Librerias

```
!pip install dataprep
```

```
Collecting dataprep
  Downloading dataprep-0.4.3-py3-none-any.whl (9.5 MB)
ents to build wheel ... etadata ... ent already satisfied: bokeh<3,>=2
in /usr/local/lib/python3.7/dist-packages (from dataprep) (2.3.3)
Collecting dask[array,dataframe,delayed]<2022.0,>=2021.11
  Downloading dask-2021.12.0-py3-none-any.whl (1.0 MB)
-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (749 kB)
e<0.9.0,>=0.8.1
  Downloading varname-0.8.3-py3-none-any.whl (21 kB)
Collecting flask_cors<4.0.0,>=3.0.10
  Downloading Flask_Cors-3.0.10-py2.py3-none-any.whl (14 kB)
Requirement already satisfied: tqdm<5.0,>=4.48 in
/usr/local/lib/python3.7/dist-packages (from dataprep) (4.64.0)
Collecting python-stdnum<2.0,>=1.16
  Downloading python_stdnum-1.17-py2.py3-none-any.whl (943 kB)
ent already satisfied: pandas<2.0,>=1.1 in
/usr/local/lib/python3.7/dist-packages (from dataprep) (1.3.5)
Requirement already satisfied: sqlalchemy<2.0.0,>=1.4.32 in
/usr/local/lib/python3.7/dist-packages (from dataprep) (1.4.36)
Collecting wordcloud<2.0,>=1.8
  Downloading wordcloud-1.8.1-cp37-cp37m-manylinux1_x86_64.whl (366
kB)
ent already satisfied: ipywidgets<8.0,>=7.5 in
/usr/local/lib/python3.7/dist-packages (from dataprep) (7.7.0)
Collecting pydantic<2.0,>=1.6
  Downloading pydantic-1.9.1-cp37-cp37m-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.1 MB)
-
manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.manylinux
2010_x86_64.whl (1.1 MB)
ent already satisfied: scipy<=1.7.1 in /usr/local/lib/python3.7/dist-
packages (from dataprep) (1.4.1)
Requirement already satisfied: numpy<2.0,>=1.21 in
/usr/local/lib/python3.7/dist-packages (from dataprep) (1.21.6)
Collecting python-crfsuite<0.10.0,>=0.9.7
  Downloading python_crfsuite-0.9.8-cp37-cp37m-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (965 kB)
etaphone<0.7,>=0.6
  Downloading Metaphone-0.6.tar.gz (14 kB)
Collecting jsonpath-ng<2.0,>=1.5
  Downloading jsonpath_ng-1.5.3-py3-none-any.whl (29 kB)
Collecting async-timeout<5.0,>=4.0.0a3
  Downloading async_timeout-4.0.2-py3-none-any.whl (5.8 kB)
Collecting frozenlist>=1.1.1
  Downloading frozenlist-1.3.0-cp37-cp37m-
manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_17_x86_64.manylinux
```

```
2014_x86_64.whl (144 kB)
ent already satisfied: charset-normalizer<3.0,>=2.0 in
/usr/local/lib/python3.7/dist-packages (from aiohttp<4.0,>=3.6-
>dataprep) (2.0.12)
Requirement already satisfied: typing-extensions>=3.7.4 in
/usr/local/lib/python3.7/dist-packages (from aiohttp<4.0,>=3.6-
>dataprep) (4.2.0)
Collecting asynctest==0.13.0
  Downloading asynctest-0.13.0-py3-none-any.whl (26 kB)
Requirement already satisfied: attrs>=17.3.0 in
/usr/local/lib/python3.7/dist-packages (from aiohttp<4.0,>=3.6-
>dataprep) (21.4.0)
Collecting multidict<7.0,>=4.5
  Downloading multidict-6.0.2-cp37-cp37m-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (94 kB)
-
manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.manylinux
2010_x86_64.whl (271 kB)
ent already satisfied: PyYAML>=3.10 in /usr/local/lib/python3.7/dist-
packages (from bokeh<3,>=2->dataprep) (3.13)
Requirement already satisfied: tornado>=5.1 in
/usr/local/lib/python3.7/dist-packages (from bokeh<3,>=2->dataprep)
(5.1.1)
Requirement already satisfied: packaging>=16.8 in
/usr/local/lib/python3.7/dist-packages (from bokeh<3,>=2->dataprep)
(21.3)
Requirement already satisfied: pillow>=7.1.0 in
/usr/local/lib/python3.7/dist-packages (from bokeh<3,>=2->dataprep)
(7.1.2)
Requirement already satisfied: python-dateutil>=2.1 in
/usr/local/lib/python3.7/dist-packages (from bokeh<3,>=2->dataprep)
(2.8.2)
Collecting partd>=0.3.10
  Downloading partd-1.2.0-py3-none-any.whl (19 kB)
Collecting fsspec>=0.6.0
  Downloading fsspec-2022.5.0-py3-none-any.whl (140 kB)
ent already satisfied: cloudpickle>=1.1.1 in
/usr/local/lib/python3.7/dist-packages (from
dask[array,dataframe,delayed]<2022.0,>=2021.11->dataprep) (1.3.0)
Requirement already satisfied: toolz>=0.8.2 in
/usr/local/lib/python3.7/dist-packages (from
dask[array,dataframe,delayed]<2022.0,>=2021.11->dataprep) (0.11.2)
Collecting Werkzeug>=2.0
  Downloading Werkzeug-2.1.2-py3-none-any.whl (224 kB)
ent already satisfied: importlib-metadata>=3.6.0 in
/usr/local/lib/python3.7/dist-packages (from flask<3,>=2->dataprep)
(4.11.3)
Collecting itsdangerous>=2.0
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting click>=8.0
```

```
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
ent already satisfied: Six in /usr/local/lib/python3.7/dist-packages
(from flask_cors<4.0.0,>=3.0.10->dataprep) (1.15.0)
Requirement already satisfied: zipp>=0.5 in
/usr/local/lib/python3.7/dist-packages (from importlib-
metadata>=3.6.0->flask<3,>=2->dataprep) (3.8.0)
Requirement already satisfied: jupyterlab-widgets>=1.0.0 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (1.1.0)
Requirement already satisfied: ipython-genutils~=0.2.0 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (0.2.0)
Requirement already satisfied: widgetsnbextension~=3.6.0 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (3.6.0)
Requirement already satisfied: nbformat>=4.2.0 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (5.4.0)
Requirement already satisfied: ipykernel>=4.5.1 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (4.10.1)
Requirement already satisfied: traitlets>=4.3.1 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (5.1.1)
Requirement already satisfied: ipython>=4.0.0 in
/usr/local/lib/python3.7/dist-packages (from ipywidgets<8.0,>=7.5-
>dataprep) (5.5.0)
Requirement already satisfied: jupyter-client in
/usr/local/lib/python3.7/dist-packages (from ipykernel>=4.5.1-
>ipywidgets<8.0,>=7.5->dataprep) (5.3.5)
Requirement already satisfied: pygments in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (2.6.1)
Requirement already satisfied: setuptools>=18.5 in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (57.4.0)
Requirement already satisfied: decorator in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (4.4.2)
Requirement already satisfied: simplegeneric>0.8 in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (0.8.1)
Requirement already satisfied: pexpect in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (4.8.0)
Requirement already satisfied: prompt-toolkit<2.0.0,>=1.0.4 in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
>ipywidgets<8.0,>=7.5->dataprep) (1.0.18)
Requirement already satisfied: pickleshare in
/usr/local/lib/python3.7/dist-packages (from ipython>=4.0.0-
```

```
>ipywidgets<8.0,>=7.5->dataprep) (0.7.5)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.7/dist-packages (from jinja2<4,>=3->dataprep)
(2.0.1)
Collecting ply
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
ent already satisfied: fastjsonschema in
/usr/local/lib/python3.7/dist-packages (from nbformat>=4.2.0-
>ipywidgets<8.0,>=7.5->dataprep) (2.15.3)
Requirement already satisfied: jupyter-core in
/usr/local/lib/python3.7/dist-packages (from nbformat>=4.2.0-
>ipywidgets<8.0,>=7.5->dataprep) (4.10.0)
Requirement already satisfied: jsonschema>=2.6 in
/usr/local/lib/python3.7/dist-packages (from nbformat>=4.2.0-
>ipywidgets<8.0,>=7.5->dataprep) (4.3.3)
Requirement already satisfied: importlib-resources>=1.4.0 in
/usr/local/lib/python3.7/dist-packages (from jsonschema>=2.6-
>nbformat>=4.2.0->ipywidgets<8.0,>=7.5->dataprep) (5.7.1)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!
=0.17.2,>=0.14.0 in /usr/local/lib/python3.7/dist-packages (from
jsonschema>=2.6->nbformat>=4.2.0->ipywidgets<8.0,>=7.5->dataprep)
(0.18.1)
Requirement already satisfied: joblib in
/usr/local/lib/python3.7/dist-packages (from nltk<4.0.0,>=3.6.7-
>dataprep) (1.1.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/usr/local/lib/python3.7/dist-packages (from packaging>=16.8-
>bokeh<3,>=2->dataprep) (3.0.9)
Requirement already satisfied: pytz>=2017.3 in
/usr/local/lib/python3.7/dist-packages (from pandas<2.0,>=1.1-
>dataprep) (2022.1)
Collecting locket
  Downloading locket-1.0.0-py2.py3-none-any.whl (4.4 kB)
Requirement already satisfied: wcwidth in
/usr/local/lib/python3.7/dist-packages (from prompt-
toolkit<2.0.0,>=1.0.4->ipython>=4.0.0->ipywidgets<8.0,>=7.5->dataprep)
(0.2.5)
Requirement already satisfied: greenlet!=0.4.17 in
/usr/local/lib/python3.7/dist-packages (from
sqlalchemy<2.0.0,>=1.4.32->dataprep) (1.1.2)
Collecting asttokens<3.0.0,>=2.0.0
  Downloading asttokens-2.0.5-py2.py3-none-any.whl (20 kB)
Collecting executing<0.9.0,>=0.8.3
  Downloading executing-0.8.3-py2.py3-none-any.whl (16 kB)
Collecting pure_eval<1.0.0
  Downloading pure_eval-0.2.2-py3-none-any.whl (11 kB)
Requirement already satisfied: notebook>=4.4.1 in
/usr/local/lib/python3.7/dist-packages (from
widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5->dataprep) (5.3.1)
Requirement already satisfied: Send2Trash in
```

```
/usr/local/lib/python3.7/dist-packages (from notebook>=4.4.1-
>widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5->dataprep) (1.8.0)
Requirement already satisfied: terminado>=0.8.1 in
/usr/local/lib/python3.7/dist-packages (from notebook>=4.4.1-
>widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5->dataprep) (0.13.3)
Requirement already satisfied: nbconvert in
/usr/local/lib/python3.7/dist-packages (from notebook>=4.4.1-
>widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5->dataprep) (5.6.1)
Requirement already satisfied: pyzmq>=13 in
/usr/local/lib/python3.7/dist-packages (from jupyter-client-
>ipykernel>=4.5.1->ipywidgets<8.0,>=7.5->dataprep) (22.3.0)
Requirement already satisfied: ptyprocess in
/usr/local/lib/python3.7/dist-packages (from terminado>=0.8.1-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (0.7.0)
Requirement already satisfied: matplotlib in
/usr/local/lib/python3.7/dist-packages (from wordcloud<2.0,>=1.8-
>dataprep) (3.2.2)
Requirement already satisfied: idna>=2.0 in
/usr/local/lib/python3.7/dist-packages (from yarl<2.0,>=1.0-
>aiohttp<4.0,>=3.6->dataprep) (2.10)
Requirement already satisfied: cycler>=0.10 in
/usr/local/lib/python3.7/dist-packages (from matplotlib-
>wordcloud<2.0,>=1.8->dataprep) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.7/dist-packages (from matplotlib-
>wordcloud<2.0,>=1.8->dataprep) (1.4.2)
Requirement already satisfied: bleach in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (5.0.0)
Requirement already satisfied: defusedxml in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (0.7.1)
Requirement already satisfied: pandocfilters>=1.4.1 in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (1.5.0)
Requirement already satisfied: entrypoints>=0.2.2 in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (0.4)
Requirement already satisfied: testpath in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (0.6.0)
Requirement already satisfied: mistune<2,>=0.8.1 in
/usr/local/lib/python3.7/dist-packages (from nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
```

```
>dataprep) (0.8.4)
Requirement already satisfied: webencodings in
/usr/local/lib/python3.7/dist-packages (from bleach->nbconvert-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets<8.0,>=7.5-
>dataprep) (0.5.1)
Building wheels for collected packages: metaphone, pystache, python-
Levenshtein
  Building wheel for metaphone (setup.py) ... etaphone:
filename=Metaphone-0.6-py3-none-any.whl size=13918
sha256=38592de617618d561a6b02ab9b992020666f67644d6e70f5aa4b0c3745d7f6f
5
  Stored in directory:
/root/.cache/pip/wheels/1d/a8/cb/6f8902aa5457bd71344e00665c230e9c45255
b3f57f2194a0f
  Building wheel for pystache (PEP 517) ... e=pystache-0.6.0-py3-none-
any.whl size=83635
sha256=4da39e99f033d29df47cbfb36b4a22013cee3ee3ff8156ef8b7ac3276e4cc47
8
  Stored in directory:
/root/.cache/pip/wheels/78/87/45/383bd15701a08a94c735e9eaf3ff329965568
4aaca63bbad96
  Building wheel for python-Levenshtein (setup.py) ...
e=python_Levenshtein-0.12.2-cp37-cp37m-linux_x86_64.whl size=149870
sha256=3e8900c6e7b88e7906d6c5528393d3cf7562acc93570e43721d7f7a82f902d4
b
  Stored in directory:
/root/.cache/pip/wheels/05/5f/ca/7c4367734892581bb5ff896f15027a932c551
080b2abd3e00d
Successfully built metaphone pystache python-Levenshtein
Installing collected packages: jinja2, locket, Werkzeug, partd,
multidict, itsdangerous, fsspec, frozenlist, click, yarl, regex, pure-
eval, ply, flask, executing, dask, asynctest, async-timeout,
asttokens, aiosignal, wordcloud, varname, python-stdnum, python-
Levenshtein, python-crfsuite, pystache, pydantic, nltk, metaphone,
jsonpath-ng, flask-cors, aiohttp, dataprep
  Attempting uninstall: jinja2
    Found existing installation: Jinja2 2.11.3
    Uninstalling Jinja2-2.11.3:
      Successfully uninstalled Jinja2-2.11.3
  Attempting uninstall: Werkzeug
    Found existing installation: Werkzeug 1.0.1
    Uninstalling Werkzeug-1.0.1:
      Successfully uninstalled Werkzeug-1.0.1
  Attempting uninstall: itsdangerous
    Found existing installation: itsdangerous 1.1.0
    Uninstalling itsdangerous-1.1.0:
      Successfully uninstalled itsdangerous-1.1.0
  Attempting uninstall: click
    Found existing installation: click 7.1.2
    Uninstalling click-7.1.2:
```

```
      Successfully uninstalled click-7.1.2
  Attempting uninstall: regex
    Found existing installation: regex 2019.12.20
    Uninstalling regex-2019.12.20:
      Successfully uninstalled regex-2019.12.20
  Attempting uninstall: flask
    Found existing installation: Flask 1.1.4
    Uninstalling Flask-1.1.4:
      Successfully uninstalled Flask-1.1.4
  Attempting uninstall: dask
    Found existing installation: dask 2.12.0
    Uninstalling dask-2.12.0:
      Successfully uninstalled dask-2.12.0
  Attempting uninstall: wordcloud
    Found existing installation: wordcloud 1.5.0
    Uninstalling wordcloud-1.5.0:
      Successfully uninstalled wordcloud-1.5.0
  Attempting uninstall: nltk
    Found existing installation: nltk 3.2.5
    Uninstalling nltk-3.2.5:
      Successfully uninstalled nltk-3.2.5
ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
datascience 0.10.6 requires folium==0.2.1, but you have folium 0.8.3
which is incompatible.
Successfully installed Werkzeug-2.1.2 aiohttp-3.8.1 aiosignal-1.2.0
asttokens-2.0.5 async-timeout-4.0.2 asynctest-0.13.0 click-8.1.3 dask-
2021.12.0 dataprep-0.4.3 executing-0.8.3 flask-2.1.2 flask-cors-3.0.10
frozenlist-1.3.0 fsspec-2022.5.0 itsdangerous-2.1.2 jinja2-3.1.2
jsonpath-ng-1.5.3 locket-1.0.0 metaphone-0.6 multidict-6.0.2 nltk-3.7
partd-1.2.0 ply-3.11 pure-eval-0.2.2 pydantic-1.9.1 pystache-0.6.0
python-Levenshtein-0.12.2 python-crfsuite-0.9.8 python-stdnum-1.17
regex-2021.11.10 varname-0.8.3 wordcloud-1.8.1 yarl-1.7.2
```

```
pip install zipfile36
```

```
Collecting zipfile36
  Downloading zipfile36-0.1.3-py3-none-any.whl (20 kB)
Installing collected packages: zipfile36
Successfully installed zipfile36-0.1.3
```

```
# Tratamiento de datos
#
================================================================================
========
import numpy as np
import pandas as pd
# import statsmodels.api as sm
import requests
import zipfile
```

```
from dataprep.eda import create_report

# Gráficos
#
==============================================================================
========
import matplotlib.pyplot as plt
import seaborn as sns

# Preprocesado y modelado
#
==============================================================================
========
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.model_selection import RepeatedKFold
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import ParameterGrid
from sklearn.feature_selection import SelectKBest
from sklearn.metrics import accuracy_score, f1_score
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
from sklearn.metrics import classification_report
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import validation_curve
import multiprocessing

# Configuración warnings
#
==============================================================================
========
import warnings
warnings.filterwarnings('once')
warnings.filterwarnings("ignore",
category=np.VisibleDeprecationWarning)

%matplotlib inline
plt.style.use('fivethirtyeight')
# pd.set_option('display.float_format', lambda x: '%.6f' % x)
# pd.options.display.float_format = '{:.2f}'.format
plt.rcParams['figure.figsize'] = (12, 9)

# tratamiento de datos
#
==============================================================================
========
!pip install researchpy
# !pip install pandas==1.2.2
# !pip install openpyxl==2.6.0
import pandas as pd
```

```
import researchpy as rp
import re
import numpy as np
from datetime import datetime

# Gráficos
#
==============================================================================
========
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud
import missingno as msno
import plotly.express as px

# procesado y modelado
#
==============================================================================
========
from string import punctuation
from nltk import word_tokenize
from nltk.util import ngrams
from nltk.corpus import stopwords
import nltk
nltk.download(['stopwords','punkt','averaged_perceptron_tagger','wordn
et'])
!pip install stop-words
from stop_words import get_stop_words
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from collections import Counter
from sklearn import svm
from scipy.spatial.distance import cosine
from sklearn.model_selection import GridSearchCV
from collections import Counter
from sklearn.model_selection import train_test_split
from sklearn.metrics import
confusion_matrix,accuracy_score,ConfusionMatrixDisplay,classification_
report
from sklearn.linear_model import LogisticRegression

import gensim
from sklearn.cluster import KMeans
import numpy as np
from nltk import word_tokenize, pos_tag
from nltk.probability import FreqDist
from nltk.collocations import *
from gensim.models.phrases import Phrases
from nltk.stem.wordnet import WordNetLemmatizer
```

```
import re

# configuracion warnings
# ======================================================================
========
import warnings
warnings.filterwarnings('ignore')

# pd.set_option('display.float_format', lambda x: '%.3f' % x)
plt.style.use('fivethirtyeight')
plt.rcParams["figure.figsize"] = (12, 9) # (w, h)
# plt.figure(figsize=(12, 9))

Requirement already satisfied: researchpy in
/usr/local/lib/python3.7/dist-packages (0.3.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-
packages (from researchpy) (1.4.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-
packages (from researchpy) (1.21.6)
Requirement already satisfied: statsmodels in
/usr/local/lib/python3.7/dist-packages (from researchpy) (0.10.2)
Requirement already satisfied: patsy in /usr/local/lib/python3.7/dist-
packages (from researchpy) (0.5.2)
Requirement already satisfied: pandas in
/usr/local/lib/python3.7/dist-packages (from researchpy) (1.3.5)
Requirement already satisfied: pytz>=2017.3 in
/usr/local/lib/python3.7/dist-packages (from pandas->researchpy)
(2022.1)
Requirement already satisfied: python-dateutil>=2.7.3 in
/usr/local/lib/python3.7/dist-packages (from pandas->researchpy)
(2.8.2)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3-
>pandas->researchpy) (1.15.0)

[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!

Requirement already satisfied: stop-words in
/usr/local/lib/python3.7/dist-packages (2018.7.23)
```

Datasets

```
# importamos archivos csv
df1 = pd.read_csv('/content/train1.csv')
df2 = pd.read_csv('/content/train2.csv',sep=";")
df1.head()
```

```
  countryName                  eprtrSectorName  \
0     Germany                 Mineral industry
1       Italy                 Mineral industry
2       Spain  Waste and wastewater management
3     Czechia                    Energy sector
4     Finland  Waste and wastewater management

                        EPRTRAnnexIMainActivityLabel  \
0  Installations for the production of cement cli...
1  Installations for the production of cement cli...
2  Landfills (excluding landfills of inert waste ...
3  Thermal power stations and other combustion in...
4                Urban waste-water treatment plants

                                    FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.ni.mu/062217...
1                        IT.CAED/240602021.FACILITY
2                        ES.CAED/001966000.FACILITY
3                    CZ.MZP.U422/CZ34736841.FACILITY
4  http://paikkatiedot.fi/so/1002031/pf/Productio...

                                         facilityName
City  \
0              Holcim (Deutschland) GmbH Werk Höver
Sehnde
1              Stabilimento di Tavernola Bergamasca  TAVERNOLA
BERGAMASCA
2                 COMPLEJO MEDIOAMBIENTAL DE ZURITA    PUERTO DEL
ROSARIO
3                                Elektrárny Prunéřov
Kadaň
4  TAMPEREEN VESI LIIKELAITOS, VIINIKANLAHDEN JÄT...
Tampere

  targetRelease              pollutant  reportingYear  MONTH  ...
CONTINENT  \
0           AIR   Carbon dioxide (CO2)           2015     10  ...
EUROPE
1           AIR  Nitrogen oxides (NOX)           2018      9  ...
EUROPE
2           AIR          Methane (CH4)           2019      2  ...
EUROPE
3           AIR  Nitrogen oxides (NOX)           2012      8  ...
EUROPE
```

```
4           AIR          Methane (CH4)           2018     12  ...
EUROPE

  max_wind_speed  avg_wind_speed  min_wind_speed  max_temp
avg_temp  \
0      15.118767       14.312541       21.419106  2.864895  4.924169

1      19.661550       19.368166       21.756389  5.462839  7.864403

2      12.729453       14.701985       17.103930  1.511201  4.233438

3      11.856417       16.122584       17.537184 10.970301 10.298348

4      17.111930       20.201604       21.536012 11.772039 11.344078


   min_temp  DAY WITH FOGS     REPORTER NAME
CITY ID
0  9.688206              2  Mr. Jacob Ortega
7cdb5e74adcb2ffaa21c1b61395a984f
1 12.023521              1    Ashlee Serrano
cd1dbabbdba230b828c657a9b19a8963
2  8.632193              2      Vincent Kemp
5011e3fa1436d15b34f1287f312fbada
3 15.179215              0        Carol Gray
37a6d7a71c4f7c2469e4f01b70dd90c2
4 16.039004              2        Blake Ford
471fe554e1c62d1b01cc8e4e5076c61a

[5 rows x 21 columns]

df1.iloc[51,:]

countryName
Estonia
eprtrSectorName
Energy sector
EPRTRAnnexIMainActivityLabel    Thermal power stations and other
combustion in...
FacilityInspireID
EE.KAUR.TTR/41.FACILITY
facilityName                    VKG Energia OÜ, Kohtla-Järve Põhja
soojuselekt...
City                                            Järve linnaosa,
Kohtla-Järve linn
targetRelease
AIR
pollutant                                                  Nitrogen
oxides (NOX)
reportingYear
```

```
2019
MONTH
12
DAY
13
CONTINENT
EUROPE
max_wind_speed
16.146999
avg_wind_speed
21.295542
min_wind_speed
27.056893
max_temp
3.947177
avg_temp
5.006776
min_temp
7.720477
DAY WITH FOGS
2
REPORTER NAME
Daniel Rowland
CITY ID
a54c29a5aa61844cc4daa83d20479434
Name: 51, dtype: object
```

```
df1.columns
```

```
Index(['countryName', 'eprtrSectorName',
'EPRTRAnnexIMainActivityLabel',
       'FacilityInspireID', 'facilityName', 'City', 'targetRelease',
       'pollutant', 'reportingYear', 'MONTH', 'DAY', 'CONTINENT',
       'max_wind_speed', 'avg_wind_speed', 'min_wind_speed',
'max_temp',
       'avg_temp', 'min_temp', 'DAY WITH FOGS', 'REPORTER NAME', 'CITY
ID'],
      dtype='object')
```

```
columns=['max_wind_speed','min_wind_speed','max_temp', 'min_temp']
```

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18563 entries, 0 to 18562
Data columns (total 21 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   countryName                    18563 non-null  object
 1   eprtrSectorName                18563 non-null  object
 2   EPRTRAnnexIMainActivityLabel   18563 non-null  object
```

```
 3   FacilityInspireID             18563 non-null  object
 4   facilityName                  18563 non-null  object
 5   City                          18563 non-null  object
 6   targetRelease                 18563 non-null  object
 7   pollutant                     18563 non-null  object
 8   reportingYear                 18563 non-null  int64
 9   MONTH                         18563 non-null  int64
 10  DAY                           18563 non-null  int64
 11  CONTINENT                     18563 non-null  object
 12  max_wind_speed                18563 non-null  float64
 13  avg_wind_speed                18563 non-null  float64
 14  min_wind_speed                18563 non-null  float64
 15  max_temp                      18563 non-null  float64
 16  avg_temp                      18563 non-null  float64
 17  min_temp                      18563 non-null  float64
 18  DAY WITH FOGS                 18563 non-null  int64
 19  REPORTER NAME                 18563 non-null  object
 20  CITY ID                       18563 non-null  object
dtypes: float64(6), int64(4), object(11)
memory usage: 3.0+ MB

df2.head()

  countryName                   eprtrSectorName  \
0     Germany  Waste and wastewater management
1      France                    Energy sector
2      France                    Energy sector
3     Germany  Waste and wastewater management
4     Estonia                    Energy sector

                         EPRTRAnnexIMainActivityLabel  \
0  Installations for the incineration of non-haza...
1  Thermal power stations and other combustion in...
2  Thermal power stations and other combustion in...
3  Landfills (excluding landfills of inert waste ...
4    Installations for gasification and liquefaction

                                    FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.hh/pf.bube-e...
1                              FR.EEA/6288.FACILITY
2                            FR.CAED/12066.FACILITY
3  https://registry.gdi-de.org/id/de.nw.inspire.p...
4                           EE.KAUR.TTR/76.FACILITY

                                         facilityName  \
0  MVR Müllverwertung Rugenberger Damm GmbH & Co. KG
1                           SOCIETE DE COGENERATION
2                                    CPCU ST-OUEN III
3         Deponie Haus Forst REMONDIS GmbH Rheinland
4  Enefit Energiatootmine AS, Auvere põlevkiviõli...
```

```
                              City targetRelease              pollutant
\
0                          Hamburg           AIR  Nitrogen oxides (NOX)

1                           TAVAUX           AIR  Nitrogen oxides (NOX)

2                        SAINT-OUEN           AIR   Carbon dioxide (CO2)

3                           Kerpen           AIR          Methane (CH4)

4  Auvere küla, Narva-Jõesuu linn           AIR   Carbon dioxide (CO2)


   reportingYear  MONTH  ...  CONTINENT max_wind_speed  avg_wind_speed
\
0           2012      4  ...     EUROPE      13.006440       17.328013

1           2007      3  ...     EUROPE      12.601338       16.415961

2           2008     11  ...     EUROPE      17.051488       18.558361

3           2009      2  ...     EUROPE       9.345776       14.584978

4           2016      7  ...     EUROPE      17.122838       18.382589


   min_wind_speed  max_temp  avg_temp  min_temp  DAY WITH FOGS
REPORTER NAME  \
0       22.819874 13.642167 13.524782 15.210716              0  Teresa
Martin
1       20.870744 12.425496 11.640683 14.170232              1  Teresa
Monroe
2       22.729832 10.676109 12.530537 14.036677              1  Brian
Johnson
3       22.153539  1.158088  1.424305  4.768707              1  David
Jackson
4       20.621925  8.620337  8.336314 12.852514              0   Holly
Graves

                             CITY ID
0  35d7df6ed3d93be2927d14acc5f1fc9a
1  8079579bf1d5379ea893be33dbb997d5
2  38fde98415bd374755bb341af3241c4f
3  8b73a54f4cb8ff07dd3e956bfa42b196
4  cffe5169a23e2951963dc5e5da3fcd97

[5 rows x 21 columns]
```

```
df2.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18564 entries, 0 to 18563
Data columns (total 21 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   countryName                     18564 non-null  object
 1   eprtrSectorName                 18564 non-null  object
 2   EPRTRAnnexIMainActivityLabel    18564 non-null  object
 3   FacilityInspireID               18564 non-null  object
 4   facilityName                    18564 non-null  object
 5   City                            18564 non-null  object
 6   targetRelease                   18564 non-null  object
 7   pollutant                       18564 non-null  object
 8   reportingYear                   18564 non-null  int64
 9   MONTH                           18564 non-null  int64
 10  DAY                             18564 non-null  int64
 11  CONTINENT                       18564 non-null  object
 12  max_wind_speed                  18564 non-null  float64
 13  avg_wind_speed                  18564 non-null  float64
 14  min_wind_speed                  18564 non-null  float64
 15  max_temp                        18564 non-null  float64
 16  avg_temp                        18564 non-null  float64
 17  min_temp                        18564 non-null  float64
 18  DAY WITH FOGS                   18564 non-null  int64
 19  REPORTER NAME                   18564 non-null  object
 20  CITY ID                         18564 non-null  object
dtypes: float64(6), int64(4), object(11)
memory usage: 3.0+ MB

# importamos archivos desde API
resp=requests.get('http://schneiderapihack-env.eba-3ais9akk.us-east-
2.elasticbeanstalk.com/first')
data = resp.json()
df3 = pd.json_normalize(data)

resp=requests.get('http://schneiderapihack-env.eba-3ais9akk.us-east-
2.elasticbeanstalk.com/second')
data = resp.json()
df4 = pd.json_normalize(data)

resp=requests.get('http://schneiderapihack-env.eba-3ais9akk.us-east-
2.elasticbeanstalk.com/third')
data = resp.json()
df5 = pd.json_normalize(data)

df3.head()

                                    CITY ID CONTINENT
City DAY  \
```

```
0  47068  4c325d62c064477ef17b4c6e4437e121    EUROPE  Europoort
Rotterdam   2
1  32952  f5e609e7095f91cc8ce9ed6d8e774a0d    EUROPE
RION   3
2  72375  cfab1ba8c67c7c838db98d666f02a132    EUROPE
--   1
3  40702  95b4e51f7b662598134e1eb956407c74    EUROPE
DRIZZONA  17
4  29884  f4433be3b1bfaeeb0633eb65d04b1325    EUROPE
Lünen   6

  DAY WITH FOGS EPRTRAnnexIMainActivityCode  \
0             1                         4(a)
1             2                         3(c)
2            12                         1(c)
3             1                         7(a)
4             0                         5(a)

                        EPRTRAnnexIMainActivityLabel
EPRTRSectorCode  \
0  Chemical installations for the production on a...              4

1  Installations for the production of cement cli...              3

2  Thermal power stations and other combustion in...              1

3  Installations for the intensive rearing of pou...              7

4  Installations for the recovery or disposal of ...              5


                                    FacilityInspireID  ...
countryName  \
0                          NL.RIVM/000019070.FACILITY  ...
Netherlands
1                             EL.CAED/100075.FACILITY  ...
Greece
2         UK.CAED/BEISOffsh-Cormorant-Alpha.FACILITY  ...  United
Kingdom
3                          IT.CAED/260342003.FACILITY  ...
Italy
4  https://registry.gdi-de.org/id/de.nw.inspire.p...  ...
Germany

                                  eprtrSectorName  \
0                               Chemical industry
1                                Mineral industry
2                                   Energy sector
3  Intensive livestock production and aquaculture
```

```
4                 Waste and wastewater management

                                        facilityName
max_temp  \
0                       Indorama Ventures Europe BV
13.256816011792559
1                  TITAN CEMENT S.A. - DREPANO PLANT
4.528859186447803
2                                    Cormorant Alpha
10.669132597893881
3  SOCIETA' AGRICOLA SPARAVALLE DI FERRARI GIUSEP...
7.095681595088376
4                      Biomassekraftwerk Lünen GmbH
9.886774464050356

       max_wind_speed            min_temp      min_wind_speed  \
0  11.019328717116156  14.696895445152332  20.899761591708206
1    14.5123950384412   9.219003402711184  23.243402867192145
2   20.26217117993502  14.715465115792192  23.956529199327292
3   18.28354666681811  13.582024001859644   26.69626609353847
4   13.75940846376134   14.00622637509683  24.768932565830674

               pollutant reportingYear targetRelease
0   Carbon dioxide (CO2)          2020           AIR
1  Nitrogen oxides (NOX)          2019           AIR
2  Nitrogen oxides (NOX)          2009           AIR
3          Methane (CH4)          2014           AIR
4   Carbon dioxide (CO2)          2015           AIR

[5 rows x 24 columns]

df3.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9500 entries, 0 to 9499
Data columns (total 24 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0                                   9500 non-null   object
 1   CITY ID                         9500 non-null   object
 2   CONTINENT                       9500 non-null   object
 3   City                            9500 non-null   object
 4   DAY                             9500 non-null   object
 5   DAY WITH FOGS                   9500 non-null   object
 6   EPRTRAnnexIMainActivityCode     9500 non-null   object
 7   EPRTRAnnexIMainActivityLabel    9500 non-null   object
 8   EPRTRSectorCode                 9500 non-null   object
 9   FacilityInspireID               9500 non-null   object
 10  MONTH                           9500 non-null   object
 11  REPORTER NAME                   9500 non-null   object
```

```
 12  avg_temp                      9500 non-null   object
 13  avg_wind_speed                9500 non-null   object
 14  countryName                   9500 non-null   object
 15  eprtrSectorName               9500 non-null   object
 16  facilityName                  9500 non-null   object
 17  max_temp                      9500 non-null   object
 18  max_wind_speed                9500 non-null   object
 19  min_temp                      9500 non-null   object
 20  min_wind_speed                9500 non-null   object
 21  pollutant                     9500 non-null   object
 22  reportingYear                 9500 non-null   object
 23  targetRelease                 9500 non-null   object
dtypes: object(24)
memory usage: 1.7+ MB

df4.head()

                                   CITY ID CONTINENT        City
DAY  \
0  66841  e8d4668a35daa00b7802cdaac2b33bab    EUROPE     SIGTUNA  18

1  43952  3a9c3ae8ea2e275700947e511afca943    EUROPE      Kaunas   3

2  77831  3d7694a841fc5d426287f208f5e04f61    EUROPE  WORKINGTON  20

3  67548  f8a4753cdbccbd64f0411a207e071aac    EUROPE        SALA   5

4  67772  ce2ddff460389bd5d9f1152dc5679d20    EUROPE     Lugnvik  26


  DAY WITH FOGS EPRTRAnnexIMainActivityCode  \
0             0                         1(c)
1             1                         1(c)
2            10                         6(b)
3             2                         5(d)
4             1                         1(c)

                        EPRTRAnnexIMainActivityLabel
EPRTRSectorCode  \
0  Thermal power stations and other combustion in...               1

1  Thermal power stations and other combustion in...               1

2  Industrial plants for the production of paper ...               6

3  Landfills (excluding landfills of inert waste ...               5

4  Thermal power stations and other combustion in...               1
```

```
             FacilityInspireID  ...     countryName  \
0    SE.CAED/10014262.Facility  ...          Sweden
1            LT.EEA/3.FACILITY  ...       Lithuania
2  UK.CAED/EW_EA-1427.FACILITY  ...  United Kingdom
3    SE.CAED/10021261.Facility  ...          Sweden
4    SE.CAED/10023054.Facility  ...          Sweden

                           eprtrSectorName              facilityName
\
0                            Energy sector              BRISTAVERKET

1                            Energy sector           Kauno elektrine

2  Paper and wood production and processing    Workington Board Mill

3          Waste and wastewater management  Isätra avfallsanläggning

4                            Energy sector            Lugnviksverket


             max_temp      max_wind_speed            min_temp  \
0  12.910353536727893  19.591151655794587   17.82215864414945
1  1.5666540044051371    9.00653292177714   4.398769749133166
2  10.241998731492398  14.076642626320474  13.447854103400092
3  12.684850988473203  18.924086287110967    17.3771676621797
4   7.826782397332164   18.59685701250554  11.731918192932916

       min_wind_speed              pollutant reportingYear
targetRelease
0  24.496400860946892   Carbon dioxide (CO2)          2010
AIR
1   20.07497599357037   Carbon dioxide (CO2)          2015
AIR
2   25.23915141001326  Nitrogen oxides (NOX)          2015
AIR
3  25.603714335587156          Methane (CH4)          2013
AIR
4  25.811989344026774   Carbon dioxide (CO2)          2014
AIR

[5 rows x 24 columns]

df4.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9500 entries, 0 to 9499
Data columns (total 24 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
```

```
 0                                  9500 non-null   object
 1   CITY ID                        9500 non-null   object
 2   CONTINENT                      9500 non-null   object
 3   City                           9500 non-null   object
 4   DAY                            9500 non-null   object
 5   DAY WITH FOGS                  9500 non-null   object
 6   EPRTRAnnexIMainActivityCode    9500 non-null   object
 7   EPRTRAnnexIMainActivityLabel   9500 non-null   object
 8   EPRTRSectorCode                9500 non-null   object
 9   FacilityInspireID              9500 non-null   object
 10  MONTH                          9500 non-null   object
 11  REPORTER NAME                  9500 non-null   object
 12  avg_temp                       9500 non-null   object
 13  avg_wind_speed                 9500 non-null   object
 14  countryName                    9500 non-null   object
 15  eprtrSectorName                9500 non-null   object
 16  facilityName                   9500 non-null   object
 17  max_temp                       9500 non-null   object
 18  max_wind_speed                 9500 non-null   object
 19  min_temp                       9500 non-null   object
 20  min_wind_speed                 9500 non-null   object
 21  pollutant                      9500 non-null   object
 22  reportingYear                  9500 non-null   object
 23  targetRelease                  9500 non-null   object
dtypes: object(24)
memory usage: 1.7+ MB
```

```
df5.head()
```

		CITY ID	CONTINENT	City	DAY	\
0	41175	7951666b94e0f0891e0c66b2381fca55	EUROPE	TORINO	24	
1	49299	33c89df2492e8d3efda719c849b530ea	EUROPE	Łódź	4	
2	34879	4a8b9d98f65af3a29bbf298d8536c142	EUROPE	Tipperary	18	
3	16905	e38f45f4d669e9f69fa97bfe049ceed6	EUROPE	REIMS	27	
4	75675	fb960490e42477cbfdcd6bab1793f31e	EUROPE	Hexham	28	

	DAY WITH FOGS	EPRTRAnnexIMainActivityCode	\
0	1	1(c)	
1	0	1(c)	
2	0	7(a)(ii)	
3	0	3(e)	
4	2	6(b)	

	EPRTRAnnexIMainActivityLabel	EPRTRSectorCode	\
0	Thermal power stations and other combustion in...	1	
1	Thermal power stations and other combustion in...	1	
2	Installations for the intensive rearing of pig...	7	

```
3  Installations for the manufacture of glass, in...               3

4  Industrial plants for the production of paper ...               6


           FacilityInspireID  ...     countryName  \
0  IT.CAED/101511001.FACILITY  ...           Italy
1        PL.EEA/1321.FACILITY  ...          Poland
2      IE.CAED/P0489.FACILITY  ...         Ireland
3       FR.CAED/3453.FACILITY  ...          France
4   UK.LAED/E375_434.FACILITY  ...  United Kingdom

                                  eprtrSectorName  \
0                                   Energy sector
1                                   Energy sector
2  Intensive livestock production and aquaculture
3                                Mineral industry
4        Paper and wood production and processing

                                       facilityName
max_temp  \
0                                Iren Energia S.p.A.
7.367005114195391
1            Dalkia Łódź S.A. Elektrociepłownia nr 3
12.764269296496483
2  Glen of Aherlow Pig Producers Co-Op Society Li...
9.278434915324674
3                       OI MANUFACTURING FRANCE REIMS
12.132209572093492
4                                 EGGER (UK) LIMITED
3.376109181863636

       max_wind_speed            min_temp      min_wind_speed  \
0  15.892428799653784  12.571580007378364  20.467596329992563
1  14.168741633729171   19.08124839205317   25.43951546576332
2  17.849097696179154  15.912359661973477  25.603903622689426
3  10.348395479695537  16.897353302304076   18.13168786166006
4  13.803120487487323   6.435357697393915   22.21506578438592

               pollutant reportingYear targetRelease
0  Nitrogen oxides (NOX)          2015           AIR
1   Carbon dioxide (CO2)          2011           AIR
2          Methane (CH4)          2011           AIR
3  Nitrogen oxides (NOX)          2014           AIR
4  Nitrogen oxides (NOX)          2012           AIR

[5 rows x 24 columns]
```

```
df5.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9501 entries, 0 to 9500
Data columns (total 24 columns):
 #   Column                          Non-Null Count  Dtype 
---  ------                          --------------  ----- 
 0                                   9501 non-null   object
 1   CITY ID                         9501 non-null   object
 2   CONTINENT                       9501 non-null   object
 3   City                            9501 non-null   object
 4   DAY                             9501 non-null   object
 5   DAY WITH FOGS                   9501 non-null   object
 6   EPRTRAnnexIMainActivityCode     9501 non-null   object
 7   EPRTRAnnexIMainActivityLabel    9501 non-null   object
 8   EPRTRSectorCode                 9501 non-null   object
 9   FacilityInspireID               9501 non-null   object
 10  MONTH                           9501 non-null   object
 11  REPORTER NAME                   9501 non-null   object
 12  avg_temp                        9501 non-null   object
 13  avg_wind_speed                  9501 non-null   object
 14  countryName                     9501 non-null   object
 15  eprtrSectorName                 9501 non-null   object
 16  facilityName                    9501 non-null   object
 17  max_temp                        9501 non-null   object
 18  max_wind_speed                  9501 non-null   object
 19  min_temp                        9501 non-null   object
 20  min_wind_speed                  9501 non-null   object
 21  pollutant                       9501 non-null   object
 22  reportingYear                   9501 non-null   object
 23  targetRelease                   9501 non-null   object
dtypes: object(24)
memory usage: 1.7+ MB
```

```
# se extraen el zip
!python -m zipfile -e /content/train6.zip /content/
```

```
!sudo apt install build-essential libpoppler-cpp-dev pkg-config
python3-dev
!pip install pdftotext
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.4ubuntu1).
pkg-config is already the newest version (0.29.1-0ubuntu2).
python3-dev is already the newest version (3.6.7-1~18.04).
python3-dev set to manually installed.
The following package was automatically installed and is no longer
required:
  libnvidia-common-460
```

```
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libpoppler-cpp0v5
The following NEW packages will be installed:
  libpoppler-cpp-dev libpoppler-cpp0v5
0 upgraded, 2 newly installed, 0 to remove and 42 not upgraded.
Need to get 36.7 kB of archives.
After this operation, 188 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64
libpoppler-cpp0v5 amd64 0.62.0-2ubuntu2.12 [28.0 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64
libpoppler-cpp-dev amd64 0.62.0-2ubuntu2.12 [8,676 B]
Fetched 36.7 kB in 1s (46.3 kB/s)
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog
based frontend cannot be used. at
/usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76, <> line 2.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (This frontend requires a controlling tty.)
debconf: falling back to frontend: Teletype
dpkg-preconfigure: unable to re-open stdin:
Selecting previously unselected package libpoppler-cpp0v5:amd64.
(Reading database ... 155629 files and directories currently
installed.)
Preparing to unpack .../libpoppler-cpp0v5_0.62.0-2ubuntu2.12_amd64.deb
...
Unpacking libpoppler-cpp0v5:amd64 (0.62.0-2ubuntu2.12) ...
Selecting previously unselected package libpoppler-cpp-dev:amd64.
Preparing to unpack .../libpoppler-cpp-dev_0.62.0-
2ubuntu2.12_amd64.deb ...
Unpacking libpoppler-cpp-dev:amd64 (0.62.0-2ubuntu2.12) ...
Setting up libpoppler-cpp0v5:amd64 (0.62.0-2ubuntu2.12) ...
Setting up libpoppler-cpp-dev:amd64 (0.62.0-2ubuntu2.12) ...
Processing triggers for libc-bin (2.27-3ubuntu1.3) ...
/sbin/ldconfig.real:
/usr/local/lib/python3.7/dist-packages/ideep4py/lib/libmkldnn.so.0 is
not a symbolic link

Collecting pdftotext
  Downloading pdftotext-2.2.2.tar.gz (113 kB)
e=pdftotext-2.2.2-cp37-cp37m-linux_x86_64.whl size=54922
sha256=21a875a5cb7a63691827f682c3e080d8b9693dcb635e2ac25f3b38d710804ee
5
  Stored in directory:
/root/.cache/pip/wheels/98/19/8e/e8648026db8b7ef3324ad9afa1f7c9109a7e7
509846f693ed9
Successfully built pdftotext
Installing collected packages: pdftotext
Successfully installed pdftotext-2.2.2
```

```
/usr/local/lib/python3.7/dist-packages/google/colab/_pip.py:87:
ResourceWarning: unclosed file <_io.TextIOWrapper
name='/usr/local/lib/python3.7/dist-packages/pdftotext-2.2.2.dist-
info/top_level.txt' mode='r' encoding='UTF-8'>
  for line in open(toplevel):
```

```
pip install pdfminer
```

```
Collecting pdfminer
  Downloading pdfminer-20191125.tar.gz (4.2 MB)
e
  Downloading pycryptodome-3.14.1-cp35-abi3-manylinux2010_x86_64.whl
(2.0 MB)
iner
  Building wheel for pdfminer (setup.py) ... iner: filename=pdfminer-
20191125-py3-none-any.whl size=6140079
sha256=31274a3ff984cd0bb909cdf195177d029d45fc57b3f2826180fa0d43ac24a88
9
  Stored in directory:
/root/.cache/pip/wheels/e3/5e/f4/d210b46e9e4a28229ea070ed5b3efa92c3c29
d1a7918dd4b97
Successfully built pdfminer
Installing collected packages: pycryptodome, pdfminer
Successfully installed pdfminer-20191125 pycryptodome-3.14.1
```

```
/usr/local/lib/python3.7/dist-packages/google/colab/_pip.py:87:
ResourceWarning: unclosed file <_io.TextIOWrapper
name='/usr/local/lib/python3.7/dist-packages/pdfminer-20191125.dist-
info/top_level.txt' mode='r' encoding='UTF-8'>
  for line in open(toplevel):
/usr/local/lib/python3.7/dist-packages/google/colab/_pip.py:87:
ResourceWarning: unclosed file <_io.TextIOWrapper
name='/usr/local/lib/python3.7/dist-packages/pycryptodome-3.14.1.dist-
info/top_level.txt' mode='r' encoding='UTF-8'>
  for line in open(toplevel):
```

```
import tempfile
from io import StringIO
import pandas as pd
import numpy as np
import pdftotext
from pdfminer.converter import TextConverter
from pdfminer.layout import LAParams
from pdfminer.pdfdocument import PDFDocument
from pdfminer.pdfinterp import PDFResourceManager, PDFPageInterpreter
from pdfminer.pdfpage import PDFPage
from pdfminer.pdfparser import PDFParser
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
from PIL import Image, ImageFont, ImageDraw
```

```
# funcion para leer los pdf
def readPdf(PDF):
    pdfminer_string = StringIO()
    # PDF=Path("/content/train6/pdfs-1.pdf")

    with open(PDF, "rb") as in_file:
        parser = PDFParser(in_file)
        doc = PDFDocument(parser)
        rsrcmgr = PDFResourceManager()
        device = TextConverter(rsrcmgr,
                               pdfminer_string,
                               laparams=LAParams())
        interpreter = PDFPageInterpreter(rsrcmgr, device)
        for page in PDFPage.create_pages(doc):
            interpreter.process_page(page)
    pdfminer_lines = pdfminer_string.getvalue().splitlines()
    pdfminer_lines = [ln for ln in pdfminer_lines if ln]


    with open(PDF, 'rb') as file:
        pdftotext_string = pdftotext.PDF(file)
    pdftotext_lines = ("\n\n".join(pdftotext_string).splitlines())
    pdftotext_lines = [ln for ln in pdftotext_lines if ln]
    return pdftotext_lines

# funcion para mapear el pdf
def getdata(pdftotext_lines):
    dic={}
    for idx, line in enumerate(pdftotext_lines[1:]):
      line_='|'.join(i for i in line.split('  ') if i !=
'').replace(':|',':')

      if idx > 8:
        line_='|'.join(i for i in line.split(' ') if i !=
'').replace(':|',':')

      # print(line_)
      if line_.find('|') != -1:
        # print(line_)
        if line_.find(':') != -1:
          line_r=line_.split('|')
          for l in line_r:
            if l.find(':') != -1:
              l=l.split(':')
              # print(l)
              dic[l[0].strip()]=l[1].strip()
        else:
          line_r=line_.split(' ')
          for l in line_r:
            l=l.split('|')
```

```
            # print(l)
            dic[l[0]]=l[1]
      else:
        # print(line_)
        if line_.find(':') != -1:
          line_r=line_.split('|')
          # print(line_r)
          for l in line_r:
            l=l.split(':')
            # print(l)
            dic[l[0].strip()]=l[1].strip()
        else:
          line_r=line_.split('  ')
          # print(line_r)
          if len(line_r) > 1:
            for l in line_r:
              l=l.split('|')
              # print(l)
              dic[l[0]]=l[1]
    return dic

import os
from pathlib import Path
root=Path("/content/train6")
# cargamos los pdf en una lista
df_dic=[]
for idx,name in enumerate(sorted(os.listdir(root))):
  path_pdf=list(sorted(root.glob("*pdf*")))[idx]
  # print(idx)
  pdf=readPdf(path_pdf)
  # print(pdf)
  df_dic.append(getdata(pdf))

# transformamos los pdf en dataframe
df6=pd.DataFrame(df_dic)
df6.head()
```

```
      nº                                    FACILITY NAME  \
0  81597  Millerhill Recycling & Energy Recovery Centre
1  81516                            Fife Ethylene Plant
2  81516                            Fife Ethylene Plant
3  81517                            Fife Ethylene Plant
4  81518                            Alloa Glass Factory

            FacilityInspireID         COUNTRY CONTINENT
CITY  \
0  UK.SEPA/200002651.Facility  United Kingdom    EUROPE  Millerhill,
Dalkeith
1  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
Cowdenbeath
2  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
```

```
Cowdenbeath
3  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
Cowdenbeath
4  UK.SEPA/200000073.Facility  United Kingdom    EUROPE
Alloa

  EPRTRSectorCode                  eprtrSectorName MainActivityCode  \
0               5  Waste and wastewater management             5(b)
1               1                    Energy sector             1(c)
2               1                    Energy sector             1(c)
3               1                    Energy sector             1(c)
4               3                 Mineral industry             3(e)

  targetRealase  ... METEOROLOGICAL max_wind_speed min_wind_speed  \
0           AIR  ...     CONDITIONS       1,79E+15        2,2E+16
1           AIR  ...     CONDITIONS       1,52E+16       2,06E+15
2           AIR  ...     CONDITIONS       1,52E+16       2,06E+15
3           AIR  ...     CONDITIONS       1,16E+16       2,18E+16
4           AIR  ...     CONDITIONS       1,11E+16       2,03E+16

  avg_wind_speed  max_temp  min_temp  avg_temp FOG     NAME  \
0       2,04E+15  1,51E+16  1,82E+15  1,71E+16  10  William
1       1,46E+16  9,61E+15  1,33E+16  8,69E+15  19    Shawn
2       1,46E+16  9,61E+15  1,33E+16  8,69E+15  19    Shawn
3       1,65E+16  8,03E+15  1,04E+16  8,94E+15  10    Aaron
4        1,6E+16  -1,9E+16  4,07E+16  1,33E+15   4    Vicki

                          CITY_ID
0  c662b4b4d859a9c224b5ac0acf221748
1  3c563ab0d76fc84128574b5da82f769a
2  3c563ab0d76fc84128574b5da82f769a
3  3c563ab0d76fc84128574b5da82f769a
4  2cc8f54182c37b8907f534011ea01e6f

[5 rows x 25 columns]

df6.head()

      nº                                  FACILITY NAME  \
0  81597  Millerhill Recycling & Energy Recovery Centre
1  81516                           Fife Ethylene Plant
2  81516                           Fife Ethylene Plant
3  81517                           Fife Ethylene Plant
4  81518                           Alloa Glass Factory

            FacilityInspireID         COUNTRY CONTINENT
CITY  \
0  UK.SEPA/200002651.Facility  United Kingdom    EUROPE  Millerhill,
Dalkeith
1  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
```

```
Cowdenbeath
2  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
Cowdenbeath
3  UK.SEPA/200000061.Facility  United Kingdom    EUROPE
Cowdenbeath
4  UK.SEPA/200000073.Facility  United Kingdom    EUROPE
Alloa

  EPRTRSectorCode                  eprtrSectorName MainActivityCode  \
0               5  Waste and wastewater management             5(b)
1               1                    Energy sector             1(c)
2               1                    Energy sector             1(c)
3               1                    Energy sector             1(c)
4               3                 Mineral industry             3(e)

  targetRealase  ... METEOROLOGICAL max_wind_speed min_wind_speed  \
0           AIR  ...     CONDITIONS       1.79E+15        2,2E+16
1           AIR  ...     CONDITIONS       1.52E+16       2,06E+15
2           AIR  ...     CONDITIONS       1.52E+16       2,06E+15
3           AIR  ...     CONDITIONS       1.16E+16       2,18E+16
4           AIR  ...     CONDITIONS       1.11E+16       2,03E+16

  avg_wind_speed  max_temp  min_temp  avg_temp FOG     NAME  \
0       2,04E+15  1,51E+16  1,82E+15  1,71E+16  10  William
1       1,46E+16  9,61E+15  1,33E+16  8,69E+15  19    Shawn
2       1,46E+16  9,61E+15  1,33E+16  8,69E+15  19    Shawn
3       1,65E+16  8,03E+15  1,04E+16  8,94E+15  10    Aaron
4        1,6E+16  -1,9E+16  4,07E+16  1,33E+15   4    Vicki

                           CITY_ID
0  c662b4b4d859a9c224b5ac0acf221748
1  3c563ab0d76fc84128574b5da82f769a
2  3c563ab0d76fc84128574b5da82f769a
3  3c563ab0d76fc84128574b5da82f769a
4  2cc8f54182c37b8907f534011ea01e6f

[5 rows x 25 columns]
```

```
# df6.max_wind_speed.astype(float)
df6['max_wind_speed']=df6.max_wind_speed.str.replace(',','.')
df6.max_wind_speed.astype(float)
```

```
0     1790000000000000.000000
1    15200000000000000.000000
2    15200000000000000.000000
3    11600000000000000.000000
4    11100000000000000.000000
               ...
77   19300000000000000.000000
78   15900000000000000.000000
```

```
79   12800000000000000.000000
80   13500000000000000.000000
81    2090000000000000.000000
Name: max_wind_speed, Length: 82, dtype: float64
```

```
float(1.79E+15)
```

```
1790000000000000.0
```

```
df6.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 82 entries, 0 to 81
Data columns (total 25 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   nº                  82 non-null     object
 1   FACILITY NAME       82 non-null     object
 2   FacilityInspireID   82 non-null     object
 3   COUNTRY             82 non-null     object
 4   CONTINENT           82 non-null     object
 5   CITY                82 non-null     object
 6   EPRTRSectorCode     82 non-null     object
 7   eprtrSectorName     82 non-null     object
 8   MainActivityCode    82 non-null     object
 9   targetRealase       82 non-null     object
 10  pollutant           82 non-null     object
 11  emissions           82 non-null     object
 12  DAY                 82 non-null     object
 13  MONTH               82 non-null     object
 14  YEAR                82 non-null     object
 15  METEOROLOGICAL      82 non-null     object
 16  max_wind_speed      82 non-null     object
 17  min_wind_speed      82 non-null     object
 18  avg_wind_speed      82 non-null     object
 19  max_temp            82 non-null     object
 20  min_temp            82 non-null     object
 21  avg_temp            82 non-null     object
 22  FOG                 82 non-null     object
 23  NAME                82 non-null     object
 24  CITY_ID             82 non-null     object
dtypes: object(25)
memory usage: 16.1+ KB
```

Analisis exploratorio de datos (EDA)

```
report1 = create_report(df1)
report1
```

```
report2 = create_report(df2)
report2
```

```
report3 = create_report(df3)
report3
```

```
report4 = create_report(df4)
report4
```

```
report5 = create_report(df5)
report5
```

```
report6 = create_report(df6)
report6
```

```
# concatenamos df1 y df2
df=pd.concat([df1,df2], axis=0, ignore_index=True)
df.head()
```

```
  countryName                 eprtrSectorName  \
0     Germany                Mineral industry
1       Italy                Mineral industry
2       Spain  Waste and wastewater management
3     Czechia                   Energy sector
4     Finland  Waste and wastewater management

                       EPRTRAnnexIMainActivityLabel  \
0  Installations for the production of cement cli...
1  Installations for the production of cement cli...
2  Landfills (excluding landfills of inert waste ...
3  Thermal power stations and other combustion in...
4                Urban waste-water treatment plants
```

```
                                  FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.ni.mu/062217...
1                        IT.CAED/240602021.FACILITY
2                        ES.CAED/001966000.FACILITY
3                    CZ.MZP.U422/CZ34736841.FACILITY
4  http://paikkatiedot.fi/so/1002031/pf/Productio...

                                       facilityName
City  \
0              Holcim (Deutschland) GmbH Werk Höver
Sehnde
1              Stabilimento di Tavernola Bergamasca  TAVERNOLA
BERGAMASCA
2                 COMPLEJO MEDIOAMBIENTAL DE ZURITA    PUERTO DEL
ROSARIO
3                               Elektrárny Prunéřov
Kadaň
4  TAMPEREEN VESI LIIKELAITOS, VIINIKANLAHDEN JÄT...
Tampere

  targetRelease              pollutant  reportingYear  MONTH  ...
CONTINENT  \
0           AIR   Carbon dioxide (CO2)           2015     10  ...
EUROPE
1           AIR  Nitrogen oxides (NOX)           2018      9  ...
EUROPE
2           AIR          Methane (CH4)           2019      2  ...
EUROPE
3           AIR  Nitrogen oxides (NOX)           2012      8  ...
EUROPE
4           AIR          Methane (CH4)           2018     12  ...
EUROPE

  max_wind_speed  avg_wind_speed  min_wind_speed  max_temp
avg_temp  \
0         15.119          14.313          21.419     2.865     4.924

1         19.662          19.368          21.756     5.463     7.864

2         12.729          14.702          17.104     1.511     4.233

3         11.856          16.123          17.537    10.970    10.298

4         17.112          20.202          21.536    11.772    11.344


   min_temp  DAY WITH FOGS     REPORTER NAME
CITY ID
```

```
0     9.688              2  Mr. Jacob Ortega
7cdb5e74adcb2ffaa21c1b61395a984f
1    12.024              1    Ashlee Serrano
cd1dbabbdba230b828c657a9b19a8963
2     8.632              2      Vincent Kemp
5011e3fa1436d15b34f1287f312fbada
3    15.179              0        Carol Gray
37a6d7a71c4f7c2469e4f01b70dd90c2
4    16.039              2        Blake Ford
471fe554e1c62d1b01cc8e4e5076c61a

[5 rows x 21 columns]
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37127 entries, 0 to 37126
Data columns (total 21 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   countryName                     37127 non-null  object
 1   eprtrSectorName                 37127 non-null  object
 2   EPRTRAnnexIMainActivityLabel  37127 non-null  object
 3   FacilityInspireID               37127 non-null  object
 4   facilityName                    37127 non-null  object
 5   City                            37127 non-null  object
 6   targetRelease                   37127 non-null  object
 7   pollutant                       37127 non-null  object
 8   reportingYear                   37127 non-null  int64
 9   MONTH                           37127 non-null  int64
 10  DAY                             37127 non-null  int64
 11  CONTINENT                       37127 non-null  object
 12  max_wind_speed                  37127 non-null  float64
 13  avg_wind_speed                  37127 non-null  float64
 14  min_wind_speed                  37127 non-null  float64
 15  max_temp                        37127 non-null  float64
 16  avg_temp                        37127 non-null  float64
 17  min_temp                        37127 non-null  float64
 18  DAY WITH FOGS                   37127 non-null  int64
 19  REPORTER NAME                   37127 non-null  object
 20  CITY ID                         37127 non-null  object
dtypes: float64(6), int64(4), object(11)
memory usage: 5.9+ MB
```

```
# concatenamos df3 df4 y df5
df_r=pd.concat([df3,df4,df5], axis=0, ignore_index=True)
df_r.head()
```

```
                                  CITY ID CONTINENT
City DAY  \
0  47068  4c325d62c064477ef17b4c6e4437e121    EUROPE  Europoort
```

```
Rotterdam   2
1  32952  f5e609e7095f91cc8ce9ed6d8e774a0d    EUROPE
RION   3
2  72375  cfab1ba8c67c7c838db98d666f02a132    EUROPE
--   1
3  40702  95b4e51f7b662598134e1eb956407c74    EUROPE
DRIZZONA  17
4  29884  f4433be3b1bfaeeb0633eb65d04b1325    EUROPE
Lünen   6

  DAY WITH FOGS EPRTRAnnexIMainActivityCode  \
0             1                         4(a)
1             2                         3(c)
2            12                         1(c)
3             1                         7(a)
4             0                         5(a)

                        EPRTRAnnexIMainActivityLabel
EPRTRSectorCode  \
0  Chemical installations for the production on a...              4

1  Installations for the production of cement cli...              3

2  Thermal power stations and other combustion in...              1

3  Installations for the intensive rearing of pou...              7

4  Installations for the recovery or disposal of ...              5


                                    FacilityInspireID  ...
countryName  \
0                          NL.RIVM/000019070.FACILITY  ...
Netherlands
1                             EL.CAED/100075.FACILITY  ...
Greece
2         UK.CAED/BEISOffsh-Cormorant-Alpha.FACILITY  ...  United
Kingdom
3                          IT.CAED/260342003.FACILITY  ...
Italy
4  https://registry.gdi-de.org/id/de.nw.inspire.p...  ...
Germany

                                   eprtrSectorName  \
0                                Chemical industry
1                                 Mineral industry
2                                    Energy sector
3  Intensive livestock production and aquaculture
4                  Waste and wastewater management
```

```
                                        facilityName
max_temp  \
0                        Indorama Ventures Europe BV
13.256816011792559
1                   TITAN CEMENT S.A. - DREPANO PLANT
4.528859186447803
2                                     Cormorant Alpha
10.669132597893881
3  SOCIETA' AGRICOLA SPARAVALLE DI FERRARI GIUSEP...
7.095681595088376
4                        Biomassekraftwerk Lünen GmbH
9.886774464050356

       max_wind_speed            min_temp      min_wind_speed  \
0  11.019328717116156  14.696895445152332  20.899761591708206
1    14.5123950384412   9.219003402711184  23.243402867192145
2   20.26217117993502  14.715465115792192  23.956529199327292
3   18.28354666681811  13.582024001859644   26.69626609353847
4   13.75940846376134   14.00622637509683  24.768932565830674

               pollutant reportingYear targetRelease
0   Carbon dioxide (CO2)          2020           AIR
1  Nitrogen oxides (NOX)          2019           AIR
2  Nitrogen oxides (NOX)          2009           AIR
3          Methane (CH4)          2014           AIR
4   Carbon dioxide (CO2)          2015           AIR

[5 rows x 24 columns]

df_r.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28501 entries, 0 to 28500
Data columns (total 24 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0                                   28501 non-null  object
 1   CITY ID                         28501 non-null  object
 2   CONTINENT                       28501 non-null  object
 3   City                            28501 non-null  object
 4   DAY                             28501 non-null  object
 5   DAY WITH FOGS                   28501 non-null  object
 6   EPRTRAnnexIMainActivityCode     28501 non-null  object
 7   EPRTRAnnexIMainActivityLabel    28501 non-null  object
 8   EPRTRSectorCode                 28501 non-null  object
 9   FacilityInspireID               28501 non-null  object
 10  MONTH                           28501 non-null  object
 11  REPORTER NAME                   28501 non-null  object
 12  avg_temp                        28501 non-null  object
```

```
 13  avg_wind_speed                28501 non-null  object
 14  countryName                   28501 non-null  object
 15  eprtrSectorName               28501 non-null  object
 16  facilityName                  28501 non-null  object
 17  max_temp                      28501 non-null  object
 18  max_wind_speed                28501 non-null  object
 19  min_temp                      28501 non-null  object
 20  min_wind_speed                28501 non-null  object
 21  pollutant                     28501 non-null  object
 22  reportingYear                 28501 non-null  object
 23  targetRelease                 28501 non-null  object
dtypes: object(24)
memory usage: 5.2+ MB
```

realmente el tipo de contaminante emitido por la empresa es independiente del componente temporal (año, mes dia) o de la velocidad del viento o la temperatura. No son una causa, no influyen en la contaminacion y no es significativo en la prediccion de la variable pollution por lo que podemos ignorarlas

```
df.head(3)
```

```
  countryName                  eprtrSectorName  \
0     Germany                Mineral industry
1       Italy                Mineral industry
2       Spain  Waste and wastewater management

                        EPRTRAnnexIMainActivityLabel  \
0  Installations for the production of cement cli...
1  Installations for the production of cement cli...
2  Landfills (excluding landfills of inert waste ...

                                  FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.ni.mu/062217...
1                        IT.CAED/240602021.FACILITY
2                        ES.CAED/001966000.FACILITY

                           facilityName                  City
targetRelease  \
0  Holcim (Deutschland) GmbH Werk Höver               Sehnde
AIR
1  Stabilimento di Tavernola Bergamasca  TAVERNOLA BERGAMASCA
AIR
2     COMPLEJO MEDIOAMBIENTAL DE ZURITA    PUERTO DEL ROSARIO
AIR

               pollutant  reportingYear  MONTH  ...  CONTINENT
max_wind_speed  \
0   Carbon dioxide (CO2)           2015     10  ...     EUROPE
15.119
1  Nitrogen oxides (NOX)           2018      9  ...     EUROPE
```

```
19.662
2          Methane (CH4)           2019      2  ...     EUROPE
12.729

   avg_wind_speed  min_wind_speed  max_temp  avg_temp  min_temp  \
0          14.313          21.419     2.865     4.924     9.688
1          19.368          21.756     5.463     7.864    12.024
2          14.702          17.104     1.511     4.233     8.632

   DAY WITH FOGS     REPORTER NAME                          CITY ID
0              2  Mr. Jacob Ortega  7cdb5e74adcb2ffaa21c1b61395a984f
1              1    Ashlee Serrano  cd1dbabbdba230b828c657a9b19a8963
2              2      Vincent Kemp  5011e3fa1436d15b34f1287f312fbada

[3 rows x 21 columns]
```

```
# añadimos las variables significativas para el modelo NLP
df['descripcion'] = df['eprtrSectorName']+' .
'+df['EPRTRAnnexIMainActivityLabel']+' . '+df['FacilityInspireID']+' .
'+df['facilityName']+' . '+df['City']
df['pollutant_code'] = df.pollutant.replace(['Carbon dioxide (CO2)',
'Nitrogen oxides (NOX)', 'Methane (CH4)'], [1,0,2])
df.head()
```

```
  countryName                 eprtrSectorName  \
0     Germany                Mineral industry
1       Italy                Mineral industry
2       Spain  Waste and wastewater management
3     Czechia                  Energy sector
4     Finland  Waste and wastewater management

                       EPRTRAnnexIMainActivityLabel  \
0  Installations for the production of cement cli...
1  Installations for the production of cement cli...
2  Landfills (excluding landfills of inert waste ...
3  Thermal power stations and other combustion in...
4               Urban waste-water treatment plants

                                  FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.ni.mu/062217...
1                      IT.CAED/240602021.FACILITY
2                      ES.CAED/001966000.FACILITY
3                  CZ.MZP.U422/CZ34736841.FACILITY
4  http://paikkatiedot.fi/so/1002031/pf/Productio...

                                      facilityName
City  \
0             Holcim (Deutschland) GmbH Werk Höver
Sehnde
1             Stabilimento di Tavernola Bergamasca  TAVERNOLA
```

```
BERGAMASCA
2                  COMPLEJO MEDIOAMBIENTAL DE ZURITA    PUERTO DEL
ROSARIO
3                                Elektrárny Prunéřov
Kadaň
4  TAMPEREEN VESI LIIKELAITOS, VIINIKANLAHDEN JÄT...
Tampere

  targetRelease              pollutant  reportingYear  MONTH  ...  \
0           AIR   Carbon dioxide (CO2)           2015     10  ...
1           AIR  Nitrogen oxides (NOX)           2018      9  ...
2           AIR          Methane (CH4)           2019      2  ...
3           AIR  Nitrogen oxides (NOX)           2012      8  ...
4           AIR          Methane (CH4)           2018     12  ...

   avg_wind_speed min_wind_speed  max_temp  avg_temp  min_temp  DAY
WITH FOGS  \
0          14.313         21.419     2.865     4.924     9.688
2
1          19.368         21.756     5.463     7.864    12.024
1
2          14.702         17.104     1.511     4.233     8.632
2
3          16.123         17.537    10.970    10.298    15.179
0
4          20.202         21.536    11.772    11.344    16.039
2

      REPORTER NAME                            CITY ID  \
0  Mr. Jacob Ortega  7cdb5e74adcb2ffaa21c1b61395a984f
1    Ashlee Serrano  cd1dbabbdba230b828c657a9b19a8963
2      Vincent Kemp  5011e3fa1436d15b34f1287f312fbada
3        Carol Gray  37a6d7a71c4f7c2469e4f01b70dd90c2
4        Blake Ford  471fe554e1c62d1b01cc8e4e5076c61a

                                        descripcion pollutant_code
0  Mineral industry . Installations for the produ...              1
1  Mineral industry . Installations for the produ...              0
2  Waste and wastewater management . Landfills (e...              2
3  Energy sector . Thermal power stations and oth...              0
4  Waste and wastewater management . Urban waste-...              2

[5 rows x 23 columns]


df_r['descripcion'] = df_r['eprtrSectorName']+' .
'+df_r['EPRTRAnnexIMainActivityLabel']+' . '+df_r['FacilityInspireID']
+' . '+df_r['facilityName']+' . '+df_r['EPRTRAnnexIMainActivityCode']
+' . '+df_r['City']
df_r['pollutant_code'] = df_r.pollutant.replace(['Carbon dioxide
```

```
(CO2)', 'Nitrogen oxides (NOX)', 'Methane (CH4)'], [1,0,2])
df_r.head()
```

```
                                  CITY ID CONTINENT
City DAY  \
0  47068  4c325d62c064477ef17b4c6e4437e121    EUROPE  Europoort
Rotterdam   2
1  32952  f5e609e7095f91cc8ce9ed6d8e774a0d    EUROPE
RION   3
2  72375  cfab1ba8c67c7c838db98d666f02a132    EUROPE
--   1
3  40702  95b4e51f7b662598134e1eb956407c74    EUROPE
DRIZZONA  17
4  29884  f4433be3b1bfaeeb0633eb65d04b1325    EUROPE
Lünen   6

  DAY WITH FOGS EPRTRAnnexIMainActivityCode  \
0             1                         4(a)
1             2                         3(c)
2            12                         1(c)
3             1                         7(a)
4             0                         5(a)

                       EPRTRAnnexIMainActivityLabel
EPRTRSectorCode  \
0  Chemical installations for the production on a...              4

1  Installations for the production of cement cli...              3

2  Thermal power stations and other combustion in...              1

3  Installations for the intensive rearing of pou...              7

4  Installations for the recovery or disposal of ...              5


                                FacilityInspireID  ...  \
0                     NL.RIVM/000019070.FACILITY  ...
1                        EL.CAED/100075.FACILITY  ...
2        UK.CAED/BEISOffsh-Cormorant-Alpha.FACILITY  ...
3                     IT.CAED/260342003.FACILITY  ...
4  https://registry.gdi-de.org/id/de.nw.inspire.p...  ...

                                      facilityName
max_temp  \
0                    Indorama Ventures Europe BV
13.256816011792559
1                TITAN CEMENT S.A. - DREPANO PLANT
4.528859186447803
```

```
2                                    Cormorant Alpha
10.669132597893881
3  SOCIETA' AGRICOLA SPARAVALLE DI FERRARI GIUSEP...
7.095681595088376
4                       Biomassekraftwerk Lünen GmbH
9.886774464050356

       max_wind_speed            min_temp      min_wind_speed  \
0  11.019328717116156  14.696895445152332  20.899761591708206
1    14.5123950384412   9.219003402711184  23.243402867192145
2   20.26217117993502  14.715465115792192  23.956529199327292
3   18.28354666681811  13.582024001859644   26.69626609353847
4   13.75940846376134   14.00622637509683  24.768932565830674

               pollutant reportingYear targetRelease  \
0   Carbon dioxide (CO2)          2020           AIR
1  Nitrogen oxides (NOX)          2019           AIR
2  Nitrogen oxides (NOX)          2009           AIR
3          Methane (CH4)          2014           AIR
4   Carbon dioxide (CO2)          2015           AIR

                                          descripcion pollutant_code
0  Chemical industry . Chemical installations for...              1
1  Mineral industry . Installations for the produ...              0
2  Energy sector . Thermal power stations and oth...              0
3  Intensive livestock production and aquaculture...              2
4  Waste and wastewater management . Installation...              1

[5 rows x 26 columns]
```

```
df_1=df[['descripcion','pollutant_code']]
df_1
```

```
                                             descripcion
pollutant_code
0      Mineral industry . Installations for the produ...
1
1      Mineral industry . Installations for the produ...
0
2      Waste and wastewater management . Landfills (e...
2
3      Energy sector . Thermal power stations and oth...
0
4      Waste and wastewater management . Urban waste-...
2
...                                                  ...             .
..
37122  Paper and wood production and processing . Ind...
0
37123  Energy sector . Thermal power stations and oth...
```

```
1
37124  Chemical industry . Chemical installations for...
0
37125  Mineral industry . Installations for the manuf...
1
37126  Mineral industry . Installations for the produ...
0

[37127 rows x 2 columns]
```

```
df_2=df_r[['descripcion','pollutant_code']]
df_2
```

```
                                            descripcion
pollutant_code
0      Chemical industry . Chemical installations for...
1
1      Mineral industry . Installations for the produ...
0
2      Energy sector . Thermal power stations and oth...
0
3      Intensive livestock production and aquaculture...
2
4      Waste and wastewater management . Installation...
1
...                                                  ...             .
..
28496  Energy sector . Thermal power stations and oth...
1
28497  Energy sector . Thermal power stations and oth...
0
28498  Waste and wastewater management . Landfills (e...
2
28499  Mineral industry . Underground mining and rela...
0
28500  Energy sector . Thermal power stations and oth...
1

[28501 rows x 2 columns]
```

```
# concatenamos todos los datasets
df_=pd.concat([df_1,df_2], axis=0, ignore_index=True)
df_.head()
```

```
                                        descripcion  pollutant_code
0  Mineral industry . Installations for the produ...               1
1  Mineral industry . Installations for the produ...               0
2  Waste and wastewater management . Landfills (e...               2
3  Energy sector . Thermal power stations and oth...               0
4  Waste and wastewater management . Urban waste-...               2
```

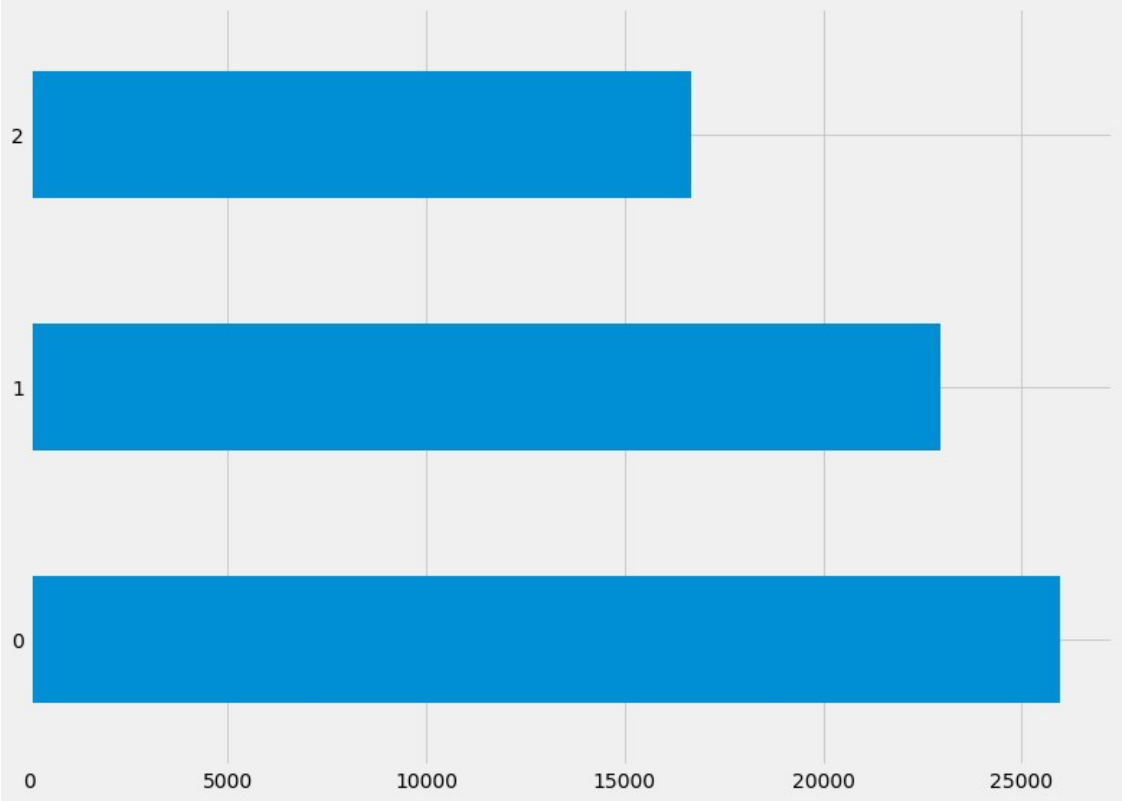
```
def limpiar_tokenizar(texto,tokenizar=True):
    '''
    Esta función limpia y tokeniza el texto en palabras individuales.
    El orden en el que se va limpiando el texto no es arbitrario.
    El listado de signos de puntuación se ha obtenido de:
print(string.punctuation)
    y re.escape(string.punctuation)
    '''

    # Se convierte todo el texto a minúsculas
    nuevo_texto = texto.lower()
    # Eliminación de páginas web (palabras que empiezan por "http")
    nuevo_texto = re.sub('http\S+', ' ', nuevo_texto)
    # Eliminación de signos de puntuación
    regex = '[\\!\\"\\#\\$\\%\\&\\'\\(\\)\\*\\+\\,\\-\\.\\/\\:\\;\\
<\\=\\>\\?\\@\\[\\\\\\]\\^_\\`\\{\\|\\}\\~]'
    nuevo_texto = re.sub(regex , ' ', nuevo_texto)
    # Eliminación de palabras que contienen números
    # nuevo_texto = re.sub("\d+", ' ', nuevo_texto)
    raw=[]
    [raw.append(t) for t in nuevo_texto.split(' ') if
(len(re.findall('\d+',t)) < 1)]
    nuevo_texto=' '.join(raw)
    # Eliminación de espacios en blanco múltiples
    nuevo_texto = re.sub("\\s+", ' ', nuevo_texto)
    # Eliminación de simbolos
    nuevo_texto = re.sub("[\\<>$+`´|]+", ' ', nuevo_texto)
    if tokenizar:
      # Tokenización por palabras individuales
      nuevo_texto = nuevo_texto.split(sep = ' ')
      # Eliminación de tokens con una longitud < 2
      nuevo_texto = [token for token in nuevo_texto if len(token) > 1]

    return(nuevo_texto)

df_['descripcion_clean'] = df_['descripcion'].apply(lambda x:
limpiar_tokenizar(x,False))
df_
```

```
                                            descripcion
pollutant_code  \
0      Mineral industry . Installations for the produ...
1
1      Mineral industry . Installations for the produ...
0
2      Waste and wastewater management . Landfills (e...
2
3      Energy sector . Thermal power stations and oth...
0
4      Waste and wastewater management . Urban waste-...
```

```
2
...                                                   ...              .
..
65623  Energy sector . Thermal power stations and oth...
1
65624  Energy sector . Thermal power stations and oth...
0
65625  Waste and wastewater management . Landfills (e...
2
65626  Mineral industry . Underground mining and rela...
0
65627  Energy sector . Thermal power stations and oth...
1

                                       descripcion_clean
0      mineral industry installations for the product...
1      mineral industry installations for the product...
2      waste and wastewater management landfills excl...
3      energy sector thermal power stations and other...
4      waste and wastewater management urban waste wa...
...                                                  ...
65623  energy sector thermal power stations and other...
65624  energy sector thermal power stations and other...
65625  waste and wastewater management landfills excl...
65626  mineral industry underground mining and relate...
65627  energy sector thermal power stations and other...

[65628 rows x 3 columns]
```

```
df_.pollutant_code.value_counts().plot(kind='barh')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f603a5ca2d0>
```




```
# se unen todos los textos de cada fila en una sola fila.
row_list =  []
codes=df_.pollutant_code.value_counts()
for j, code in enumerate(codes.index):
  row1 = ' . '.join(df_[df_.pollutant_code == code]
['descripcion_clean'])
  row_list.append([row1])

data =
pd.DataFrame(row_list,index=codes.index,columns=['descripcion_clean'])
data
```

```
                                  descripcion_clean
0  mineral industry installations for the product...
1  mineral industry installations for the product...
2  waste and wastewater management landfills excl...
```

```
stop_words = set(stopwords.words('english') + list(punctuation))

# Creación de la matriz tf-idf
#=================================================================
=========
tfidf_vectorizador = TfidfVectorizer(
                        tokenizer  = limpiar_tokenizar,
                        min_df     = 3,
```

```
                        stop_words = stop_words
                    )
# tfidf_vectorizador.fit_transform(X_train.apply(lambda x:np.str_(x)))
# X =
tfidf_vectorizador.fit_transform(df_clean[~df_clean.Descripción.isna()
].Descripción.apply(lambda x:np.str_(x)))
X =
tfidf_vectorizador.fit_transform(df_['descripcion_clean'].apply(lambda
x:np.str_(x)))

# X = tfidf_vectorizador.fit_transform(data.Descripcion.apply(lambda
x:np.str_(x)))
MNYT = X.toarray()
#Obtenemos los feature names
feature_names = np.array(tfidf_vectorizador.get_feature_names())
data_tfidf = pd.DataFrame(MNYT,
columns=tfidf_vectorizador.get_feature_names())
# print(feature_names)
print(MNYT)
```

```
[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
```

```
# Reparto train y test
#
================================================================================
========
```

```
X_train, X_test, y_train, y_test = train_test_split(
    MNYT,
    df_.pollutant_code,
    test_size = 0.2,
    random_state = 123
)

value, counts = np.unique(y_train, return_counts=True)
print(dict(zip(value, 100 * counts / sum(counts))))
value, counts = np.unique(y_test, return_counts=True)
print(dict(zip(value, 100 * counts / sum(counts))))
```

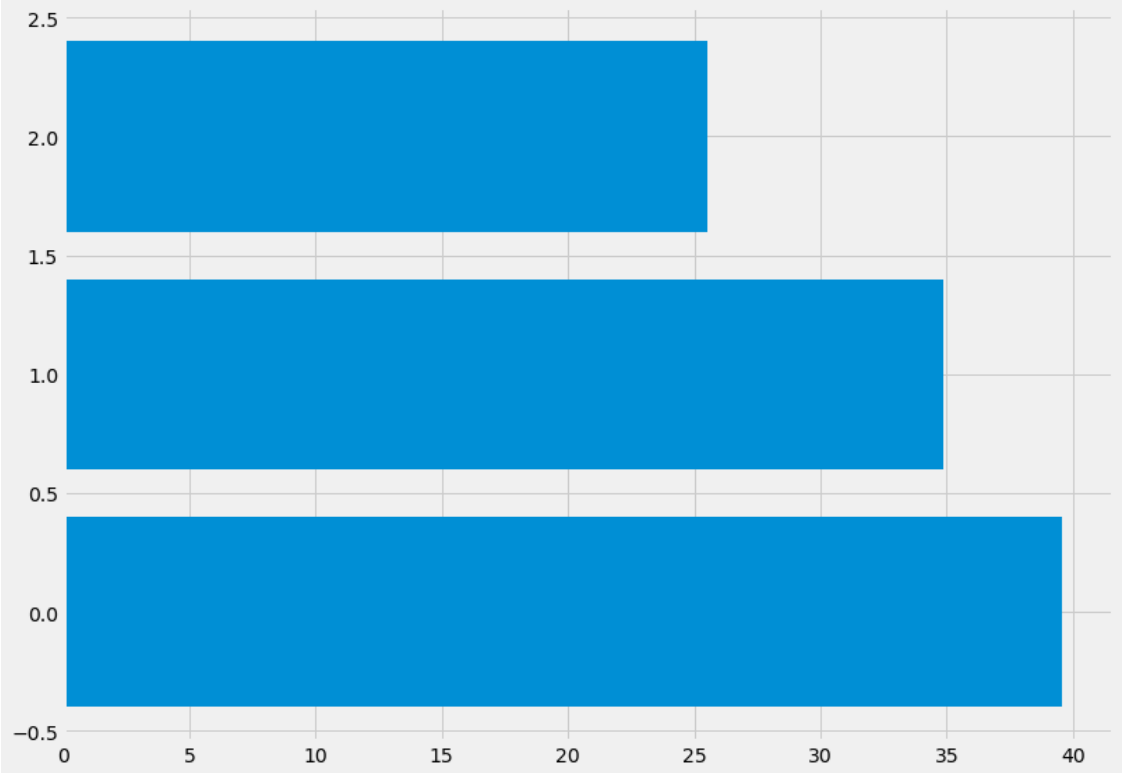
```
{0: 39.59658679669346, 1: 34.88819473543865, 2: 25.51521846786789}
{0: 39.562699984763064, 1: 35.4030169129971, 2: 25.03428310223983}
```

```
# df.ETIQUETADO_PREVIO.value_counts().sort_values().plot(kind='barh')
df_.pollutant_code.value_counts()
```

```
0    25982
1    22964
2    16682
Name: pollutant_code, dtype: int64
```

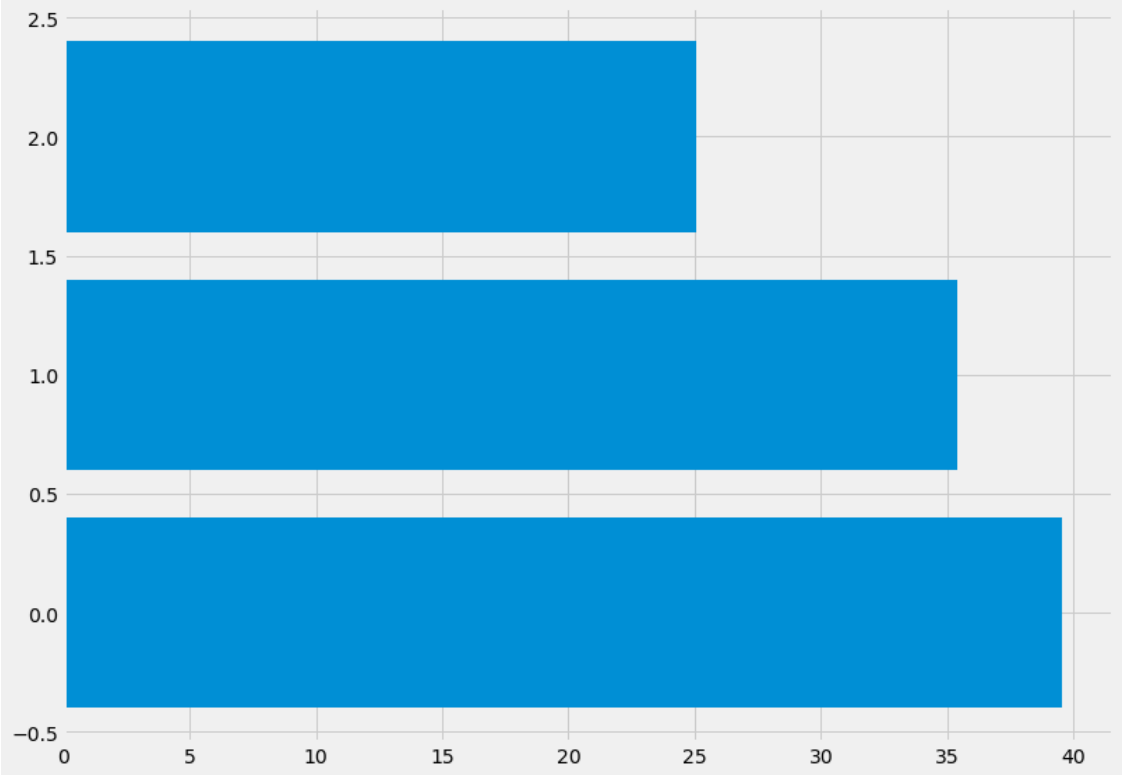
```
value, counts = np.unique(y_train, return_counts=True)
print(dict(zip(value, 100 * counts / sum(counts))))
D =dict(zip(value, 100 * counts / sum(counts)))
plt.barh(*zip(*D.items()))
# plt.xticks(rotation=90)
plt.show()
```

```
{0: 39.59658679669346, 1: 34.88819473543865, 2: 25.51521846786789}
```




```
value, counts = np.unique(y_test, return_counts=True)
print(dict(zip(value, 100 * counts / sum(counts))))
D =dict(zip(value, 100 * counts / sum(counts)))
plt.barh(*zip(*D.items()))
# plt.xticks(rotation=90)
plt.show()
```

```
{0: 39.562699984763064, 1: 35.4030169129971, 2: 25.03428310223983}
```




```
print(f" Número de tokens creados:
{len(tfidf_vectorizador.get_feature_names())}")
print(tfidf_vectorizador.get_feature_names()[:10])
```

```
 Número de tokens creados: 11111
['aabenraa', 'aak', 'aalborg', 'aalen', 'aalst', 'aan', 'aarhus',
'ab', 'abaixo', 'abajas']
```

Logistic regresion

```
#2. Importar clasificador

#Establecemos el método de clasificación (método logistic regression)
classifier = LogisticRegression()
#Entrenamiento del clasificador aplicando el método al corpus de
entrenamiento
log_model = classifier.fit(X=X_train, y=y_train)

y_test.size
y_train.size
```

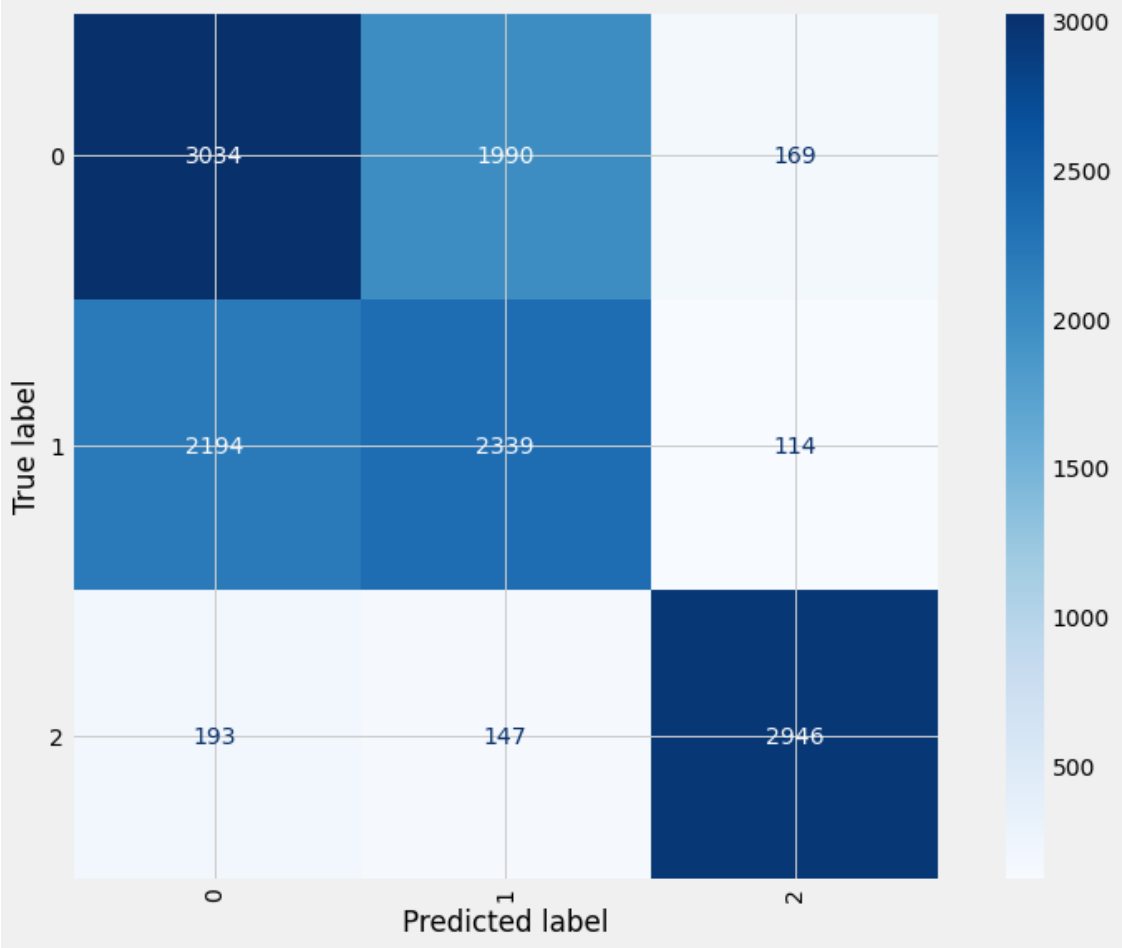
```
52502
```

```
#Resultados de la clasificación/predicción del clasificador entrenado
al corpus de test
y_pred = log_model.predict(X_test)

print("\nCLASES DEL CORPUS DE TEST:\n===============")
```

```
print(y_pred)

def most_informative_feature_for_binary_classification(vectorizer,
classifier, n=10):
    class_labels = classifier.classes_
    feature_names = vectorizer.get_feature_names()
    topn_class1 = sorted(zip(classifier.coef_[0], feature_names))[:n]
    topn_class2 = sorted(zip(classifier.coef_[0], feature_names))[-n:]

    for coef, feat in topn_class1:
        print (class_labels[0], coef, feat)

    for coef, feat in reversed(topn_class2):
        print (class_labels[1], coef, feat)

print("\nFEATURES MÁS INFORMATIVOS EN LAS CLASES:\n===============")
print(most_informative_feature_for_binary_classification(tfidf_vectori
zador, classifier))


CLASES DEL CORPUS DE TEST:
===============
[0 0 0 ... 1 1 2]

FEATURES MÁS INFORMATIVOS EN LAS CLASES:
===============
0 -2.6027490742222343 intensive
0 -2.1459132788383632 ch
0 -1.973309310812712 guardian
0 -1.9225038469894602 kings
0 -1.865359521109773 compressor
0 -1.851958876013746 pigs
0 -1.8448651500807625 kwk
0 -1.8429568722913658 dyke
0 -1.8372957546782422 gazu
0 -1.7562828783329392 bacton
1 2.5265301766927735 roxby
1 2.2064517782678696 es
1 2.147039854699695 landfill
1 2.133847482063194 animal
1 2.0880748474406103 glass
1 1.9412940661852842 cement
1 1.90134957899636 incineration
1 1.87164387446861 mucking
1 1.855989846207769 including
1 1.8442218040866636 hempsted
None

# Error predicciones test
#
```

```
================================================================================
========
print("-------------")
print("Error de test")
print("-------------")

print(f"Número de clasificaciones erróneas de un total de
{len(y_test)} " \
      f"clasificaciones: {(y_test != y_pred).sum()}"
)
print(f"% de error: {100*(y_test != y_pred).mean()}")
print('Accuracy: {}'.format(accuracy_score(y_test,y_pred)))
print("")
print("--------------------")
print("Matriz de confusión")
print("--------------------")
cm=confusion_matrix(y_true = y_test, y_pred= y_pred)

ls=classifier.classes_

disp = ConfusionMatrixDisplay(confusion_matrix=cm,
                              display_labels=ls)

#
sns.heatmap(disp.confusion_matrix,annot=True,cmap='Blues',square=True,
fmt='d')
disp.plot(include_values=True,
cmap='Blues',values_format='d',xticks_rotation='vertical')
plt.show()
print(f"El accuracy de test es: {100 * f1_score(y_test, y_pred,
average='macro')} %")
print(classification_report(y_test,y_pred))
```

```
-------------
Error de test
-------------
Número de clasificaciones erróneas de un total de 13126
clasificaciones: 4807
% de error: 36.62197165930215
Accuracy: 0.6337802834069786

--------------------
Matriz de confusión
--------------------
```




```
El accuracy de test es: 66.29474000636478 %
              precision    recall  f1-score   support

           0       0.56      0.58      0.57      5193
           1       0.52      0.50      0.51      4647
           2       0.91      0.90      0.90      3286

    accuracy                           0.63     13126
   macro avg       0.66      0.66      0.66     13126
weighted avg       0.63      0.63      0.63     13126
```

SVM

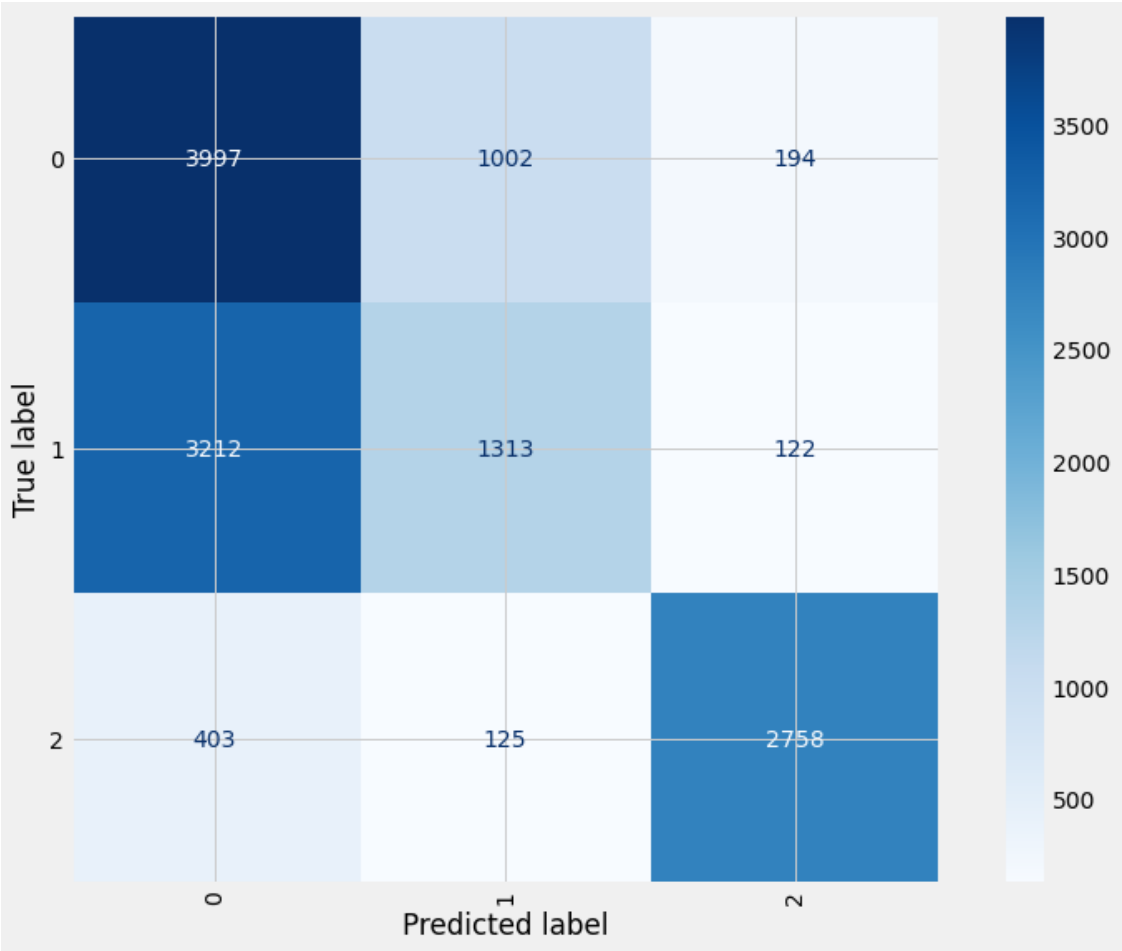
```
# Entrenamiento del modelo SVM
#
================================================================================
========
modelo_svm_lineal_tarea = svm.SVC(kernel= "linear", C =
2.1544346900318821)
modelo_svm_lineal_tarea.fit(X=X_train, y= y_train)
```

```
SVC(C=2.154434690031882, kernel='linear')
```

```
# Error predicciones test
#
==========================================================================
========
y_pred = modelo_svm_lineal_tarea.predict(X=X_test)

print("-------------")
print("Error de test")
print("-------------")

print(f"Número de clasificaciones erróneas de un total de
{X_test.shape[0]} " \
      f"clasificaciones: {(y_test != y_pred).sum()}"
)
print(f"% de error: {100*(y_test != y_pred).mean()}")
print('Accuracy: {}'.format(accuracy_score(y_test,y_pred)))
print("")
print("--------------------")
print("Matriz de confusión")
print("--------------------")
cm=confusion_matrix(y_true = y_test, y_pred= y_pred)
# pd.DataFrame(cm,
#              columns= list(y_test.unique()),
#              index = list(y_test.unique()))

# cm=confusion_matrix(y_pred.argmax(axis=1),test_y.argmax(axis=1))
# ls=df_clean.ETIQUETADO_previo_2.unique().tolist()
ls=modelo_svm_lineal_tarea.classes_
disp = ConfusionMatrixDisplay(confusion_matrix=cm,
                              display_labels=ls)

#
sns.heatmap(disp.confusion_matrix,annot=True,cmap='Blues',square=True,
fmt='d')
disp.plot(include_values=True,
cmap='Blues',values_format='d',xticks_rotation='vertical')
plt.show()

print(classification_report(y_test,y_pred))
```

```
-------------
Error de test
-------------
Número de clasificaciones erróneas de un total de 13126
clasificaciones: 5058
% de error: 38.534206917568184
Accuracy: 0.6146579308243182
```

```
--------------------
Matriz de confusión
--------------------
```



	precision	recall	f1-score	support
0	0.53	0.77	0.62	5193
1	0.54	0.28	0.37	4647
2	0.90	0.84	0.87	3286
accuracy			0.61	13126
macro avg	0.65	0.63	0.62	13126
weighted avg	0.62	0.61	0.60	13126

```
print(classification_report(y_test,y_pred))
```

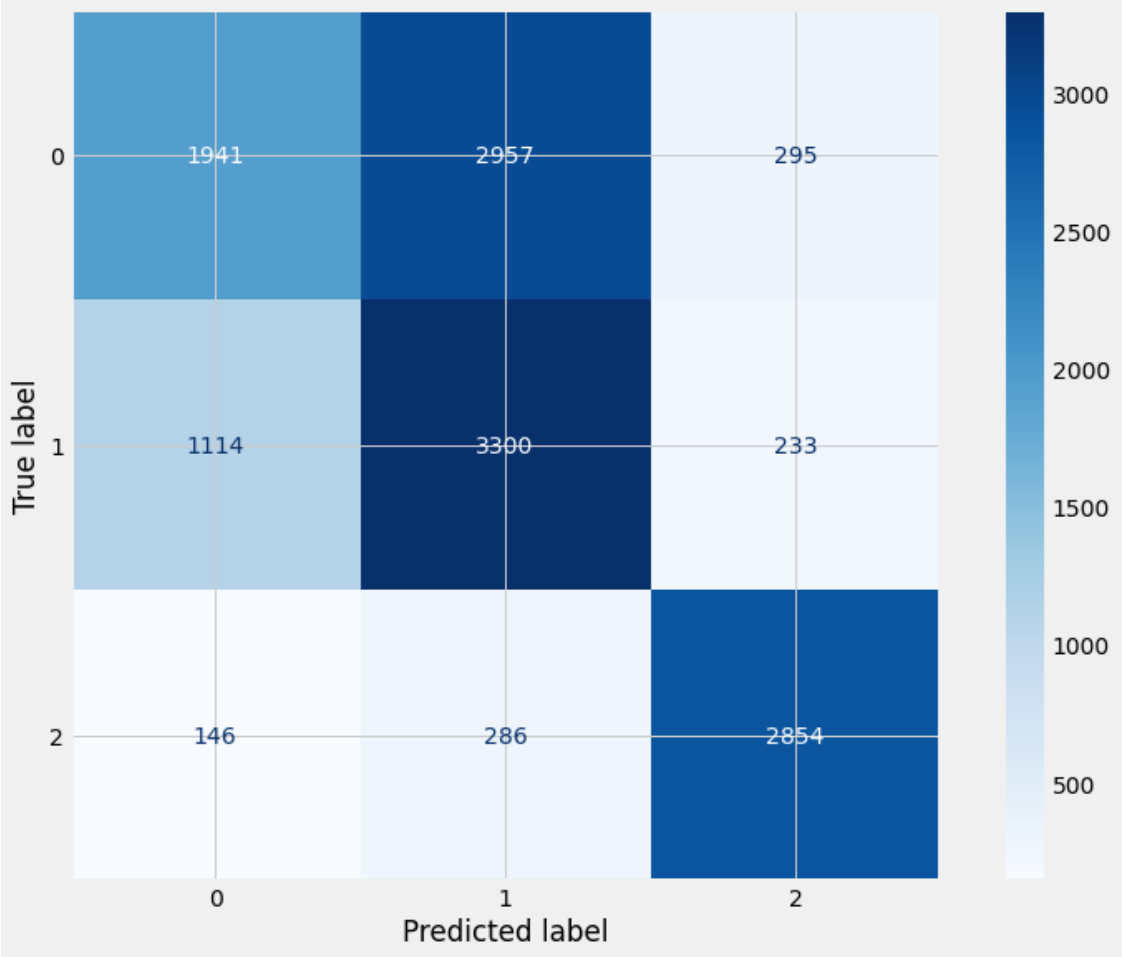
	precision	recall	f1-score	support
0	0.52	0.75	0.62	2898
1	0.55	0.31	0.40	2685
2	0.90	0.84	0.87	1843

```
    accuracy                           0.62      7426
   macro avg       0.66      0.64      0.63      7426
weighted avg       0.63      0.62      0.60      7426
```

Random Forest

```
forest =
RandomForestClassifier(class_weight='balanced',random_state=123)
# balanced", "balanced_subsample"
modelF = forest.fit(X_train, y_train)
predicciones = modelF.predict(X_test)

mat_confusion = confusion_matrix(
                    y_true    = y_test,
                    y_pred    = predicciones
                )

accuracy = accuracy_score(
            y_true    = y_test,
            y_pred    = predicciones,
            normalize = True
           )

print("Matriz de confusión")
print("-------------------")
# print(mat_confusion)
disp = ConfusionMatrixDisplay(confusion_matrix=mat_confusion,
display_labels=modelF.classes_)
disp.plot(cmap=plt.cm.Blues)
plt.show()
print("")
print(f"El accuracy de test es: {100 * f1_score(y_test, predicciones,
average='macro')} %")
```

```
Matriz de confusión
-------------------
```



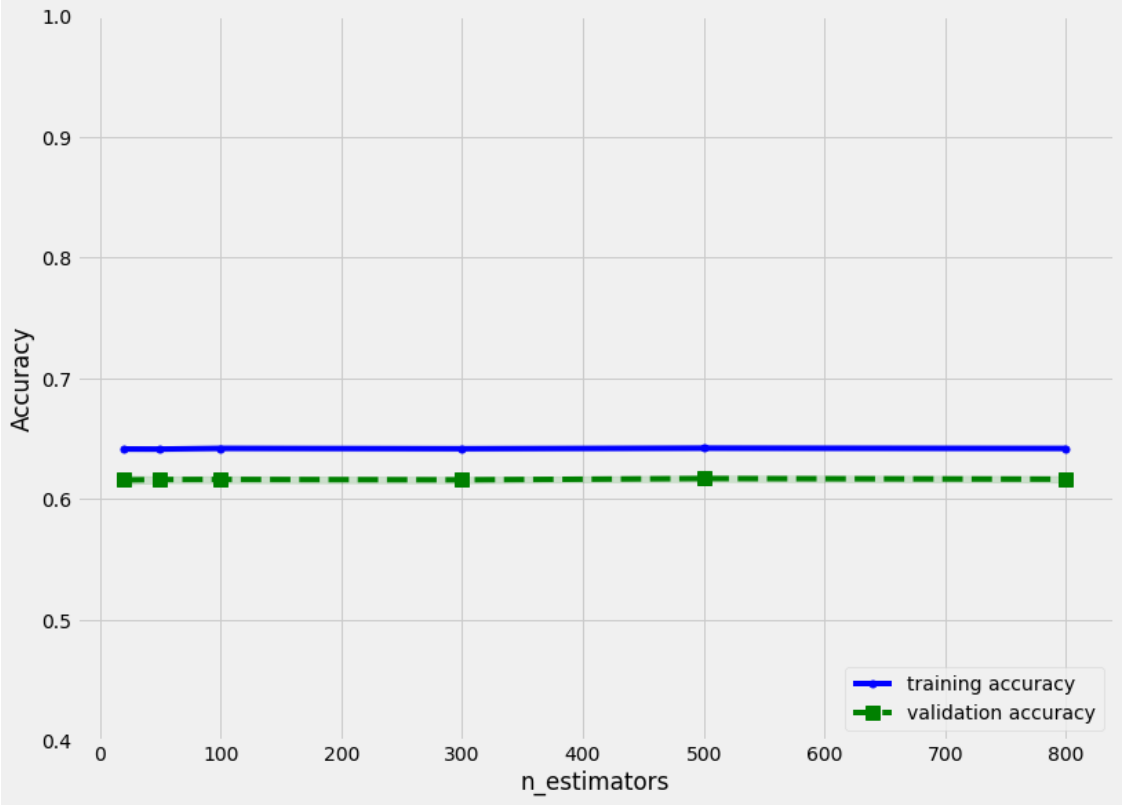

```
El accuracy de test es: 63.61047739400717 %
```

```
print(
    classification_report(
        y_true = y_test,
        y_pred = predicciones
    )
)
```

```
              precision    recall  f1-score   support

           0       0.61      0.37      0.46      5193
           1       0.50      0.71      0.59      4647
           2       0.84      0.87      0.86      3286

    accuracy                           0.62     13126
   macro avg       0.65      0.65      0.64     13126
weighted avg       0.63      0.62      0.61     13126
```

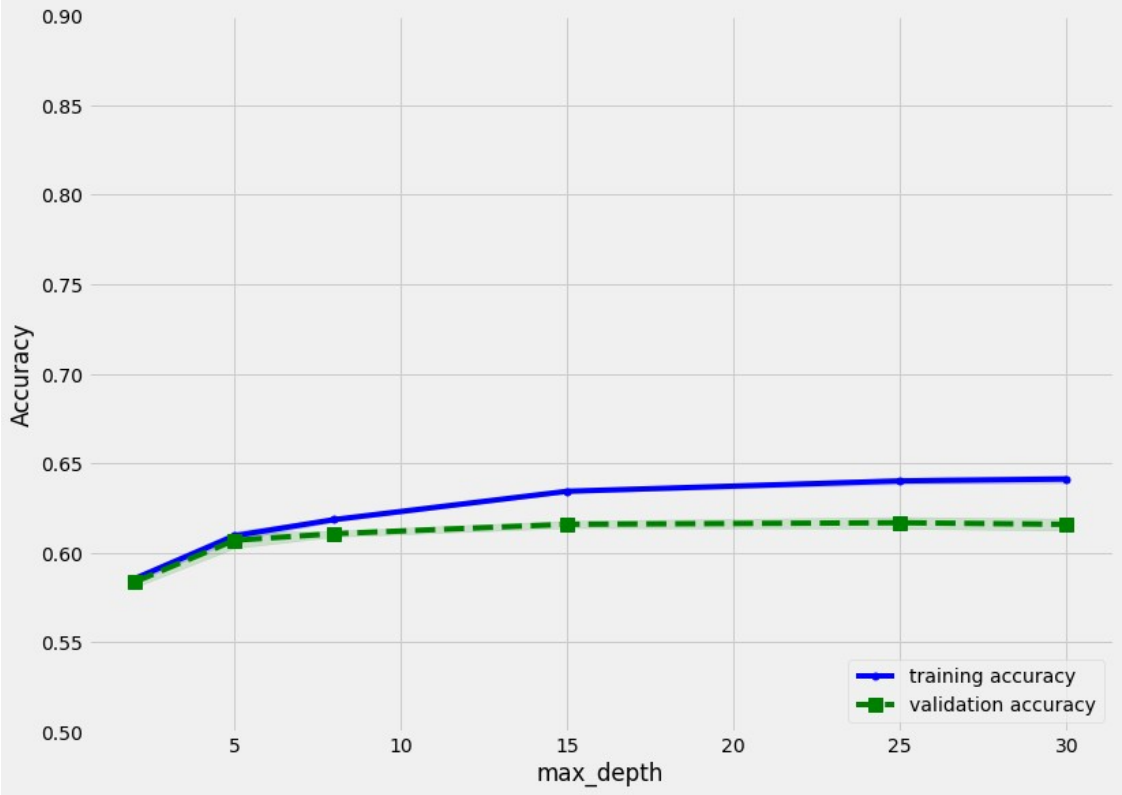
buscamos los mejores valores de los hiperparametros

```
n_estimators = [20, 50, 100, 300, 500, 800]
max_depth = [2, 5, 8, 15, 25, 30]
min_samples_split = [2, 5, 10, 15]
min_samples_leaf = [1, 2, 4, 10]

train_scoreNum, test_scoreNum = validation_curve(

RandomForestClassifier(class_weight='balanced'),
                                X = X_train, y = y_train,
                                param_name = 'n_estimators',
                                param_range = n_estimators, cv = 3)

train_mean = np.mean(train_scoreNum, axis=1)
train_std = np.std(train_scoreNum, axis=1)
test_mean = np.mean(test_scoreNum, axis=1)
test_std = np.std(test_scoreNum, axis=1)

plt.plot(n_estimators, train_mean,    color='blue', marker='o',
    markersize=5,
    label='training accuracy')
plt.fill_between(n_estimators, train_mean + train_std,
    train_mean - train_std, alpha=0.15,
    color='blue')
plt.plot(n_estimators, test_mean,
    color='green', linestyle='--',
    marker='s', markersize=10,
    label='validation accuracy')
plt.fill_between(n_estimators,
    test_mean + test_std,
    test_mean - test_std,
    alpha=0.15, color='green')
plt.grid(True)
plt.legend(loc='lower right')
plt.xlabel('n_estimators')
plt.ylabel('Accuracy')
plt.ylim([0.4, 1.0])
plt.show()
```



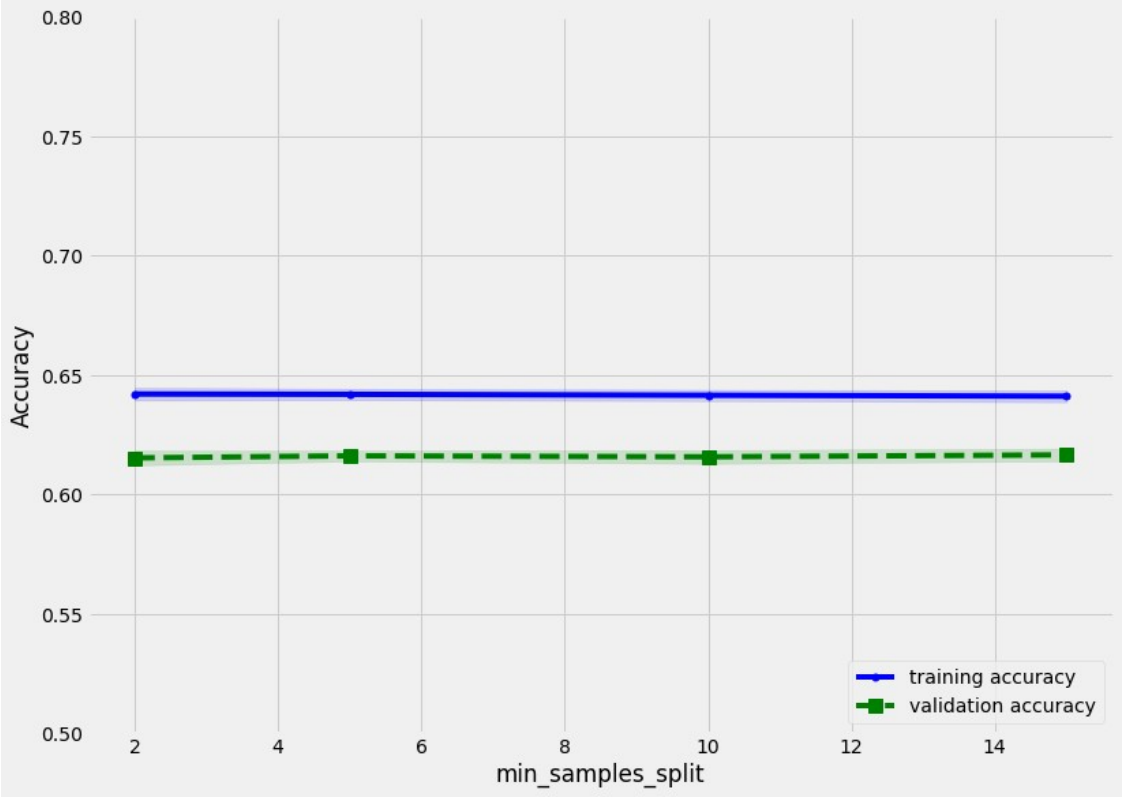

```
train_scoreNum, test_scoreNum = validation_curve(

RandomForestClassifier(class_weight='balanced'),
                                X = X_train, y = y_train,
                                param_name = 'max_depth',
                                param_range = max_depth, cv = 3)

train_mean = np.mean(train_scoreNum, axis=1)
train_std = np.std(train_scoreNum, axis=1)
test_mean = np.mean(test_scoreNum, axis=1)
test_std = np.std(test_scoreNum, axis=1)

plt.plot(max_depth, train_mean,    color='blue', marker='o',
    markersize=5,
    label='training accuracy')
plt.fill_between(max_depth, train_mean + train_std,
    train_mean - train_std, alpha=0.15,
    color='blue')
plt.plot(max_depth, test_mean,
    color='green', linestyle='--',
    marker='s', markersize=10,
    label='validation accuracy')
plt.fill_between(max_depth,
    test_mean + test_std,
    test_mean - test_std,
    alpha=0.15, color='green')
```

```
plt.grid(True)
plt.legend(loc='lower right')
plt.xlabel('max_depth')
plt.ylabel('Accuracy')
plt.ylim([0.5, 0.9])
plt.show()
```



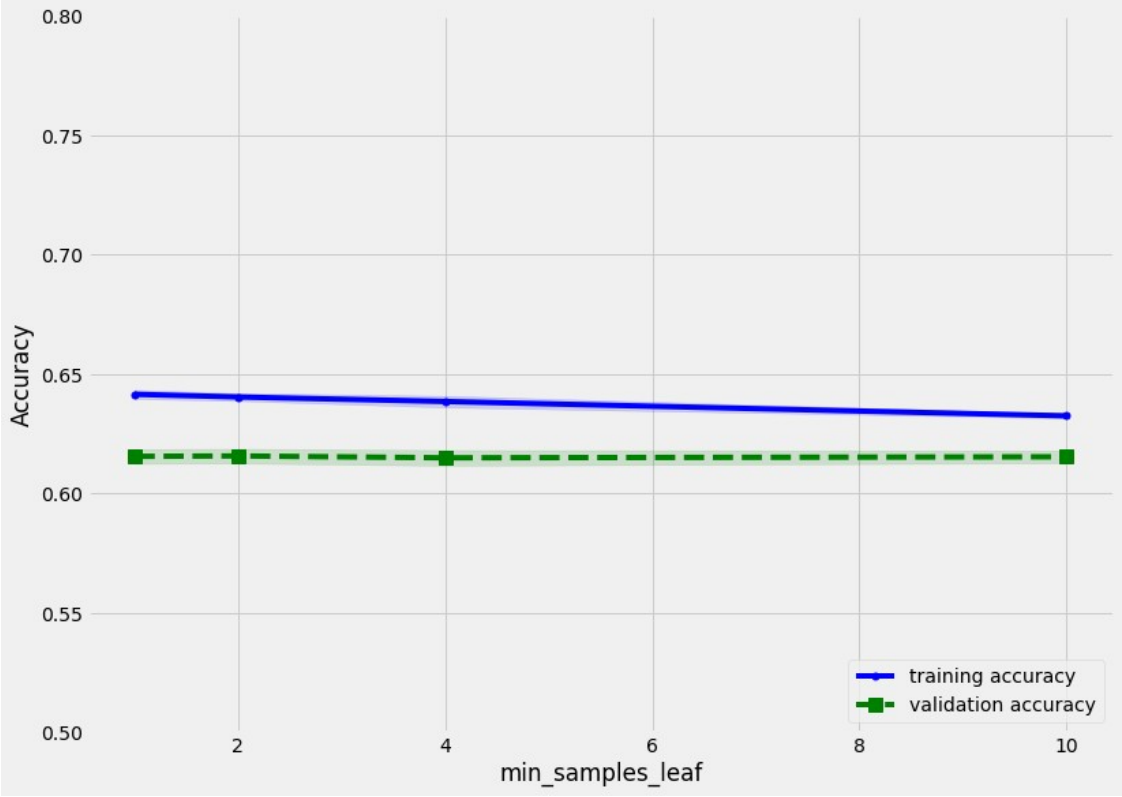

```
train_scoreNum, test_scoreNum = validation_curve(

RandomForestClassifier(class_weight='balanced'),
                                X = X_train, y = y_train,
                                param_name = 'min_samples_split',
                                param_range = min_samples_split, cv =
3)

train_mean = np.mean(train_scoreNum, axis=1)
train_std = np.std(train_scoreNum, axis=1)
test_mean = np.mean(test_scoreNum, axis=1)
test_std = np.std(test_scoreNum, axis=1)

plt.plot(min_samples_split, train_mean,    color='blue', marker='o',
    markersize=5,
    label='training accuracy')
plt.fill_between(min_samples_split, train_mean + train_std,
    train_mean - train_std, alpha=0.15,
    color='blue')
```

```
plt.plot(min_samples_split, test_mean,
    color='green', linestyle='--',
    marker='s', markersize=10,
    label='validation accuracy')
plt.fill_between(min_samples_split,
    test_mean + test_std,
    test_mean - test_std,
    alpha=0.15, color='green')
plt.grid(True)
plt.legend(loc='lower right')
plt.xlabel('min_samples_split')
plt.ylabel('Accuracy')
plt.ylim([0.5, 0.8])
plt.show()
```




```
train_scoreNum, test_scoreNum = validation_curve(

RandomForestClassifier(class_weight='balanced'),
                                X = X_train, y = y_train,
                                param_name = 'min_samples_leaf',
                                param_range = min_samples_leaf, cv =
3)

train_mean = np.mean(train_scoreNum, axis=1)
train_std = np.std(train_scoreNum, axis=1)
test_mean = np.mean(test_scoreNum, axis=1)
```

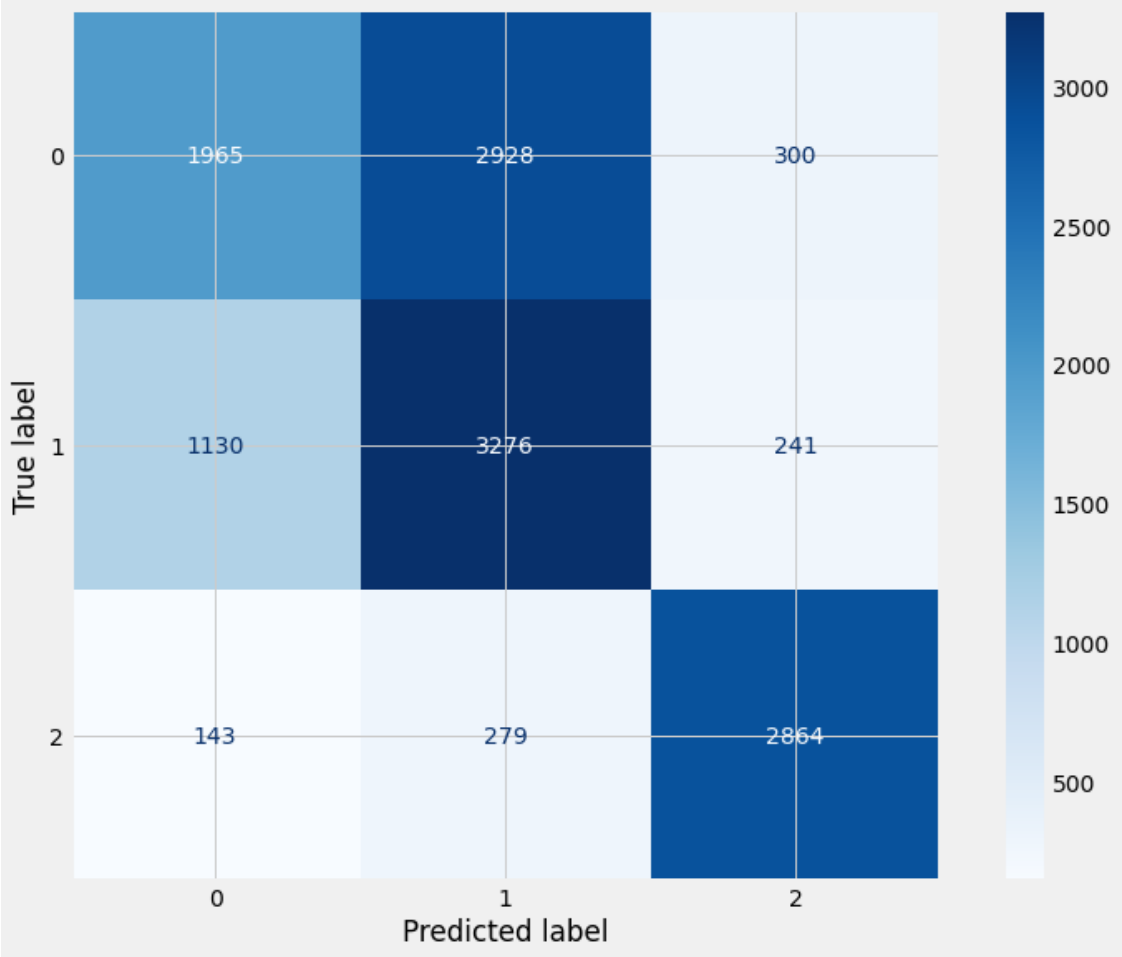
```
test_std = np.std(test_scoreNum, axis=1)

plt.plot(min_samples_leaf, train_mean,    color='blue', marker='o',
    markersize=5,
    label='training accuracy')
plt.fill_between(min_samples_leaf, train_mean + train_std,
    train_mean - train_std, alpha=0.15,
    color='blue')
plt.plot(min_samples_leaf, test_mean,
    color='green', linestyle='--',
    marker='s', markersize=10,
    label='validation accuracy')
plt.fill_between(min_samples_leaf,
    test_mean + test_std,
    test_mean - test_std,
    alpha=0.15, color='green')
plt.grid(True)
plt.legend(loc='lower right')
plt.xlabel('min_samples_leaf')
plt.ylabel('Accuracy')
plt.ylim([0.5, 0.8])
plt.show()
```




```
# forest =
RandomForestClassifier(class_weight='balanced',random_state=123)
forestVC = RandomForestClassifier(random_state = 123,
```

```
                                    n_estimators = 100,
                                    max_depth = 30,
                                    min_samples_split = 2,
                                    min_samples_leaf = 1,
                                    class_weight='balanced')
# balanced", "balanced_subsample"
modelFVC = forestVC.fit(X_train, y_train)
predicciones = modelFVC.predict(X_test)

mat_confusion = confusion_matrix(
                    y_true    = y_test,
                    y_pred    = predicciones
                )

accuracy = accuracy_score(
             y_true    = y_test,
             y_pred    = predicciones,
             normalize = True
            )

print("Matriz de confusión")
print("-------------------")
# print(mat_confusion)
disp = ConfusionMatrixDisplay(confusion_matrix=mat_confusion,
display_labels=modelFVC.classes_)
disp.plot(cmap=plt.cm.Blues)
plt.show()
print("")
print(f"El accuracy de test es: {100 * f1_score(y_test, predicciones,
average='macro')} %")
```

```
Matriz de confusión
-------------------
```



El accuracy de test es: 63.69638153873267 %

```
print(
    classification_report(
        y_true = y_test,
        y_pred = predicciones
    )
)
```

```
              precision    recall  f1-score   support

           0       0.61      0.38      0.47      5193
           1       0.51      0.70      0.59      4647
           2       0.84      0.87      0.86      3286

    accuracy                           0.62     13126
   macro avg       0.65      0.65      0.64     13126
weighted avg       0.63      0.62      0.61     13126
```

Recuperamos el mejor modelo (regresion logistica) y realizamos la prediccion sobre el conjunto de test

```
test=pd.read_csv('/content/test_x.csv')
test.head()
```

```
   test_index  countryName  EPRTRSectorCode
eprtrSectorName  \
0           0       Poland                3                 Mineral
industry
1           1   Luxembourg                5  Waste and wastewater
management
2           2  Netherlands                1                    Energy
sector
3           3       Sweden                5  Waste and wastewater
management
4           4     Portugal                1                    Energy
sector

  EPRTRAnnexIMainActivityCode  \
0                       3(a)
1                       5(d)
2                       1(c)
3                       5(d)
4                       1(c)

                      EPRTRAnnexIMainActivityLabel  \
0         Underground mining and related operations
1  Landfills (excluding landfills of inert waste ...
2  Thermal power stations and other combustion in...
3  Landfills (excluding landfills of inert waste ...
4  Thermal power stations and other combustion in...

            FacilityInspireID  \
0    PL.MŚ/000002357.FACILITY
1  LU.CAED/000012000.FACILITY
2      NL.EEA/212857.FACILITY
3   SE.CAED/10013901.Facility
4      PT.EEA/133926.FACILITY

                                          facilityName       City
targetRelease  \
0  Polska Grupa Górnicza sp. z o.o. Oddział KWK R...  Rydułtowy
AIR
1                                              Sidec   Diekirch
AIR
2              Nuon Power Generation BV (Eemshaven)  Eemshaven
AIR
3                      HÖGBYTORPS AVFALLSANLÄGGNING        BRO
AIR
```

```
4  SPCG - Sociedade Portuguesa de Co-Geração Eléc...    SETÚBAL
AIR

   ...  CONTINENT  max_wind_speed  avg_wind_speed min_wind_speed
max_temp  \
0  ...     EUROPE          14.080          14.856         18.475
10.279
1  ...     EUROPE          16.052          17.624         22.623
6.626
2  ...     EUROPE          13.647          15.542         17.819
5.669
3  ...     EUROPE          16.337          17.458         19.962
6.161
4  ...     EUROPE          21.517          20.532         21.617
10.964

   avg_temp  min_temp  DAY WITH FOGS     REPORTER NAME  \
0    11.381    13.481              1     Brittany Buck
1     8.840    13.423              0     Lauren Fisher
2     8.403    11.276              2    Linda Thompson
3     7.572     9.444              2  Bethany Mcmillan
4    11.548    12.624              2     Sarah Hoffman

                            CITY ID
0  826b1de9dad293ae3e4f9cbaf6cf3420
1  ed30a6667b40ba0a66198b3173e7353f
2  78e1082c3cfef3bdf3554da8d6afcc34
3  27f959641950d381869d746d7d0e7d4e
4  1cb71655d9e0bd5cedb2320bf5fdd8f7

[5 rows x 23 columns]
```

```
test['descripcion'] = test['eprtrSectorName']+' .
'+test['EPRTRAnnexIMainActivityLabel']+' . '+test['FacilityInspireID']
test.head()
```

```
   test_index  countryName  EPRTRSectorCode
eprtrSectorName  \
0           0       Poland                3                 Mineral
industry
1           1   Luxembourg                5  Waste and wastewater
management
2           2  Netherlands                1                    Energy
sector
3           3       Sweden                5  Waste and wastewater
management
4           4     Portugal                1                    Energy
sector

  EPRTRAnnexIMainActivityCode  \
```

```
0                       3(a)
1                       5(d)
2                       1(c)
3                       5(d)
4                       1(c)

                      EPRTRAnnexIMainActivityLabel  \
0         Underground mining and related operations
1  Landfills (excluding landfills of inert waste ...
2  Thermal power stations and other combustion in...
3  Landfills (excluding landfills of inert waste ...
4  Thermal power stations and other combustion in...

            FacilityInspireID  \
0    PL.MŚ/000002357.FACILITY
1  LU.CAED/000012000.FACILITY
2      NL.EEA/212857.FACILITY
3   SE.CAED/10013901.Facility
4      PT.EEA/133926.FACILITY

                                          facilityName       City
targetRelease  \
0  Polska Grupa Górnicza sp. z o.o. Oddział KWK R...  Rydułtowy
AIR
1                                              Sidec   Diekirch
AIR
2              Nuon Power Generation BV (Eemshaven)  Eemshaven
AIR
3                      HÖGBYTORPS AVFALLSANLÄGGNING        BRO
AIR
4  SPCG - Sociedade Portuguesa de Co-Geração Eléc...    SETÚBAL
AIR

   ...  max_wind_speed  avg_wind_speed  min_wind_speed max_temp
avg_temp  \
0  ...          14.080          14.856          18.475   10.279
11.381
1  ...          16.052          17.624          22.623    6.626
8.840
2  ...          13.647          15.542          17.819    5.669
8.403
3  ...          16.337          17.458          19.962    6.161
7.572
4  ...          21.517          20.532          21.617   10.964
11.548

   min_temp  DAY WITH FOGS     REPORTER NAME  \
0    13.481              1     Brittany Buck
1    13.423              0     Lauren Fisher
2    11.276              2    Linda Thompson
```

```
3     9.444              2  Bethany Mcmillan
4    12.624              2     Sarah Hoffman

                           CITY ID  \
0  826b1de9dad293ae3e4f9cbaf6cf3420
1  ed30a6667b40ba0a66198b3173e7353f
2  78e1082c3cfef3bdf3554da8d6afcc34
3  27f959641950d381869d746d7d0e7d4e
4  1cb71655d9e0bd5cedb2320bf5fdd8f7

                                        descripcion
0  Mineral industry . Underground mining and rela...
1  Waste and wastewater management . Landfills (e...
2  Energy sector . Thermal power stations and oth...
3  Waste and wastewater management . Landfills (e...
4  Energy sector . Thermal power stations and oth...

[5 rows x 24 columns]
```

```
test_ = test[['test_index','descripcion']]
test_.head()
```

```
   test_index                                        descripcion
0           0  Mineral industry . Underground mining and rela...
1           1  Waste and wastewater management . Landfills (e...
2           2  Energy sector . Thermal power stations and oth...
3           3  Waste and wastewater management . Landfills (e...
4           4  Energy sector . Thermal power stations and oth...
```

```
test_['descripcion_clean'] = test_['descripcion'].apply(lambda x:
limpiar_tokenizar(x,False))
test_.head()
```

```
   test_index                                        descripcion  \
0           0  Mineral industry . Underground mining and rela...
1           1  Waste and wastewater management . Landfills (e...
2           2  Energy sector . Thermal power stations and oth...
3           3  Waste and wastewater management . Landfills (e...
4           4  Energy sector . Thermal power stations and oth...

                                  descripcion_clean
0  mineral industry underground mining and relate...
1  waste and wastewater management landfills excl...
2  energy sector thermal power stations and other...
3  waste and wastewater management landfills excl...
4  energy sector thermal power stations and other...
```

```
test_m = tfidf_vectorizador.transform(test_['descripcion'])
test_m = test_m.toarray()
test_m
```

```
array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])
```

```
predicciones_test = log_model.predict(test_m)
predicciones_test
```

```
array([2, 2, 1, ..., 2, 1, 2])
```

```
test_['pollutant']=predicciones_test
test_.head()
```

```
   test_index                                         descripcion  \
0           0  Mineral industry . Underground mining and rela...
1           1  Waste and wastewater management . Landfills (e...
2           2  Energy sector . Thermal power stations and oth...
3           3  Waste and wastewater management . Landfills (e...
4           4  Energy sector . Thermal power stations and oth...

                                 descripcion_clean  pollutant
0  mineral industry underground mining and relate...          2
1  waste and wastewater management landfills excl...          2
2  energy sector thermal power stations and other...          1
3  waste and wastewater management landfills excl...          2
4  energy sector thermal power stations and other...          0
```

```
test_.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24480 entries, 0 to 24479
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   test_index         24480 non-null  int64 
 1   descripcion        24480 non-null  object
 2   descripcion_clean  24480 non-null  object
 3   pollutant          24480 non-null  int64 
dtypes: int64(2), object(2)
memory usage: 765.1+ KB
```

```
# guardamos formato csv
test_.drop(columns=['descripcion','descripcion_clean']).to_csv('predic
tions.csv', index=False)
```

```
result =
test_.drop(columns=['descripcion','descripcion_clean']).to_json(orient
="columns")
result
```

```
{"type":"string"}
```

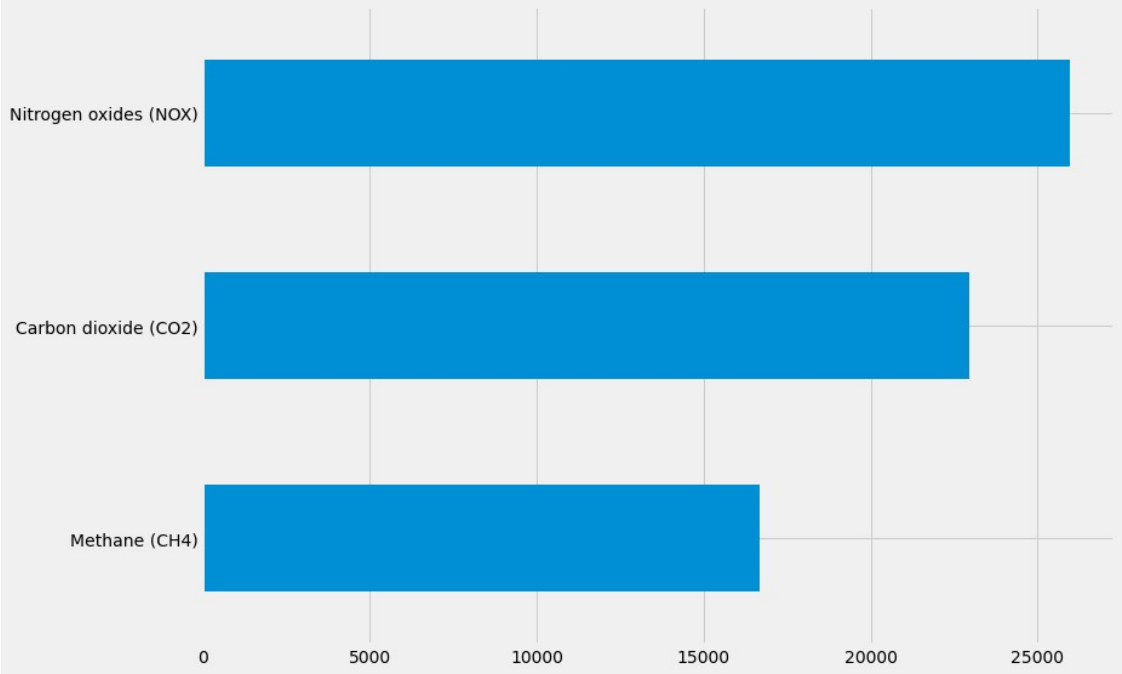
```
#guardamos en formato json
import json
with open('predictions.json', 'w') as fp:
    parsed = json.loads(result)
    json.dump(parsed, fp,indent=2)
```

```
df_final=pd.concat([df,df_r], axis=0, ignore_index=True)
df_final.head()
```

```
  countryName                  eprtrSectorName  \
0     Germany                 Mineral industry
1       Italy                 Mineral industry
2       Spain  Waste and wastewater management
3     Czechia                    Energy sector
4     Finland  Waste and wastewater management

                        EPRTRAnnexIMainActivityLabel  \
0  Installations for the production of cement cli...
1  Installations for the production of cement cli...
2  Landfills (excluding landfills of inert waste ...
3  Thermal power stations and other combustion in...
4                Urban waste-water treatment plants

                                    FacilityInspireID  \
0  https://registry.gdi-de.org/id/de.ni.mu/062217...
1                        IT.CAED/240602021.FACILITY
2                        ES.CAED/001966000.FACILITY
3                    CZ.MZP.U422/CZ34736841.FACILITY
4  http://paikkatiedot.fi/so/1002031/pf/Productio...

                                         facilityName
City  \
0              Holcim (Deutschland) GmbH Werk Höver
Sehnde
1              Stabilimento di Tavernola Bergamasca  TAVERNOLA
BERGAMASCA
2                 COMPLEJO MEDIOAMBIENTAL DE ZURITA    PUERTO DEL
ROSARIO
3                                Elektrárny Prunéřov
Kadaň
4  TAMPEREEN VESI LIIKELAITOS, VIINIKANLAHDEN JÄT...
Tampere

  targetRelease              pollutant reportingYear MONTH  ...
avg_temp  \
0           AIR   Carbon dioxide (CO2)          2015    10  ...
4.924
1           AIR  Nitrogen oxides (NOX)          2018     9  ...
```

```
7.864
2           AIR          Methane (CH4)          2019     2  ...
4.233
3           AIR  Nitrogen oxides (NOX)          2012     8  ...
10.298
4           AIR          Methane (CH4)          2018    12  ...
11.344

  min_temp DAY WITH FOGS     REPORTER NAME
CITY ID  \
0    9.688              2  Mr. Jacob Ortega
7cdb5e74adcb2ffaa21c1b61395a984f
1   12.024              1    Ashlee Serrano
cd1dbabbdba230b828c657a9b19a8963
2    8.632              2      Vincent Kemp
5011e3fa1436d15b34f1287f312fbada
3   15.179              0        Carol Gray
37a6d7a71c4f7c2469e4f01b70dd90c2
4   16.039              2        Blake Ford
471fe554e1c62d1b01cc8e4e5076c61a

                                        descripcion pollutant_code
\
0  Mineral industry . Installations for the produ...              1
NaN
1  Mineral industry . Installations for the produ...              0
NaN
2  Waste and wastewater management . Landfills (e...              2
NaN
3  Energy sector . Thermal power stations and oth...              0
NaN
4  Waste and wastewater management . Urban waste-...              2
NaN

  EPRTRAnnexIMainActivityCode EPRTRSectorCode
0                         NaN             NaN
1                         NaN             NaN
2                         NaN             NaN
3                         NaN             NaN
4                         NaN             NaN

[5 rows x 26 columns]
```

```
# mayor tipo de contaminacion
_=df_final.pollutant.value_counts().sort_values().plot(kind='barh')
```



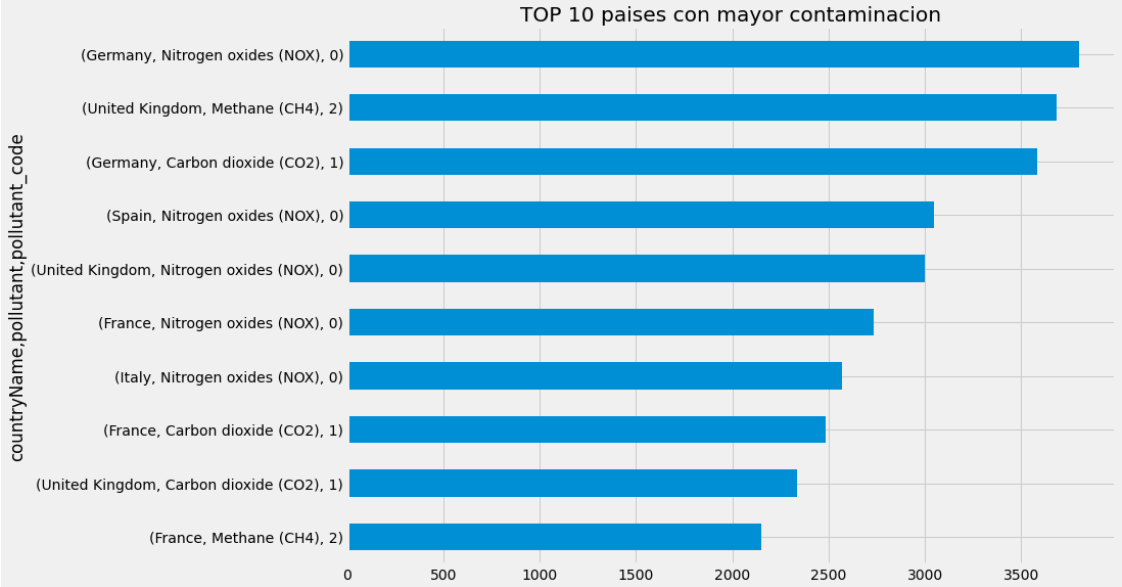

```
# mayor tipo de contaminacion
df_final.pollutant.value_counts().sort_values().plot(kind='barh')
```

```
df_final.columns
```

```
Index(['countryName', 'eprtrSectorName',
'EPRTRAnnexIMainActivityLabel',
       'FacilityInspireID', 'facilityName', 'City', 'targetRelease',
       'pollutant', 'reportingYear', 'MONTH', 'DAY', 'CONTINENT',
       'max_wind_speed', 'avg_wind_speed', 'min_wind_speed',
'max_temp',
       'avg_temp', 'min_temp', 'DAY WITH FOGS', 'REPORTER NAME', 'CITY
ID',
       'descripcion', 'pollutant_code', '',
'EPRTRAnnexIMainActivityCode',
       'EPRTRSectorCode'],
      dtype='object')
```

```
df_final.groupby(['countryName','pollutant'],observed=True).pollutant_
code.value_counts().sort_values(ascending=False).head(10).sort_values(
).plot(kind='barh',title='TOP 10 paises con mayor contaminacion')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6032e17a50>
```




```
df_final.groupby(['countryName','pollutant'],observed=True).pollutant_
code.mean().sort_values(ascending=False).head(10).sort_values().plot(k
ind='barh',title='TOP 10 paises con mayor contaminacion')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6032c6d910>
```