

ApiRest en servidor web local con Streamlit

El objetivo de esta guía es indicar los pasos necesarios para simular la puesta en producción, con un servidor web local, de un modelo entrenado y que permita a los/as usuarios/as finales validar el modelo.

Se utilizan dos librerías diferentes para exportar(serializar) e importar(deserializar) el modelo (objetos):

- Ejemplo_3_11_CropPrediction.ipynb, Librería **pickle**
- Ejemplo_3_3_v2_Clasificación_con_Naive_Bayes_(Heart_Diseases).ipynb: librería **joblib**

Importante: Los pasos indicados a continuación para ambas librerías son iguales, tan sólo cambia el uso de una librería u otra

En nuestro caso se recomienda utilizar joblib, ya que está más extendida para serializar/exportar modelos muy grandes de ML.

Título: Pickle or Joblib for saving ML Models?

Url: <https://technorep.in/2023/04/06/pickle-or-joblib-for-saving-ml-models/#text=joblib%20is%20generally%20faster%20than%20pickle%20may%20not%20be%20significant>.

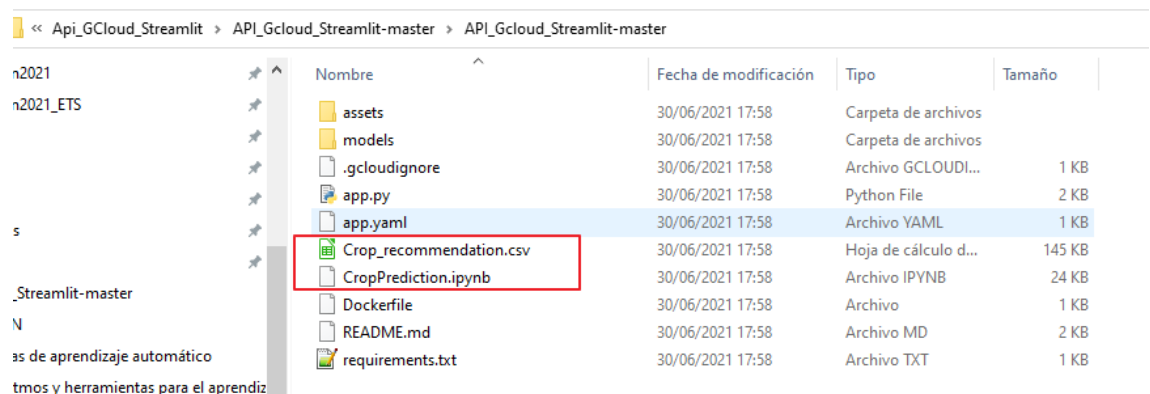
La tecnología para montar el servidor local está disponible en <https://streamlit.io/>

Este documento refleja los pasos realizados siguiendo como referencia las pautas indicadas en el siguiente video aunque ha sido necesario realizar algún pequeño ajuste derivado de variaciones y errores causados por las librerías propuestas e instaladas según se indica en el video.

<https://www.youtube.com/watch?v=vibDbEBnyV4> – Parte 1

Pasos para ejemplo con pickle:

1. Acceder al siguiente github y descargar el código fuente
https://github.com/DavidReveloLuna/API_Gcloud_Streamlit
2. Crear en GooglerColab un cuaderno con el proyecto que se suministra en el fichero descargado en el punto anterior y ejecutarlo para verificar su funcionamiento:



« Api_GCloud_Streamlit > API_Gcloud_Streamlit-master > API_Gcloud_Streamlit-master				
	Nombre	Fecha de modificación	Tipo	Tamaño
n2021	assets	30/06/2021 17:58	Carpeta de archivos	
n2021_ETS	models	30/06/2021 17:58	Carpeta de archivos	
	.gcloudignore	30/06/2021 17:58	Archivo GCLLOUDI...	1 KB
	app.py	30/06/2021 17:58	Python File	2 KB
s	app.yaml	30/06/2021 17:58	Archivo YAML	1 KB
	Crop_recommendation.csv	30/06/2021 17:58	Hoja de cálculo d...	145 KB
_Streamlit-master	CropPrediction.ipynb	30/06/2021 17:58	Archivo IPYNB	24 KB
N	Dockerfile	30/06/2021 17:58	Archivo	1 KB
as de aprendizaje automático	README.md	30/06/2021 17:58	Archivo MD	2 KB
tmos y herramientas para el aprendiz	requirements.txt	30/06/2021 17:58	Archivo TXT	1 KB

Nota: En mi caso comenté la línea siguiente para ejecutarlo con la versión de ScikitLearn que actualmente está instalada en GoogleColab

```
1. Instalación scikit-learn

[1] pip list

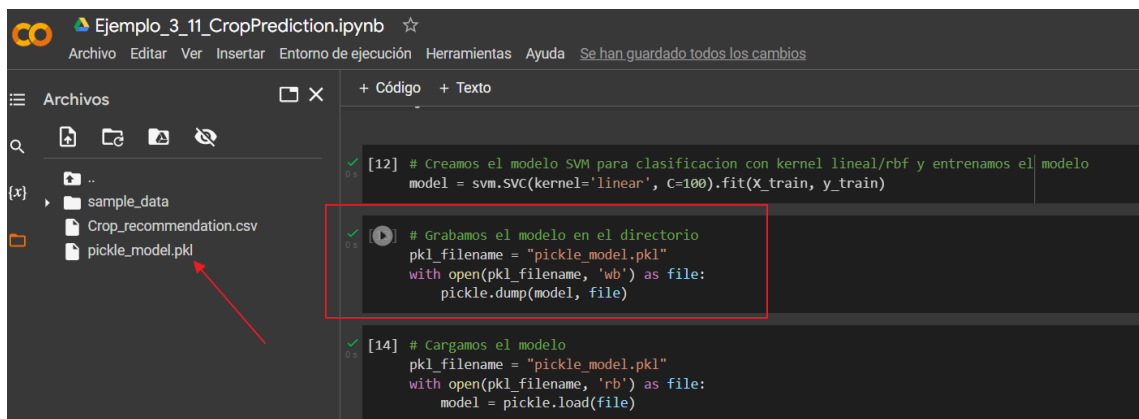
[ ] #!pip install scikit-learn==0.24.1
```

El cuaderno lo tienes disponible en:

Título: Ejemplo_3_11_CropPrediction.ipynb

Url: <https://colab.research.google.com/drive/1C-vhgSfYMciipKCPd72w5Z-Me1215TJr?usp=sharing>

3. Una vez has **entrenado** el modelo descargar el fichero con los parámetros del modelo



```
Ejemplo_3_11_CropPrediction.ipynb
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Se han guardado todos los cambios

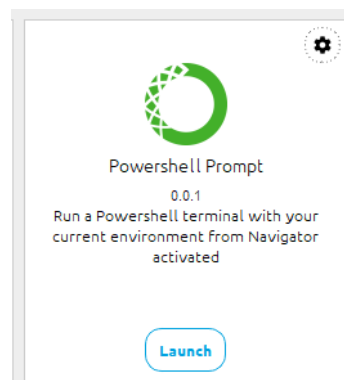
Archivos
sample_data
Crop_recommendation.csv
pickle_model.pkl

[12] # Creamos el modelo SVM para clasificacion con kernel lineal/rbf y entrenamos el modelo
model = svm.SVC(kernel='linear', C=100).fit(X_train, y_train)

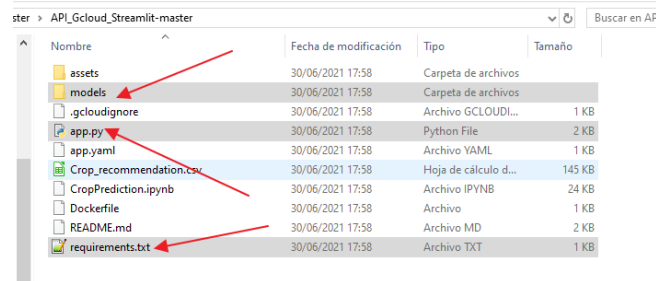
[13] # Grabamos el modelo en el directorio
pkl_filename = "pickle_model.pkl"
with open(pkl_filename, 'wb') as file:
    pickle.dump(model, file)

[14] # Cargamos el modelo
pkl_filename = "pickle_model.pkl"
with open(pkl_filename, 'rb') as file:
    model = pickle.load(file)
```

4. A partir de este punto trabajamos en el PC local

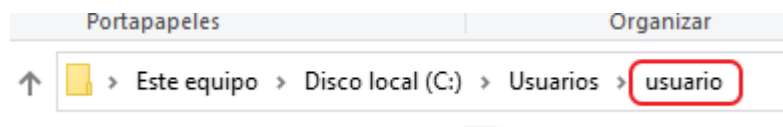


- a. Arrancar la consola de ProwerShell del Anaconda
- b. Copiar en la carpeta del usuario con el que te has logueado en Windows los siguientes ficheros:

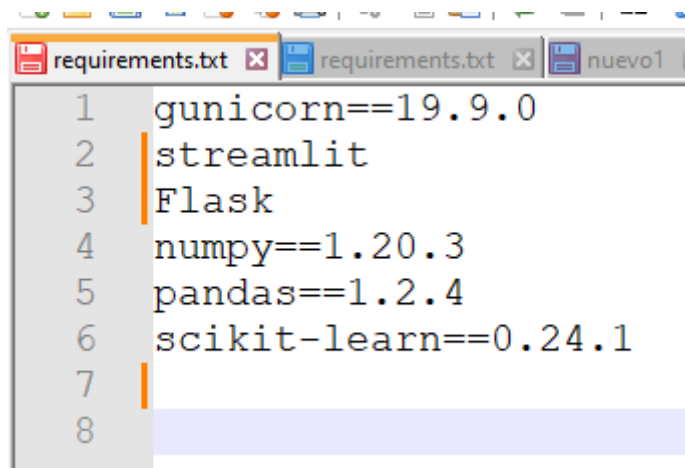


Nombre	Fecha de modificación	Tipo	Tamaño
assets	30/06/2021 17:58	Carpeta de archivos	
models	30/06/2021 17:58	Carpeta de archivos	
.gitignore	30/06/2021 17:58	Archivo GLOU...	1 KB
app.py	30/06/2021 17:58	Python File	2 KB
app.yaml	30/06/2021 17:58	Archivo YAM...	1 KB
Crop_recommendation.csv	30/06/2021 17:58	Hoja de cálculo d...	145 KB
CropPrediction.ipynb	30/06/2021 17:58	Archivo IPYNB	24 KB
Dockerfile	30/06/2021 17:58	Archivo	1 KB
README.md	30/06/2021 17:58	Archivo MD	2 KB
requirements.txt	30/06/2021 17:58	Archivo TXT	1 KB

En mi caso esta es la carpeta que utilizo en mi portátil:



En el fichero requeriments.txt tenemos el siguiente contenido:



```
1 19.9.0
2 streamlit
3 Flask
4 numpy==1.20.3
5 pandas==1.2.4
6 scikit-learn==0.24.1
7
8
```

IMPORTANTE: Ha de tener dichos módulos, en dicho orden y versiones

c. Desde el **PowerShell** iniciado ejecutar los siguientes comandos

i. *Creamos el entorno y lo activamos:*

conda create -n ApiCrop

conda activate ApiCrop

Nota: Para salir de n environment: conda deactivate ApiCrop

Para borrar un environment: conda remove ApiCrop

ii. *Instalamos las librerías que necesitamos:*

conda install python=3.9

pip install -r requirements.txt

iii. *Arrancamos el servidor streamlit:*

streamlit run app.py

```
(ApiCrop) PS C:\Users\usuario> streamlit run app.py
You can now view your Streamlit app in your browser.
Local URL: http://localhost:8501
Network URL: http://192.168.0.101:8501
```

Y tenemos la aplicación ejecutándose:

← → ↻ 🏠 localhost:8501 🔍 📄 ☆ ⚙️

salvador.bordon@g... Test de Velocidad... moodle local IESEIRincon MongoDB Herramientas Gobierno de Canari... Detalles del servicio... Java

CULTIVO

Nitrógeno:

Fósforo:

Potasio:

Temperatura:

Humedad:

pH:

Lluvia:

Predicción :

EL CULTIVO RECOMENDADO ES: KIDNEYBEANS

Pasos para ejemplo con joblib:

Este gui3n ha sido preparado con el proyecto:

Título: Ejemplo_3_3_v2_Clasificaci3n_con_Naive_Bayes_(Heart_Diseases).ipynb

Url: https://colab.research.google.com/drive/1rz8Hs6H10wsUKofKdl4O0G8m_V5ae3h3?usp=sharing

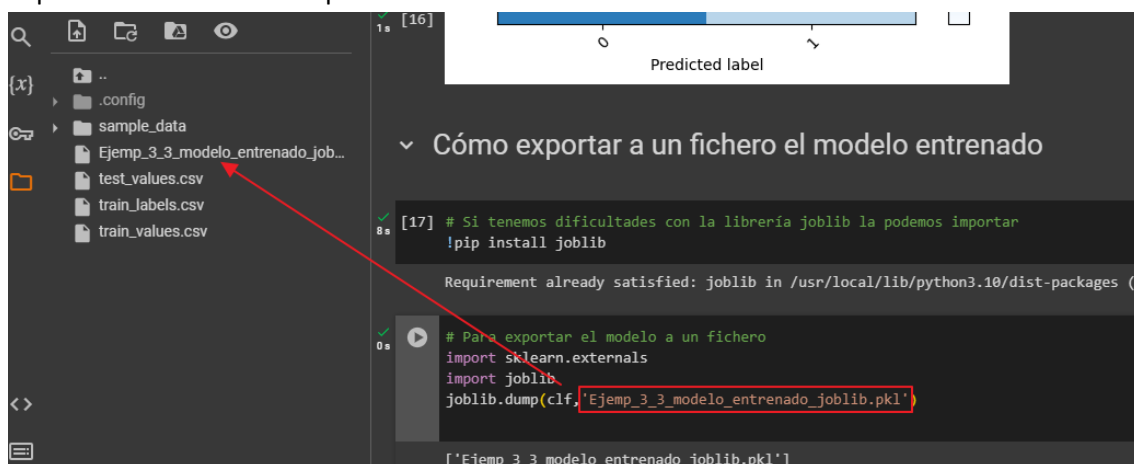
Se adjunta tambi3n un fichero .zip con recursos preparados para ser utilizados:

Recursos_ApiHeartDiseases para Steamlit.zip

Nombre	Fecha de modificaci3n	Tipo	Tama1o
models	15/01/2024 1:01	Carpeta de archivos	
appHeartDiseases.py	08/01/2023 20:46	Python File	2 KB
requeriments.txt	15/01/2024 1:01	Archivo TXT	1 KB

Los pasos son:

1. En Google Colab crear/abrir el proyecto y entrenar el modelo
2. Exportar el fichero con las par3metros del modelo entrenado



The screenshot shows the Google Colab interface. On the left, the file explorer displays a directory structure with files: `.config`, `sample_data`, `Ejemp_3_3_modelo_entrenado.joblib`, `test_values.csv`, `train_labels.csv`, and `train_values.csv`. A red arrow points from the code cell to the `Ejemp_3_3_modelo_entrenado.joblib` file. The main code cell is titled "C3mo exportar a un fichero el modelo entrenado" and contains the following code:

```
[17] # Si tenemos dificultades con la librería joblib la podemos importar
!pip install joblib

Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (1

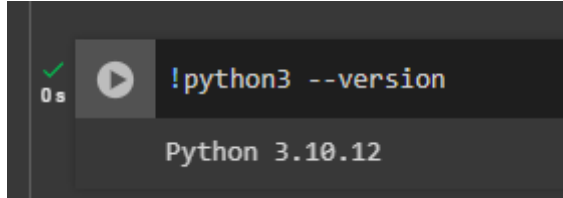
# Para exportar el modelo a un fichero
import sklearn.externals
import joblib
joblib.dump(clf, 'Ejemp_3_3_modelo_entrenado_joblib.pkl')

['Ejemp_3_3_modelo_entrenado_joblib.pkl']
```

3. Realizar un inventario de las librerías que utilizas y sus versiones para instalarlas en el PC Local

→ Versión de Python: `!python3 --version`

Importante: en el entorno en local instalar la versión 3.9 de Python. Es como funciona bien las librerías de Streamlit.



```
!python3 --version

Python 3.10.12
```

→ Añadir en el fichero llamado requirements.txt las diferentes librerías y versiones utilizadas en el cuaderno, ya las que sean necesarias (*las destacadas en negritas son necesarias para ejecutar la api rest en local*):

Importante: Dejarlas en este orden en el fichero credentials.txt

gunicorn==19.9.0

streamlit

Flask

numpy==1.23.5

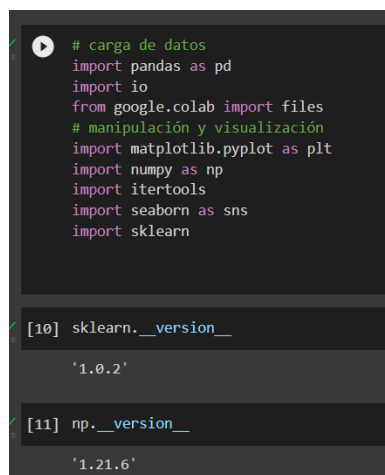
pandas==1.2.4

scikit-learn==1.2.2

protobuf==3.20.*

joblib==1.2.2

A modo de ejemplo:



```
# carga de datos
import pandas as pd
import io
from google.colab import files
# manipulación y visualización
import matplotlib.pyplot as plt
import numpy as np
import itertools
import seaborn as sns
import sklearn

[10] sklearn.__version__

'1.0.2'

[11] np.__version__

'1.21.6'
```

4. En **Anaconda PowerShell** realizar la instalación del entorno y de las librerías para nuestro proyecto:

i. *Creamos el entorno y lo activamos:*

`conda create -n ApiHeartDiseases`

`conda activate ApiHeartDiseases`

ii. *Instalamos las librerías que necesitamos:*

Aunque en GoogleColab tenemos la 3.10, en local instalo la 3.9, para no tener que actualizar las librerías de Microsoft Visual C++ (En fin... las dependencias)

`conda install python=3.9`

`pip install -r requirements.txt`

iii. *Copiamos el fichero .pkl generado en GoogleColab*

iv. *Arrancamos el servidor streamlit:*

`streamlit run appHeartDiseases.py`

```
(ApiHeartDiseases) PS C:\Users\usuario> cd .\SNS\APiRest_Ejemplo_3_3\  
(ApiHeartDiseases) PS C:\Users\usuario\SNS\APiRest_Ejemplo_3_3> streamlit run app.py  
  
You can now view your Streamlit app in your browser.  
  
Local URL: http://localhost:8501  
Network URL: http://192.168.0.101:8501
```

Estudio de enfermedades cardíacas

Valor de resting_blood_pressure:

120

Valor de serum_cholesterol_mg_per_dl:

178

Valor de max_heart_rate_achieved:

120

Predicción :

LA PREDICCIÓN DE DOLENCIA CARDIACA ES: 1

ApiRest en Google Cloud

- <https://www.youtube.com/watch?v=o4sxWISxpXk> – Parte 2 – En GoogleCloud