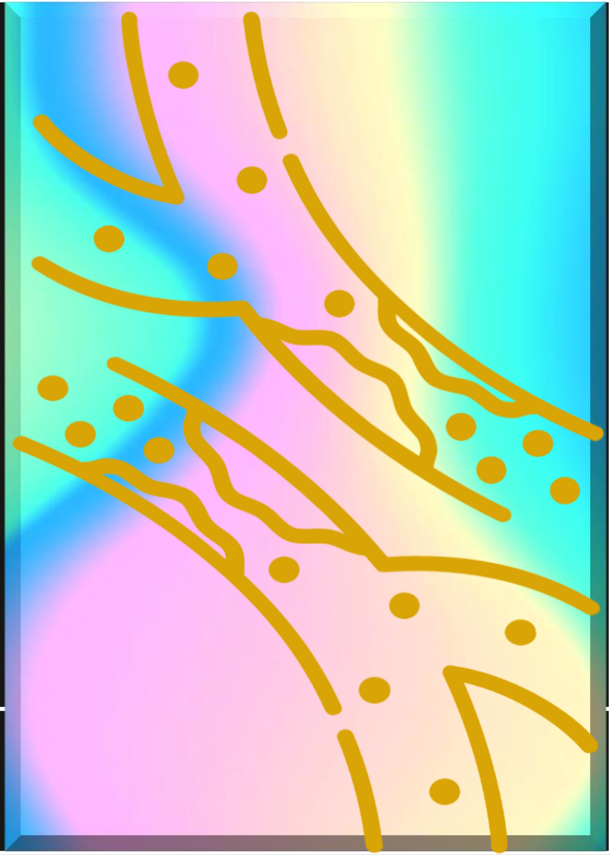
Thorreznou Library

For Python Programming Language



Data Science Bootcamp (promo NOV21)

✳ Objetivo libreria

Proporcionar a cualquier usuario de Python, una librería con funciones que faciliten su trabajo de programación, reduciendo la cantidad de código que requieren para obtener unos resultados excelentes.

Miguel Batalla
Manager Proyecto

La biblioteca contiene funciones agrupadas en:

- Exploratory Data Analysis
- Vizualization
- Feature Engineering & Machine Learning models

Index →

01 Exploratory Data Analysis

02 VisualizeME

03 Machine Learning

04 Production

01

Exploratory Data Analysis

Into the wild!

Objetivo de Exploratory Data Analysis

Nuestro objetivo es automatizar y reducir las acciones repetitivas y tediosas que siempre ocurren durante un EDA

Equipo EDA

David Guix
Manager EDA

Jorge
Arranz

Isabella

*Funciones Img

Funcion 1:
Resize Image

Funcion 2:
Reduce Image

Funcioin 3:
Process Color

Funcion 4:
Reduce Color Palettte

Funciones Imágenes

Funciones:

- Resize Image

Input: directory path

Output: Resized images list

- Reduce Image

Input: Image, height

Output: The reduced image

- Process Color

Input: cannel value, bins

Output: Processed value

- Reduce Color Palette

Input: Image, bins

Output: Image

✳ Funciones Datos

Funcion 1:
Math Expectation

Funcion 2:
DF Overview

Funcion 3:
Outlier Removal

Funcion 4:
Datetime

Funcion 5:
Missing Data

Funcion 6:
Standarize Numbers

....

Funciones Datos

Funciones:

- **Math Expectation**

Input: df, columns

Output: df with new ME column

- **DF Overview**

Input: df

Output: df with overview

- **Outlier Removal**

Input: df, columns

Output: df

- **Datetime**

Input: df, column

Output: df

- **Coordinates**

Input: lat, long

Output: lat, long

- **Missing data**

Input: df, columns

Output: df

- **Standarize numbers**

Input: list of numbers to standarize

Output: The reduced image

*

02

VisualizeME

Your favorite graphics!

Objetivo de Visualización

Obtener gráficos complejos de forma sencilla, incluyendo tablas con métricas descriptivas que complementen lo que visualmente se ve

Equipo Visualización

Marta Buesa
Manager
Visualization

→

Natalia
Fernández

→

Oscar
Olaechea

Thorreznou Library

Funciones

1. VisualizeME palettes and colors

When working on:

Exploratory Data Analysis

2. VisualizeME and describe Violinbox
3. VisualizeME and describe barplot
4. VisualizeME Figure Words
5. VisualizeME Bagel look top
6. VisualizeME Select graph

Feature Engineering & development of Machine Learning Models

7. VisualizeME and describe Spidey
8. VisualizeME c Matrix
9. VisualizeME scores models

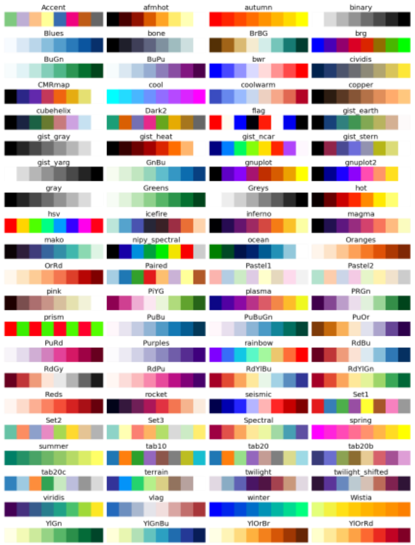
1. VisualizeME palettes and colors

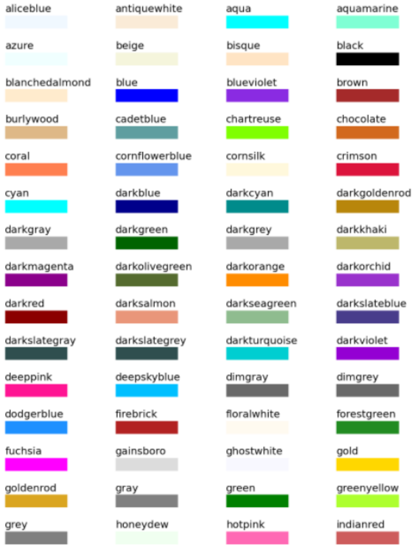
Parámetros entrada (2):

- `selection = 'palettes'`
- `quantity_colors= 8`

Return (1):

- `Image palettes Seaborn / Image colors Seaborn`





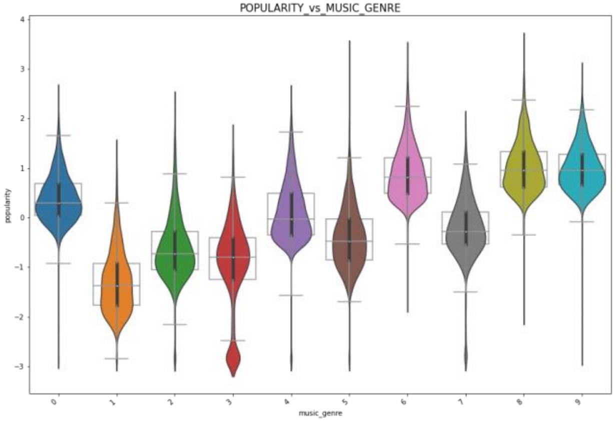
2. VisualizeME and describe Violinbox

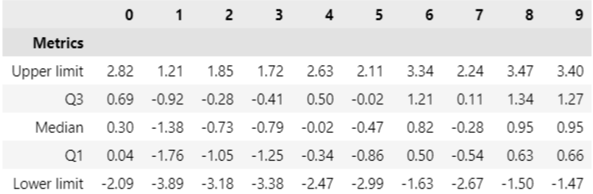
Parámetros entrada (5):

- dataframe
- categ_var
- numeric_var
- palette= 'tab10'
- save= True

Return (2):

- Graph Violin & Boxplot
- Table with metrics





	0	1	2	3	4	5	6	7	8	9
Metrics										
Upper limit	2.82	1.21	1.85	1.72	2.63	2.11	3.34	2.24	3.47	3.40
Q3	0.69	-0.92	-0.28	-0.41	0.50	-0.02	1.21	0.11	1.34	1.27
Median	0.30	-1.38	-0.73	-0.79	-0.02	-0.47	0.82	-0.28	0.95	0.95
Q1	0.04	-1.76	-1.05	-1.25	-0.34	-0.86	0.50	-0.54	0.63	0.66
Lower limit	-2.09	-3.89	-3.18	-3.38	-2.47	-2.99	-1.63	-2.67	-1.50	-1.47

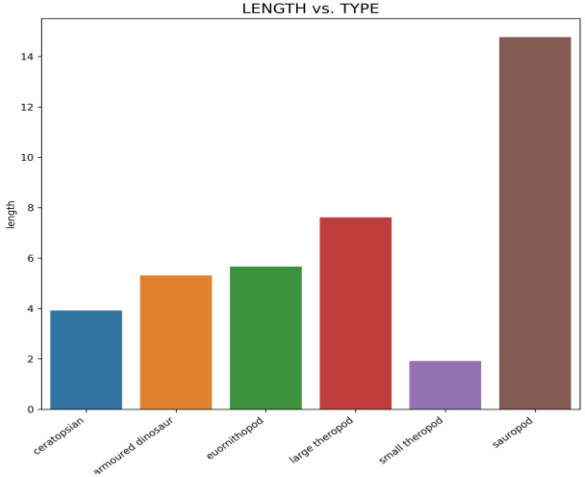
3. VisualizeME and describe barplot

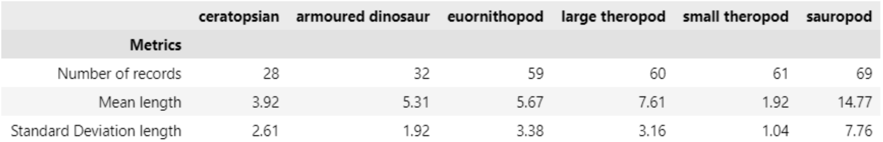
Parámetros entrada (5):

- dataframe
- categ_var
- numeric_var
- palette= 'tab10'
- save= True

Return (2):

- Graph Barplot
- Table with metrics





	ceratopsian	armoured dinosaur	euornithopod	large theropod	small theropod	sauropod
Metrics						
Number of records	28	32	59	60	61	69
Mean length	3.92	5.31	5.67	7.61	1.92	14.77
Standard Deviation length	2.61	1.92	3.38	3.16	1.04	7.76

4. VisualizeME Figure Words

Parámetros entrada (9):

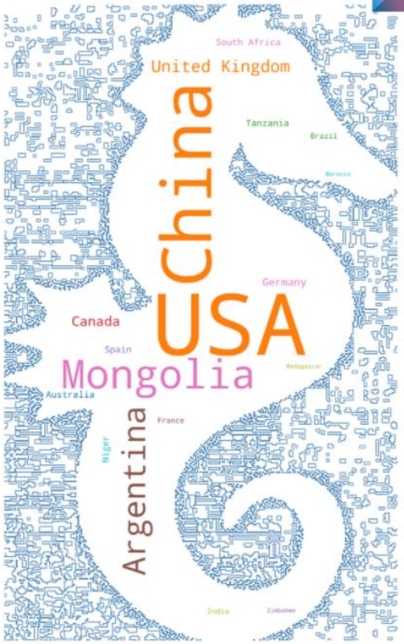
- Dataframe
- categ_var
- shape= 'seahorse
- cmap= 'tab10'
- contour= 'steelblue'
- back_color = 'white'
- height= 18
- width = 20
- save= True

Return (1):

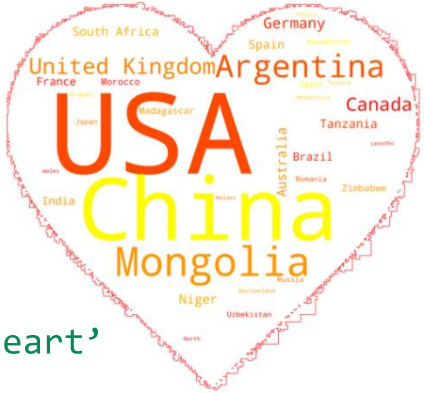
- Graph Figure Words

Figure words disponibles:

'seahorse'



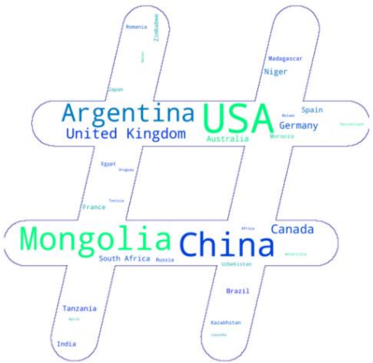
'heart'



4. VisualizeME Figure Words

Figure words disponibles:

‘hashtag’



United Kingdom
Mongolia
USA
Canada
China
Argentina
Germany
Spain
Tanzania
South Africa
Niger
Brazil
Zimbabwe
France

‘dino’

‘star’

Spain
Canada
Australia
South Africa
United
China
USA
Mongolia
Argentina
Niger
Tanzania
France
Brazil

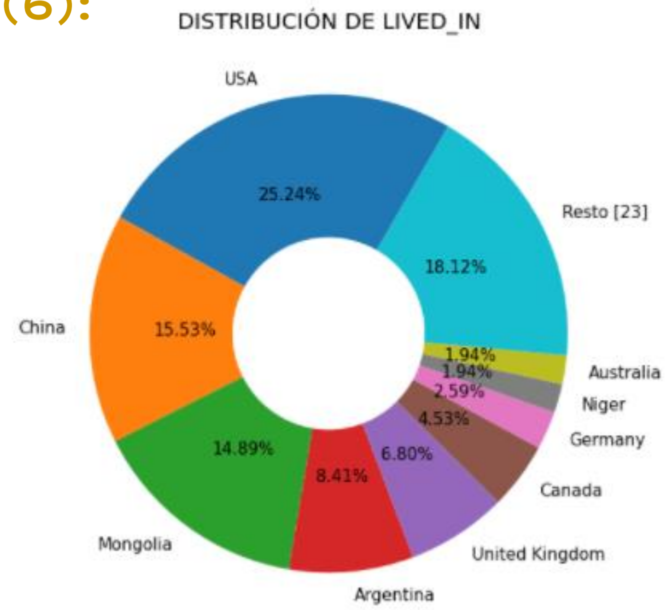
5. VisualizeME Bagel look top

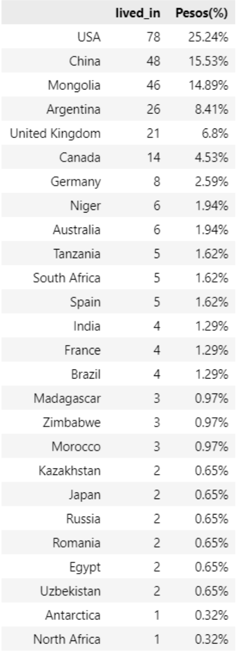
Parámetros entrada (6):

- `Dataframe`
- `categ_var`
- `top=0`
- `cmap = 'tab10’`
- `circle=True`
- `save=True`

Return (2):

- `Graph Bagel`
- `Table with metrics`





	lived_in	Pesos(%)
USA	78	25.24%
China	48	15.53%
Mongolia	46	14.89%
Argentina	26	8.41%
United Kingdom	21	6.8%
Canada	14	4.53%
Germany	8	2.59%
Niger	6	1.94%
Australia	6	1.94%
Tanzania	5	1.62%
South Africa	5	1.62%
Spain	5	1.62%
India	4	1.29%
France	4	1.29%
Brazil	4	1.29%
Madagascar	3	0.97%
Zimbabwe	3	0.97%
Morocco	3	0.97%
Kazakhstan	2	0.65%
Japan	2	0.65%
Russia	2	0.65%
Romania	2	0.65%
Egypt	2	0.65%
Uzbekistan	2	0.65%
Antarctica	1	0.32%
North Africa	1	0.32%

5. VisualizeME select graph

Obtener tu gráfico a la carta según tu necesidad.

PASO 1: ¿Cuántas variables quieres utilizar en tu gráfico? (1, 2 o 3)

PASO 2: ¿De qué tipo son tus variables? (categóricas o numéricas)

PASO 3: ¿Qué gráfico prefieres utilizar? (Ejemplos)

Return (1):

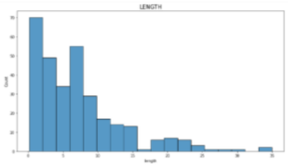
- Función que debe llamar para generar su gráfico

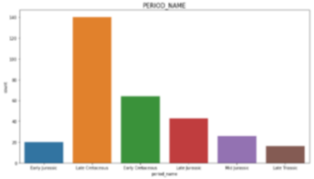
Opciones gráficos:

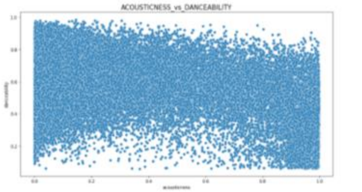
- COUNTPLOT
- PIE
- BOXPLOT
- VIOLINPLOT
- BARPLOT
- SCATTERPLOT
- LINE PLOT
- HISTPLOT
- DENSIDAD

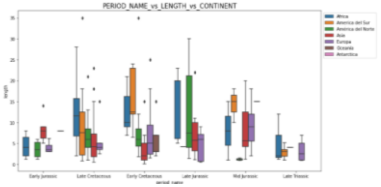
5. VisualizeME select graph

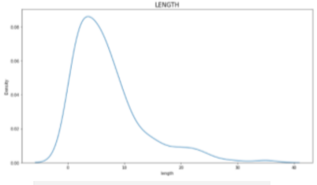
Ejemplos gráficos obtenidos:

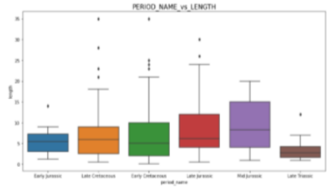


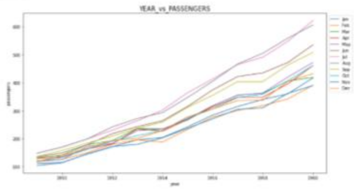


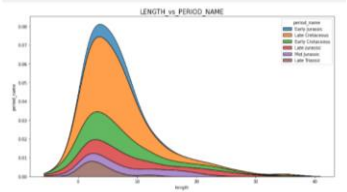


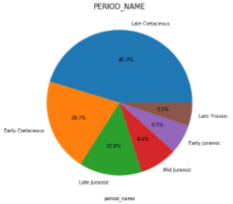


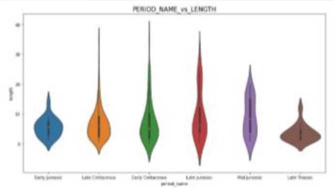


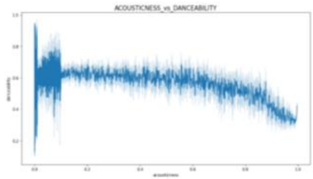


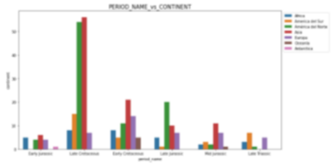












7. VisualizeME and describe Spidey

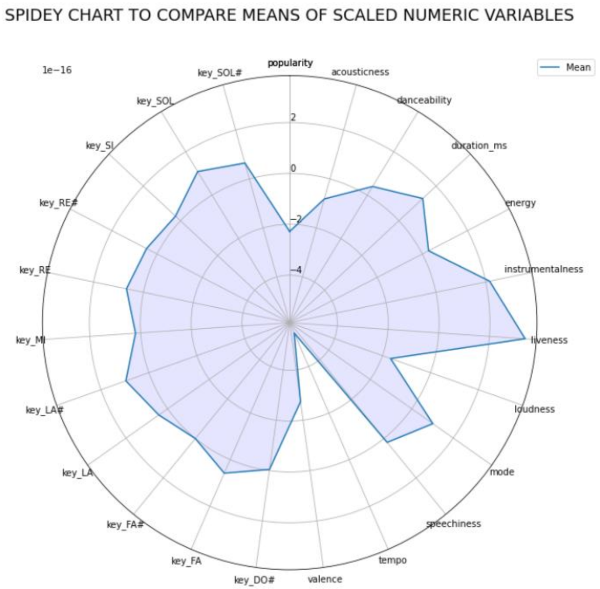
Esta función se utilizará con los datos escalados

Parámetros entrada (6):

- Dataframe
- save= True

Return (2):

- Polar graph
- Table with means of variables



	Means
popularity	-2.282667e-16
acousticness	-8.157494e-17
danceability	3.989301e-17
duration_ms	1.284447e-16
energy	2.772832e-17
instrumentalness	2.153865e-16
liveness	3.406111e-16
loudness	-1.663699e-16
mode	1.016109e-16
speechiness	1.985706e-17
tempo	-5.431173e-16
valence	-2.736159e-16
key_DO#	-4.382864e-18
key_FA	5.814003e-17
key_FA#	3.577848e-19
key_LA	4.239750e-17
key_LA#	9.624411e-17
key_MI	2.003595e-17
key_RE	6.797911e-17
key_RE#	4.651202e-17
key_SI	2.683386e-17
key_SOL	1.087666e-16
key_SOL#	6.511683e-17

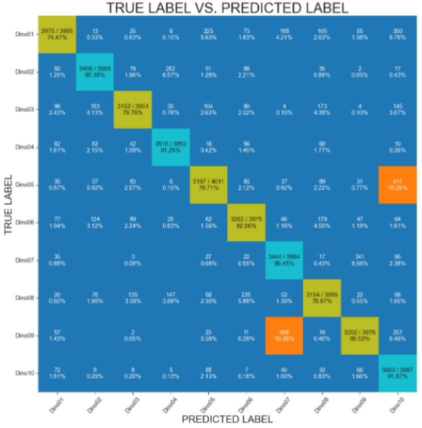
8. VisualizeME c Matrix

Parámetros entrada (9):

- `y_true,`
- `y_pred,`
- `title='',`
- `categories=[],`
- `rotate=False,`
- `cmap='',`
- `cbar=True,`
- `metrics=True,`
- `save=True`

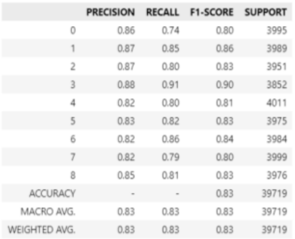
Return (2):

- Confusion Matrix
- Table with scores

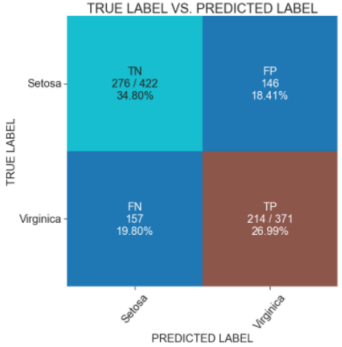

TRUE LABEL VS. PREDICTED LABEL

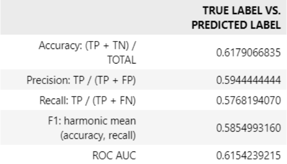
TRUE LABEL	Dino01	Dino02	Dino03	Dino04	Dino05	Dino06	Dino07	Dino08	Dino09	Dino10
Dino01	2975 / 3995 74.47%	13 0.33%	25 0.63%	6 0.15%	225 5.63%	73 1.83%	168 4.21%	105 2.63%	55 1.38%	350 8.76%
Dino02	50 1.25%	3406 / 3989 85.38%	78 1.96%	262 6.57%	51 1.28%	88 2.21%		35 0.88%	2 0.05%	17 0.43%
Dino03	96 2.43%	163 4.13%	3152 / 3951 79.78%	30 0.76%	104 2.63%	80 2.02%	4 0.10%	173 4.38%	4 0.10%	145 3.67%
Dino04	62 1.61%	83 2.15%	42 1.09%	3515 / 3852 91.25%	16 0.42%	56 1.45%		68 1.77%		10 0.26%
Dino05	35 0.87%	37 0.92%	83 2.07%	6 0.15%	3197 / 4011 79.71%	85 2.12%	37 0.92%	89 2.22%	31 0.77%	411 10.25%
Dino06	77 1.94%	124 3.12%	89 2.24%	25 0.63%	62 1.56%	3262 / 3975 82.06%	46 1.16%	179 4.50%	47 1.18%	64 1.61%
Dino07	35 0.88%		3 0.08%		27 0.68%	22 0.55%	3444 / 3984 86.45%	17 0.43%	341 8.56%	95 2.38%
Dino08	20 0.50%	76 1.90%	135 3.38%	147 3.68%	92 2.30%	235 5.88%	52 1.30%	3154 / 3999 78.87%	22 0.55%	66 1.65%
Dino09	57 1.43%		2 0.05%		23 0.58%	11 0.28%	408 10.26%	16 0.40%	3202 / 3976 80.53%	257 6.46%
Dino10	72 1.81%	8 0.20%	8 0.20%	5 0.13%	85 2.13%	7 0.18%	40 1.00%	33 0.83%	66 1.66%	3663 / 3987 91.87%

TRUE LABEL
PREDICTED LABEL



	PRECISION	RECALL	F1-SCORE	SUPPORT
0	0.86	0.74	0.80	3995
1	0.87	0.85	0.86	3989
2	0.87	0.80	0.83	3951
3	0.88	0.91	0.90	3852
4	0.82	0.80	0.81	4011
5	0.83	0.82	0.83	3975
6	0.82	0.86	0.84	3984
7	0.82	0.79	0.80	3999
8	0.85	0.81	0.83	3976
ACCURACY	-	-	0.83	39719
MACRO AVG.	0.83	0.83	0.83	39719
WEIGHTED AVG.	0.83	0.83	0.83	39719


TRUE LABEL VS. PREDICTED LABEL



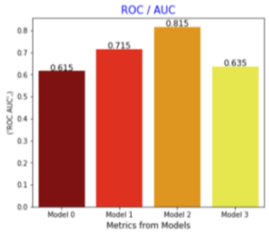
	TRUE LABEL VS. PREDICTED LABEL
Accuracy: (TP + TN) / TOTAL	0.6179066835
Precision: TP / (TP + FP)	0.5944444444
Recall: TP / (TP + FN)	0.5768194070
F1: harmonic mean (accuracy, recall)	0.5854993160
ROC AUC	0.6154239215

9. VisualizeME scores models

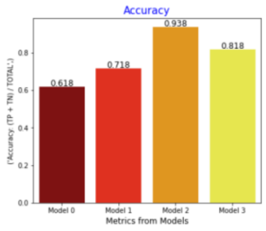
Parámetros entrada (5):

- y_true
- Models (dictionary format)
- bin_multi_classifier=True
- pallete='tab10’
- save=True

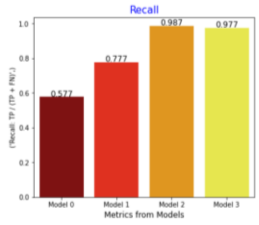
```
The best ROC / AUC value is 0.8154239215 from Model: Model 1
```



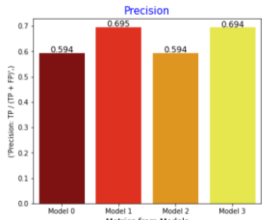
```
The best Accuracy value is 0.9379066835 from Model: 2
```



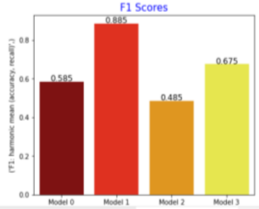
```
The best Recall value is 0.986819407 from Model: Model 1
```



```
The best Precision value is 0.6954444444 from Model: Model 1
```

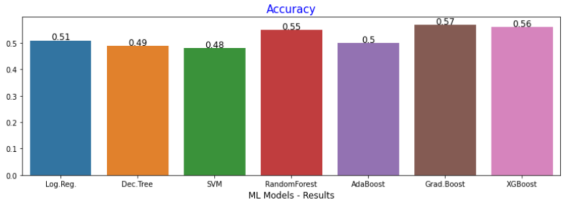


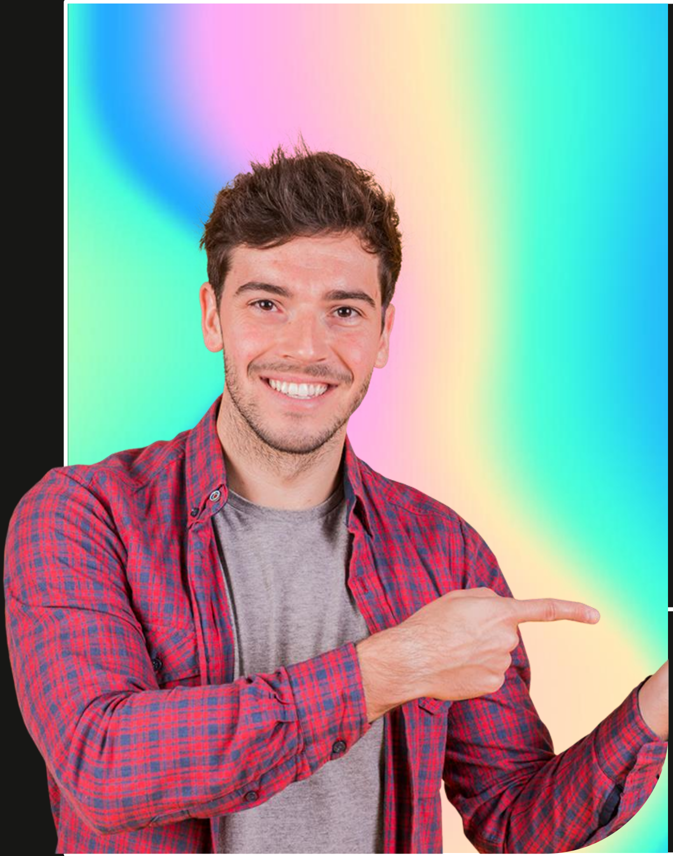
```
The best F1 Score value is 0.885499316 from Model: Model 1
```



Return (1):

- Graphs barplot comparing scores from different models





03

Machine Learning

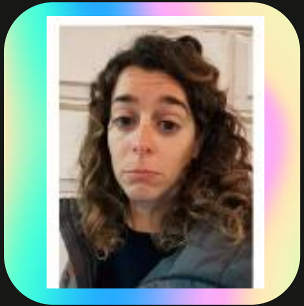
Where the magic Is made

Objetivo de Machine Learning

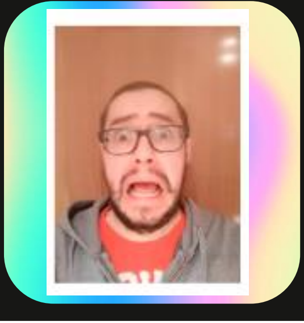
El objetivo principal de la parte de Machine Learning es acelerar la toma de decisiones sobre el modelo a implementar

✳ Equipo Machine Learning

Daniel Vivas
Manager ML



Ana
Genua



Fernando
Bielza

✳ Funciones

1. PREPROCESSING:
   - División train/test
   - Preparación dataset

2. REGRESORES:
   - Regresión lineal
   - Regresión polinómica

3. CLASIFICADORES:
   - Regresión Logística
   - RandomForest
   - SVC
   - KNN

4. SCORING:
   - Métricas regresores
   - Métricas clasificadores

1. PREPARACIÓN DATOS

Funciones:

- Divide

Input: dataset

Output: X, y

- prepare_data

Input: dataset

Output: list of train/test sets

División X y

Balanceado

RandomUnderSampler / SMOTE

División train/test

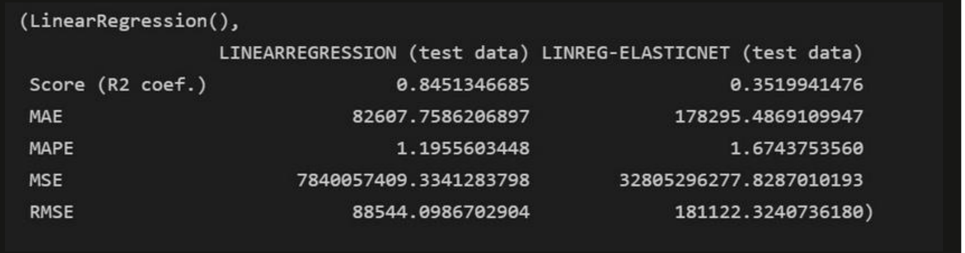
Estandarización

StandardScaler / MinMaxScaler

2. REGRESORES

Funciones:

- Regresión lineal
- Regresión polinómica



```
(LinearRegression(),
                   LINEARREGRESSION (test data) LINREG-ELASTICNET (test data)
 Score (R2 coef.)                 0.8451346685                0.3519941476
 MAE                           82607.7586206897            178295.4869109947
 MAPE                             1.1955603448                1.6743753560
 MSE                      7840057409.3341283798      32805296277.8287010193
 RMSE                          88544.0986702904            181122.3240736180)
```

Input: X_train, X_test, y_train, y_test, regular_type

Output: model, metrics

Regularización

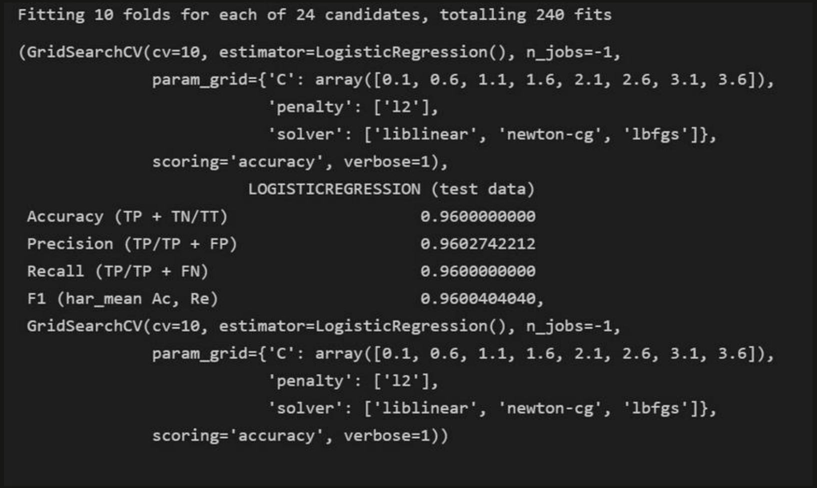
3. CLASIFICADORES

Funciones:

- Regresión logística
- RandomForest
- SVC
- KNN

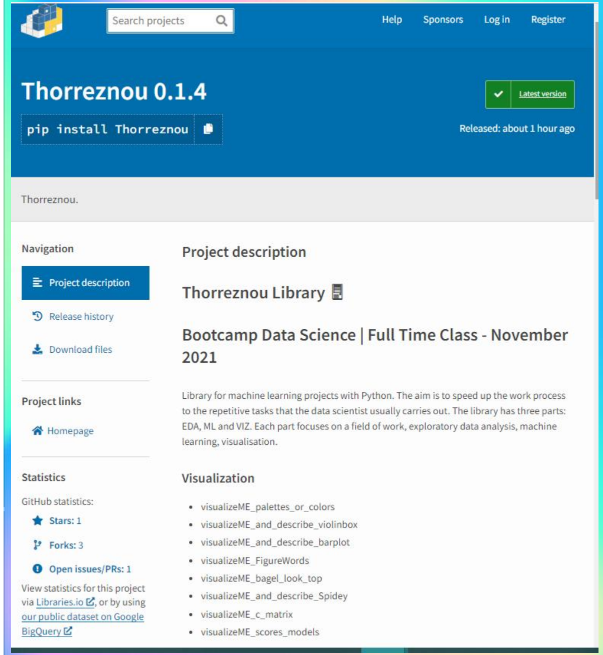
Input: X_train, X_test, y_train, y_test

Output: model, metrics, grid



```
Fitting 10 folds for each of 24 candidates, totalling 240 fits

(GridSearchCV(cv=10, estimator=LogisticRegression(), n_jobs=-1,
              param_grid={'C': array([0.1, 0.6, 1.1, 1.6, 2.1, 2.6, 3.1, 3.6]),
                          'penalty': ['l2'],
                          'solver': ['liblinear', 'newton-cg', 'lbfgs']},
              scoring='accuracy', verbose=1),
                        LOGISTICREGRESSION (test data)
 Accuracy (TP + TN/TT)                   0.9600000000
 Precision (TP/TP + FP)                  0.9602742212
 Recall (TP/TP + FN)                     0.9600000000
 F1 (har_mean Ac, Re)                    0.9600404040,
 GridSearchCV(cv=10, estimator=LogisticRegression(), n_jobs=-1,
              param_grid={'C': array([0.1, 0.6, 1.1, 1.6, 2.1, 2.6, 3.1, 3.6]),
                          'penalty': ['l2'],
                          'solver': ['liblinear', 'newton-cg', 'lbfgs']},
              scoring='accuracy', verbose=1))
```



Thorreznou 0.1.4

```
pip install Thorreznou
```

Released: about 1 hour ago

Thorreznou.

Statistics

GitHub statistics:

- Stars: 1
- Forks: 3
- Open issues/PRs: 1

View statistics for this project via Libraries.io, or by using our public dataset on Google BigQuery

Project description

Thorreznou Library

Bootcamp Data Science | Full Time Class - November 2021

Library for machine learning projects with Python. The aim is to speed up the work process to the repetitive tasks that the data scientist usually carries out. The library has three parts: EDA, ML and VIZ. Each part focuses on a field of work, exploratory data analysis, machine learning, visualisation.

Visualization

- visualizeME_palettes_or_colors
- visualizeME_and_describe_violinbox
- visualizeME_and_describe_barplot
- visualizeME_FigureWords
- visualizeME_bagel_look_top
- visualizeME_and_describe_Spidey
- visualizeME_c_matrix
- visualizeME_scores_models

*

04

Producción

Make it come true!

Objetivo de Producción

Creación /instalación del package

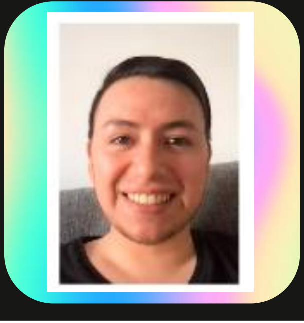
✳ Equipo Producción

Javier Mesonero
Manager
Producción

→

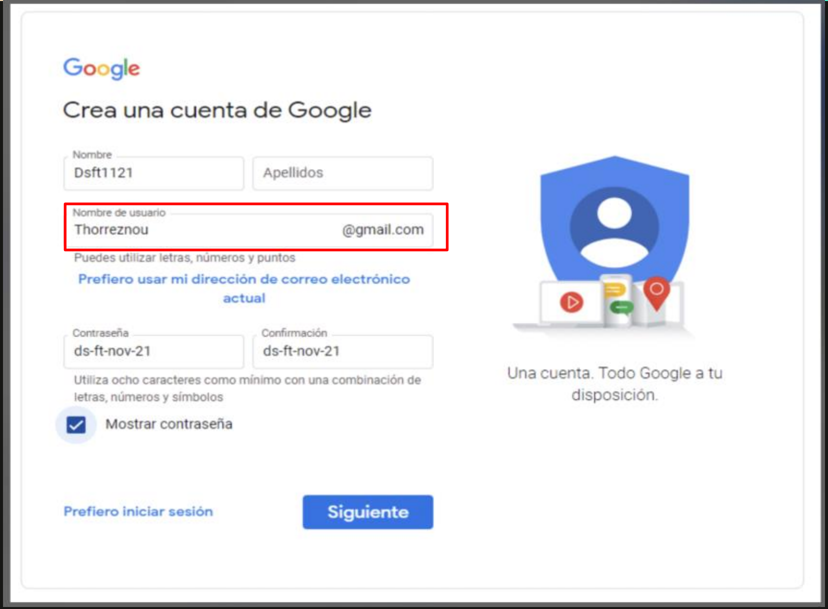
→

Federico
Ruiz



Erik
Urresta

1. Crear una cuenta Gmail



Google

Crea una cuenta de Google

Nombre: Dsft1121

Apellidos

Nombre de usuario: Thorreznou @gmail.com

Puedes utilizar letras, números y puntos

Prefiero usar mi dirección de correo electrónico actual

Contraseña: ds-ft-nov-21

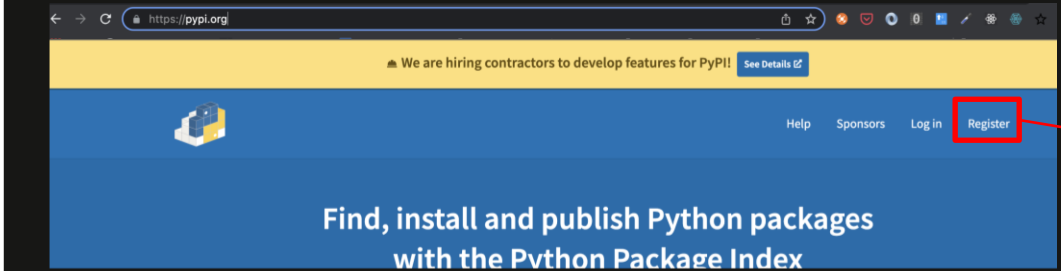
Confirmación: ds-ft-nov-21

Utiliza ocho caracteres como mínimo con una combinación de letras, números y símbolos

☑ Mostrar contraseña

Una cuenta. Todo Google a tu disposición.

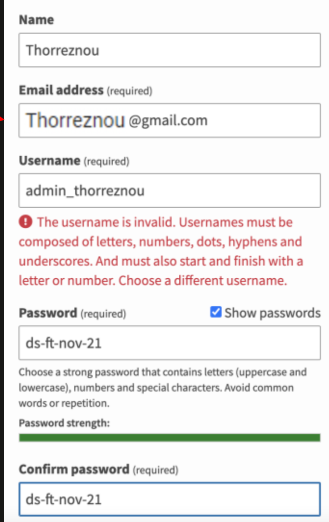
2. Crear una cuenta https://pypi.org/



https://pypi.org

We are hiring contractors to develop features for PyPI! See Details

Find, install and publish Python packages with the Python Package Index



Name

Thorreznou

Email address (required)

Thorreznou @gmail.com

Username (required)

admin_thorreznou

The username is invalid. Usernames must be composed of letters, numbers, dots, hyphens and underscores. And must also start and finish with a letter or number. Choose a different username.

Password (required) ☑ Show passwords

ds-ft-nov-21

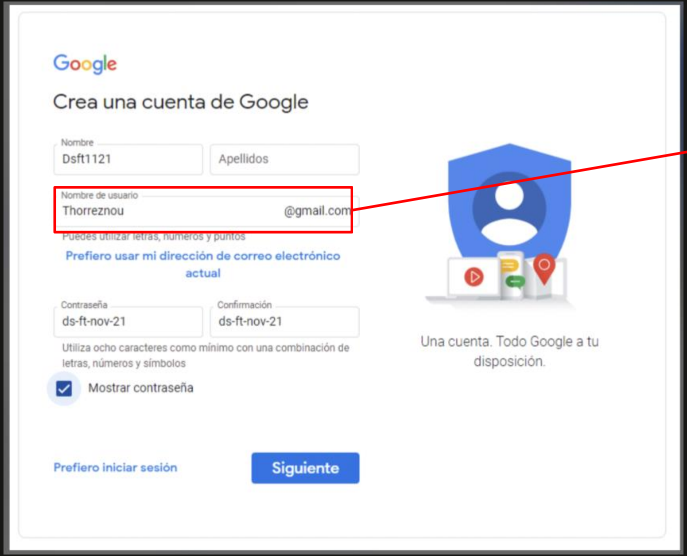
Choose a strong password that contains letters (uppercase and lowercase), numbers and special characters. Avoid common words or repetition.

Password strength:

Confirm password (required)

ds-ft-nov-21

Usar la cuenta de Gmail al registrarnos en https://pypi.org/



Google

Crea una cuenta de Google

Nombre: Dsft1121

Apellidos

Nombre de usuario: Thorreznou @gmail.com

Puedes utilizar letras, números y puntos

Prefiero usar mi dirección de correo electrónico actual

Contraseña: ds-ft-nov-21

Confirmación: ds-ft-nov-21

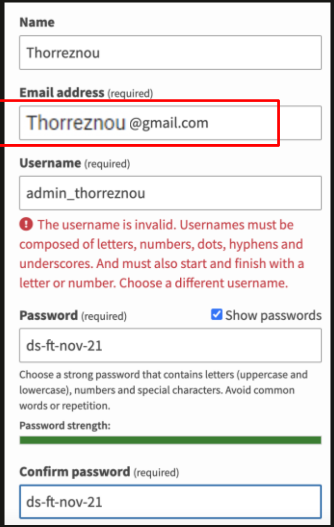
Utiliza ocho caracteres como mínimo con una combinación de letras, números y símbolos

☑ Mostrar contraseña

Prefiero iniciar sesión

Siguiente

Una cuenta. Todo Google a tu disposición.



Name

Thorreznou

Email address (required)

Thorreznou @gmail.com

Username (required)

admin_thorreznou

The username is invalid. Usernames must be composed of letters, numbers, dots, hyphens and underscores. And must also start and finish with a letter or number. Choose a different username.

Password (required) ☑ Show passwords

ds-ft-nov-21

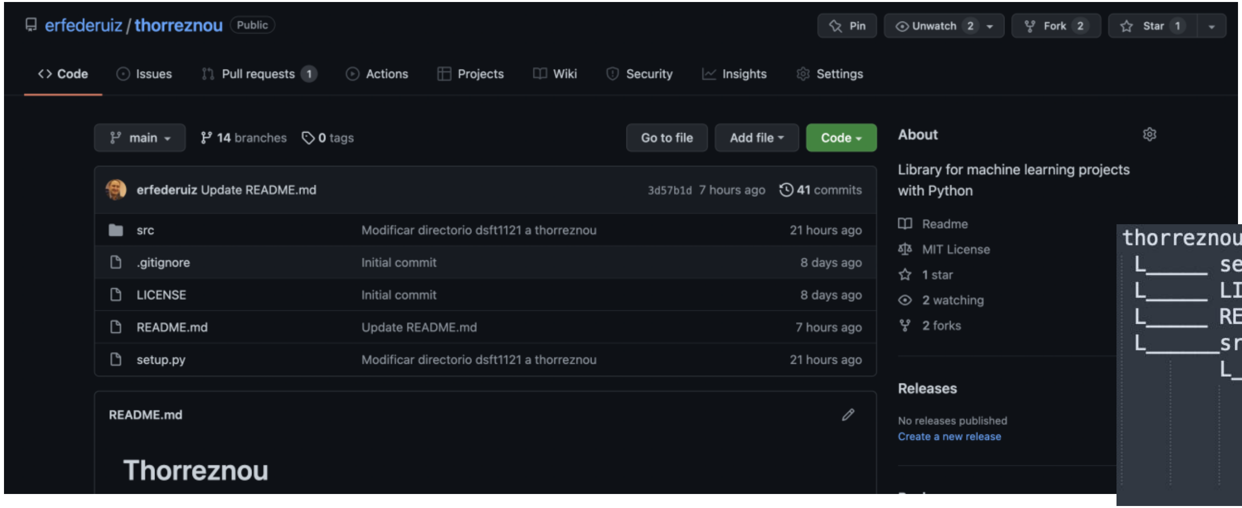
Choose a strong password that contains letters (uppercase and lowercase), numbers and special characters. Avoid common words or repetition.

Password strength:

Confirm password (required)

ds-ft-nov-21

3. Git clone del repo



erfederuiz / thorreznou Public

erfederuiz Update README.md 3d57b1d 7 hours ago 41 commits

src	Modificar directorio dsft1121 a thorreznou	21 hours ago
.gitignore	Initial commit	8 days ago
LICENSE	Initial commit	8 days ago
README.md	Update README.md	7 hours ago
setup.py	Modificar directorio dsft1121 a thorreznou	21 hours ago

README.md

Thorreznou

About

Library for machine learning projects with Python

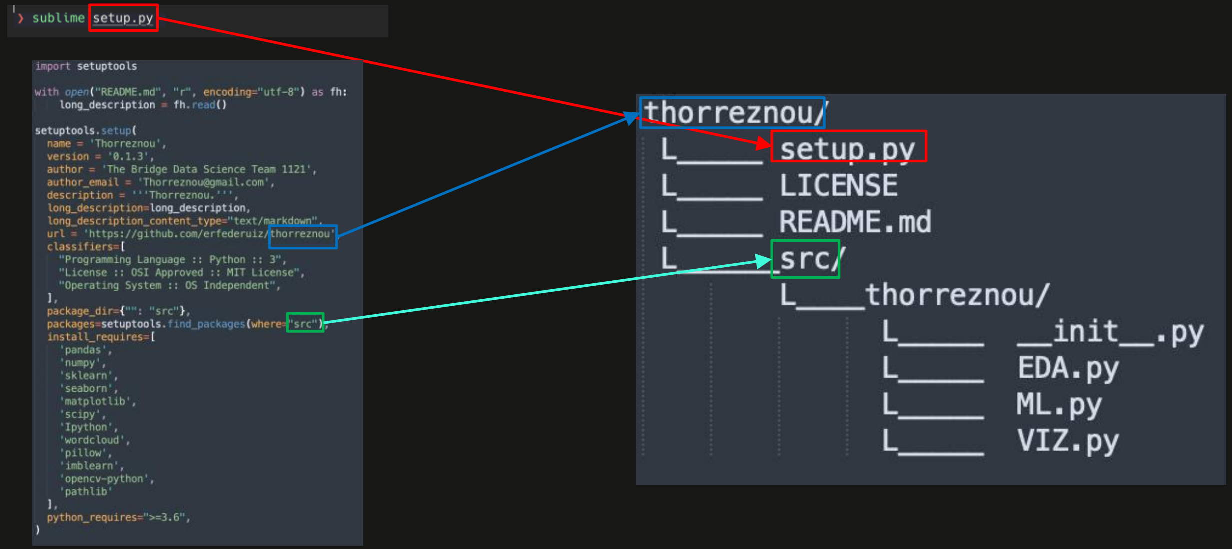
Readme
MIT License
1 star
2 watching
2 forks

Releases

No releases published
Create a new release

```
thorreznou/
 L______ setup.py
 L______ LICENSE
 L______ README.md
 L_______src/
          L____thorreznou/
                L______   __init__.py
                L______   EDA.py
                L______   ML.py
                L______   VIZ.py
```

4. Configuración del archivo de contrucción, setup.py



```
❯ sublime setup.py
```

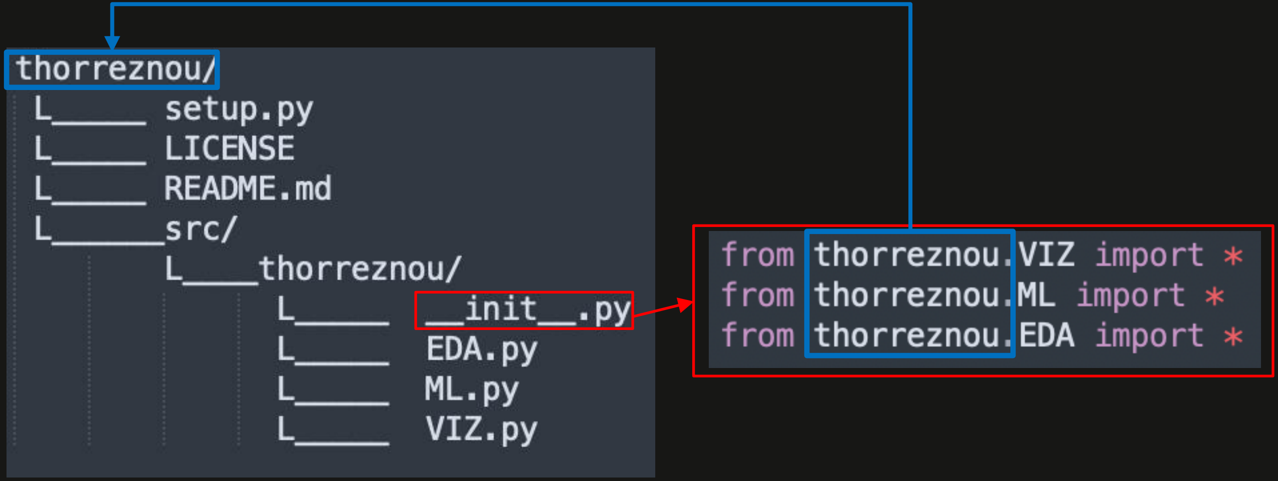
```python
import setuptools

with open("README.md", "r", encoding="utf-8") as fh:
    long_description = fh.read()

setuptools.setup(
  name = 'Thorreznou',
  version = '0.1.3',
  author = 'The Bridge Data Science Team 1121',
  author_email = 'Thorreznou@gmail.com',
  description = '''Thorreznou.''',
  long_description=long_description,
  long_description_content_type="text/markdown",
  url = 'https://github.com/erfederuiz/thorreznou',
  classifiers=[
    "Programming Language :: Python :: 3",
    "License :: OSI Approved :: MIT License",
    "Operating System :: OS Independent",
  ],
  package_dir={"": "src"},
  packages=setuptools.find_packages(where="src"),
  install_requires=[
    'pandas',
    'numpy',
    'sklearn',
    'seaborn',
    'matplotlib',
    'scipy',
    'Ipython',
    'wordcloud',
    'pillow',
    'imblearn',
    'opencv-python',
    'pathlib'
  ],
  python_requires=">=3.6",
)
```

```
thorreznou/
  L______ setup.py
  L______ LICENSE
  L______ README.md
  L______ src/
          L____thorreznou/
                L______  __init__.py
                L______  EDA.py
                L______  ML.py
                L______  VIZ.py
```

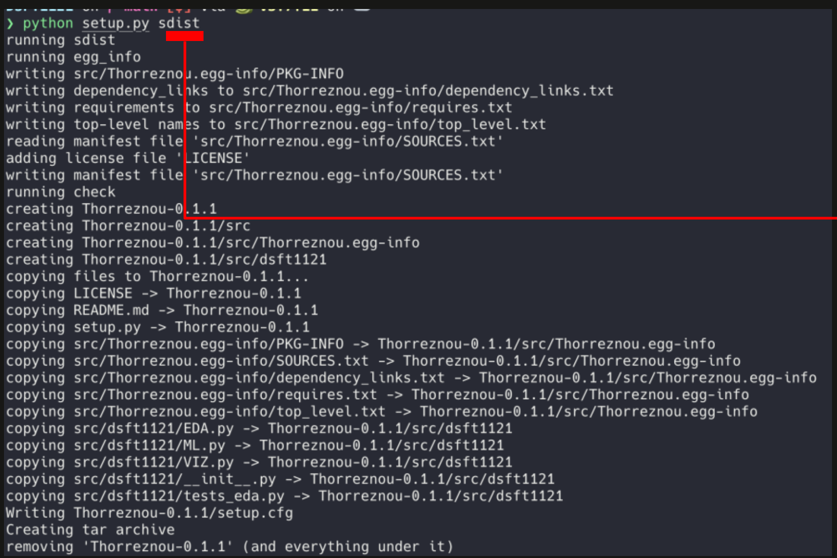
5. Configurar el archivo __init__.py



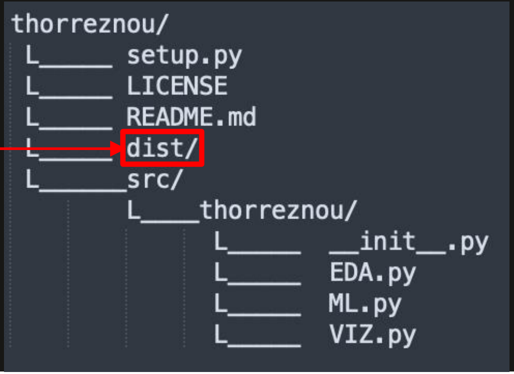
```
thorreznou/
 L_____ setup.py
 L_____ LICENSE
 L_____ README.md
 L______src/
           L____thorreznou/
                    L_____ __init__.py
                    L_____  EDA.py
                    L_____  ML.py
                    L_____  VIZ.py
```

```python
from thorreznou.VIZ import *
from thorreznou.ML import *
from thorreznou.EDA import *
```

6. Generación local del package



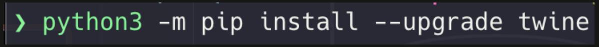
```
❯ python setup.py sdist
running sdist
running egg_info
writing src/Thorreznou.egg-info/PKG-INFO
writing dependency_links to src/Thorreznou.egg-info/dependency_links.txt
writing requirements to src/Thorreznou.egg-info/requires.txt
writing top-level names to src/Thorreznou.egg-info/top_level.txt
reading manifest file 'src/Thorreznou.egg-info/SOURCES.txt'
adding license file 'LICENSE'
writing manifest file 'src/Thorreznou.egg-info/SOURCES.txt'
running check
creating Thorreznou-0.1.1
creating Thorreznou-0.1.1/src
creating Thorreznou-0.1.1/src/Thorreznou.egg-info
creating Thorreznou-0.1.1/src/dsft1121
copying files to Thorreznou-0.1.1...
copying LICENSE -> Thorreznou-0.1.1
copying README.md -> Thorreznou-0.1.1
copying setup.py -> Thorreznou-0.1.1
copying src/Thorreznou.egg-info/PKG-INFO -> Thorreznou-0.1.1/src/Thorreznou.egg-info
copying src/Thorreznou.egg-info/SOURCES.txt -> Thorreznou-0.1.1/src/Thorreznou.egg-info
copying src/Thorreznou.egg-info/dependency_links.txt -> Thorreznou-0.1.1/src/Thorreznou.egg-info
copying src/Thorreznou.egg-info/requires.txt -> Thorreznou-0.1.1/src/Thorreznou.egg-info
copying src/Thorreznou.egg-info/top_level.txt -> Thorreznou-0.1.1/src/Thorreznou.egg-info
copying src/dsft1121/EDA.py -> Thorreznou-0.1.1/src/dsft1121
copying src/dsft1121/ML.py -> Thorreznou-0.1.1/src/dsft1121
copying src/dsft1121/VIZ.py -> Thorreznou-0.1.1/src/dsft1121
copying src/dsft1121/__init__.py -> Thorreznou-0.1.1/src/dsft1121
copying src/dsft1121/tests_eda.py -> Thorreznou-0.1.1/src/dsft1121
Writing Thorreznou-0.1.1/setup.cfg
Creating tar archive
removing 'Thorreznou-0.1.1' (and everything under it)
```



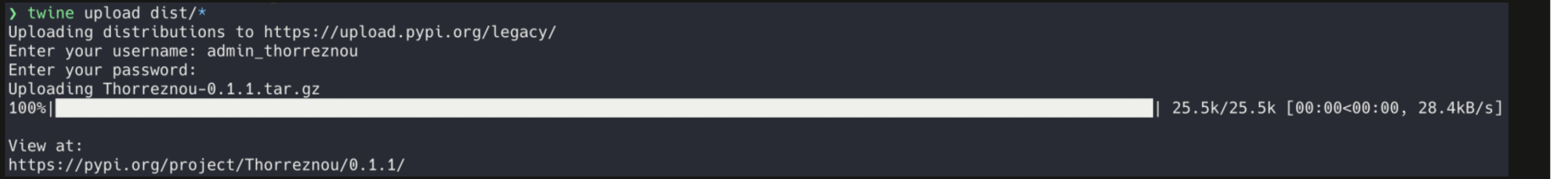
```
thorreznou/
 L_____ setup.py
 L_____ LICENSE
 L_____ README.md
 L_____ dist/
 L______src/
        L____thorreznou/
             L_____  __init__.py
             L_____  EDA.py
             L_____  ML.py
             L_____  VIZ.py
```

El directorio dist no existe hasta ejecutar python setup.py sdist !!

7. Publicación del package en PyPI



```
❯ python3 -m pip install --upgrade twine
```

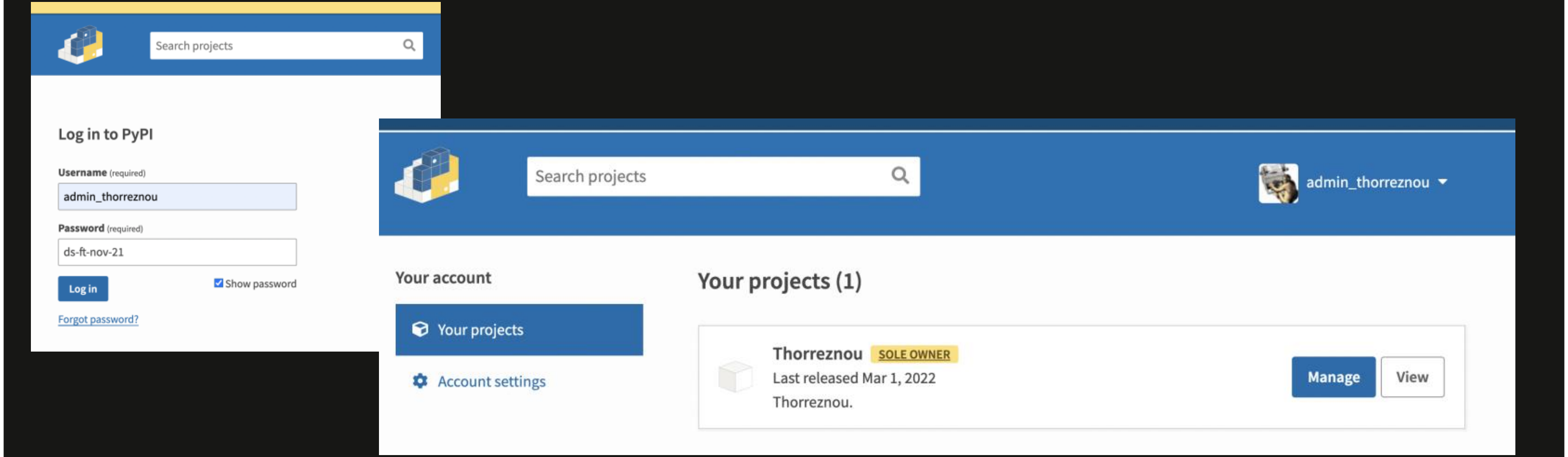


```
❯ twine upload dist/*
Uploading distributions to https://upload.pypi.org/legacy/
Enter your username: admin_thorreznou
Enter your password:
Uploading Thorreznou-0.1.1.tar.gz
100%|████████████████████████████████████████████████████████| 25.5k/25.5k [00:00<00:00, 28.4kB/s]

View at:
https://pypi.org/project/Thorreznou/0.1.1/
```

8. Comprobación de la subida a PyPI

https://pypi.org/



Log in to PyPI

Username (required)

admin_thorreznou

Password (required)

ds-ft-nov-21

Your account

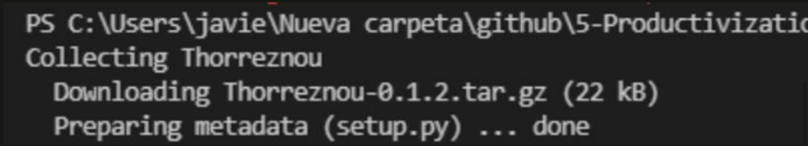
Your projects (1)

Thorreznou SOLE OWNER

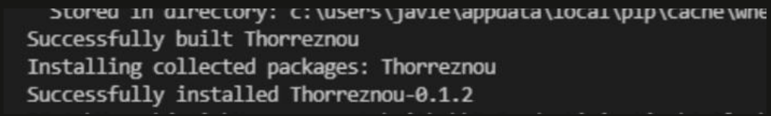
Last released Mar 1, 2022

Thorreznou.

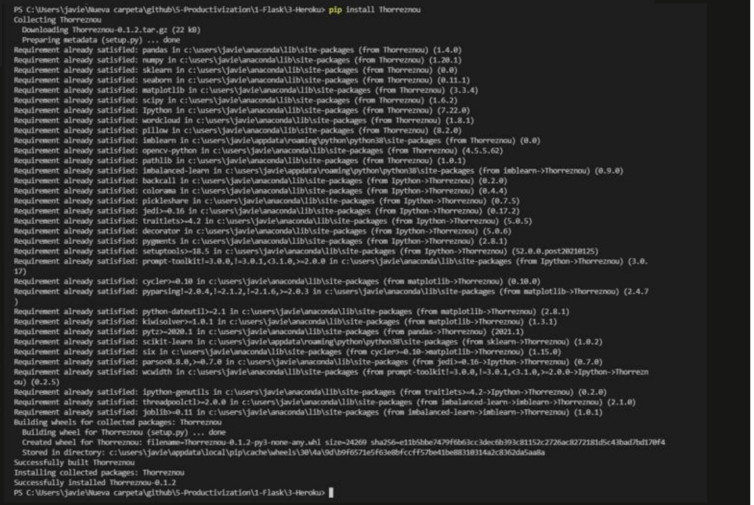
9. Instalación del package en nuestro Python



```
PS C:\Users\javie\Nueva carpeta\github\5-Productivization\1-Flask\3-Heroku> pip install Thorreznou
Collecting Thorreznou
  Downloading Thorreznou-0.1.2.tar.gz (22 kB)
  Preparing metadata (setup.py) ... done
```

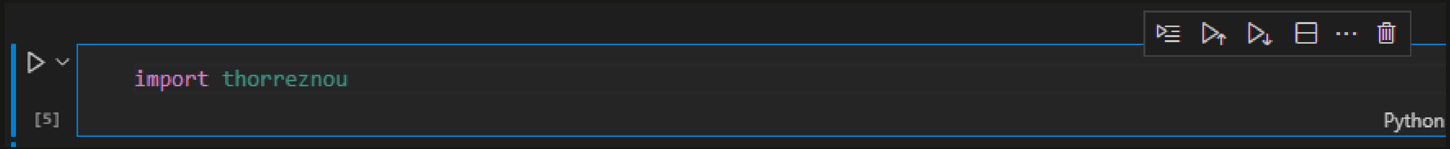


```
  Stored in directory: c:\users\javie\appdata\local\pip\cache\whe
Successfully built Thorreznou
Installing collected packages: Thorreznou
Successfully installed Thorreznou-0.1.2
```



```
PS C:\Users\javie\Nueva carpeta\github\5-Productivization\1-Flask\3-Heroku> pip install Thorreznou
Collecting Thorreznou
  Downloading Thorreznou-0.1.2.tar.gz (22 kB)
  Preparing metadata (setup.py) ... done
Requirement already satisfied: pandas in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (1.4.0)
Requirement already satisfied: numpy in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (1.20.1)
Requirement already satisfied: sklearn in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (0.0)
Requirement already satisfied: seaborn in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (0.11.1)
Requirement already satisfied: matplotlib in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (3.3.4)
Requirement already satisfied: scipy in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (1.6.2)
Requirement already satisfied: Ipython in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (7.22.0)
Requirement already satisfied: wordcloud in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (1.8.1)
Requirement already satisfied: pillow in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (8.2.0)
Requirement already satisfied: imblearn in c:\users\javie\appdata\roaming\python\python38\site-packages (from Thorreznou) (0.0)
Requirement already satisfied: opencv-python in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (4.5.5.62)
Requirement already satisfied: pathlib in c:\users\javie\anaconda\lib\site-packages (from Thorreznou) (1.0.1)
Requirement already satisfied: imbalanced-learn in c:\users\javie\appdata\roaming\python\python38\site-packages (from imblearn->Thorreznou) (0.9.0)
Requirement already satisfied: backcall in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (0.2.0)
Requirement already satisfied: colorama in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (0.4.4)
Requirement already satisfied: pickleshare in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (0.7.5)
Requirement already satisfied: jedi>=0.16 in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (0.17.2)
Requirement already satisfied: traitlets>=4.2 in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (5.0.5)
Requirement already satisfied: decorator in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (5.0.6)
Requirement already satisfied: pygments in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (2.8.1)
Requirement already satisfied: setuptools>=18.5 in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (52.0.0.post20210125)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in c:\users\javie\anaconda\lib\site-packages (from Ipython->Thorreznou) (3.0.
17)
Requirement already satisfied: cycler>=0.10 in c:\users\javie\anaconda\lib\site-packages (from matplotlib->Thorreznou) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\users\javie\anaconda\lib\site-packages (from matplotlib->Thorreznou) (2.4.7
)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\javie\anaconda\lib\site-packages (from matplotlib->Thorreznou) (2.8.1)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\javie\anaconda\lib\site-packages (from matplotlib->Thorreznou) (1.3.1)
Requirement already satisfied: pytz>=2020.1 in c:\users\javie\anaconda\lib\site-packages (from pandas->Thorreznou) (2021.1)
Requirement already satisfied: scikit-learn in c:\users\javie\appdata\roaming\python\python38\site-packages (from sklearn->Thorreznou) (1.0.2)
Requirement already satisfied: six in c:\users\javie\anaconda\lib\site-packages (from cycler>=0.10->matplotlib->Thorreznou) (1.15.0)
Requirement already satisfied: parso<0.8.0,>=0.7.0 in c:\users\javie\anaconda\lib\site-packages (from jedi>=0.16->Ipython->Thorreznou) (0.7.0)
Requirement already satisfied: wcwidth in c:\users\javie\anaconda\lib\site-packages (from prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->Ipython->Thorrezn
ou) (0.2.5)
Requirement already satisfied: ipython-genutils in c:\users\javie\anaconda\lib\site-packages (from traitlets>=4.2->Ipython->Thorreznou) (0.2.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\javie\anaconda\lib\site-packages (from imbalanced-learn->imblearn->Thorreznou) (2.1.0)
Requirement already satisfied: joblib>=0.11 in c:\users\javie\anaconda\lib\site-packages (from imbalanced-learn->imblearn->Thorreznou) (1.0.1)
Building wheels for collected packages: Thorreznou
  Building wheel for Thorreznou (setup.py) ... done
  Created wheel for Thorreznou: filename=Thorreznou-0.1.2-py3-none-any.whl size=24269 sha256=e11b5bbe7479f6b63cc3dec6b393c81152c2726ac8272181d5c43bad7bd170f4
  Stored in directory: c:\users\javie\appdata\local\pip\cache\wheels\30\4a\9d\b9f6571e5f63e8bfccff57be41be88310314a2c8362da5aa8a
Successfully built Thorreznou
Installing collected packages: Thorreznou
Successfully installed Thorreznou-0.1.2
PS C:\Users\javie\Nueva carpeta\github\5-Productivization\1-Flask\3-Heroku> 
```

10. Uso del package instalado



```
import thorreznou
```

¡Gracias!

Thorreznou Library

Puedes visitar la libraría aquí:

https://pypi.org/project/Thorreznou/

https://github.com/erfederuiz/thorreznou

Y tras usarla irte a descansar y no quedarte así: