

El Tutorial De Python

"This book introduces you to R, RStudio, and the tidyverse, a collection of R packages designed to work together to make data science fast, fluent, and fun. Suitable for readers with no previous programming experience"--

El Tutorial de Python Python Tutorial Release 3. 6. 6rc1 Createspace Independent Publishing Platform

This is the eBook of the printed book and may not include any media, website access codes, or print supplements that may come packaged with the bound book. Used by sites as varied as Twitter, GitHub, Disney, and Airbnb, Ruby on Rails is one of the most popular frameworks for developing web applications, but it can be challenging to learn and use. Whether you're new to web development or new only to Rails, *Ruby on Rails™ Tutorial, Fourth Edition*, is the solution. Best-selling author and leading Rails developer Michael Hartl teaches Rails by guiding you through the development of three example applications of increasing sophistication. The tutorial's examples focus on the general principles of web development needed for virtually any kind of website. The updates to this edition include full compatibility with Rails 5, a division of the largest chapters into more manageable units, and a huge number of new exercises interspersed in each chapter for maximum reinforcement of the material. This indispensable guide provides integrated tutorials not only for Rails, but also for the essential Ruby, HTML, CSS, and SQL skills you need when developing web applications. Hartl explains how each new technique solves a real-world problem, and then he demonstrates it with bite-sized code that's simple enough to understand, yet novel enough to be useful. Whatever your previous web development experience, this book will guide you to true Rails mastery. This book will help you install and set up your Rails development environment, including pre-installed integrated development environment (IDE) in the cloud. Go beyond generated code to truly understand how to build Rails applications from scratch. Learn testing and test-driven development (TDD). Effectively use the Model-View-Controller (MVC) pattern. Structure applications using the REST architecture. Build static pages and transform them into dynamic ones. Master the Ruby programming skills all Rails developers need. Create high-quality site layouts and data models. Implement registration and authentication systems, including validation and secure passwords. Update, display, and delete users. Upload images in production using a cloud storage service. Implement account activation and password reset, including sending email with Rails. Add social features and microblogging, including an introduction to Ajax. Record version changes with Git and create a secure remote repository at Bitbucket. Deploy your applications early and often with Heroku.

Invent Your Own Computer Games with Python will teach you how to make computer games using the popular Python programming language—even if you've never programmed before! Begin by building classic games like

Hangman, Guess the Number, and Tic-Tac-Toe, and then work your way up to more advanced games, like a text-based treasure hunting game and an animated collision-dodging game with sound effects. Along the way, you'll learn key programming and math concepts that will help you take your game programming to the next level. Learn how to: –Combine loops, variables, and flow control statements into real working programs –Choose the right data structures for the job, such as lists, dictionaries, and tuples –Add graphics and animation to your games with the pygame module –Handle keyboard and mouse input –Program simple artificial intelligence so you can play against the computer –Use cryptography to convert text messages into secret code –Debug your programs and find common errors As you work through each game, you'll build a solid foundation in Python and an understanding of computer science fundamentals. What new game will you create with the power of Python? The projects in this book are compatible with Python 3.

Este libro fue pensado para quién desea iniciarse en el mundo de la programación de computadoras. Conceptos básicos de programación – tales como expresiones, variables, repeticiones, decisiones, listas, funciones, archivos y banco de datos con SQLite 3 – son presentados uno por uno con ejemplos y ejercicio claros. El objetivo de esta obra es explorar la programación de computadoras como una herramienta para el día a día. Puede ser útil para un curso de introducción a la programación de computadoras y ser usado como guía de estudios por autodidactas. Para lograr un aprovechamiento pleno del contenido de la obra son necesarios conocimientos básicos de informática, tales como escribir textos, abrir y guardar archivos. Todo el software utilizado en el libro puede ser bajado gratuitamente y ejecutarse en Windows, Linux y Mac OS X. Aunque el lenguaje Python (versión 3.x) es muy poderoso y repleto de recursos modernos de programación, este libro pretende enseñar a programar y no el lenguaje en sí. Algunos recursos del lenguaje no han sido utilizados para privilegiar los ejercicios de lógica de programación y ofrecer al lector una preparación más amplia para otros lenguajes. Esta decisión no afecta la presentación de recursos poderosos del lenguaje, aunque el libro no debe ser considerado una obra de referencia definitiva y sí un guía introductorio a este tema.

El deep learning es un tipo de machine learning o aprendizaje automático cuyo fin es entrenar a una PC para que sea capaz de realizar tareas como los seres humanos, por ejemplo, el reconocimiento del habla o la identificación de imágenes. En este Informe Users se presenta esta tecnología para enseñarte a crear los rudimentos de tu primera red neuronal.

Python for Everybody is designed to introduce students to programming and software development through the lens of exploring data. You can think of the Python programming language as your tool to solve data problems that are beyond the capability of a spreadsheet. Python is an easy to use and easy to learn programming language that is freely available on Macintosh, Windows, or

Linux computers. So once you learn Python you can use it for the rest of your career without needing to purchase any software. This book uses the Python 3 language. The earlier Python 2 version of this book is titled "Python for Informatics: Exploring Information". There are free downloadable electronic copies of this book in various formats and supporting materials for the book at www.pythonlearn.com. The course materials are available to you under a Creative Commons License so you can adapt them to teach your own Python course.

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python Web site, <https://www.python.org/>, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation. The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications. This tutorial introduces the reader informally to the basic concepts and features of the python language and system. It helps to have a Python interpreter handy for hands-on experience, but all examples are self contained, so the tutorial can be read off-line as well. For a description of standard objects and modules, see [library-index](#). [reference-index](#) gives a more formal definition of the language. To write extensions in C or C++, read [extending-index](#) and [c-api-index](#). There are also several books covering Python in depth. This tutorial does not attempt to be comprehensive and cover every single feature, or even every commonly used feature. Instead, it introduces many of Python's most noteworthy features, and will give you a good idea of the language's flavor and style. After reading it, you will be able to read and write Python modules and programs, and you will be ready to learn more about the various Python library modules described in [library-index](#). The Glossary is also worth going through.

Python. Aplicaciones prácticas es un libro para entusiastas de la programación, estudiantes y profesionales en el mundo Python, los capítulos inician con prácticas sencillas que aumentan de complejidad gradualmente y está desarrollado en el lenguaje de programación Python. Python está escrito en el lenguaje C, por lo que se puede extender a través de su api en C o C++ y escribir nuevos tipos de datos, funciones, etc. En la actualidad hay dos vertientes la versión 2.x y 3.x, al final llegara el momento que se integraran estas dos versiones, es recomendable utilizar la última versión estable 3.x. Algunas de las características más importantes es que Python es multiparadigma: Programación estructurada, Programación Orientada a Objetos y Programación Funcional. El objetivo de este libro es brindar al lector los fundamentos para introducir o reforzar conocimientos en temas como: Kivy, Django, Juegos

Pygame, Introducción Arduino con Python, Matemática Científica -Scipy Inteligencia Artificial, Procesamiento de Imágenes, Forense,Blockchain. El libro contiene material adicional que podrá descargar accediendo a la ficha del libro en www.ra-ma.es. Este material incluye la construcción y código propuestos en esta obra.

The second edition of this best-selling Python book (over 500,000 copies sold!) uses Python 3 to teach even the technically uninclined how to write programs that do in minutes what would take hours to do by hand. There is no prior programming experience required and the book is loved by liberal arts majors and geeks alike. If you've ever spent hours renaming files or updating hundreds of spreadsheet cells, you know how tedious tasks like these can be. But what if you could have your computer do them for you? In this fully revised second edition of the best-selling classic Automate the Boring Stuff with Python, you'll learn how to use Python to write programs that do in minutes what would take you hours to do by hand--no prior programming experience required. You'll learn the basics of Python and explore Python's rich library of modules for performing specific tasks, like scraping data off websites, reading PDF and Word documents, and automating clicking and typing tasks. The second edition of this international fan favorite includes a brand-new chapter on input validation, as well as tutorials on automating Gmail and Google Sheets, plus tips on automatically updating CSV files. You'll learn how to create programs that effortlessly perform useful feats of automation to:

- Search for text in a file or across multiple files
- Create, update, move, and rename files and folders
- Search the Web and download online content
- Update and format data in Excel spreadsheets of any size
- Split, merge, watermark, and encrypt PDFs
- Send email responses and text notifications
- Fill out online forms

Step-by-step instructions walk you through each program, and updated practice projects at the end of each chapter challenge you to improve those programs and use your newfound skills to automate similar tasks. Don't spend your time doing work a well-trained monkey could do. Even if you've never written a line of code, you can make your computer do the grunt work. Learn how in Automate the Boring Stuff with Python, 2nd Edition.

This document is a self learning document for a course in Python programming. This course contains (1) a part for beginners, (2) a discussion of several advanced topics that are of interest to Python programmers, and (3) a Pythonworkbook with lots of exercises.

"This manual is part of the official reference documentation for Python, an object-oriented programming language created by Guido van Rossum. Python is free software. The term "free software" refers to your freedom to run, copy, distribute, study, change and improve the software. With Python you have all these freedoms. You can support free software by becoming an associate member of the Free Software Foundation. The Free Software Foundation is a tax-exempt charity dedicated to promoting the right to use, study, copy, modify, and redistribute computer programs. It also helps to spread awareness of the ethical and political issues of freedom in the use of software. For more information visit the website www.fsf.org. The development of Python itself is supported by the Python Software Foundation. Companies using Python can invest in the language by becoming sponsoring members of this group. Donations can also be made online through the Python website. Further information is available at <http://www.python.org/psf/>."--Page 1.

Raspberry Pi, fundamentos y aplicaciones, es un libro que pretende mostrar al lector los fundamentos sobre los que se basan las nuevas tecnologías en el desarrollo del hardware abierto implementando como plataforma la placa Raspberry Pi. El libro es adecuado para lectores interesados en el mundo Raspberry Pi pero también en el conocimiento de placas de hardware abierto en general. Está pensado para estudiantes de tecnología, ingenieros, técnicos e informáticos así como profesores y docentes de cualquiera de estas disciplinas pero también para cualquier interesado en el mundo de la electrónica y de la actualidad de estos entornos de desarrollo.

Raspberry Pi, está desarrollado de tal forma que el lector asimile de una forma rápida los fundamentos que le harán falta para la comprensión o desarrollo posterior de cualquier proyecto que encuentre o quiera diseñar, siendo muy representativo para cualquier otra placa hardware con la que desee trabajar. El libro incluye al final ejemplos prácticos para la asimilación de los conceptos y la aplicación en la capa física a través del GPIO donde también se expone la iniciación y aplicación concreta con Raspberry Pi en el campo del Internet de las cosas (IoT), campo que en la actualidad presenta muchas aplicaciones y en crecimiento constante.

El número creciente de personas que colaboran en Internet, deliberada o accidentalmente, ha creado un enorme conjunto de datos alimentados por millones de interacciones dentro de la experiencia de usuario, marketing, gustos personales y comportamiento humano en general. Este libro proporciona una introducción al emergente campo de la inteligencia colectiva. Se abarcan las formas de procesar conjuntos de datos de muchos sitios web interesantes de los cuales probablemente habremos oído, y los temas relacionados con el cómo recolectar datos de los usuarios de tus propias aplicaciones, y muchas diferentes formas de analizar y comprender los datos una vez que los encuentras. El objetivo de este libro es llevarte más allá de las aplicaciones sencillas de base de datos y enseñarte a escribir programas inteligentes para tomar ventaja de la información que tú y otros entregan cada día.

This book shows readers how they can successfully analyze data using only two core machine learning algorithms---and how to do so using the popular Python programming language. These algorithms deal with common scenarios faced by all data analysts and data scientists. This book focuses on two algorithm families (linear methods and ensemble methods) that effectively predict outcomes. This type of problem covers a multitude of use cases (what ad to place on a web page, predicting prices in securities markets, detecting credit card fraud, etc.). The focus on two families gives enough room for full descriptions of the mechanisms at work in the algorithms. Then the code examples serve to illustrate the workings of the machinery with specific hackable code. The author will explain in simple terms, using no complex math, how these algorithms work, and will then show how to apply them in Python. He will also provide advice on how to select from among these algorithms, and will show how to prepare the data, and how to use the trained models in practice. The author begins with an overview of the two core algorithms, explaining the types of problems solved by each one. He then introduces a core set of Python programming techniques that can be used to apply these algorithms. The author shows various techniques for building predictive models that solve a range of problems, from simple to complex; he also shows how to measure the performance of each model to ensure you use the right one. The following chapters provide a deep dive into each of the two algorithms: penalized linear regression and

ensemble methods. Chapters will show how to apply each algorithm in Python. Readers can directly use the sample code to build their own solutions. Python Essential Reference is the definitive reference guide to the Python programming language — the one authoritative handbook that reliably untangles and explains both the core Python language and the most essential parts of the Python library. Designed for the professional programmer, the book is concise, to the point, and highly accessible. It also includes detailed information on the Python library and many advanced subjects that is not available in either the official Python documentation or any other single reference source. Thoroughly updated to reflect the significant new programming language features and library modules that have been introduced in Python 2.6 and Python 3, the fourth edition of Python Essential Reference is the definitive guide for programmers who need to modernize existing Python code or who are planning an eventual migration to Python 3. Programmers starting a new Python project will find detailed coverage of contemporary Python programming idioms. This fourth edition of Python Essential Reference features numerous improvements, additions, and updates: Coverage of new language features, libraries, and modules Practical coverage of Python's more advanced features including generators, coroutines, closures, metaclasses, and decorators Expanded coverage of library modules related to concurrent programming including threads, subprocesses, and the new multiprocessing module Up-to-the-minute coverage of how to use Python 2.6's forward compatibility mode to evaluate code for Python 3 compatibility Improved organization for even faster answers and better usability Updates to reflect modern Python programming style and idioms Updated and improved example code Deep coverage of low-level system and networking library modules — including options not covered in the standard documentation

Este texto es el resultado de nuestra labor como orientadores del curso de métodos numéricos en la Institución Universitaria Politecnico Grancolombiano durante varios años. Aunque inicialmente solamente se disponía de notas de clase construidas de manera informal para los cursos, con el tiempo surgió la necesidad de consolidarlas en un solo documento. El libro contiene métodos computacionales para resolver problemas esenciales en el campo de ingeniería o matemática aplicada. En cada método se pretende dar al lector una visión de su esencia, necesidad, ventajas y desventajas. En algunos casos, alejándose de presentaciones rigurosas, pero sin dejar de ser correctas. El objetivo final es proporcionar los elementos necesarios para la aplicación adecuada de los algoritmos. El capítulo 1 expone los principales procedimientos computacionales para la solución de ecuaciones de una variable, describiendo sus ventajas y desventajas computacionales. El capítulo 2 presenta las herramientas básicas para la estimación de nuevos datos a partir de un conjunto de puntos. Se abordan dos enfoques de interpolación: exacta y ajuste de mínimo cuadrados. El capítulo 3 comprende los algoritmos de factorización LY, Jacobi y Gauss-Seidel para la solución de sistemas de ecuaciones lineales. El capítulo 4 se encuentra dedicado a los métodos numéricos para la solución de ecuaciones diferenciales, iniciando con el método de Euler y finalizando con el método RKA. Al final de libro, el lector puede encontrar todos los algoritmos presentados a lo largo del texto desarrollados en Python, un lenguaje de programación flexible y que responde a las necesidades didácticas del curso. Para terminar, deseamos expresar nuestros

agradecimientos a los alumnos y compañeros que, a través de sus inquietudes, permitieron la construcción de los contenidos de este documento.

This book offers a concise and gentle introduction to finite element programming in Python based on the popular FEniCS software library. Using a series of examples, including the Poisson equation, the equations of linear elasticity, the incompressible Navier–Stokes equations, and systems of nonlinear advection–diffusion–reaction equations, it guides readers through the essential steps to quickly solving a PDE in FEniCS, such as how to define a finite variational problem, how to set boundary conditions, how to solve linear and nonlinear systems, and how to visualize solutions and structure finite element Python programs. This book is open access under a CC BY license.

Python es un lenguaje de programación multiplataforma, consistente y maduro, en el cual confían con éxito las Empresas y organizaciones mundiales más prestigiosas: Google, la NASA, YouTube, Intel y Yahoo! Su éxito está vinculado tanto al hecho de que favorece la productividad, haciendo más sencillo el desarrollo de sistemas de software sin tener en cuenta su complejidad, como al hecho de que tiene múltiples entornos de uso: aplicaciones web, juegos y multimedia, interfaces gráficas, networking, aplicaciones científicas, inteligencia artificial y programación de sistemas, entre muchos otros. El gran libro de Python es el más completo, moderno y detallado de entre los volúmenes dedicados a Python que pueden encontrarse actualmente en el mercado. Actualizado a la versión 3.4 del lenguaje, lanzada en enero de 2014. Su composición es muy detallada y sigue un curso gradual elaborado en torno a una amplia serie de ejemplos y ejercicios: parte de las bases del lenguaje, sin dar nada por sabido, hasta llegar a los argumentos considerados más difíciles, incluso por los programadores más experimentados. Soporte al libro disponible online:

[code.google.com/p/the\[phytonic\]way/](http://code.google.com/p/the[phytonic]way/) Aspectos destacados: - Introducción al lenguaje Python, a su sintaxis, a sus construcciones fundamentales y a la librería estándar. - Funciones y modos de emparejamiento de argumentos, generadores, corrutinas, archivos, comodines y expresiones regulares. - Módulos y paquetes, entornos y espacios de nombres, ambientes virtuales, instalación y distribución de aplicaciones. - Prueba de validación de cadenas de documentación y desarrollo guiado por pruebas. - Programación orientada a objetos en Python: clases, herencia, gestión de las excepciones, patrón y antipatrón, propiedades y decoradores. - Modelo a objetos de Python, atributos mágicos, descriptores y metaclasses. Marco Buttu: ingeniero electrónico. Tras haber trabajado en el campo del Network on Chip y de las nuevas arquitecturas para la elaboración paralela, aterrizó en el Istituto Nazionale di Astrofisica, en el cual se ocupa del desarrollo del software de control del Sardinia Radio Telescope, el radiotelescopio europeo más moderno y uno de los más grandes del mundo. En 2004 tuvo su primer encuentro con Python y desde entonces no ha podido separarse de él.

Written for statisticians, computer scientists, geographers, research and applied scientists, and others interested in visualizing data, this book presents a unique foundation for producing almost every quantitative graphic found in scientific journals, newspapers, statistical packages, and data visualization systems. It was designed for a distributed computing environment, with special attention given to conserving computer code and system resources. While the tangible result of this work is a Java production

graphics library, the text focuses on the deep structures involved in producing quantitative graphics from data. It investigates the rules that underlie pie charts, bar charts, scatterplots, function plots, maps, mosaics, and radar charts. These rules are abstracted from the work of Bertin, Cleveland, Kosslyn, MacEachren, Pinker, Tufte, Tukey, Tobler, and other theorists of quantitative graphics.

Master Bayesian Inference through Practical Examples and Computation—Without Advanced Mathematical Analysis Bayesian methods of inference are deeply natural and extremely powerful. However, most discussions of Bayesian inference rely on intensely complex mathematical analyses and artificial examples, making it inaccessible to anyone without a strong mathematical background. Now, though, Cameron Davidson-Pilon introduces Bayesian inference from a computational perspective, bridging theory to practice—freeing you to get results using computing power. Bayesian Methods for Hackers illuminates Bayesian inference through probabilistic programming with the powerful PyMC language and the closely related Python tools NumPy, SciPy, and Matplotlib. Using this approach, you can reach effective solutions in small increments, without extensive mathematical intervention. Davidson-Pilon begins by introducing the concepts underlying Bayesian inference, comparing it with other techniques and guiding you through building and training your first Bayesian model. Next, he introduces PyMC through a series of detailed examples and intuitive explanations that have been refined after extensive user feedback. You'll learn how to use the Markov Chain Monte Carlo algorithm, choose appropriate sample sizes and priors, work with loss functions, and apply Bayesian inference in domains ranging from finance to marketing. Once you've mastered these techniques, you'll constantly turn to this guide for the working PyMC code you need to jumpstart future projects. Coverage includes

- Learning the Bayesian “state of mind” and its practical implications
- Understanding how computers perform Bayesian inference
- Using the PyMC Python library to program Bayesian analyses
- Building and debugging models with PyMC
- Testing your model’s “goodness of fit”
- Opening the “black box” of the Markov Chain Monte Carlo algorithm to see how and why it works
- Leveraging the power of the “Law of Large Numbers”
- Mastering key concepts, such as clustering, convergence, autocorrelation, and thinning
- Using loss functions to measure an estimate’s weaknesses based on your goals and desired outcomes
- Selecting appropriate priors and understanding how their influence changes with dataset size
- Overcoming the “exploration versus exploitation” dilemma: deciding when “pretty good” is good enough
- Using Bayesian inference to improve A/B testing
- Solving data science problems when only small amounts of data are available

Cameron Davidson-Pilon has worked in many areas of applied mathematics, from the evolutionary dynamics of genes and diseases to stochastic modeling of financial prices. His contributions to the open source community include lifelines, an implementation of survival analysis in Python. Educated at the University of Waterloo and at the Independent University of Moscow, he currently works with the online commerce leader Shopify.

Get a comprehensive, in-depth introduction to the core Python language with this hands-on book. Based on author Mark Lutz’s popular training course, this updated fifth edition will help you quickly write efficient, high-quality code with Python. It’s an ideal way to begin, whether you’re new to programming or a professional developer versed in other

languages. Complete with quizzes, exercises, and helpful illustrations, this easy-to-follow, self-paced tutorial gets you started with both Python 2.7 and 3.3—the latest releases in the 3.X and 2.X lines—plus all other releases in common use today. You'll also learn some advanced language features that recently have become more common in Python code. Explore Python's major built-in object types such as numbers, lists, and dictionaries Create and process objects with Python statements, and learn Python's general syntax model Use functions to avoid code redundancy and package code for reuse Organize statements, functions, and other tools into larger components with modules Dive into classes: Python's object-oriented programming tool for structuring code Write large programs with Python's exception-handling model and development tools Learn advanced Python tools, including decorators, descriptors, metaclasses, and Unicode processing

sí que no habéis programado nunca... A medida que avanzamos en este libro, intentaremos enseñaros a programar. Vosotros tenéis que leer código y escribir código (así denominamos a los programas). Veremos un montón de código. Para entenderlo tendréis que copiar este código, probarlo y observar que ocurre. Jugad y haced cambios. Lo peor que os puede pasar es que no funcione. Python es un lenguaje de programación de alto nivel, interpretado, imperativo, orientado a objetos, funcional, de tipado dinámico y fuerte. Fue lanzada por Guido van Rossum en 1991. Actualmente tiene un modelo de desarrollo comunitario, abierto y gestionado por la organización sin fines lucrativos Python Software Foundation. A pesar de que varias partes del lenguaje tienen estándares y especificaciones formales, el lenguaje como un todo no está formalmente especificado. El estándar de hecho es la implementación CPython. Una de sus principales características es permitir la fácil lectura del código y exigir pocas líneas de código si comparamos el mismo programa con otros lenguajes de programación. Debido a sus características, se utiliza principalmente utilizada para el procesamiento de textos, datos científicos y creación de CGIs para páginas dinámicas de la web.

This book offers a highly accessible introduction to natural language processing, the field that supports a variety of language technologies, from predictive text and email filtering to automatic summarization and translation. With it, you'll learn how to write Python programs that work with large collections of unstructured text. You'll access richly annotated datasets using a comprehensive range of linguistic data structures, and you'll understand the main algorithms for analyzing the content and structure of written communication. Packed with examples and exercises, Natural Language Processing with Python will help you: Extract information from unstructured text, either to guess the topic or identify "named entities" Analyze linguistic structure in text, including parsing and semantic analysis Access popular linguistic databases, including WordNet and treebanks Integrate techniques drawn from fields as diverse as linguistics and artificial intelligence This book will help you gain practical skills in natural language processing using the Python programming language and the Natural Language Toolkit (NLTK) open source library. If you're interested in developing web applications, analyzing multilingual news sources, or documenting endangered languages -- or if you're simply curious to have a programmer's perspective on how human language works -- you'll find Natural Language Processing with Python both fascinating and immensely useful.

El aprendizaje automático está invadiendo el mundo del software. Si quieres entender y trabajar la vanguardia del aprendizaje automático, las redes neuronales y el aprendizaje profundo, esta segunda edición del bestseller Python Machine Learning, es tu libro. Modernizado y ampliado para incluir las tecnologías de código abierto más recientes, como scikit-learn, Keras y TensorFlow, este manual proporciona el conocimiento práctico y las técnicas necesarias para crear eficaces aplicaciones de aprendizaje automático y aprendizaje profundo en Python. El conocimiento y la experiencia únicos de Sebastian Raschka y Vahid Mirjalili presentan los algoritmos de aprendizaje automático y aprendizaje profundo, antes de continuar con temas avanzados en análisis de datos. Combinan los principios teóricos del aprendizaje automático con un enfoque práctico de codificación para una comprensión completa de la teoría del aprendizaje automático y la implementación con Python. Aprenderás a:

- Explorar y entender los frameworks clave para la ciencia de datos, el aprendizaje automático y el aprendizaje profundo
- Formular nuevas preguntas sobre datos con modelos de aprendizaje automático y redes neuronales
- Aprovechar el poder de las últimas librerías de código abierto de Python para aprendizaje automático
- Dominar la implementación de redes neuronales profundas con la librería de TensorFlow
- Incrustar modelos de aprendizaje automático en aplicaciones web accesibles
- Predecir resultados objetivos continuos con análisis de regresión
- Descubrir patrones ocultos y estructuras en datos con agrupamientos
- Analizar imágenes mediante técnicas de aprendizaje profundo
- Profundizar en datos de medios sociales y textuales con el análisis de sentimientos

En los últimos años, Python se ha convertido en un lenguaje muy adoptado por la industria de la seguridad informática, debido a su simpleza, practicidad, además de ser un lenguaje tanto interpretado como de scripting. Su integración con multitud de librerías de terceros hace pensar en Python como un lenguaje con múltiples posibilidades tanto desde el punto de vista ofensivo como defensivo de la seguridad y ha sido utilizado para un gran número de proyectos incluyendo programación Web, herramientas de seguridad, scripting y automatización de tareas. El objetivo del libro es capacitar a aquellos interesados en la seguridad, a aprender a utilizar Python como lenguaje de programación, no solo para poder construir aplicaciones, sino también para automatizar y especificar muchas de las tareas que se realizan durante un proceso de auditoría de seguridad. Repasaremos desde los conceptos básicos de programación hasta construir nuestra propia herramienta de análisis y extracción de información. Con el objetivo de extraer información de servidores y servicios que están ejecutando, información como nombres de dominio y banners, conoceremos los módulos que ofrece python para extraer información que los servidores exponen de forma pública y veremos los módulos que permiten extraer metadatos de documentos e imágenes, así como extraer información de geolocalización a partir de direcciones IP y nombres de dominio. También analizaremos conceptos más avanzados, como implementar nuestro propio escáner de puertos con comandos nmap y scapy, además de cómo conectarnos desde python con servidores FTP, SSH, SNMP, Metasploit y escáneres de vulnerabilidades como nexpose.

Get complete instructions for manipulating, processing, cleaning, and crunching datasets in Python. Updated for Python 3.6, the second edition of this hands-on guide is packed with practical case studies that show you how to solve a broad set of data

analysis problems effectively. You'll learn the latest versions of pandas, NumPy, IPython, and Jupyter in the process. Written by Wes McKinney, the creator of the Python pandas project, this book is a practical, modern introduction to data science tools in Python. It's ideal for analysts new to Python and for Python programmers new to data science and scientific computing. Data files and related material are available on GitHub. Use the IPython shell and Jupyter notebook for exploratory computing Learn basic and advanced features in NumPy (Numerical Python) Get started with data analysis tools in the pandas library Use flexible tools to load, clean, transform, merge, and reshape data Create informative visualizations with matplotlib Apply the pandas groupby facility to slice, dice, and summarize datasets Analyze and manipulate regular and irregular time series data Learn how to solve real-world data analysis problems with thorough, detailed examples

The programming language Python was conceived in the late 1980s, [1] and its implementation was started in December 1989[2] by Guido van Rossum at CWI in the Netherlands as a successor to the ABC (programming language) capable of exception handling and interfacing with the Amoeba operating system.[3] Van Rossum is Python's principal author, and his continuing central role in deciding the direction of Python is reflected in the title given to him by the Python community, Benevolent Dictator for Life (BDFL).[4][5] Python was named for the BBC TV show Monty Python's Flying Circus.[6] Python 2.0 was released on October 16, 2000, with many major new features, including a cycle-detecting garbage collector (in addition to reference counting) for memory management and support for Unicode. However, the most important change was to the development process itself, with a shift to a more transparent and community-backed process.[7] Python 3.0, a major, backwards-incompatible release, was released on December 3, 2008[8] after a long period of testing. Many of its major features have also been backported to the backwards-compatible Python 2.6 and 2.7.[9] In February 1991, van Rossum published the code (labeled version 0.9.0) to alt.sources.[10] Already present at this stage in development were classes with inheritance, exception handling, functions, and the core datatypes of list, dict, str and so on. Also in this initial release was a module system borrowed from Modula-3; Van Rossum describes the module as "one of Python's major programming units." [1] Python's exception model also resembles Modula-3's, with the addition of an else clause.[3] In 1994 comp.lang.python, the primary discussion forum for Python, was formed, marking a milestone in the growth of Python's userbase.[1] Python reached version 1.0 in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce. Van Rossum stated that "Python acquired lambda, reduce(), filter() and map(), courtesy of a Lisp hacker who missed them and submitted working patches." [11] The last version released while Van Rossum was at CWI was Python 1.2. In 1995, Van Rossum continued his work on Python at the Corporation for National Research Initiatives (CNRI) in Reston, Virginia whence he released several versions. By version 1.4, Python had acquired several new features. Notable among these are the Modula-3 inspired keyword arguments (which are also similar to Common Lisp's keyword arguments) and built-in support for complex numbers. Also included is a basic form of data hiding by name mangling, though this is easily bypassed.[12] During Van Rossum's stay at CNRI, he launched the Computer Programming for Everybody (CP4E) initiative, intending to make programming more accessible to more people, with

a basic "literacy" in programming languages, similar to the basic English literacy and mathematics skills required by most employers. Python served a central role in this: because of its focus on clean syntax, it was already suitable, and CP4E's goals bore similarities to its predecessor, ABC. The project was funded by DARPA.[13] As of 2007, the CP4E project is inactive, and while Python attempts to be easily learnable and not too arcane in its syntax and semantics, reaching out to non-programmers is not an active concern.[14] Here are what people are saying about the book: This is the best beginner's tutorial I've ever seen! Thank you for your effort. -- Walt Michalik The best thing i found was "A Byte of Python," which is simply a brilliant book for a beginner. It's well written, the concepts are well explained with self evident examples. -- Joshua Robin Excellent gentle introduction to programming #Python for beginners -- Shan Rajasekaran Best newbie guide to python -- Nickson Kaigi start to love python with every single page read -- Herbert Feutl perfect beginners guide for python, will give u key to unlock magical world of python

Build real-world Artificial Intelligence applications with Python to intelligently interact with the world around you About This Book Step into the amazing world of intelligent apps using this comprehensive guide Enter the world of Artificial Intelligence, explore it, and create your own applications Work through simple yet insightful examples that will get you up and running with Artificial Intelligence in no time Who This Book Is For This book is for Python developers who want to build real-world Artificial Intelligence applications. This book is friendly to Python beginners, but being familiar with Python would be useful to play around with the code. It will also be useful for experienced Python programmers who are looking to use Artificial Intelligence techniques in their existing technology stacks. What You Will Learn Realize different classification and regression techniques Understand the concept of clustering and how to use it to automatically segment data See how to build an intelligent recommender system Understand logic programming and how to use it Build automatic speech recognition systems Understand the basics of heuristic search and genetic programming Develop games using Artificial Intelligence Learn how reinforcement learning works Discover how to build intelligent applications centered on images, text, and time series data See how to use deep learning algorithms and build applications based on it In Detail Artificial Intelligence is becoming increasingly relevant in the modern world where everything is driven by technology and data. It is used extensively across many fields such as search engines, image recognition, robotics, finance, and so on. We will explore various real-world scenarios in this book and you'll learn about various algorithms that can be used to build Artificial Intelligence applications. During the course of this book, you will find out how to make informed decisions about what algorithms to use in a given context. Starting from the basics of Artificial Intelligence, you will learn how to develop various building blocks using different data mining techniques. You will see how to implement different algorithms to get the best possible results, and will understand how to apply them to real-world scenarios. If you want to add an intelligence layer to any application that's based on images, text, stock market, or some other form of data, this exciting book on Artificial Intelligence will definitely be your guide! Style and approach This highly practical book will show you how to implement Artificial Intelligence. The book provides multiple examples enabling you to create smart applications to meet the needs of your organization. In every chapter, we

explain an algorithm, implement it, and then build a smart application.

Este trabajo tiene como objetivo principal el análisis de un hardware comercial recientemente aparecido en el mercado, denominado Pyboard v.1.0, que se basa en un microcontrolador ARM de 32 bits y una versión reducida y eficiente de Python, denominado Micro Python. A partir de este TFG, los profesores de la ETSEIB podrán decidir si este hardware y software es adecuado para la enseñanza de microcontroladores y sus aplicaciones en distintas asignaturas impartidas en la escuela. El producto resultante de este trabajo también debe ser la síntesis en forma de tutorial, de toda una experiencia vivida como alumno de la ETSEIB y para un usuario del mismo tipo. El método que se ha utilizado en el análisis de este trabajo comienza con seguir los primeros pasos indicados en el tutorial de Micro Python. Una vez acabada esta pequeña introducción, se pasa a elaborar ejemplos propios, que hagan uso del hardware diverso de la placa, y comprobar su funcionamiento y eficiencia. Se han probado los temporizadores, interrupciones, los dos pulsadores, la UART, el bus CAN y el USB, así como el acelerómetro y los LEDs de la placa. Se analizan, por lo tanto, factores como la sencillez de uso. Se comprueba el funcionamiento de los comandos y se anotan los errores que se puedan encontrar. También se investiga en funcionalidades poco explicadas y definidas, a través de una documentación que, en muchos casos, es difícil de digerir, además de estar escrita totalmente en inglés. Los resultados obtenidos son satisfactorios hasta cierto punto. La sencillez característica de Python contrasta con cierta lentitud en algunos procesos, pudiendo llegar a ser tres veces más lento el intérprete ejecutando Python que ensamblador, en la ejecución de instrucciones sencillas. Se han encontrado errores en el firmware, algunos de ellos se han podido solucionar y otros no, como por ejemplo la recepción de mensajes a través del bus CAN. También se ha indagado sin mucho éxito en la programación del firmware, buscando una manera de añadirle funciones compiladas en C, para después ejecutarlas desde el intérprete de Python. Aunque no se logra ninguna utilidad determinada, sí que se abre la senda en ese sentido. Puede convertirse en un programador de Python expertos en menos de una semana * ATENCIN * Este libro fue traducido directamente desde el inglés y puede contener errores gramaticales. El lenguaje de programación Python ha sido visto como uno de los mejores a utilizar. Tiene una gran biblioteca a usar, es fácil de leer, y tiene todas las ventajas que vas a necesitar cuando primero aprender cómo se trabaja con la programación. Es todo listo para que su uso, y solo tienes que dar los primeros pasos! Esta guía va a ayudar a dar estos primeros pasos le muestra exactamente cómo empezar con el lenguaje de programación Python. Si usted ha trabajado en la programación antes o solo busca para empezar, esta guía tiene todos los temas y pasos que necesitas para obtener tu primer código escrito en ningún momento. Dentro de esta guía usted aprender: Por qué debo aprender sobre Python? Las partes básicas del código Python Trabajar con clases y objetos Trabajo sobre herencias Manejo de excepciones Trabajo con estructuras de control de la decisión La importancia de los lazos Qué archivo de entrada y salida significa en esta lengua Los diferentes operadores disponibles para hacer el código más fuerte Práctica escribir los códigos para que sea divertido juegos en su propio interior de Python. Trabajando en el Python lenguaje puede ser una de las experiencias más gratificantes. Hay mucha energía que puede estar detrás de estos programas pero es lo suficientemente simple como para

incluso un principiante poder utilizar en su propio. Cuando ests listo para empezar a trabajar con Python y algunos de sus propios cdigos, asegrese de que revisa esta gua y ver cun grande se puede hacer todo esto con Python!

Take your Python skills to the next level to develop scalable, real-world applications for local as well as cloud deployment Key Features All code examples have been tested with Python 3.7 and Python 3.8 and are expected to work with any future 3.x release Learn how to build modular and object-oriented applications in Python Discover how to use advanced Python techniques for the cloud and clusters Book Description Python is a multipurpose language that can be used for multiple use cases. Python for Geeks will teach you how to advance in your career with the help of expert tips and tricks. You'll start by exploring the different ways of using Python optimally, both from the design and implementation point of view. Next, you'll understand the life cycle of a large-scale Python project. As you advance, you'll focus on different ways of creating an elegant design by modularizing a Python project and learn best practices and design patterns for using Python. You'll also discover how to scale out Python beyond a single thread and how to implement multiprocessing and multithreading in Python. In addition to this, you'll understand how you can not only use Python to deploy on a single machine but also use clusters in private as well as in public cloud computing environments. You'll then explore data processing techniques, focus on reusable, scalable data pipelines, and learn how to use these advanced techniques for network automation, serverless functions, and machine learning. Finally, you'll focus on strategizing web development design using the techniques and best practices covered in the book. By the end of this Python book, you'll be able to do some serious Python programming for large-scale complex projects. What you will learn Understand how to design and manage complex Python projects Strategize test-driven development (TDD) in Python Explore multithreading and multiprocessing in Python Use Python for data processing with Apache Spark and Google Cloud Platform (GCP) Deploy serverless programs on public clouds such as GCP Use Python to build web applications and application programming interfaces Apply Python for network automation and serverless functions Get to grips with Python for data analysis and machine learning Who this book is for This book is for intermediate-level Python developers in any field who are looking to build their skills to develop and manage large-scale complex projects. Developers who want to create reusable modules and Python libraries and cloud developers building applications for cloud deployment will also find this book useful. Prior experience with Python will help you get the most out of this book.

A inicios de los años 90, el holandés Guido van Rossum desarrolló el lenguaje de programación Python que rápidamente se convirtió en una herramienta popular para escribir scripts de grabación de audio y para el desarrollo web. A pesar de que en la actualidad hay más alternativas que nunca, el popular lenguaje apenas ha perdido adeptos en sus tres décadas de existencia. Da igual si hablamos de aplicaciones web, sistemas embebidos (p. ej. ordenadores pequeños en electrodomésticos) o software empresarial, los programadores de Python actúan en los sectores más diversos y están muy solicitados. Una de las razones para el éxito tan prolongado de Python es su sencillez a la hora de iniciarse desde cero, ya que es mucho más fácil aprender y aplicar Python que

muchas de sus alternativas. Este tutorial de Python explica por qué es tan sencillo y cuál es la mejor forma de iniciarse en este lenguaje de programación. For many researchers, Python is a first-class tool mainly because of its libraries for storing, manipulating, and gaining insight from data. Several resources exist for individual pieces of this data science stack, but only with the Python Data Science Handbook do you get them all—IPython, NumPy, Pandas, Matplotlib, Scikit-Learn, and other related tools. Working scientists and data crunchers familiar with reading and writing Python code will find this comprehensive desk reference ideal for tackling day-to-day issues: manipulating, transforming, and cleaning data; visualizing different types of data; and using data to build statistical or machine learning models. Quite simply, this is the must-have reference for scientific computing in Python. With this handbook, you'll learn how to use: IPython and Jupyter: provide computational environments for data scientists using Python NumPy: includes the ndarray for efficient storage and manipulation of dense data arrays in Python Pandas: features the DataFrame for efficient storage and manipulation of labeled/columnar data in Python Matplotlib: includes capabilities for a flexible range of data visualizations in Python Scikit-Learn: for efficient and clean Python implementations of the most important and established machine learning algorithms

Desde su aparición en la década de los 60, los algoritmos genéticos han ido ganando popularidad, gracias al frenético crecimiento de la capacidad computacional en los últimos años. Finalmente se han abierto camino en el ámbito de la ingeniería como una de las herramientas más prometedoras para resolver problemas de gran complejidad, inabordables desde los enfoques clásicos de la ingeniería. Los algoritmos genéticos son estrategias de resolución de problemas de optimización basados en la teoría de la selección natural de Darwin, mediante la cual aquellos individuos más aptos para sobrevivir tienen una mayor probabilidad de crear descendencia y transmitir su información genética. Partiendo de esta base, son muchas las propuestas que se han desarrollado para abordar una gran cantidad de problemas de diferentes áreas de la ingeniería. En este libro le proponemos adentrarte en el mundo de los algoritmos genéticos utilizando Python, uno de los lenguajes de programación más populares en la actualidad y con más crecimiento durante los últimos años. Los contenidos del libro se han diseñado para que sean sencillos, concisos y fáciles de implementar, con ejemplos directos de aplicación para que pueda practicar desde la primera página. Con este libro aprenderá a: - Entender la naturaleza y el funcionamiento de los algoritmos genéticos, comprendiendo las diferentes operaciones y procesos que lo componen. - Conocer las diferentes implementaciones de los algoritmos genéticos de mayor relevancia, así como identificar las ventajas e inconvenientes de cada uno para determinar su potencial para resolver un determinado problema. - Conocer a fondo y utilizar los diferentes operadores (selección, mutación y cruce) que la librería deap pone a su disposición. - Desarrollar un algoritmo genético desde cero en Python y

utilizarlo para resolver sus propios problemas de ingeniería. - Conocer y estudiar aplicaciones de relevancia de algoritmos genéticos en el ámbito de la ingeniería, tales como la gestión del despacho económico, el diseño de plantas hidroeléctricas o la disposición de sensores inalámbricos.

The Hitchhiker's Guide to Python takes the journeyman Pythonista to true expertise. More than any other language, Python was created with the philosophy of simplicity and parsimony. Now 25 years old, Python has become the primary or secondary language (after SQL) for many business users. With popularity comes diversity—and possibly dilution. This guide, collaboratively written by over a hundred members of the Python community, describes best practices currently used by package and application developers. Unlike other books for this audience, The Hitchhiker's Guide is light on reusable code and heavier on design philosophy, directing the reader to excellent sources that already exist.

An introduction to a broad range of topics in deep learning, covering mathematical and conceptual background, deep learning techniques used in industry, and research perspectives. “Written by three experts in the field, Deep Learning is the only comprehensive book on the subject.” —Elon Musk, cochair of OpenAI; cofounder and CEO of Tesla and SpaceX Deep learning is a form of machine learning that enables computers to learn from experience and understand the world in terms of a hierarchy of concepts. Because the computer gathers knowledge from experience, there is no need for a human computer operator to formally specify all the knowledge that the computer needs. The hierarchy of concepts allows the computer to learn complicated concepts by building them out of simpler ones; a graph of these hierarchies would be many layers deep. This book introduces a broad range of topics in deep learning. The text offers mathematical and conceptual background, covering relevant concepts in linear algebra, probability theory and information theory, numerical computation, and machine learning. It describes deep learning techniques used by practitioners in industry, including deep feedforward networks, regularization, optimization algorithms, convolutional networks, sequence modeling, and practical methodology; and it surveys such applications as natural language processing, speech recognition, computer vision, online recommendation systems, bioinformatics, and videogames. Finally, the book offers research perspectives, covering such theoretical topics as linear factor models, autoencoders, representation learning, structured probabilistic models, Monte Carlo methods, the partition function, approximate inference, and deep generative models. Deep Learning can be used by undergraduate or graduate students planning careers in either industry or research, and by software engineers who want to begin using deep learning in their products or platforms. A website offers supplementary material for both readers and instructors.

El presente documento se encarga de mostrar el proceso de montaje y programación de drones de código libre utilizando Python como interfaz. Esta

clase de drones ha ganado popularidad en la actualidad debido a su reducido costo y fácil personalización; es posible cambiar cualquier aspecto del controlador de vuelo, así como realizar el montaje físico que sea más adecuado para la situación que se encuentre. Como proyecto base se utilizará Betaflight (un firmware para controladores de vuelo), debido a su avanzado estado de desarrollo y su énfasis en control de bajo nivel. Para el proceso de montaje, se muestra un tutorial paso a paso que incluye las instrucciones de montaje electrónico, así como las modificaciones en Betaflight y el proceso de compilación. A causa de las imperfecciones que pueden llegar a tener los drones de bajo costo (como el mostrado en el tutorial), se incluyeron también las modificaciones necesarias para mejorar la calidad de vuelo y respuesta del drone. Finalmente, se construyó una suite de control manual y autónomo utilizando Django y MSP, la cual puede ser utilizada como base para proyectos futuros en automatización y modelos de enjambre de drones.

[Copyright: fe5726eece4648d5dce52b2ed9cf8d3e](#)