Python en la Ciencia de Datos

Universidad
Carlos III de Madrid

Fabio Scielzo Ortiz

Marcos Álvarez Martín

Indice

Una advertencia

- No somos expertos en los temas que vamos a mencionar, solo meros aficionados. Por tanto las cosas que aquí expondremos no deberían tomarse como dogma de fe, ni mucho menos.
- Recomendamos que tras esta presentación investigueis por vuestra cuenta sobre estos temas, acudiendo a otras fuentes.

`Python` en la ciencia de datos

- `Python` es un lenguaje de programación de **proposito general** creado en 1991 por *Guido van Rossum*
- En los últimos años ha tenido mucho desarrollo orientado a la **ciencia de datos** $\Rightarrow$ Aparición y desarrollo de muchas librerias orientadas a la ciencia de datos.

`Python` en la ciencia de datos

- Algunas de las liberias de `Python` más usadas para ciencia de datos son:
  - Pandas
  - Numpy
  - Scipy
  - Matplotlib , Seaborn , Plotly
  - Sklearn , Statmodels

Python en la ciencia de datos

Pandas

- Libreria orientada al manejo de data-frames. Es la libreria básica y fundamental, aquella que todo científico de datos debe dominar.

Numpy

- Libreria orientada al manejo de vectores y matrices en Python. Fundamental a la hora de programar procedimientos y algoritmos relacionados con la ciencia de datos. Es además notablemente más eficiente que Pandas. En este artículo hago una muestra de ello.

Scipy

Es la libreria fundamental de Python para matemáticas computacionales. Provee módulos para optimización, integración, interpolación, problemas de autovalores, ecuaciones algebraicas, ecuaciones diferenciales, estadística y más.

Python en la ciencia de datos

Matplotlib

- Es la libreria más conocida para la visualización de datos en Python.

Seaborn

- Es otra libreria muy popular para visualización de datos. Se complementa bien con Matplotlib.

Plotly

- Es una de las librerias más usadas para la visualizacion interactiva de datos (visualización en movimiento).

Python en la ciencia de datos

Sklearn

- La libreria de referencia para la aplicación de modelos de machine learning en Python. Cubre una gran cantidad de algoritmos.
  Como curiosidad, uno de sus principales creadores es madrileño, Fabian Pedregosa.

Statmodels

- Libreria orientada a la implementación de modelos de tradición estadística. Cubre algunos modelos que no son cubiertos por Sklearn.

Python en la ciencia de datos

- Algunas librerias más especializadas en cosas concretas:
  - JobLib
  - SymPy
  - Skforcast, PyGam, PyClustering
  - Pytorch , NLTK, TensorFlow
  - BeautifulSoup , Selenium
  - Pyomo, Gurobi
  - Shiny , FastAPI
  - MLflow

JobLib

Es una de las librerias más populares para paralelizar código de forma sencilla.

SymPy

La libreria más relevante de Python para matemáticas simbólicas. Los que esteis cursando la asignatura de optimizacion estais usando matematicas simbolicas en Matlab con la libreria (toolbox) Symbolic Math. Esta es la alternativa mas potente en Python.

Python en la ciencia de datos

TensorFlow

Libreria de Python desarrollada por Google orientada al deep learning.

Pytorch

Libreria orientada principalmente a la aplicación de deep learning a imagenes y videos (vision artifical). Aunque tambien se aplica en problemas de audio y texto.

NLKT

Libreria orientada al PLN (Procesamiento de Lenguaje Natural).

Python en la ciencia de datos

BeautifulSopu , Selenium

Librerias orientadas al web scrapping.

Pyomo, Gurobi

Librerias orientadas a la investigación operativa.

Python en la ciencia de datos

Skforcast

Libreria para predicción de series temporales.

PyGam

Libreria que permite implementar modelos GAM (Modelos Aditivos Generalizados).

PyClustering

Libreria orientada a la aplicación de algoritmos de clustering.

Python en la ciencia de datos

Shiny

- Libreria, hasta hace poco exclusiva de R, que permite la creación de aplicaciones interactivas orientadas a la ciencia de datos. Desde hace unos meses también está en Python.

FastAPI

- Otra libreria popular para la creación de API's con Python.

MLflows

- Libreria que permite hacer ML Ops.
- ML Ops es un conjunto de prácticas que tiene como objetivo implementar y mantener modelos de aprendizaje automático en producción de manera confiable y eficiente

Python en la ciencia de datos

- Hay más de **137000** librerias en Python, asi que esta ha sido una minuscula muestra de librerias, pero muchas de ellas son cruciales en ciencia de datos.
- Os recomendamos investigar por vuestra cuenta, Python es un **mundo de posibilidades**, tanto en el ambito de la ciencia de datos como fuera de él.
- Google es nuestro amigo.

¿Por qué dar el salto de R a Python?

- `R` es un lenguaje de programación enfocado en el **análisis estadístico**.
- **¿Debería aprender más R o empezar a aprender Python?**
  - *Opción ideal* : empezar a aprender Python, sin dejar de aprender R
  - *Realidad* : el tiempo es finito, en muchas ocasiones es dificil lo anterior. En esos caso recomendamos priorizar Python sobre R.

A continuación argumentamos por qué.

¿Por qué dar el salto de R a Python?

- **Motivos por los que empezar a aprender Python**
  - Abre Linkedin, busca ofertas de empleo como cientifico de datos, estadistico, analista de datos, ingeniero de datos o cosas similares.
  - ¿En cuantas ofertas exigen conocimientos de R? ⇒ en pocas.
  - ¿ En cuantas ofertas exigen conocimientos de Python? ⇒ en muchas.
- Si sabes Python es más facil encontrar trabajo.
  - Esto ya es un buen motivo para aprender Python.

¿Por qué dar el salto de R a Python?

Fuente : ver las encuestas que hace la web Kaggle cada año en relación a estas cuestiones.

¿Por qué dar el salto de R a Python?

- **Más motivos para aprender Python**
  - A nuestro criterio R tiene una sola ventaja clara respecto de Python: en algunos campos de la estadistica R tiene un mayor desarrollo que Python, debido a su mayor uso en el ámbito academico.
    - Pero en los campos más demandados por las empresas Python está más desarrollado.
  - Python es un lenguaje de proposito general, no solo es util para hacer cosas de estadistica.

¿Por qué dar el salto de R a Python?

- **Más motivos para aprender Python**
  - Aunque no os lo creais muchas personas consideran que Python es un lenguaje **más sencillo** de aprender que R.

¿Por qué dar el salto de R a Python?

Motivo clave

- La ciencia de datos es bastante más que coger un data-set, entrenar un modelo y hacer predicciones con él para una muestra de test. Hay toda un parte denominada puesta en producción de modelos que es igual o más importante que la anterior para las empresas.
- Poner un modelo en producción significa que el modelo va a operar en la vida real. Va a recibir datos, va a predecir en tiempo real y estos resultados van a ser usados.

¿Por qué dar el salto de R a Python?

- Poner un modelo de machine learning en producción es un proceso complejo que involucra diferentes etapas y herramientas como Docker, Kubernetes, computación en la nube (Azure, Google Cloud, Amazon Web Service), y más.

¿Por qué dar el salto de R a Python?

- Python está mucho más desarrollado que R para la puesta en producción de modelos de machine learning, la libreria de MLflow es un buen ejemplo de ello.
- Este es el punto fuerte de Python desde el punto de vista empresarial.
    - Quiza el factor que mejor explica el por qué del auge de Python en detrimento de R en los últimos años, en el ambito de la ciencia de datos.

¿Por qué dar el salto de R a Python?

- Ahora quiza podemos entender mejor por qué, pese a que Python arrasa a R en el mundo empresarial, en la academia R sigue reinando, aunque es posible que cada vez lo haga con menor fuerza.
- En la academia pocas veces es relevante la parte de puesta en producción de un modelo, por lo que una de las mayores virtudes de Python sobre R se disipa.
- Quiza ahora también pueda entenderse algo mejor por qué no hemos visto **nada** de Python en este grado, hasta el día de hoy.

La transición de R a Python

Nunca es tarde para empezar.

Además, tenemos buenas noticias:

- El hecho de manejar con fluidez otro lenguaje de programación, como en nuestro caso es R, ayuda **muchisimo** a la hora de aprender un nuevo lenguaje, en este caso Python.

La transición de R a Python

- En mi caso particular escribí mi primera sentencia en código Python en junio de este año (2022). Y creo haber aprendido en apenas 5 meses al menos igual de Python que lo que aprendí de R en 3 años de carrera.

- ¿ Cual es el secreto ?
  - Luego lo comentaremos.

La transición de R a Python

¿ Por dónde empezar con Python ? ⇒ Parte I

- Existen varias formas de usar Python en nuestro ordenador, lo mas común es usando un IDE (Integrated Development Environment).
- Los IDE's más usados son:
  - Jupyter Notebook (a través de Anaconda).
  - Spider (parecido a R-studio)
  - R-studio (¿O debería decir Posit ?)
  - Visual Studio Code (el presente y futuro de los IDE's)

La transición de R a Python

¿ Por dónde empezar con Python ? ⇒ Parte I

- Visual Studio Code (el presente y futuro de los IDE's).
  - Soporta multitud de lenguajes como Python, C, C++ , Java, R, HTML, CSS, Julia, Markdown, Go, Docker. Esta presentación esta hecha usando markdown en VS code !
  - Muchas extensiones útiles.
  - Buena integración con GitHub y Docker.

La transición de R a Python

¿ Por dónde empezar con Python ? ⇒ Parte II

Tres librarias de Python que nos podrian ser utiles en un comienzo:

- dfply
- plotnine
- rpy2

Y también hay una libreria de R que podria ser de ayuda:

- reticulate

La transición de R a Python

dfply

- Libreria para manejo de data-frames inspirada en la sintaxis de dplyr, una de las librerias de R más usadas para el manejo de data-frames

plotnine

- Libreria para visualización de datos que utiliza la gramática de ggplot (una de las librerias de R más usadas para visualización de datos). Básicamente es la version de ggplot en Python.

La transición de R a Python

rpy2

- Esta libreria de Python permite usar R dentro de Python.

reticulate

- Esta libreria de R permite usar Python dentro de R.

La transición de R a Python

¿ Por dónde seguir ?

- Recomendamos que, tras una etapa inicial usando dfply y plotnine, pasemos a sus versiones canonicas en Python, es decir, a **Pandas** y **Seaborn** + **Matplotlib**.
- Despues recomendamos tener una primera aproximacion a **Numpy**, ver cosas básicas.
- Luego estaria bien empezar a soltarnos con **Sklearn** y **Statmodels**.

La transición de R a Python

¿ Cuál es el secreto para aprender Python ?

Nuestra respuesta personal a esta pregunta es:

- Haz proyectos de ciencia de datos que te motiven.
  - Si te interesan temas especificos, como los deportes, el cine o los videojuegos, usa herramientas que la ciencia de datos te da para analizar estos temas, a través de Python.
  - Si te interesa cierto algoritmo o procedimiento, aplicalo a un caso real a través de Python.

La transición de R a Python

- Puedes crear un blog con estos proyectos, o subirlos a Linkedin. Es una manera de hacer crecer tu marca personal mientras aprendes.
    - Para ello lo típico fuera de la ciencia de datos es usar webs como Wordpress o Wix.
    - Pero dentro de la ciencia de datos se estila mucho crear blogs (o páginas webs en general) a través de **github pages**.
    - Si os poneis con ello, recomendamos **github pages**.

La transición de R a Python

- Apoyate en libros, documentos y webs útiles.
  - Ejemplos de webs de interés :
    - **Cienciadedatos.net**
      - Creada por Joaquin Amat Rodrigo.
      - Articulos sobre estadística y machine learning aplicado tanto en Python como en R.
      - Recomendable como introducción práctica a la estadística y machine learning.

La transición de R a Python

- Apoyate en libros, documentos y webs útiles.
  - Ejemplos de webs de interés :
    - **Documentación oficial de Sklearn**
      - Tiene tanto explicaciones teoricas como aplicaciones prácticas.
      - Va al grano.

La transición de R a Python

- Algunos ejemplos de webs de interés :
  - **Blog de Ander Fernandez Jauregui**
    - Filosofia similar a Cienciadedatos.net
    - Prestá más atencion a la programación de algoritmos y a la puesta en producción de modelos. aprte de puesta en producción de modelos.
    - Recomendable para introducirse en ML Ops y mejorar tú programación.

La transición de R a Python

- Algunos ejemplos de webs de interés :
  - **Estadistica4all**
    - El blog de Fabio Scielzo Ortiz sobre ciencia de datos (un poco de autobombo).
    - Filosofia similar a Cienciadedatos.net y al blog de Ander Fernandez.
    - El proyecto está en construcción, tiene unos pocos meses.
    - Se espera que mejore notablemente en calidad de contenido en un futuro.

La transición de R a Python

- Ejemplos de libros de interés:
  - **An Introduction to Statistical Learning**
    - Fue liberado por sus autores en la red.
      - *G. James, D. Witten, T. Hastie, R. Tibshirani*
    - Introducción de *nivel intermedio* al machine learning.
    - Bueno para entender a nivel teorico los principales algoritmos de machine learning.
    - Ejemplos prácticos en R ⇒ Es un buen ejercicio replicarlos en Python.

La transición de R a Python

- Ejemplos de libros de interés:
  - **The Elementos of Statistical Learning**
    - Fue liberado por sus autores en la red:
      - *J.H. Friedman*, *R. Tibshirani* y *T. Hastie*
    - Introducción de *nivel intermedio-alto* al machine learning.
    - Bueno para entender a nivel teorico los principales algoritmos de machine learning. Pero algo más duro.

La transición de R a Python

- Ejemplos de libros de interés:
  - **Probabilistic machine learning: An introduction**
    - Fue liberado por sus autores en la red:
      - *Kevin Murphy*
    - Introducción de *nivel avanzado* al machine learning.
    - Bueno para entender a nivel teorico los principales algoritmos de machine learning. Pero bastante duro.
    - Aplicado en Python.

Gracias por la atención !