HITO GRUPAL

PROGRAMACIÓN



Trabajo realizado por: Mario Alajarín,Adriel Sadia y Jesús Cabrera.

**Fase 1.(Adriel Sadia y Jesús Cabrera)**

**1. Hablamos de fuentes de datos. De grandes volúmenes de datos. Por ejemplo, data Lake o similar.También es importante tratar la diferencia entre datos estructurados y no estructurados en relación al bigData.**

**(Jesús Cabrera)**

Un data lake es un sistema de almacenamiento de datos no estructurado que permite recolectar grandes volúmenes de datos de diferentes fuentes, y almacenarlos de manera no estructurada (sin necesidad de pre-procesamiento). Los data lakes son ideales para almacenar datos de gran volumen, variedad y velocidad, como datos de sensores, datos de redes sociales, datos de transacciones comerciales, entre otros.La diferencia entre datos estructurados y no estructurados es que los datos estructurados son aquellos que se encuentran en un formato estandarizado y pueden ser fácilmente procesados por una computadora, como tablas de bases de datos relacionales, hojas de cálculo, entre otros. Por otro lado, los datos no estructurados son aquellos que no se encuentran en un formato estandarizado y son más difíciles de procesar, como textos, imágenes, videos, audio, entre otros.



En el contexto del big data, los datos no estructurados son más difíciles de analizar y procesar que los datos estructurados, ya que requieren de herramientas y técnicas especializadas para poder ser convertidos en información útil. Sin embargo, los datos no estructurados también pueden ser valiosos para la toma de decisiones, ya que pueden proporcionar información no disponible en los datos estructurados.



**2. Entre las herramientas más interesantes a la hora de gestionar grandes volúmenes de datos nos encontramos con Hadoop y Spark. Habría que tratar sus características y finalidad.(Jesús Cabrera)**

Hadoop es un framework de software de código abierto que permite el procesamiento distribuido de grandes volúmenes de datos en clusters de computadoras. Utiliza el almacenamiento distribuido para guardar y procesar datos, y su sistema de ficheros HDFS (Hadoop Distributed File System) permite el almacenamiento de datos no estructurados y estructurados. Hadoop también incluye un motor de procesamiento de datos llamado MapReduce, que permite el procesamiento paralelo de datos en varios nodos del clúster.



Por otro lado, Apache Spark es un framework de procesamiento de datos de código abierto que proporciona una plataforma para el procesamiento en tiempo real, el análisis iterativo y la generación de informes de grandes volúmenes de datos. A diferencia de Hadoop, Spark utiliza la memoria en lugar de solo el disco para el almacenamiento y procesamiento de datos, lo que lo hace más rápido y eficiente. También incluye un motor de procesamiento de datos llamado RDD (Resilient Distributed Datasets), que permite el procesamiento de datos en varios nodos del clúster de manera similar a Hadoop.



En resumen, Hadoop es una plataforma de almacenamiento y procesamiento de datos distribuidos de código abierto, mientras que Spark es un framework de procesamiento de datos distribuido de código abierto, que se enfoca en proporcionar alta velocidad y eficiencia en el procesamiento de datos en memoria.



**3. Existen lenguajes de programación “recomendables” para gestionar datos. Entre ellos, están Python y Scala. Sería explicar brevemente por qué.(Adriel Sadia)**

Python es un lenguaje de programación de alto nivel, con una sintaxis clara y concisa, y una amplia gama de librerías y herramientas para trabajar con datos. Python es especialmente popular en el análisis de datos y la ciencia de datos, y tiene una gran comunidad de usuarios que han desarrollado y mantienen bibliotecas especializadas en estos campos, como NumPy, pandas y Matplotlib. Además, Python es un lenguaje interpretado, lo que lo hace fácil de aprender y usar.



Scala es un lenguaje de programación funcional de alto nivel, que se ejecuta en JVM (Java Virtual Machine) y se integra perfectamente con Java. Scala es un lenguaje de programación de propósito general, pero es muy popular en el procesamiento de grandes volúmenes de datos, y es el lenguaje principal utilizado en Apache Spark, una de las principales plataformas de procesamiento de datos en tiempo real. Scala también es un lenguaje muy versátil que permite a los desarrolladores combinar programación funcional y orientada a objetos.



**4. En la parte de visualización de datos, de mostrar dashboards nos encontramos con PowerBI y Tableau entre otros. Debemos explicar qué son.**

**(Adriel Sadia)**

PowerBI es una herramienta de visualización de datos desarrollada por Microsoft, diseñada para ayudar a las empresas a transformar datos en información valiosa. PowerBI es una herramienta intuitiva y fácil de usar, y es adecuada para equipos de pequeñas y medianas empresas que desean analizar y visualizar sus datos de manera efectiva.



Tableau es una herramienta de visualización de datos altamente avanzada que permite a los usuarios crear visualizaciones atractivas e interactivas de sus datos. Esta es personalizable, que permite a los usuarios crear visualizaciones únicas y atractivas que se adaptan a sus necesidades específicas. Tableau es una herramienta popular en equipos de grandes empresas, y es especialmente útil en industrias como la consultoría, el sector de servicios financieros y la banca.



**Fase 2 Mario Alajarín Escobar**

En esta segunda fase se realiza la implementación de la investigación. En concreto sería acceder a un volumen de datos y mostrarlo. Podríamos utilizar Scala o Python y mostrar el resultado en PowerBI o Tableau. La idea es que sea algo muy impactante por la calidad de contenido tratado, velocidad de acceso, volumen de datos…

Lo primero que he hecho ha sido descargar un csv que se adapte a lo que quiero.

Me he encontrado con un archivo que contenía las siguientes columnas:



Decidí establecer una relación y crear una tabla entre la industria y el número de empleados totales.

Antes de ir directo a PowerBi hice la gráfica en PyCharm ya que utilicé Python y estoy más o menos familiarizado con el programa.





Leemos el csv con Pandas.

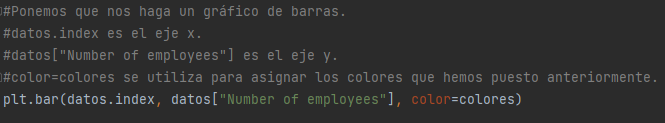


Al hacer esto podemos ver el número total de empleados que tiene una industria, si no hubiésemos hecho esto nos saldría una columna del gráfico por cada fila (no se podría ver), por lo que queremos que las industrias de cada fila que tengan el mismo nombre se agrupen.

(Aquí tuve ciertos problemas que explicaré en la fase 3).

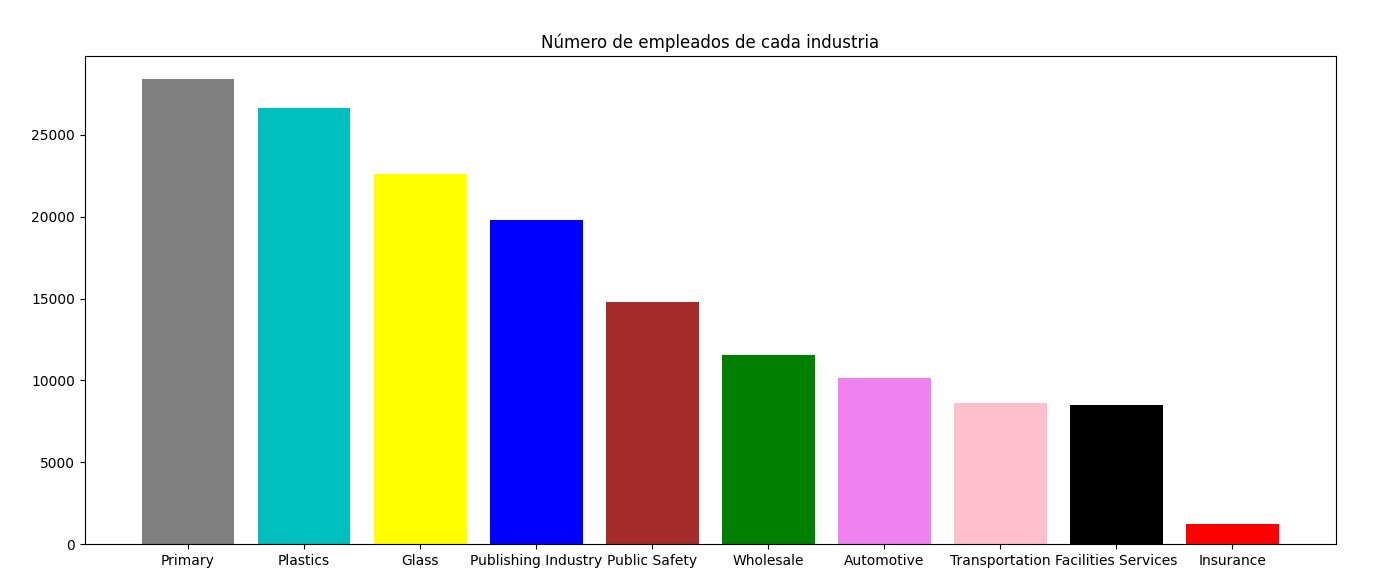






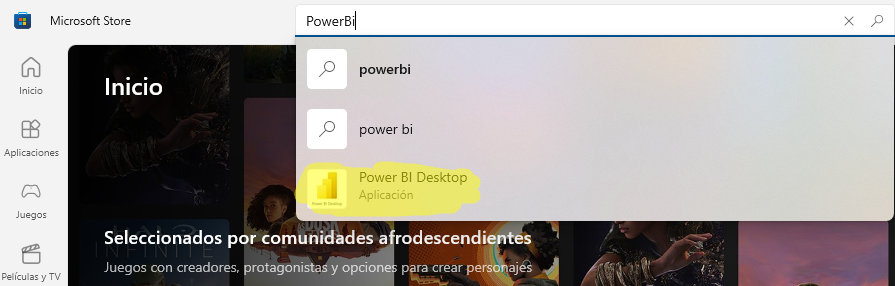


Como PyCharm es capaz de reproducir imágenes a partir de código al darle a ejecutar el programa podremos ver la gráfica.



Es hora de pasarlo a Power Bi.

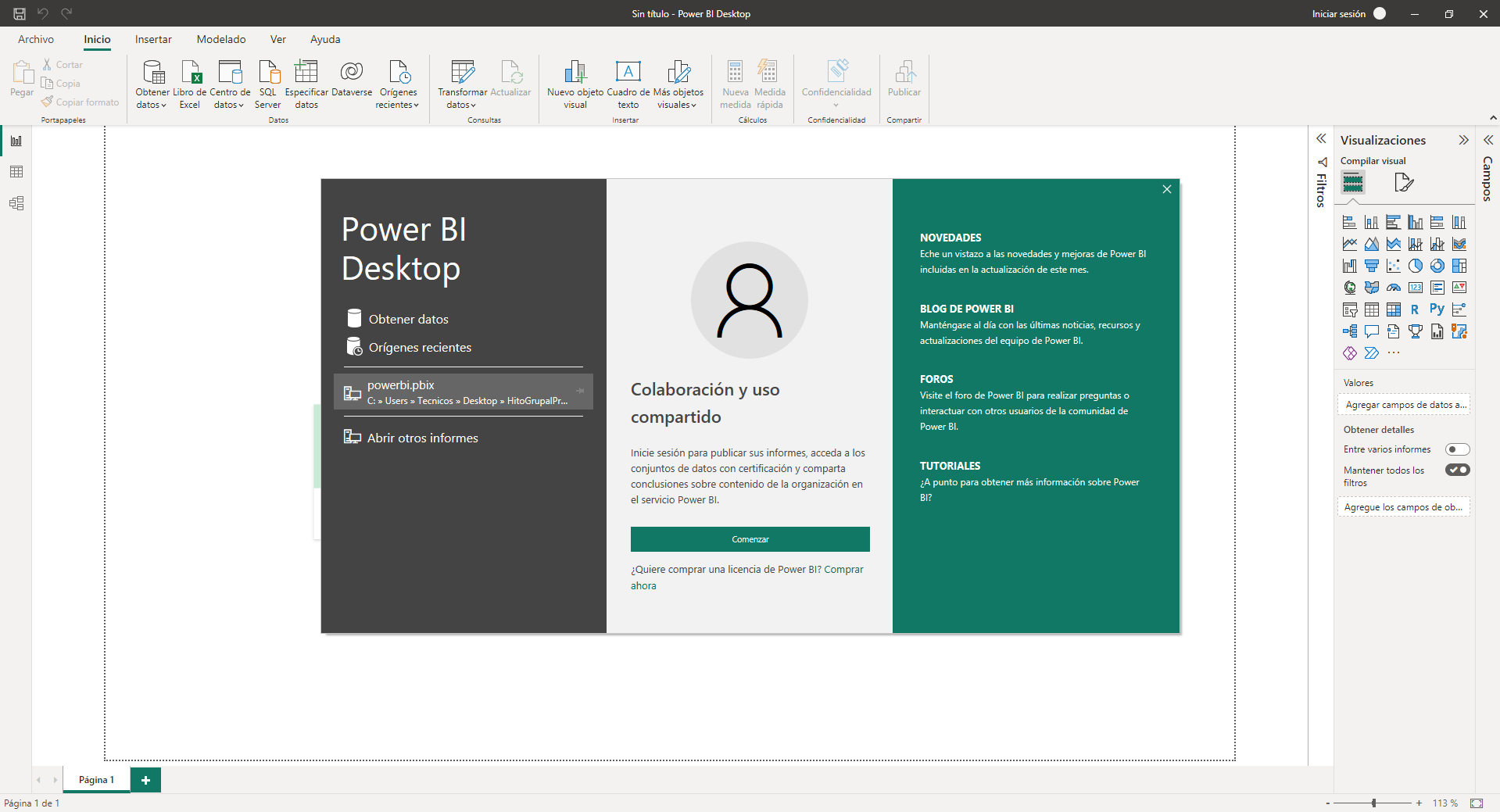
Power Bi es una aplicación de Microsoft, por lo que se puede descargar desde la Microsoft Store de Windows.



Abrimos la Microsoft Store y escribimos Power Bi, seleccionamos Power Bi Desktop y le damos a Obtener.



Una vez descargado lo abrimos.

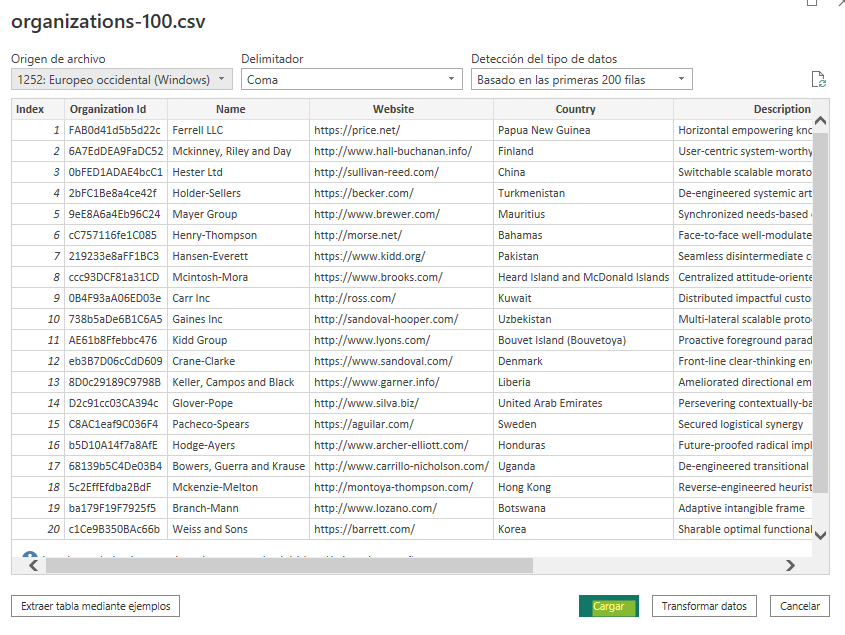


Cerramos la ventana que nos ha salido en el medio.

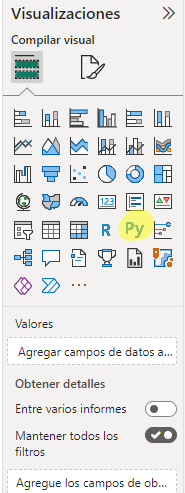


Le damos a Importar datos de Excel.

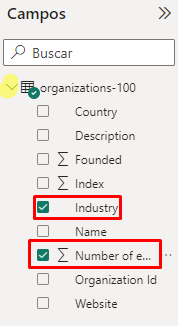
Seleccionamos nuestro csv y le damos a abrir, al hacer esto nuestro csv se conectará a Power Bi pudiendo hacer gráficas desde la aplicación.



Le damos a Cargar.

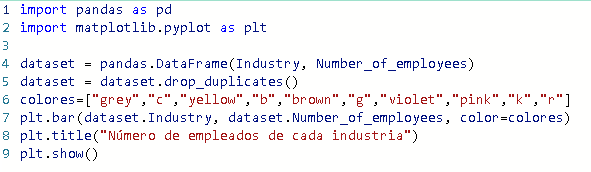


Al cargar vamos a ver en la parte de “Visualizaciones” el símbolo de Python (Py), gracias a esto podremos realizar el gráfico usando Python.

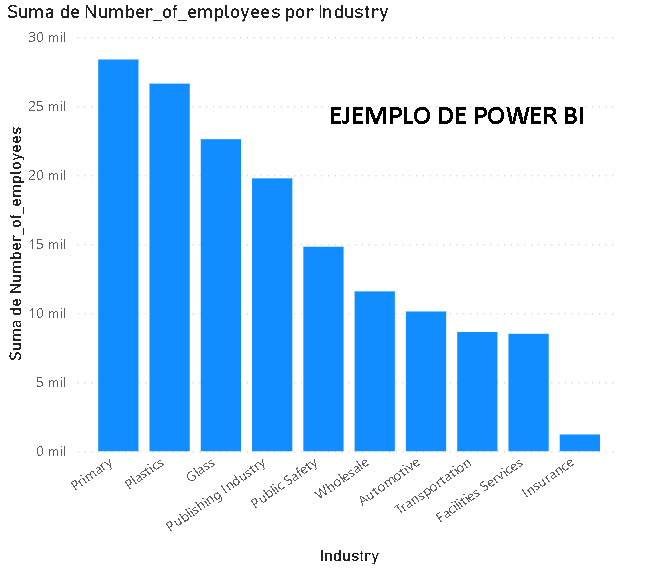


En el apartado de campos podremos ver que se nos ha cargado el csv, le damos al botón desplegable y seleccionamos “Industry” y “Number of employees”.

Se nos abrirá un editor de scripts de Python, ahí podremos escribir el código, aunque habrá que modificar algunas cosas.



Como se puede ver, ya no hace falta la función groupby() y sum() ya que Power Bi nos la hace automáticamente, a parte, utilizamos dataset para mostrar los datos, “Industry” para el eje x y “Number\_of\_employees” para el eje y.



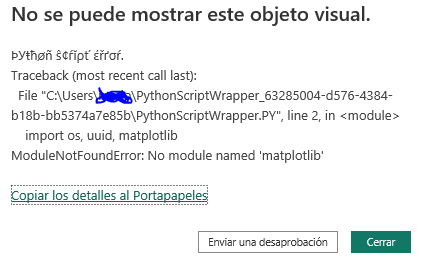
A parte de no tener que utilizar groupby() y sum(), Power Bi nos lo ordena automáticamente.

Le damos a ejecutar el código.





Sin embargo nos sale este error.



Al darle a “Ver detalles” podemos ver que resulta que no reconoce la librería de matplotlib (aunque la tengo instalada).

(En posibles errores (Fase 3) completo esta información).

**Fase 3 Mario Alajarín Escobar**

Para finalizar, realizamos una evaluación o consideraciones de cómo ha evolucionado el acceso a datos en los últimos años. Desde acceso a ficheros, pasando por base de datos y consumiendo APIs.

Acceso a ficheros:

Base de datos:

Al principio las bases de datos se registraban de manera manual, hasta los años 60 no se creó una base de datos computarizada que coincidió con la aparición de los primeros ordenadores personales.

En los años 70 se desarrolla el modelo relacional, además, aparece el SQL (Structured Query Language), que facilita la estructuración y el acceso a la información de una base de datos.

En los años 80 y 90 se crea el lenguaje de consultas, además, Microsoft lanza Microsoft Access, un gestor de bases de datos para reunir información desde plataformas como Excel.

A partir de los 2000 las bases de datos están lideradas por Oracle, IBM y Microsoft, donde los sistemas son más sofisticados y seguros.

[Webgrafía Bases de datos](https://www.adsalsa.com/evolucion-de-las-bases-de-datos)

APIs:

El desarrollo de los API comenzó en los años 70 gracias a los sistemas distribuidos.

A finales de los años 80 se creó la Programación Orientada a Objetos (POO), que permitía que las aplicaciones complejas se pudieran organizar en objetos que recogían los datos y los procedimientos.

En los años 90, los sistemas distribuidos se hicieron más comunes gracias a la WWW.

Desde 2010 la prevalencia de la API pública se ha disparado, ya que su integración pudo hacer posibles negocios online de gran escala.

Las cuatro áreas principales en los las API han tenido un gran empuje son:

1. Comercio electrónico.
2. Redes sociales.
3. Cloud Computing.
4. Movilidad.

[Webgrafía APIs](https://recluit.com/historia-y-evolucion-de-las-api/#.Y-t3D3bMKUk)

**Posibles problemas**

Problemas con el groupby() y sum():

Cuando me puse a hacer la gráfica en PyCharm y representarla me salía un resultado para cada fila, sin embargo, había filas con la misma industria que no se me juntaban, para eso tenía que utilizar groupby() y sum(), me vi un tutorial para intentar solucionarlo y funcionó.

[Tutorial](https://sparkbyexamples.com/pandas/pandas-groupby-sum-examples/?utm_content=cmp-true)

Problemas con las librerías y Power Bi:

Al hacer la gráfica en PyCharm las librerías son importadas casi inmediatamente, sin tener que hacer nada prácticamente, pero a la hora de pasarlo a Power Bi no se me importaban las librerías ya que nos las tenía instaladas, por lo que tenía que descargarlas, para eso tenía que abrir el CMD y poner “pip install pandas” y “pip install matplotlib”, sin embargo, me daba el siguiente error:



Esto se soluciona buscando la carpeta Scripts de Python y añadiendo la carpeta al Path (se hace en variables de entorno del sistema).

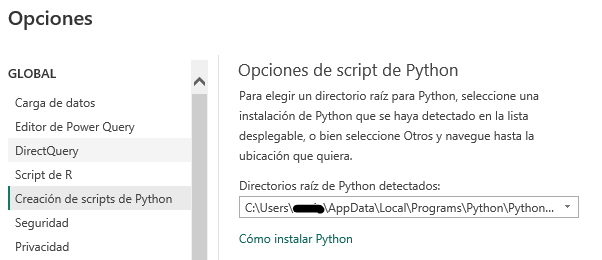
Problemas a la hora de mostrar la gráfica:

A la hora de mostrar la gráfica tuve un error que no pude solucionar y es que no reconocía la librería de matplotlib, estuve viendo foros de Microsoft (desarrolladora de Power Bi), sin embargo no lo podía arreglar.

[Intento número uno de solucionar el problema de matplotlib](https://community.powerbi.com/t5/Desktop/Now-able-to-run-python-script/m-p/610764)

[Intento número dos de solucionar el problema de matplotlib](https://community.powerbi.com/t5/Desktop/Can-not-run-Python-script-in-powerbi-desktop/td-p/1334179)

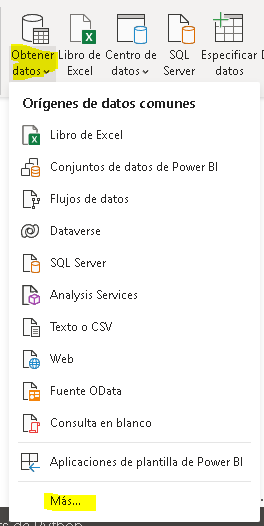
En algunas páginas me decía de actualizar pip y reinstalar matplotlib, lo cual hice y no funcionó, otras de las posibles soluciones era cambiar la ruta de Python en Power Bi (en la ventana “Opciones”).

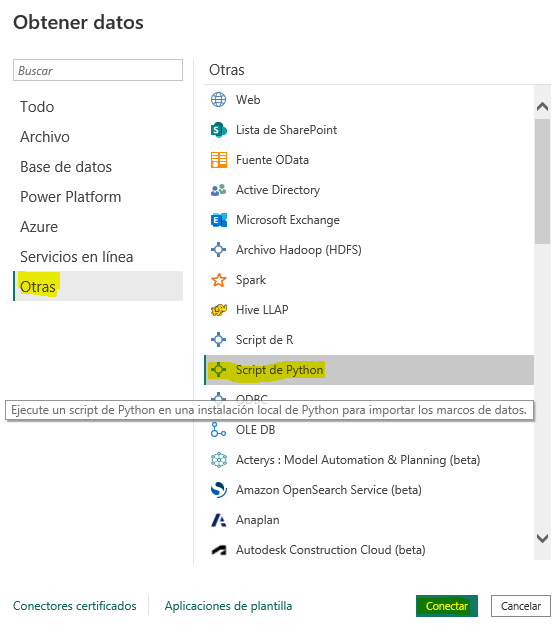


Puse la ruta de Python310 y la de los Scripts de Python310 y aún así no funcionaba.

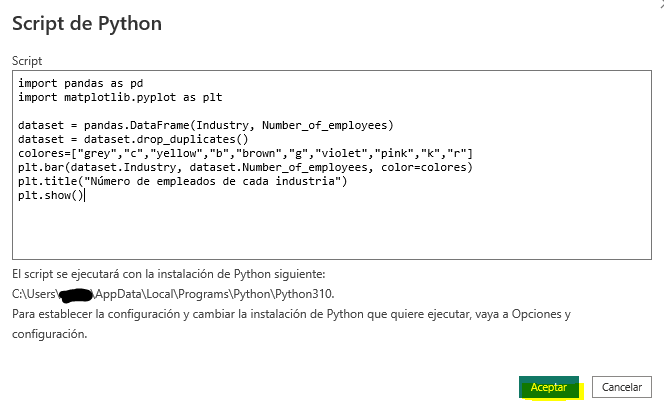
También probé a instalar matplotlib con pip3 en vez de pip, pero no dio resultado.

Como última opción probé a ejecutar el código directamente, para ello seleccioné “Obtener datos” y le di a “Más…”.

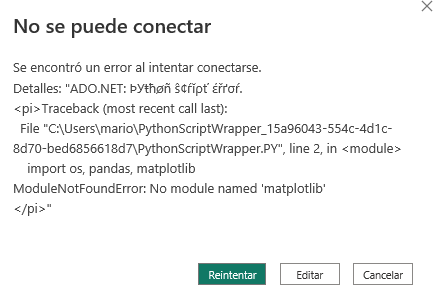




Después le di a “Otras”, luego a “Script de Python” y a “Conectar”.



Puse el código y le di a “Aceptar”.



Pero seguía sin funcionar, por lo que pensé que era problema de la ruta, o que tenía algún archivo corrupto.

Podcast: [Ivoox](https://go.ivoox.com/sq/1835217)