará uso del lenguaje Python y paquetes especializados en Machine Learning.

Objetivos

En este curso el estudiante será capaz de:

1. Entender la necesidad de la utilización de modelos, algoritmos, software especial para un correcto análisis de los datos.
2. Entender las diferencias entre: estadística, análisis de datos, recuperación de la información, ML – “Machine Learning”, minería de datos y Ciencia de Datos.
3. Conocer los principales modelos, técnicas y algoritmos utilizados en los métodos No Supervisados.
4. Efectuar Análisis Exploratorio de Datos en Python.
5. Utilizar Python en métodos no supervisados de análisis de datos, como lo son Análisis en Componentes Principales, Clusterización Jerárquica, K-Medias, entre otros.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar y programar proyectos de Machine Learning relacionados con aprendizaje No Supervisado como por ejemplo segmentación de carteras de clientes.

Contenido

1. **Conceptos de Machine Learning.**
 - a. Definiciones básicas en Machine Learning y Ciencia de Datos.
2. **Análisis Exploratorio de Datos.**
 - a. Tipos de variables.
 - b. Estadísticas básicas y matriz de correlaciones.
 - c. Tablas de datos y datos atípicos.
 - d. Uso de NumPy.
 - e. Manipulación de Datos con Pandas.
 - f. Visualización de Datos con Matplotlib.
3. **Métodos de condensación de la información.**
 - a. Análisis en Componentes Principales – ACP (PCA, Karhunen–Loeve o K-L Method).
 - b. Método t-SNE.
 - c. Método UMAP.
 - d. Plano principal.
 - e. Círculo de correlaciones.
 - f. Dualidad y sobre-posición de gráficos.
 - g. Análisis Factorial de Correspondencias Múltiples.

4. Clustering Jerárquico Aglomerativo.

- a. ¿Qué es “cluster analysis”?
- b. Clusterización Jerárquica Aglomerativa.
- c. Distancias y matrices de distancias.
- d. Agregaciones.
- e. Jerarquías binarias.
- f. Jerarquías Binarias sobre las Componentes Principales.

5. Método de k-medias (k-means).

- a. Inercia total, inercia inter-clases e inercia intra-clases.
- b. Teorema de Fisher.
- c. Problema combinatorio.
- d. Método de Forgy.
- e. Método de las nubes dinámicas.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Giuseppe Bonaccorso. Hands-On Unsupervised Learning with Python: Implement machine learning and deep learning models using Scikit-Learn, TensorFlow, and more. Google-Books-ID: lgyLDwAAQBAJ. Packt Publishing Ltd, Feb. 28, 2019. 375 pp. isbn: 978-1-78934-927-6.
- [2] M. Emre Celebi and Kemal Aydin, eds. Unsupervised Learning Algorithms. Cham: Springer International Publishing, 2016. isbn: 978-3-319-24209-5 978-3-319-24211-8. doi: 10.1007/978-3-319-24211-8. url: <http://link.springer.com/10.1007/978-3-319-24211-8> (visited on 10/21/2022).
- [3] Benjamin Johnston, Aaron Jones, and Christopher Kruger. Applied Unsupervised Learning with Python: Discover hidden patterns and relationships in unstructured data with Python. Google-Books-ID: l66aDwAAQBAJ. Packt Publishing Ltd, May 28, 2019. 483 pp. isbn: 978-1-78995-837-9.
- [4] Ankur A. Patel. Hands-On Unsupervised Learning Using Python: How to Build Applied Machine Learning Solutions from Unlabeled Data. “O’Reilly Media, Inc.”, Feb. 21, 2019. 362 pp. isbn: 978-1-4920-3561-9.
- [5] Fabian Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: MACHINE LEARNING IN PYTHON (), p. 6.
- [6] Zhi-Hua Zhou. Machine Learning. Springer Nature, Aug. 20, 2021. 460 pp. isbn: 9789811519673.



ML3006

Métodos Supervisados en Python

Descripción

En este curso se presentarán los principales métodos en Machine Learning, especialmente enfocados en métodos predictivos, conocidos también como métodos de aprendizaje supervisado. El énfasis principal del curso será examinar dichos métodos desde un punto de vista algorítmico y de sus aplicaciones en casos reales. Se le dará especial importancia al uso de los conceptos de machine learning en aplicaciones reales en Python.

Objetivos

En este curso el estudiante será capaz de:

1. Comprender la diferencia entre modelos de aprendizaje supervisado y modelos de aprendizaje no supervisado.
2. Comprender la necesidad de la utilización de modelos, algoritmos, software para predecir el comportamiento futuro.
3. Comprender la diferencia entre los datos de "train" y los datos de "testing".
4. Conocer los principales modelos predictivos, técnicas y algoritmos utilizados para predecir conductas a partir de un conjunto de datos históricos.
5. Utilizar el lenguaje Python para analizar y desarrollar ejemplos con datos reales.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.

3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar y programar proyectos de Machine Learning relacionados con aprendizaje Supervisado utilizando modelos predictivos.

Contenido

1. Conceptos de Aprendizaje Supervisado.

- a. Conceptos y diferencias entre aprendizaje supervisado y aprendizaje no supervisado.
- b. Diseño de bases de datos de aprendizaje.
- c. Diseño de bases de datos de testing.
- d. Variables cuantitativas y variables cualitativas.
- e. ¿Cómo evaluar la calidad de un modelo predictivos?
- f. Cálculo de la Matriz de confusión e índices de calidad.

2. Método de los K vecinos más cercanos.

- a. Estructura General del método.
- b. El mejor valor de K.
- c. Algoritmo de Aprendizaje.
- d. Aplicación con datos reales en Python y Scikit-Learn.

3. Máquinas de Soporte Vectorial.

- a. Hiperplano de separación de las clases.
- b. Vectores de soporte.
- c. Función discriminante lineal.
- d. ¿Cómo resolver un Problema Optimización?
- e. MVS no linealmente separables.
- f. Núcleos en Máquinas de Soporte Vectorial.
- g. Aplicación con datos reales en Python y Scikit-Learn.

4. Árboles de Decisión (Método CART).

- a. Algoritmos ID3, C4.5, C5.0 y CART.
- b. Árboles de auto-regresión.
- c. Aplicación con datos reales en Python y Scikit-Learn.

5. Métodos de Consenso y de Potenciación.

- a. Métodos de Consenso (Bagging).
- b. Bosques Aleatorios (Random forests).
- c. Métodos de impulso (Boosting).
- d. Métodos de Potenciación (ADA Boosting).
- e. Métodos de Potenciación (eXtreme Gradient Boosting).
- f. Aplicación con datos reales en Python y Scikit-Learn.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

Leo Breiman. "Random Forests". In: Machine Learning 45.1 (Oct. 1, 2001), pp. 5–32. issn: 1573-0565. doi: 10.1023/A:1010933404324. url: <https://doi.org/10.1023/A:1010933404324> (visited on 10/21/2022).

[2] JIANGANG Hao and Tin Kam Ho. "Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language". In: Journal of Educational and Behavioral Statistics 44.3 (June 1, 2019). Publisher: American Educational Research Association, pp. 348–361. issn: 1076-9986. doi: 10.3102/1076998619832248. url: <https://doi.org/10.3102/1076998619832248> (visited on 10/21/2022).

[3] Benjamin Johnston and Ishita Mathur. Applied Supervised Learning with Python: Use scikit-learn to build predictive models from real-world datasets and prepare yourself for the future of machine learning. Google-Books-ID: I_eVDwAAQBAJ. Packt Publishing Ltd, Apr. 27, 2019. 404 pp. isbn: 978-1-78995-583-5.

[4] William S. Noble. "What is a support vector machine?" In: Nature Biotechnology 24.12 (Dec. 2006). Number: 12 Publisher: Nature Publishing Group, pp. 1565–1567. issn: 1546- 1696. doi: 10.1038/nbt1206-1565. url: <https://www.nature.com/articles/nbt1206-1565> (visited on 10/21/2022).

[5] Shan Suthaharan. "Decision Tree Learning". In: Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning. Ed. by Shan Suthaharan. Integrated Series in Information Systems. Boston, MA: Springer US, 2016, pp. 237–269. isbn: 978-1-4899-7641-3. doi: 10.1007/978- 1- 4899- 7641- 3_10. url: https://doi.org/10.1007/978-1-4899-76413_10 (visited on 10/21/2022).

[6] Vaibhav Verdhhan. Supervised Learning with Python: Concepts and Practical Implementation Using Python. Berkeley, CA: Apress, 2020. isbn: 978-1-4842-6155-2 978-1-4842- 6156-9. doi: 10.1007/978-1-4842-6156-9. url: <http://link.springer.com/10.1007/978-1-4842-6156-9> (visited on 10/21/2022).



ML2006 Visualización de Datos en Python

Descripción

Al hacer análisis de datos se invierten muchos recursos en la correcta preparación de los datos y modelado; no obstante, es primordial presentar los resultados de forma tal que sea muy rápido y fácil interpretarlos; como bien se dice, una buena imagen (gráfico) es más valiosa que mil palabras. De esta forma, en este curso se presentarán los fundamentos del lenguaje Python para visualizar la información por medio del uso de herramientas gráficas.

Adicionalmente se usarán paquetes que permiten visualizarde mejor forma los datos según su tipo y dominio. Es importante recordar que la visualización de la información no es únicamente para presentar resultados finales, sino también para conocer los datos sobre los cuales se plantean los modelos.

Objetivos

En este curso el estudiante será capaz de:

1. Comprender y aplicar correctamente los principios de visualización de datos.
2. Aplicar correctamente los principios de la teoría del color en el campo de la visualización de datos.
3. Seleccionar el gráfico correcto según los datos que se deban representar.
4. Diseñar gráficos estáticos utilizando las librerías matplotlib y seaborn.
5. Comprender y aplicar los principios de la teoría de la gramática de los gráficos.
6. Crear gráficos interactivos con Plotly.
7. Visualizar información geoespacial utilizando mapas estáticos e interactivos.
8. Diseñar reportes profesionales utilizando Jupyter Notebooks.
9. Diseñar aplicaciones web con Dash.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar proyectos de Minería de Datos que involucren aspectos de visualización de datos, además será capaz de desarrollar reportes o aplicaciones con gráficos estáticos o interactivos.

Contenido

1. **Fundamentos de la visualización de datos.**
 - a. Historia de la visualización de datos.
 - b. Principios de la codificación visual.
 - I. Forma.
 - II. Posición.
 - III. Color.
 - IV. Tamaño.
 - V. Transparencia.
 - VI. Grupos.
 - c. Principios de la teoría del color aplicados a la visualización de datos.
 - d. Teoría de la gramática de los datos.
 - e. Selección de tipos de gráficos según los datos a representar.
 - f. Buenas prácticas para la presentación de datos.
 - I. Orden de los datos.
 - II. Títulos y anotaciones.
 - III. Uso correcto de las tipografías.
 - IV. Selección de colores aptos para personas con daltonismo.

2. Visualización de datos geoespaciales.

- a. Sistemas de coordenadas.
- b. Manejo de datos geoespaciales con python.
- c. Mapas coropléticos.
- d. Mapas de calor o densidad.
- e. Análisis de las limitaciones del uso de divisiones político-administrativas para la visualización de datos.
- f. Gráficos alternativos para datos geoespaciales.
- g. Mapas interactivos con python.

3. Gráficos interactivos.

- a. Introducción a Plotly.
- b. Estructura básica de un gráfico interactivo en Plotly.
- c. Desarrollo de gráficos interactivos en Plotly.
- d. Configuración de gráficos interactivos en Plotly.

4. Creación de reportes.

- a. Creación de reportes con Jupyter Notebooks.
- b. Personalización de los reportes para adaptar al estilo de tu organización.
- c. Exportar reportes en distintos formatos.

5. Desarrollo de aplicaciones.

- a. Introducción a Dash.
- b. Layout.
- c. Interactividad.
- d. Introducción a Dash Bootstrap Components.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Abha Belorkar et al. Interactive Data Visualization with Python: Present your data as an effective and compelling story, 2nd Edition. Google-Books-ID: _1PdDwAAQBAJ. Packt Publishing Ltd, Apr. 14, 2020. 362 pp. isbn: 978-1-80020-106-4.
- [2] Ekaba Bisong. "Matplotlib and Seaborn". In: Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners. Ed. by Ekaba Bisong. Berkeley, CA: Apress, 2019, pp. 151-165. isbn: 978-1-4842-4470-8. doi: 10.1007/978-1-4842-4470-8_12. url: https://doi.org/10.1007/978-1-4842-4470-8_12 (visited on 10/21/2022).
- [3] Elias Dabbas. Interactive Dashboards and Data Apps with Plotly and Dash: Harness the power of a fully fledged frontend web framework in Python – no JavaScript required. Google-Books-ID: b_gqEAAAQBAJ. Packt Publishing Ltd, May 21, 2021. 364 pp. isbn: 978-1-80056-035-2.
- [4] Sandro Tosi. Matplotlib for Python Developers. Google-Books-ID: 8Fs3AgAAQBAJ. Packt Publishing Ltd, Nov. 9, 2009. 525 pp. isbn: 978-1-84719-791-7.
- [5] Claus O. Wilke. Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures. Google-Books-ID: XmmNDwAAQBAJ. "O'Reilly Media, Inc.", Mar. 18, 2019. 390 pp. isbn: 978-1-4920-3105-5.



ML3000

Modelos Analíticos Avanzados en Python

Descripción

Al hacer análisis de datos se invierten muchos recursos en la correcta preparación de los datos y modelado; no obstante, es primordial presentar los resultados de forma tal que sea muy rápido y fácil interpretarlos; como bien se dice, una buena imagen (gráfico) es más valiosa que mil palabras. De esta forma, en este curso se presentarán los fundamentos del lenguaje Python para visualizar la información por medio del uso de herramientas gráficas.

Adicionalmente se usarán paquetes que permiten visualizar de mejor forma los datos según su tipo y dominio. Es importante recordar que la visualización de la información no es únicamente para presentar resultados finales, sino también para conocer los datos sobre los cuales se plantean los modelos.

Objetivos

En este curso el estudiante será capaz de:

1. Generar un modelo predictivo en Python con Scikit-Learn.
2. Aplicar Validación Cruzada (Cross-Validation) y Remuestreo (bootstrapping) para la medición correcta del error de un modelo predictivo.
3. Calibrar adecuadamente tanto métodos exploratorios como métodos predictivos en Machine Learning.
4. Seleccionar el mejor modelo predictivo dado un conjunto de datos.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una video conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.

2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar proyectos de Machine Learning que involucren evaluar, calibrar y seleccionar métodos exploratorios y predictivos en código Python.

Contenido

1. **Modelos supervisados en clasificación.**
 - a. Método de Bayes.
 - b. Análisis Discriminante Lineal.
 - c. Análisis Discriminante Cuadrático.
2. **Modelos supervisados en regresión.**
 - a. Regresión Clásica - Mínimos Cuadrados.
 - b. Regresión Ridge y LASSO.
 - c. Regresión con árboles de decisión, bosques aleatorios y potenciación.
 - d. Regresión usando Máquinas de Soporte Vectorial.
3. **Medir la calidad de un modelo supervisado.**
 - a. Training-testing-validación (the validation test approach).
 - b. Dejando uno afuera (Leave One Out).
 - c. Validación cruzada (cross-validation).
 - d. Curva ROC.
4. **Calibración y selección en modelos no supervisados y en modelos supervisados.**
 - a. Calibración de los parámetros de los supervisados.
 - b. Selección de los modelos más adecuados para el análisis una determinada base de datos.
 - c. Implementación óptima y paralela en Python.
 - d. Probabilidad de Corte como método para enfrentar problemas desbalanceados.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Benjamin Johnston and Ishita Mathur. Applied Supervised Learning with Python: Use scikit-learn to build predictive models from real-world datasets and prepare yourself for the future of machine learning. Google-Books-ID: l_eVDwAAQBAJ. Packt Publishing Ltd, Apr. 27, 2019. 404 pp. isbn: 978-1-78995-583-5.
- [2] Yuxi (Hayden) Liu. Python Machine Learning By Example. Google-Books-ID: 0nc5DwAAQBAJ. Packt Publishing Ltd, May 31, 2017. 249 pp. isbn: 978-1-78355-312-9.
- [3] Osvaldo Martin. Bayesian Analysis with Python. Google-Books-ID: t6PcDgAAQBAJ. Packt Publishing Ltd, Nov. 25, 2016. 282 pp. isbn: 978-1-78588-985-1.
- [4] Andreas C. Müller and Sarah Guido. Introduction to Machine Learning with Python: A Guide for Data Scientists. "O'Reilly Media, Inc.", Sept. 26, 2016. 400 pp. isbn: 978-1-4493-6990-3.
- [5] Tzu-Tsung Wong. "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation". In: Pattern Recognition 48.9 (Sept. 1, 2015), pp. 2839-2846. issn: 0031-3203. doi: 10.1016/j.patcog.2015.03.009. url: <https://www.sciencedirect.com/science/article/pii/S0031320315000989> (visited on 10/19/2022).



ML3003

Deep Learning en Python

Descripción

Cada vez más es necesario utilizar modelos basados en Deep Learning para resolver problemas, esto debido a que ahora es más común ver problemas que no provienen de una tabla como lo son las imágenes y sonidos. Por tanto, en este curso se presentarán métodos en Machine Learning enfocados en métodos de "Deep Learning".

Objetivos

En este curso el estudiante será capaz de:

1. Conocer el significado de Deep Learning y por qué es ampliamente usado en problemas de Big Data.
2. Conocer los distintos tipos de métodos basados en Deep Learning.
3. Conocer las principales modelos empleados en Deep Learning y su aplicación en clasificación, regresión y series de tiempo.
4. Entender el funcionamiento y método de aprendizaje de las Redes Neuronales clásicas y Convolucionales.
5. Desarrollar modelos de Deep Learning con Redes Neuronales empleando las librerías Keras y TensorFlow.
6. Usar estructuras especializadas de Redes Neuronales para la aplicación en problemas de texto, imágenes y sonidos.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar proyectos de Machine Learning que involucren evaluar, calibrar y seleccionar métodos predictivos basados en Deep Learning en código Python.

Contenido

- 1. Fundamentos matemáticos del Deep Learning (aprendizaje profundo).**
 - a. Fundamentos del Álgebra Lineal.
 - b. Tensores en múltiples dimensiones.
 - c. Fundamentos de Optimización – Método del Descenso del Gradiente.
 - d. Desmitificando el descenso del gradiente.
 - e. Descenso de gradiente versus descenso de gradiente estocástico.
 - f. Fundamentos de Probabilidad.
- 2. Redes Neuronales Clásicas.**
 - a. Algoritmo del Perceptrón.
 - b. Redes neuronales de múltiples capas.
 - c. Aplicaciones en casos reales con Python, paquetes Keras y Scikit-Learn.
- 3. Redes Neuronales Convolucionales (CNN).**
 - a. Redes Neuronales Convolucionales - Convolutional Neural Networks (CNN).
 - b. La necesidad de usar CNN.
 - c. Los tipos de datos utilizados en las CNN.
 - d. Entendiendo las CNN.
 - e. Entrenamiento y optimización.
 - f. Arquitecturas populares.

4. Redes Neuronales Recurrentes (RNN).

- a. Redes Neuronales Recurrentes – Recurrent Neural Networks (RNN).
- b. La necesidad de usar RNN.
- c. Los tipos de datos utilizados en las RNN.
- d. Entendiendo a los RNN.
- e. Memoria a corto plazo.
- f. Unidades recurrentes cerradas.
- g. RNN profundos.
- h. Las redes neuronales de memoria de corto y largo plazo – Long-Short Term Memory (LSTM).
- i. Entrenamiento y optimización.
- j. Arquitecturas populares.

5. Otros modelos de Deep Neural Networks..

- a. Autoencoders.
- b. Métodos adaptativos de descenso de gradientes.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Rowel Atienza. Advanced Deep Learning with Keras: Apply deep learning techniques, autoencoders, GANs, variational autoencoders, deep reinforcement learning, policy gradients, and more. Google-Books-ID: Vuh1DwAAQBAJ. Packt Publishing Ltd, Oct. 31, 2018. 369 pp. isbn: 978-1-78862-453-4.
- [2] Jason Brownlee. Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python. Google-Books-ID: o5qnDwAAQBAJ. Machine Learning Mastery, Aug. 30, 2018. 572 pp.
- [3] Chollet François. Deep Learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek. Google-Books-ID: ouVcDwAAQBAJ. MITP, May 24, 2018. 618 pp. isbn: 978-3-95845-840-6.
- [4] Sebastian Raschka and Vahid Mirjalili. Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow 2. Third edition. Expert insight. Birmingham Mumbai: Packt, 2019. 741 pp. isbn: 978-1-78995-575-0.
- [5] Sudharsan Ravichandiran. Hands-On Deep Learning Algorithms with Python: Master deep learning algorithms with extensive math by implementing them using TensorFlow. Google-Books-ID: 8DqIDwAAQBAJ. Packt Publishing Ltd, July 25, 2019. 498 pp. isbn: 978-1-78934-451-6.
- [6] Mohit Sewak, Md Rezaul Karim, and Pradeep Pujari. Practical Convolutional Neural Networks: Implement advanced deep learning models using Python. Google-Books-ID: bOIODwAAQBAJ. Packt Publishing Ltd, Feb. 27, 2018. 211 pp. isbn: 978-1-78839-414-7.
- [7] Ivan Vasilev et al. Python Deep Learning: Exploring deep learning techniques and neural network architectures with PyTorch, Keras, and TensorFlow, 2nd Edition. Google-Books-ID: ESKEDwAAQBAJ. Packt Publishing Ltd, Jan. 16, 2019. 379 pp. isbn: 978-1-78934-970-2.



ML3006 Web Mining en Python

Descripción

En este curso se presentarán los conceptos y técnicas necesarios para extraer datos de la web. Se estudian los fundamentos de HTML, CSS y JavaScript para entender el funcionamiento de una página web. Se estudiará el uso de expresiones regulares y su aplicación en la minería de texto para la correcta extracción de datos de la web. Se aprenderá a extraer datos tanto de páginas web estáticas como dinámicas, lo cual incluye redes sociales.

Objetivos

En este curso el estudiante será capaz de:

1. Entender los elementos básicos de un página web.
2. Aprender a extraer información de páginas web estáticas.
3. Aprender a extraer información de páginas web dinámicas.
4. Aprender a extraer información de Redes Sociales.
5. Manejar paginación y navegación de páginas web de forma automatizada.
6. Aplicar expresiones regulares para la limpieza y extracción de datos.
7. Analizar información obtenida de páginas web.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.

2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar y programar proyectos de Machine Learning que involucren extraer, limpiar, corregir y analizar información de sitios web utilizando código automatizado en Python.

Contenido

1. Conceptos básicos de una página web.

- a. ¿Qué es una página web?
- b. Diferencias entre una página web estática y una dinámica.
- c. ¿Qué es Web Mining, Web Scraping y Text Mining?
- d. HTML (Hyper Text Markup Language).
- e. CSS (Cascading Style Sheets).
- f. JavaScript.

2. Extracción de datos de la web (Web Scraping).

- a. XPath como lenguaje de consultas para documentos WEB.
- b. BeautifulSoup como herramienta para extraer datos de páginas web estáticas.
- c. Scrapy-Splash como herramientas para extraer datos de páginas web dinámicas.
- d. Manejar paginación y navegación.
- e. Minando las Redes Sociales.

3. Minería de texto (Text Mining).

- a. Expresiones regulares.
 - I. Clases de caracteres.
 - II. Cuantificadores.
 - III. Aserciones.
- b. Manipulación y limpieza de textos.
 - I. Detección de patrones de texto.
 - II. Extracción de patrones de texto.
 - III. Eliminación de patrones de texto.
 - IV. Modificación de patrones de texto.

4. Análisis de datos obtenidos de la web.

- a. KNN.
- b. Árboles.
- c. Bosques Aleatorios.
- d. Potenciación y XGBoosting.

5. Calibración y Selección de Modelos Predictivos desde la consola de R.

- a. Preparar los datos para un análisis de métodos no supervisados.
- b. Preparar los datos para un análisis de métodos supervisados.
- c. Nubes de palabras (wordclouds).
- d. Análisis de Sentimientos.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Seppe Vanden Broucke and Bart Baesens. Practical web scraping for data science: best practices and examples with Python. In collab. with Seppe Vanden Broucke and Bart Baesens. CreateSpace, 2017. 306 pp. url: <https://eprints.soton.ac.uk/425855/> (visited on 10/10/2022).
- [2] Markus Hofmann and Andrew Chisholm. Text Mining and Visualization: Case Studies Using Open-Source Tools. Google-Books-ID: JfQYCwAAQBAJ. CRC Press, Jan. 5, 2016. 337 pp. isbn: 978-1-4822-3758-0.
- [3] Dimitrios Kouzis-Loukas. Learning Scrapy. Google-Books-ID: EF8dDAAAQBAJ. Packt Publishing Ltd, Jan. 30, 2016. 270 pp. isbn: 978-1-78439-091-4.
- [4] Bing Liu. Web Data Mining. Berlin, Heidelberg: Springer, 2011. isbn: 978-3-642-19459-7 978-3-642-19460-3. doi: 10.1007/978-3-642-19460-3. url: <http://link.springer.com/10.1007/978-3-642-19460-3> (visited on 10/10/2022).
- [5] Mining the Social Web, 3rd Edition [Book]. ISBN: 9781491985045. url: <https://www.oreilly.com/library/view/mining-the-social/9781491973547/> (visited on 10/10/2022).
- [6] Ryan Mitchell. Web Scraping with Python: Collecting More Data from the Modern Web. Google-Books-ID: TYtSDwAAQBAJ. "O'Reilly Media, Inc.", Mar. 21, 2018. 329 pp. isbn: 978-1-4919-8552-6..



ML3009

Series de Tiempo en Python

Descripción

En este curso se presentarán los principales conceptos, algoritmos y métodos en Series de Tiempo. Para ello, con la ayuda del lenguaje Python y paquetes relacionados con series de tiempo se verá como limpiar, manipular y crear una serie de tiempo para que pueda ser analizada y posteriormente poder aplicar modelos predictivos que sean especialmente para datos de tipo series de tiempo.

Objetivos

En este curso el estudiante será capaz de:

1. Entender la necesidad de la utilización de modelos, algoritmos, software especial para la predicción en Series de Tiempo.
2. Limpiar, manipular y corregir datos de tipo series de tiempo.
3. Aprender las técnicas estadísticas aplicables a datos tipo series de tiempo para preparar pronósticos.
4. Aprender procesos y técnicas para preparar y visualizar series de tiempo en Python.
5. Aprender los métodos basados en regresión, suavizado exponencial, Método de Holt Winters, métodos ARIMA y Deep Learning.
6. Utilizar Python para analizar datos reales de Series de Tiempo.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.

2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar y programar proyectos de Machine Learning que involucren extraer, limpiar, corregir, analizar y predecir problemas relacionados con Series de Tiempo.

Contenido

1. **Series de tiempo.**
 - a. Conceptos básicos de una serie de tiempo.
 - b. Características de las series de tiempo.
 - c. Instalación de los paquetes necesarios en Python para series de tiempo.
 - d. Corregir series de tiempo.
 - e. Crear series de tiempo en Python.
 - f. Visualización en Python de las Series de Tiempo.
 - g. Pronósticos y cálculo del error.
 - h. Filtrado Lineal.
 - i. Descomposición de una Serie de Tiempo: tendencia, ciclos y estacionalidades.
 - j. Métodos basados en Regresión.
 - k. Aplicaciones en casos reales con Python.
2. **Ajuste exponencial y método de Holt-Winters.**
 - a. Suavizado exponencial y predicción de Series de tiempo.
 - b. Método de Holt-Winters.
 - c. Aplicaciones en casos reales con Python.
3. **Métodos ARIMA.**
 - a. Análisis de las autocorrelaciones y autocorrelaciones parciales.
 - b. Aplicaciones en casos reales con Python.
 - c. Estimación de parámetros en los modelos ARIMA.
 - d. Comprobación y diagnóstico.

- e. Predicción mediante los Modelos ARIMA.
- f. Periodograma y Transformaciones de Fourier Discretas.

4. Métodos Basados en Redes Neuronales y Deep Learning.

- a. Uso de las redes Neuronales aplicados a modelos de series de tiempo.
- b. Redes Neuronales Recurrentes en series de tiempo.

5. Utilización de Reglas para mejorar la predicción.

- a. Reglas con criterio experto.
- b. Reglas basadas en análisis histórico de los casos especiales.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] Jason Brownlee. Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python. Google-Books-ID: o5qnDwAAQBAJ. Machine Learning Mastery, Aug. 30, 2018. 572 pp.
- [2] Jason Brownlee. Introduction to Time Series Forecasting With Python: How to Prepare Data and Develop Models to Predict the Future. Machine Learning Mastery, Feb. 16, 2017. 359 pp.
- [3] Dr Avishek Pal and Dr PKS Prakash. Practical Time Series Analysis: Master Time Series Data Processing, Visualization, and Modeling using Python. Google-Books-ID: EJpGDwAAQBAJ. Packt Publishing Ltd, Sept. 28, 2017. 238 pp. isbn: 978-1-78829-419-5.
- [4] B V Vishwas and Ashish Patel. Hands-on Time Series Analysis with Python: From Basics to Bleeding Edge Techniques. Berkeley, CA: Apress, 2020. isbn: 978-1-4842-5991-7978-1-4842-5992-4. doi: 10.1007/978-1-4842-5992-4. url: <http://link.springer.com/10.1007/978-1-4842-5992-4> (visited on 10/21/2022).



ML9000

Proyecto Final de Graduación

Descripción

En este curso el estudiante desarrollará de principio a fin, siguiendo las buenas prácticas de gestión de proyectos en Machine Learning, su propio proyecto con datos de la empresa en donde trabaja o datos en los cuales está particularmente interesado(a).

Objetivos

En este curso el estudiante será capaz de:

1. Utilizar adecuadamente la metodología CRISP-DM para el desarrollo de proyectos en Machine Learning.
2. Distinguir entre un proyecto de segmentación, clasificación, regresión y series de tiempo.
3. Elegir, para un problema concreto, qué técnicas de Machine Learning son las más apropiadas.
4. Determinar la herramienta de software más adecuada para el enfrentar el problema planteado.
5. Generar los modelos y patrones elegidos utilizando una herramienta o paquete de Machine Learning.
6. Evaluar la calidad de un modelo, utilizando técnicas sencillas de evaluación, validación cruzada.
7. Integrar sistemas de Bases de Datos desde Python.
8. Presentar resultados en aplicaciones Web.

Metodología

Basado en la teoría y en la aplicación directa de los conceptos aprendidos. Para esto se dispondrán de las siguientes herramientas:

1. Una vídeo conferencia semanal, las cuales quedarán grabadas en Zoom, para que los alumnos la puedan acceder en cualquier momento.
2. Trabajos prácticos semanales.
3. Foros para plantear dudas al tutor y compañeros.
4. Aula virtual en Moodle.

Luego de este curso el estudiante será capaz de:

Desarrollar proyectos de Machine Learning de principio a fin, es decir, iniciando con la obtención y limpieza de datos hasta la generación de conocimiento derivados de los datos de la organización.

Contenido

1. Fundamentos de la metodología para la implementación de proyectos de Machine Learning CRISP-DM.
2. Integración de sistemas de bases de datos en Python, para usarlos como fuente directa de datos.
3. Integración de los resultados de un proceso de Machine Learning en una aplicación web.
4. Estudio comparativo de diferentes tipos problemas en Machine Learning.

Evaluación

El curso se evalúa con 4 tareas, una por semana, cada tarea tiene un valor de 25 puntos. La nota mínima de aprobación es de 70.

Bibliografía

- [1] José Alberto Gallardo Arancibia. "Metodología para la definición de requisitos en proyectos de data mining (er-dm)". <http://purl.org/dc/dcmitype/Text>. Universidad Politécnica de Madrid, 2009. url: <https://dialnet.unirioja.es/servlet/tesis?codigo=20961> (visited on 10/19/2022).
- [2] Ana Azevedo and Manuel Filipe Santos. "KDD, SEMMA and CRISP-DM: a parallel overview". In: IADS - DM (2008). Accepted: 2012-06-14T09:51:14Z. url: <https://recipp.ipp.pt/handle/10400.22/136> (visited on 10/19/2022).
- [3] Davy Cielen and Arno Meysman. Introducing Data Science: Big data, machine learning, and more, using Python tools. Google-Books-ID: bTozEAAAQBAJ. Simon and Schuster, May 2, 2016. 475 pp. isbn: 978-1-63835-249-5.
- [4] CRISP-DM 1.0: Step-by-step Data Mining Guide. Google-Books-ID: po7FtgAACAAJ. SPSS, 2000. book.
- [5] Elias Dabbas. Interactive Dashboards and Data Apps with Plotly and Dash: Harness the power of a fully fledged frontend web framework in Python – no JavaScript required. Google-Books-ID: b_gqEAAAQBAJ. Packt Publishing Ltd, May 21, 2021. 364 pp. isbn: 978-1-80056-035-2.
- [6] Albert Lukaszewski and Antony Reynolds. MySQL for Python. Google-Books-ID: SWa1AAAAQBAJ. Packt Publishing Ltd, Sept. 21, 2010. 771 pp. isbn: 978-1-84951-019-6.



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