UCAB UNIVERSIDAD CATÓLICA ANDRÉS BELLO

INTRODUCCIÓN A PYTHON Y R PARA ANÁLISIS DE DATOS

Tabla de contenido

Objetivo

- Conocer dos herramientas fundamentales para el análisis de datos, como son Python y R, el entorno de desarrollo para R y para Python y las librerías para ciencia de datos.

Introducción

El rol de un analista o de un científico de datos en los proyectos de *big data* está orientado principalmente a la exploración y al análisis de datos.

En la última década, Python, como lenguaje de programación, ha despertado mucho interés en su aplicación para el análisis de datos, siendo comparado con herramientas dedicadas como SAS & R. La principal ventaja es que el desarrollo y uso de módulos específicos para tareas asociadas al análisis de datos, llamados "bibliotecas" que están accesibles en repositorios públicos, aumentan los casos de uso donde se puede aplicar este lenguaje.

Otra de las alternativas muy utilizadas en *big data*, especialmente por los matemáticos y expertos en estadística, ha sido R, un lenguaje de programación que también presenta múltiples ventajas. Uno de sus puntos fuertes es la visualización de datos; muchas empresas y universidades utilizan este lenguaje para sus análisis estadísticos, por su robustez, su gran modularidad y posibilidades que se adaptan a todo tipo de necesidades en el manejo de datos complejos.

Python y R se consideran lenguajes de programación esenciales para la ciencia de datos. Lo ideal sería conocer ambos para tener una base de programación completa.

Python, bibliotecas y anaconda

Python es un lenguaje de programación interpretado (que no requiere compilación) muy flexible, multipropósito, multiplataforma y de *software* gratuito, que cada día que pasa va tomando mucho más crédito dentro de la ciencia de la analítica de datos (Hostinger, 2022).

Según los datos de Stack Overflow, Python es el lenguaje de programación de más rápido crecimiento en todo el mundo. Es muy accesible para los principiantes y ofrece el tipo de versatilidad que los desarrolladores web y científicos de datos necesitan (EDX, 2021). Varios índices de lenguajes de programación confirman este crecimiento como se puede ver en la siguiente imagen:

Worldwide, Mar 2022 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	28.27 %	-2.0 %
2		Java	18.03 %	+0.8 %
3		JavaScript	8.86 %	+0.4 %
4		C#	7.51 %	+0.6 %
5		C/C++	7.32 %	+0.6 %
6		PHP	5.71 %	-0.4 %

Fuente: Tabla de lenguajes de programación más utilizados por PYPL. Extraída de Geekflare (2022)

La razón principal es su sintaxis sencilla y elegante, junto con una gran colección de bibliotecas de terceros.

- Bibliotecas:

Una vez que se manejen las bases de este lenguaje de programación, es necesario utilizar librerías o bibliotecas para ejecutar tareas específicas en *big data*. No es suficiente con aprender Python para ponerlo en práctica en el análisis de datos; lo ideal es conocer y elegir correctamente los recursos a utilizar, para así orientar el aprendizaje hacia estos, ya que es un lenguaje de propósito general donde se pueden programar un sinfín de casos de *software* fuera de este ámbito (Código Fuente, 2018).

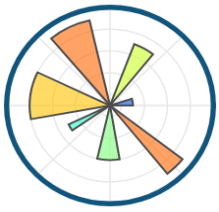
En este sentido, las bibliotecas de Python más utilizas para el análisis de datos son:

NumPy: significa *Numerical Python* y su objetivo es dar facilidades para el análisis numérico. La característica más poderosa de NumPy es la matriz n-dimensional. Esta biblioteca también contiene funciones básicas de álgebra lineal, transformadas de Fourier, capacidades avanzadas de números aleatorios y herramientas para la integración con otros lenguajes de bajo nivel como Fortran, C y C ++.[1]

https://numpy.org/

SciPy: significa *Scientific Python*. SciPy se basa en NumPy y está orientada a la computación científica. Es una de las bibliotecas más útiles para una variedad de módulos de ciencia e ingeniería de alto nivel como la transformada de Fourier discreta, el álgebra lineal, la optimización y las matrices dispersas.

https://scipy.org/



Matplotlib: esta biblioteca permite crear visualizaciones estáticas, animadas e interactivas en Python e incrustar una gran variedad de gráficos, desde histogramas hasta trazados de líneas para gráficos de calor. En su página hay un enlace a una prueba con binder:

https://matplotlib.org/

Pandas: su nombre viene de *Pyton Data Analisys* y está basada en NumPy para operaciones de datos estructurados y manipulaciones. Esta biblioteca es ampliamente utilizada para la recopilación de datos y la preparación. Permite leer y escribir fácilmente archivos en formato CSV, Excel y bases de datos SQL; ofrece métodos para reordenar, dividir y combinar conjuntos de datos y permite trabajar con series temporales, todo de manera eficiente. Esta ha sido fundamental para impulsar el uso de Python en la comunidad científica de datos.

https://pandas.pydata.org/

Scikit Learn: para el aprendizaje automático y el análisis predictivo. Construida sobre NumPy, SciPy y matplotlib, esta biblioteca contiene una gran cantidad de herramientas eficientes para el aprendizaje automático y el modelado estadístico, que incluyen clasificación, regresión, agrupación y reducción de dimensionalidad.

https://scikit-learn.org/stable/

Statsmodels: para el modelado estadístico. Es un módulo de Python que permite a los usuarios explorar datos, estimar modelos estadísticos y realizar pruebas estadísticas. Se encuentra disponible una lista extensa de estadísticas descriptivas, pruebas estadísticas, funciones de trazado y estadísticas de resultados para diferentes tipos de datos y cada estimador.

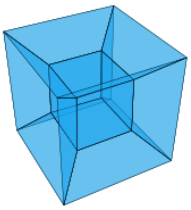
https://www.statsmodels.org/stable/index.html

Seaborn: para visualización de datos estadísticos. Seaborn es una biblioteca de visualización de datos de Python basada en matplotlib. Proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos e informativos; tiene como objetivo hacer de la visualización una parte central de la exploración y comprensión de los datos.

https://seaborn.pydata.org/

Bokeh: para crear paneles interactivos, paneles de control y aplicaciones de datos en navegadores web modernos. Permite al usuario generar gráficos elegantes y concisos al estilo de D3.js. Además, tiene la capacidad de interactividad de alto rendimiento en datasets muy grandes o de transmisión.

https://bokeh.org/



Blaze: amplía la capacidad de Numpy y Pandas para la distribución y transmisión de datos. Puede usarse para acceder a datos de una multitud de fuentes, incluyendo Bcolz, MongoDB, SQLAlchemy, Apache Spark, PyTables, etc. Junto con Bokeh, Blaze puede actuar como una herramienta muy poderosa para crear visualizaciones y cuadros de mando efectivos en grandes cantidades de datos.

https://pypi.org/project/blaze/

TensorFlow: una biblioteca para computación numérica (usando grafos de flujo de datos), creada por Google Brain y que permite, principalmente, procesar de forma sencilla matrices o tensores (de 2 dimensiones). Su objetivo principal es facilitar la programación de redes neuronales. Además, con Tensorflow Lite podemos entrenar y optimizar modelos para ser ejecutados en smartphones o dispositivos IoT (*Internet of Things*) y Tensorflow.js permite la ejecución de los modelos en Web Browsers.

https://www.tensorflow.org/

Keras: es una biblioteca para trabajar con redes neuronales, pero a más alto nivel. Keras requiere un motor computacional que se ejecute debajo, como Tensorflow, Caffe, Theano o CNTK (y otros). Creada por François Chollet, un ingeniero de Google, es muy fácil de usar y permite una implementación muy rápida. Enfatiza el minimalismo por el hecho que se puede construir una red neuronal con muy pocas líneas de código.

https://keras.io/

Scrapy para rastreo web: es un marco muy útil para obtener patrones específicos de datos. Tiene la capacidad de comenzar en la URL de inicio de un sitio web y luego profundizar en las páginas web del sitio para recopilar información.

https://scrapy.org/

SymPy para computación simbólica: tiene una amplia gama de capacidades, desde aritmética simbólica básica hasta cálculo, álgebra, matemáticas

discretas y física cuántica. Otra característica útil es la capacidad de formatear el resultado de los cálculos como código LaTeX.

https://www.sympy.org/

- Anaconda y cuadernos de Jupyter:

Anaconda es una distribución libre y abierta que viene lista para programas con los lenguajes Python y R, muy utilizada en ciencia de datos y en el aprendizaje automático (*machine learning*). Ya viene con bibliotecas incorporadas, como las expuestas anteriormente, y que podemos observar en la imagen, permitiendo el procesamiento de grandes volúmenes de información, análisis predictivos y cómputos científicos (Anaconda, 2022).

Las diferentes versiones de los paquetes se administran mediante el sistema de gestión de paquetes Conda, el cual lo hace bastante sencillo de instalar, correr, y actualizar las bibliotecas de análisis y visualización de datos.

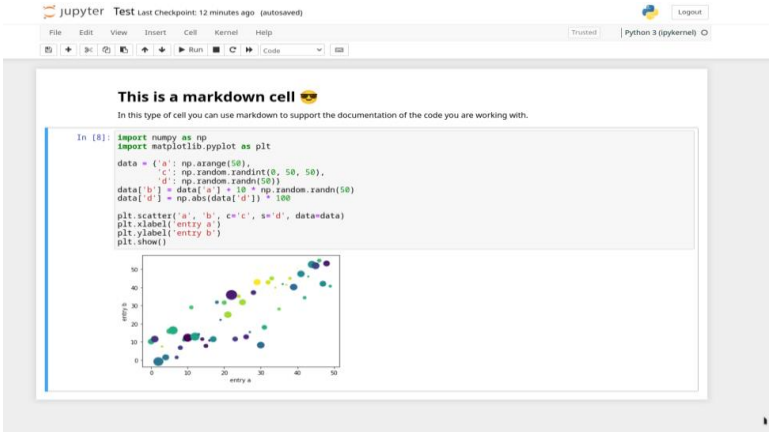
Fuente: Repositorio de bibliotecas o paquetes de Anaconda. Extraída de Anaconda (2022)

1. Cuadernos de Jupyter:

Uno de los componentes integrados a Anaconda son los cuadernos de Jupyter o Jupyter Notebook (anteriormente llamado IPython Notebook), aunque lo puedes instalar aparte. Esta es una aplicación web de código abierto que permite crear y compartir documentos que contienen código en vivo, ecuaciones, visualizaciones y texto narrativo. Además, es una aplicación muy utilizada en el campo de la ciencia de datos (*Data Science*) para crear y compartir documentos que incluyen: limpieza y transformación de datos, simulación numérica, modelado

estadístico, visualización de datos, aprendizaje automático y mucho más. En la siguiente imagen tenemos un ejemplo de un cuaderno de Jupyter.

Permite editar y ejecutar documentos con instrucciones a través de cualquier navegador web o en un escritorio local, por lo que no requiere acceso a internet; también puede instalarse en un servidor remoto y acceder a través de internet.



Fuente: Cuaderno Jupyter. Extraída de Geekflare (s.f.)

Jupyter Notebook es ideal para los siguientes casos de uso:

- Aprendizaje de Python
- Procesamiento / transformación de datos
- Simulación numérica
- Modelado estadístico
- Aprendizaje automático.

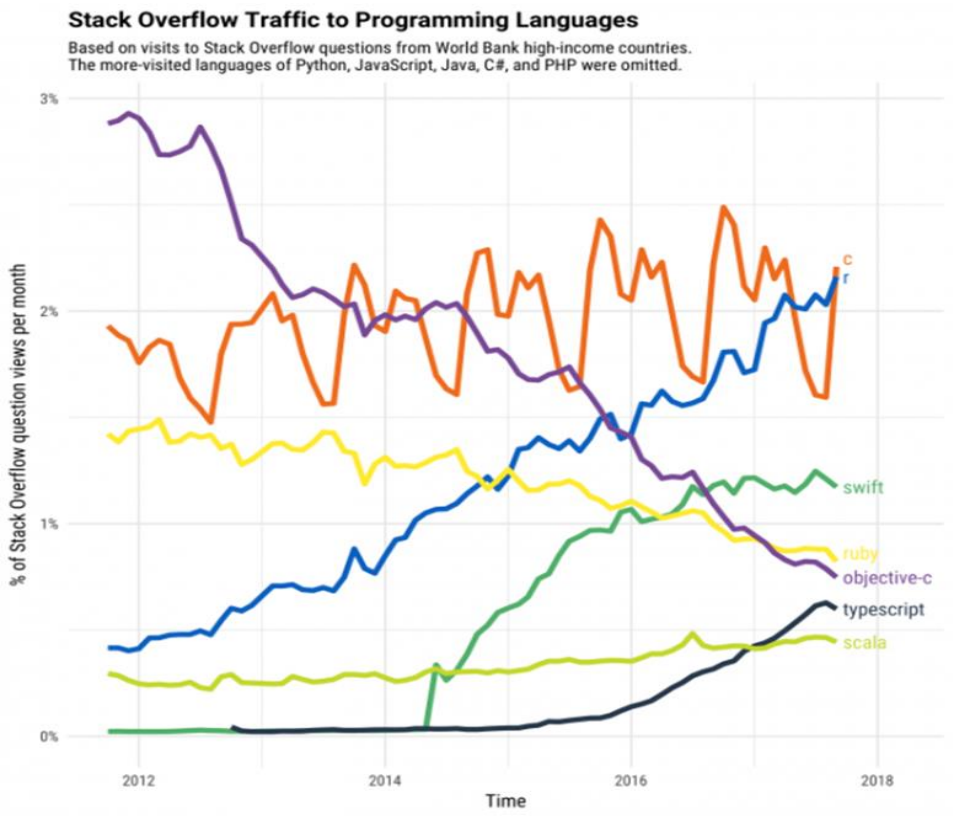
Para iniciar rápidamente proyectos de ciencia de datos se recomienda usar Anaconda en conjunto con los cuadernos de Jupyter, ya que facilita el aprendizaje y el trabajo en conjunto en proyectos de analítica de datos.

R, extensiones y paquetes y RStudio

R es un lenguaje de programación con un enfoque al análisis estadístico y matemático. Fue creado por Ross Ihaka y Robert Gentleman, como una reimplementación de *software* libre del lenguaje S.

Está pensado para cálculos estadísticos y matemáticos y para la creación de gráficos. Se trata de uno de los lenguajes de programación más utilizados en investigación científica, siendo además muy popular en los campos de aprendizaje automático (*machine learning*), minería de datos, investigación biomédica, bioinformática y matemáticas financieras. A esto contribuye la posibilidad de cargar diferentes bibliotecas o paquetes.

Según un análisis de Stack Overflow, la popularidad de R ha aumentado en el transcurso de los últimos años.



Fuente: La creciente popularidad de R. Extraída de Geekflare (s.f.)

R es parte del sistema GNU y se distribuye bajo la licencia GNU GPL. Está disponible para los sistemas operativos Windows, Macintosh, Unix y GNU/Linux. Para descargar R, elija un espejo CRAN en la página del proyecto (R, 2022).

Este programa proporciona una amplia variedad de técnicas estadísticas como modelado lineal y no lineal, pruebas estadísticas clásicas, análisis de series temporales, clasificación, agrupamiento, entre otras, y técnicas gráficas; además es altamente extensible (R, 2022). Para la documentación R tiene su propio formato similar a LaTeX, muy usado por los matemáticos y que se utiliza para proporcionar documentación completa, tanto en línea como en varios formatos (por ejemplo, papel).

1. Extensiones y paquetes:

R forma parte de un proyecto colaborativo y abierto. Los paquetes en este lenguaje son colecciones de funciones y conjuntos de datos desarrollados por la comunidad. Estos incrementan la potencialidad de este lenguaje, mejorando las funcionalidades base o añadiendo nuevas. Por ejemplo, en el campo de ciencia de los datos cuando trabajamos con data. frames, probablemente usaremos los paquetes dplyr o data.table, dos de los más populares en la comunidad.

Actualmente, el repositorio oficial CRAN recoge cerca de 10.000 paquetes publicados y, además, existen muchos más publicados en internet (Sánchez, 2021).

Debido a la gran cantidad de nuevos paquetes que se generan constantemente, estos se han organizado en vistas (o temas) que permiten agruparlos según su naturaleza y función. Por ejemplo, hay grupos de paquetes relacionados con estadística bayesiana, econometría, series temporales, etc. Para facilitar el desarrollo de nuevos paquetes, se ha puesto a servicio de la comunidad una forja de desarrollo que facilita las tareas relativas a dicho proceso.

2. Rstudio:

RStudio, ahora llamado Posit, es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para el lenguaje de programación R, dedicado a la computación estadística y gráficos. Incluye una consola, editor de sintaxis que apoya la ejecución de código, así como herramientas para el trazado, la depuración y la gestión del espacio de trabajo.

Este IDE está disponible para Windows, Mac y Linux o para navegadores conectados a RStudio Server o RStudio Server Pro. Con el cambio de nombre se está agregando soporte también a Python en este entorno (Posit, 2022).

Python o R

Aunque existen muchos otros lenguajes con los que puedes trabajar para realizar análisis de datos, se han descrito dos de los más importantes, pero puede surgir la duda de cuál de los dos usar. A continuación, algunos consejos para elegir entre Python y R:

La gente que elige Python:	Las personas que eligen R:
• Trabajan en la ciencia de datos orientada al negocio • Crean algoritmos de aprendizaje automático • Trabajan en una variedad de industrias • Requieren un lenguaje flexible • Planean crear proyectos que escalen	• Trabajan en áreas de análisis o de ciencia de datos con estadística • Trabajan en el mundo académico • Necesitan la sintaxis específica del lenguaje de los procesos estadísticos • Realizan análisis estadísticos o trabajos analíticos especializados • Necesitan una salida dinámica para comunicar los resultados
Es mejor elegir Python si:	**Es mejor elegir R si:**
• No tiene experiencia en programación • El objetivo principal es la producción o el despliegue • Quieres construir nuevos modelos desde cero • El código de los proyectos debe ser legible	• Tiene previsto trabajar en la investigación o en el mundo académico • El trabajo tiene un fuerte componente estadístico y de análisis • Desea hacer uso de amplias bibliotecas para soluciones existentes • Desea hacer uso de amplias bibliotecas para soluciones existentes • Las características específicas de la sintaxis son importantes • La comunicación de resultados complejos es clave

Fuente: Phyton o R. Extraída de EDX (2021)

Cierre

En este contenido te hemos presentado dos importantes lenguajes para trabajar con los datos en *big data*, bien sea para los análisis como para el aprendizaje, y hemos hecho una comparación de cuándo usar cada uno. Ahora es tu turno de aplicar estos conocimientos en tus actividades prácticas.

Referencias

Anaconda (2022). *Distribucion Anaconda*. https://www.anaconda.com/products/distribution

Código Fuente (2018). *Bibliotecas para hacer análisis de datos en Python*. https://www.codigofuente.org/bibliotecas-analisis-datos-python/

EDX (2021). *R vs. Python para la ciencia de datos*. https://blog.edx.org/es/r-vs-python-para-la-ciencia-de-datos-explicacion-y-consejos-de-aprendizaje

Hostinger (2022). *Tutoriales*. https://www.hostinger.es/tutoriales/que-es-python

Posit (2022). *RStudio is now Posit*. https://posit.co/

R (2022). *¿Qué es R?* https://www.r-project.org/about.html

R (2022). *The R Project for Statistical Computing*. https://www.r-project.org/

Sánchez, R. (2021). *Ciencia de datos con R*. https://rsanchezs.gitbooks.io/ciencia-de-datos-con-r/content/paquetes/paquetes.html

Referencias de las imágenes

Anaconda (2022). Repositorio de bibliotecas o paquetes de Anaconda [Imagen]. Disponible en: *Distribucion Anaconda*. https://www.anaconda.com/products/distribution

EDX (2021). Phyton o R [Imagen]. Disponible en: https://blog.edx.org/es/r-vs-python-para-la-ciencia-de-datos-explicacion-y-consejos-de-aprendizaje

Geekflare (s.f.). Cuaderno de Jupyter [Imagen]. Disponible en: https://geekflare.com/wp-content/uploads/2022/04/jupyter-968x628.png

Geekflare (s.f.). La creciente popularidad de R [Imagen]. Disponible en: https://geekflare.com/wp-content/uploads/2022/04/R-popularity-628x628.png

Geekflare (2022). Tabla de lenguajes de programación más utilizados por PYPL [Imagen]. Disponible en: https://geekflare.com/es/data-science-programming-languages/