

# PRÁCTICA FINAL, INTRODUCCIÓN A ML UTILIZANDO ANACONDA



Presentado por:

José Luis López López

Camilo Muñoz

Daniel Fernando Perdomo

Camilo Alejandro Vivas

Presentado a:

MSc. Ing. Gustavo Adolfo Ramirez Gonzalez

Electiva Plataformas Ubicuas IoT

Facultad de Ingeniería Electrónica y Telecomunicaciones

Universidad del Cauca

marzo, 2020

<b>INTRODUCCIÓN</b>	<b>3</b>
Descargar e instalar Anaconda:	<b>4</b>
Probar algunas de las características de las librerías mediante los IDE:	<b>5</b>

## INTRODUCCIÓN

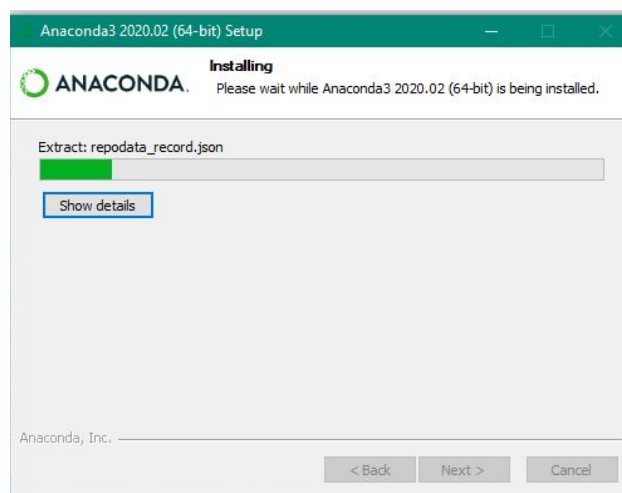
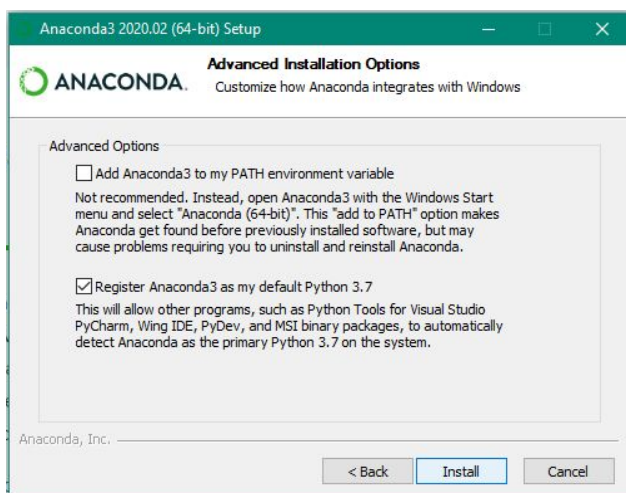
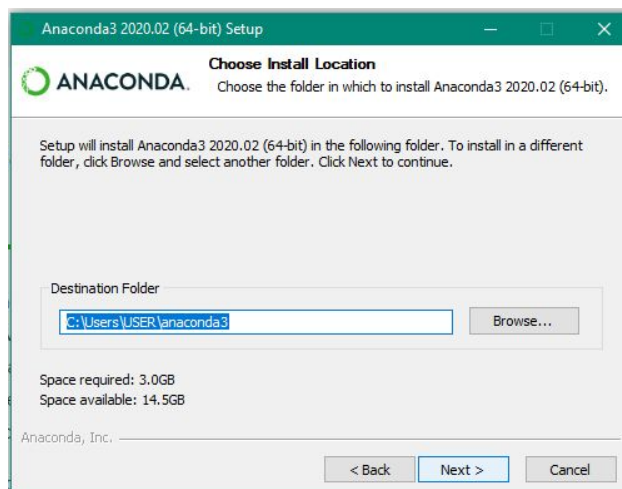
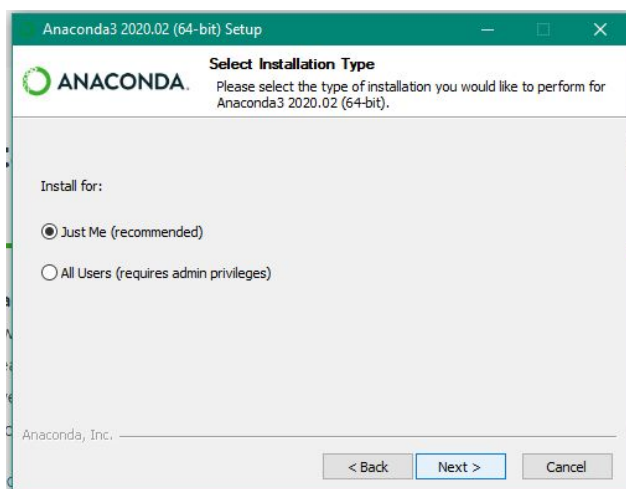
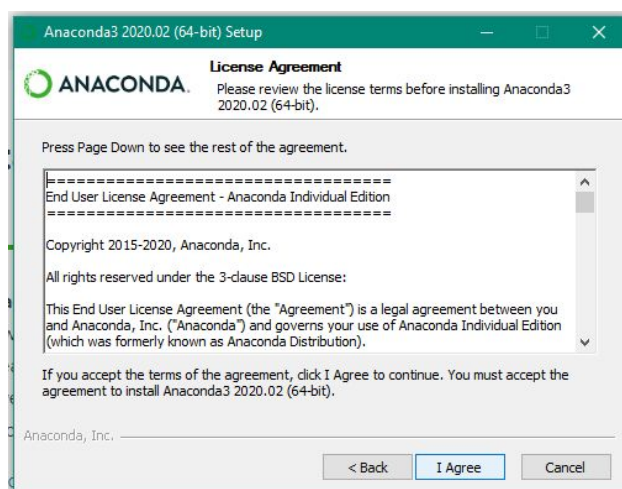
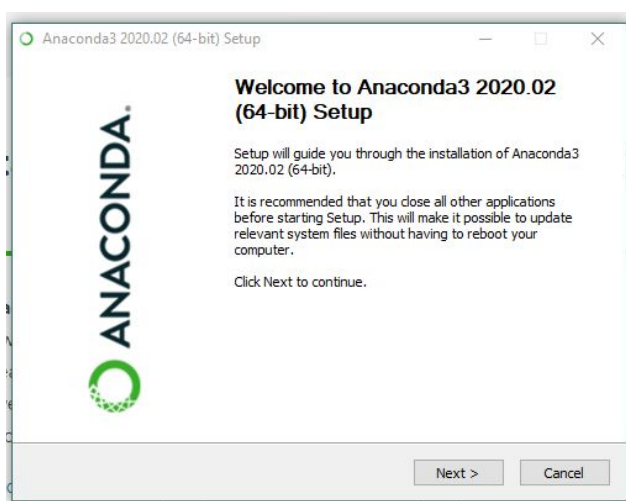
Machine Learning (ML) o aprendizaje de máquina es el proceso de predicción de datos tomados de un conjunto de ejemplos, dentro del proceso el ML descubre los patrones que hay detrás de los datos y los utiliza para hacer predicciones de nuevas circunstancias. Los lenguajes de programación enfocados en esta rama de la inteligencia artificial son: Python, Matlab, R, entre otros. En este caso y como objetivo de la práctica se busca tener un primer acercamiento al ML mediante Python como lenguaje y sus librerías NumPy, Pandas, SciKit, entre otras. que vienen previamente incluidas en la plataforma Anaconda disponible en este [enlace](#). De acuerdo con su sitio web: La edición individual de código abierto Anaconda Individual (formalmente Anaconda Distribution) es la forma más fácil de realizar ciencia de datos Python / R y aprendizaje automático en Linux, Windows y Mac OS X. Con más de 19 millones de usuarios en todo el mundo, es el estándar de la industria para el desarrollo , pruebas y capacitación en una sola máquina, lo que permite que los científicos de datos individuales:

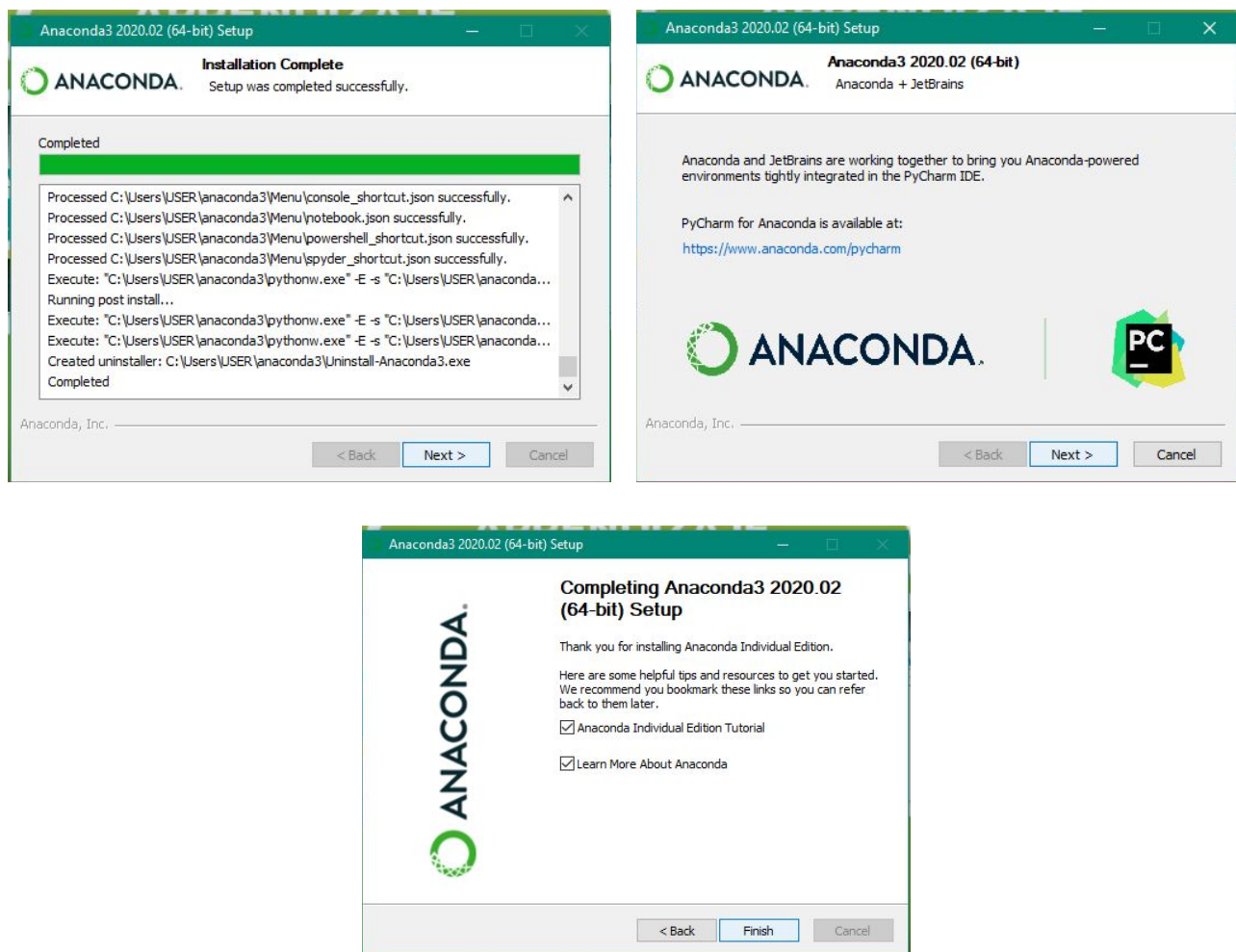
- Descarguen rápidamente más de 7,500 paquetes de ciencia de datos Python / R
- Administren bibliotecas, dependencias y entornos con Conda.
- Desarrollen y entrenen modelos de aprendizaje automático y aprendizaje profundo con scikit-learn, TensorFlow y Theano
- Analicen datos con escalabilidad y rendimiento con Dask, NumPy, pandas y Numba
- Visualicen resultados con Matplotlib, Bokeh, Datashader y Holoviews

Teniendo en cuenta lo anterior y para tener un acercamiento a lo que es ML se considera supremamente útil la plataforma Anaconda pues integra muchas características que proporcionan las herramientas necesarias en un proyecto de ML, por lo que ahorra tiempo en instalar librerías y softwares por separado. A continuación se presenta una práctica básica en la cual se prueba el funcionamiento del IDE Spyder y la librería NumPy, además de el entorno Web Jupyter Notebook y se ejecutan códigos sencillos como una introducción a las herramientas que generalmente se trabajan en ML.

## Descargar e instalar Anaconda:

Como primera parte de la práctica se debe descargar la plataforma Anaconda, haciendo clic en el siguiente [enlace](#), una vez descargado se procede a ejecutar el instalador haciendo doble clic sobre el archivo recién descargado. Aparecerán las siguientes ventanas en las cuales debemos dar clic en “continuar” (Next), o “de acuerdo” (I Agree), donde sea necesario para una instalación estándar, o recomendada por los mismos desarrolladores. Seguidamente, comienza la instalación.

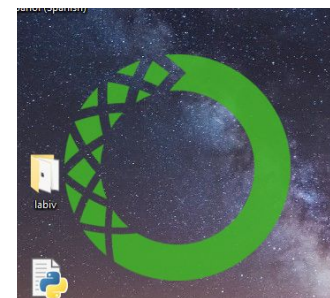
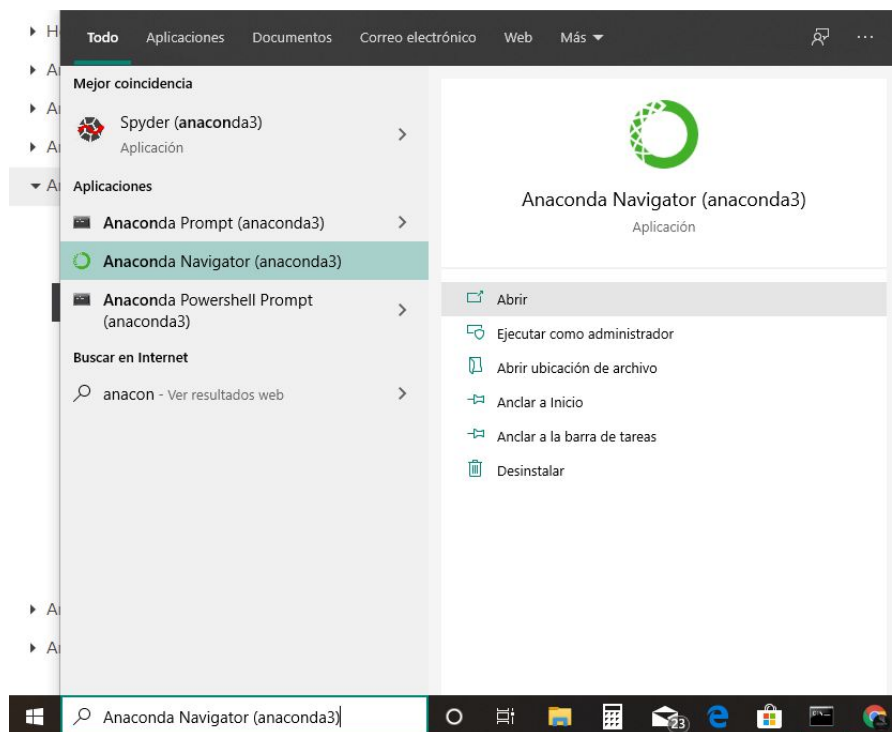




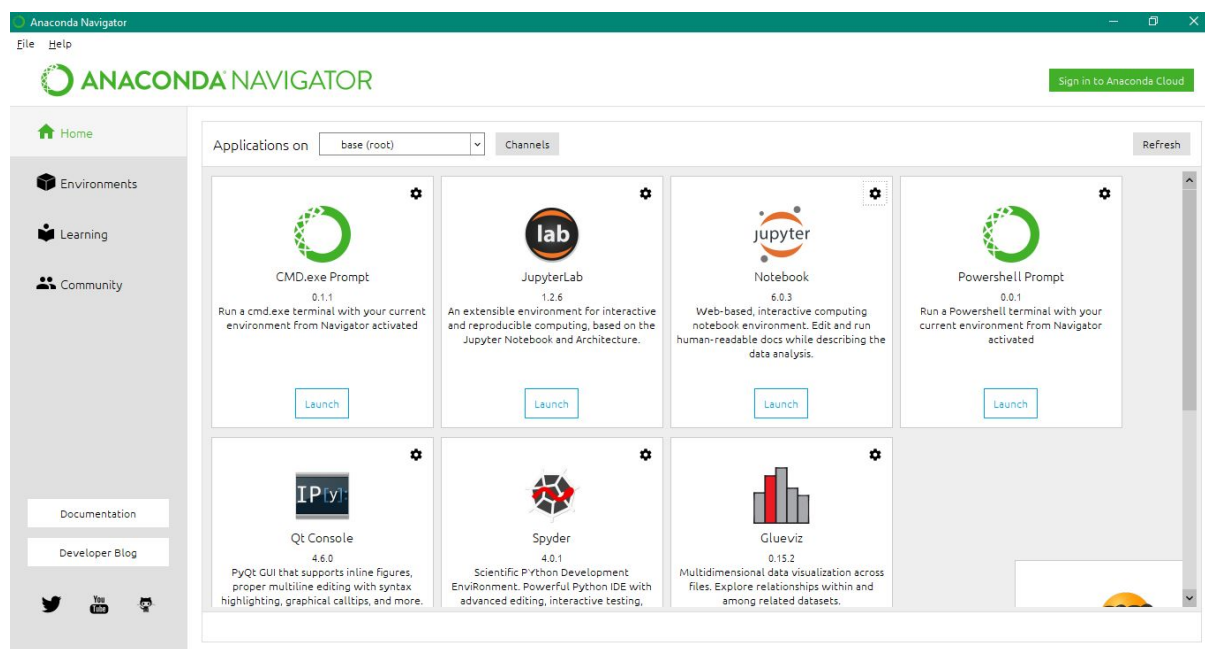
## Probar algunas de las características de las librerías mediante los IDE:

La plataforma Anaconda tiene varios componentes, entre ellos un entorno de desarrollo integrado (IDE) llamado Spyder además de librerías como NumPy, Pandas, Matplotlib, entre otras, que son útiles en proyectos de Machine Learning. A continuación, se presenta la forma de acceder a Anaconda y al IDE Spyder, después se realiza una prueba de la librería NumPy, que es una librería optimizada para estructurar arreglos de datos y operar dichos arreglos en menos tiempo en comparación con los métodos tradicionales de Python.

Para acceder a Anaconda, basta con teclear este nombre en el buscador de Windows y dando clic sobre el botón abrir accedemos al navegador de Anaconda.



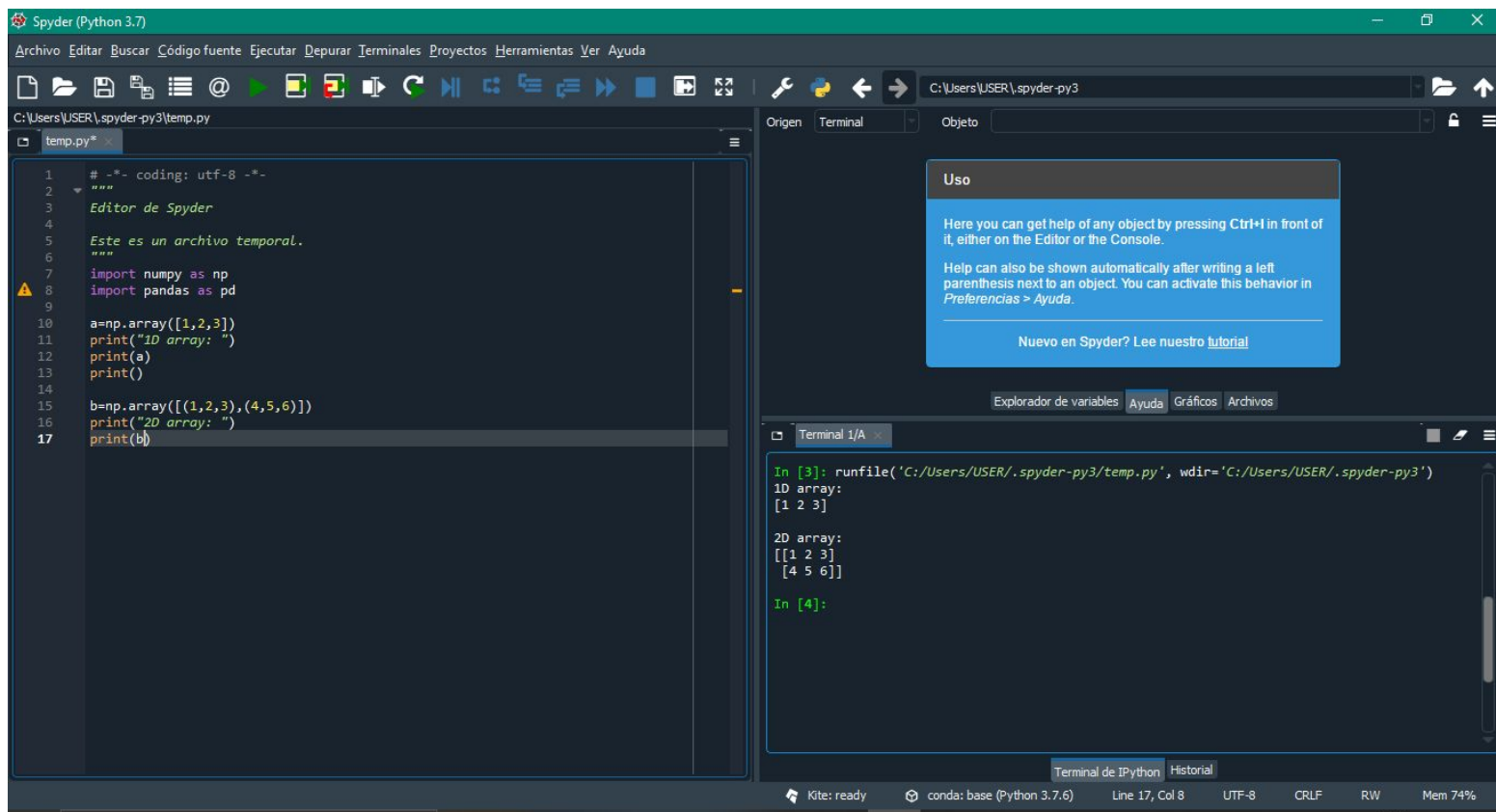
Dentro de Anaconda se accede al IDE Spyder dando doble clic sobre el cajón correspondiente a su nombre.



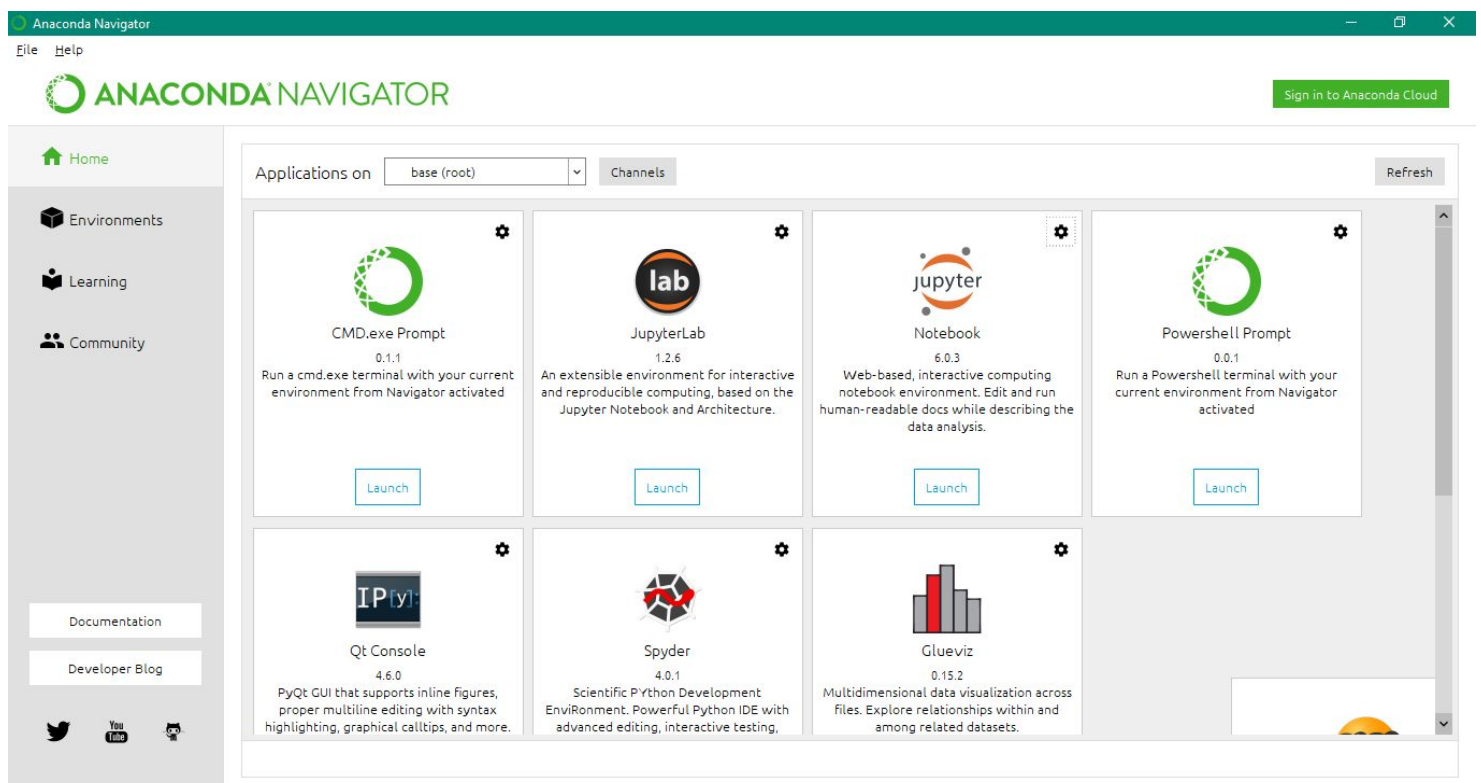
Dentro del IDE de Spyder se procede a probar con un pequeño ejemplo el comando array. El código es el que aparece en la siguiente pantalla a la izquierda, a la

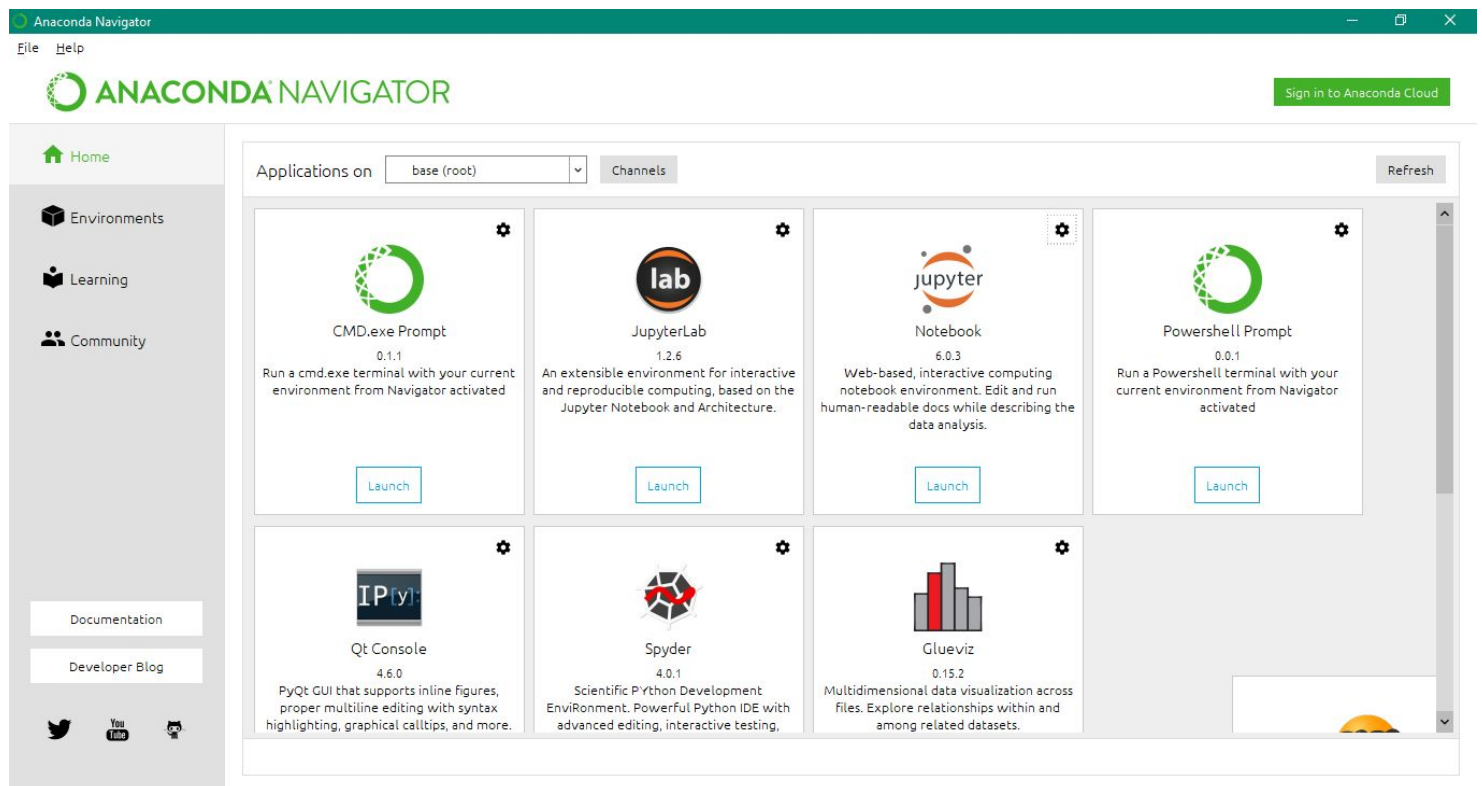


derecha de la misma se encuentra el navegador de variables y abajo de ese la salida de la ejecución, la cual se realiza presionando F5.



Otro de los componentes de Anaconda es el entorno basado en la Web Jupyter Notebook, mediante el cual haremos un ejemplo más relacionado con ML, pues se utilizará el método Classifier de la librería SciKit (Sklearn) para clasificar datos en un ejemplo de datos muy simple. Para acceder a Jupyter se debe iniciar el servidor:





Posteriormente en el prompt se teclea el comando “jupyter notebook” y se obtiene el enlace o url para acceder a través un navegador al entorno de jupyter. El resultado es el siguiente:

```

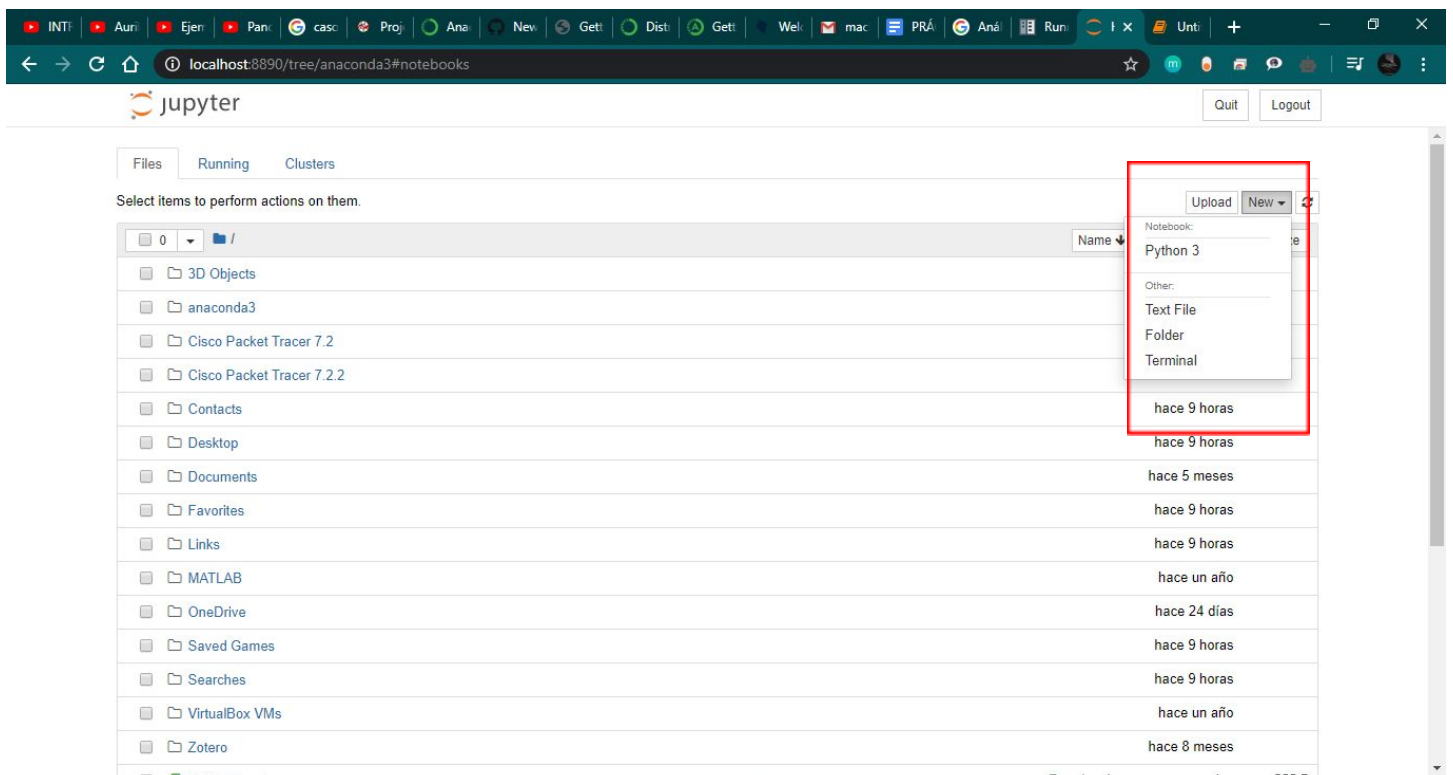
C:\> Seleccionar C:\WINDOWS\system32\cmd.exe - jupyter notebook

Microsoft Windows [Versión 10.0.18362.720]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

(base) C:\Users\USER>jupyter notebook
[I 22:50:37.031 NotebookApp] The port 8888 is already in use, trying another port.
[I 22:50:37.572 NotebookApp] JupyterLab extension loaded from C:\Users\USER\anaconda3\lib\site-packages\jupyterlab
[I 22:50:37.573 NotebookApp] JupyterLab application directory is C:\Users\USER\anaconda3\share\jupyter\lab
[I 22:50:37.602 NotebookApp] Serving notebooks from local directory: C:\Users\USER
[I 22:50:37.603 NotebookApp] The Jupyter Notebook is running at:
[I 22:50:37.603 NotebookApp] http://localhost:8889/?token=e55f7a0dd5dbff4a5d9ffd8a1d14487d52f629833b9f2a5b
[I 22:50:37.603 NotebookApp] or http://127.0.0.1:8889/?token=e55f7a0dd5dbff4a5d9ffd8a1d14487d52f629833b9f2a5b
[I 22:50:37.604 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 22:50:37.695 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/USER/AppData/Roaming/jupyter/runtime/nbserver-6892-open.html
Or copy and paste one of these URLs:
http://localhost:8889/?token=e55f7a0dd5dbff4a5d9ffd8a1d14487d52f629833b9f2a5b
or http://127.0.0.1:8889/?token=e55f7a0dd5dbff4a5d9ffd8a1d14487d52f629833b9f2a5b
  
```





Después se crea un nuevo Notebook, dando clic en New → Python 3 y se abrirá una nueva pestaña en la cual se escribe el siguiente código el cual se ejecuta para comprobar el funcionamiento del comando classifier para predicción de datos.

The image shows a Jupyter Notebook interface. The top bar displays the Jupyter logo, 'Untitled', 'Last Checkpoint: hace 19 minutos (unsaved changes)', and 'Logout'. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. Below the menu bar is a toolbar with various icons. The main area shows a code cell with the following code:

```
In [3]: from sklearn import tree

features = [[7,0.6,40],[7,0.6,41],[37,0.8,37],[37,0.8,38]]
labels = [0,0,1,1]

classifier = tree.DecisionTreeClassifier()
classifier.fit(features,labels)

res = classifier.predict([[7,0.6,41]])
print(res)

[0]
```

features corresponde a las características y labels a la etiqueta que se le da a ese “objeto” con características. En este [enlace](#) se encuentra el video donde se explica con más detalle este pequeño código.

A continuación otro [enlace](#) supremamente útil de un curso básico de ML cuyos primeros videos se resumieron en esta práctica, pero se recomienda ver dichos videos para entender un poco más a profundidad el mundo de ML y los métodos y algoritmos.

Finalmente, y a manera de complementar la respuesta a la pregunta para qué sirve el ML, se muestra en este [enlace](#) el trabajo de dos compañeros de la Universidad que a partir de un dataset obtenido por otros estudiantes de la universidad, pudieron determinar la aplicación y protocolo que más se demanda por parte de los usuarios de la red de la universidad. El dataset se encuentra adjunto al proyecto en GitHub.