Python基础与回归机器学习模型

唐 刚，邵明

2024.07

学术之友微信公众号(专注于分享理论计算教程)：dft_family

VASPKIT微信公众号(分享VASPKIT软件最新动态)：VASPKIT

机器学习势 QQ群(MLP各类软件交流学习)：783405103

主要内容：

01
OPTION

Python环境搭建

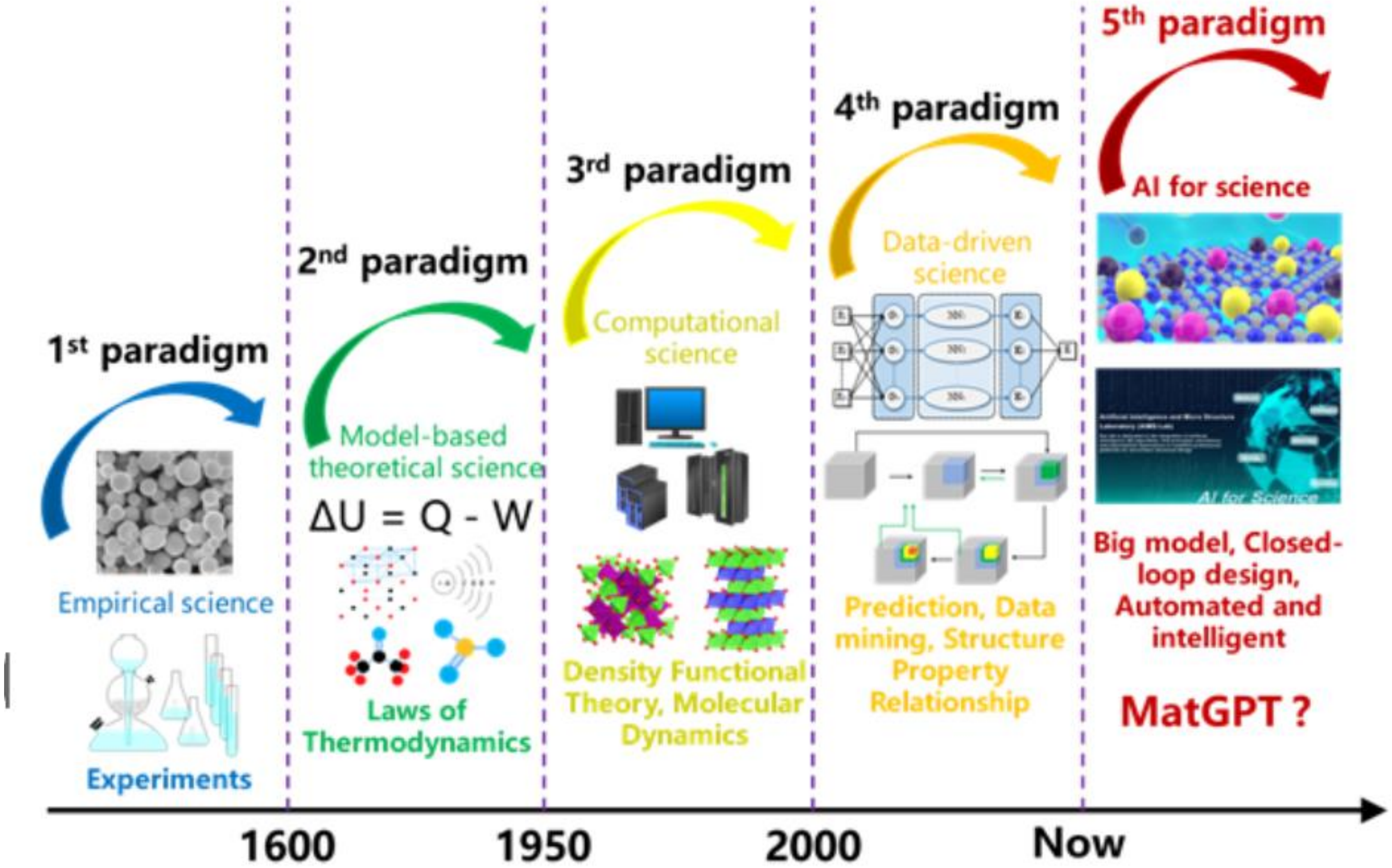
02
OPTION

基本Python库介绍

03
OPTION

机器学习基本流程介绍

基本背景—材料研究的新范式

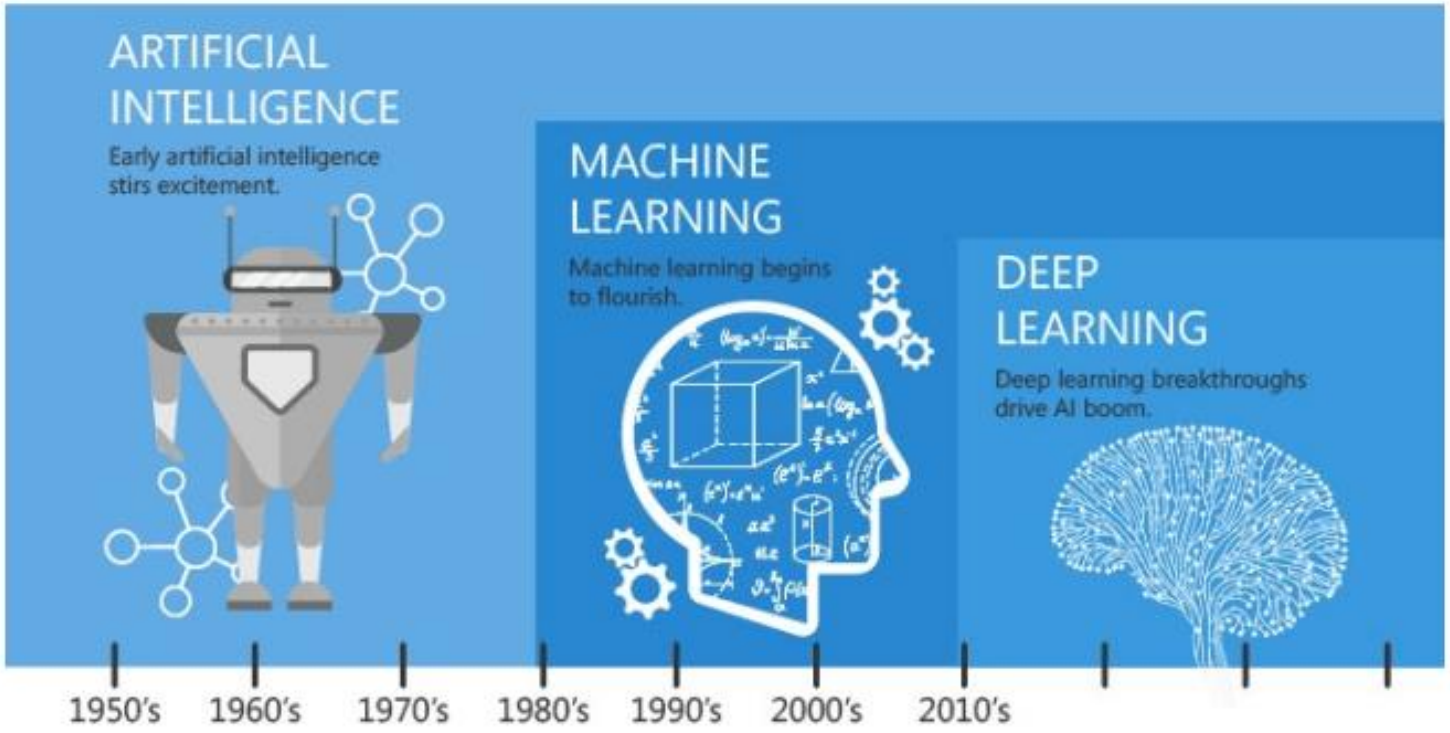


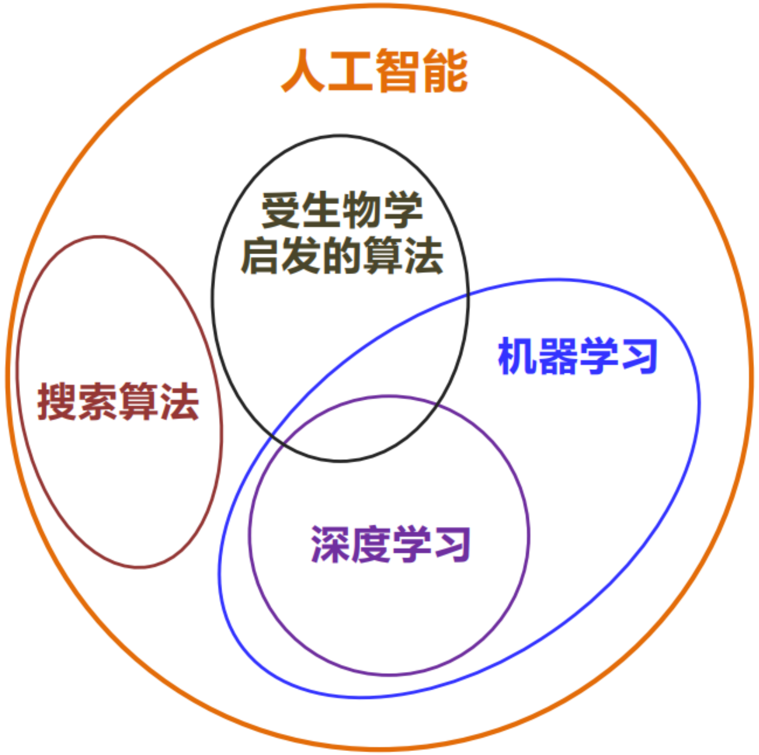
Advanced Materials, 2024, 36, 2306733.

基本背景—人工智能 vs 机器学习基本定义

✧ 人工智能(AI)，即机器的智慧，以区别于人类和动物具备的自然智慧。广义而言，机器帮助人类所完成的一切都属于人工智能。

✧ 机器学习是一类从数据学习规律的算法，它让机器像人类一样从经验中学习。机器学习通常不依赖于事先设定的方程，而是采用统计学方法来训练模型从数据中学习，即依据一定的“学习方法”直接从训练数据集中学习、 生成模型，然后对未知数据做出预测或判断。

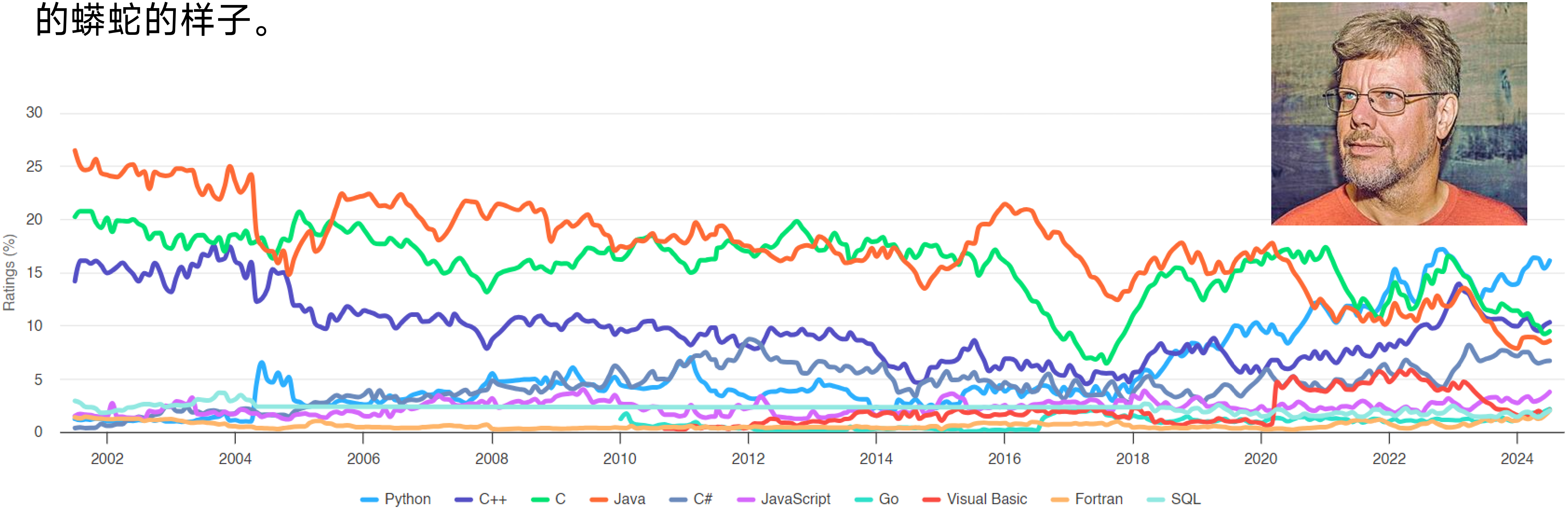

ARTIFICIAL INTELLIGENCE
Early artificial intelligence stirs excitement.
MACHINE LEARNING
Machine learning begins to flourish.
DEEP LEARNING
Deep learning breakthroughs drive AI boom.
1950's
1960's
1970's
1980's
1990's
2000's
2010's



Python语言与其它语言的对比

✧ Python是一种计算机编程语言，由荷兰的吉多·范罗苏姆(出生于1956年)于1990年代初设计。

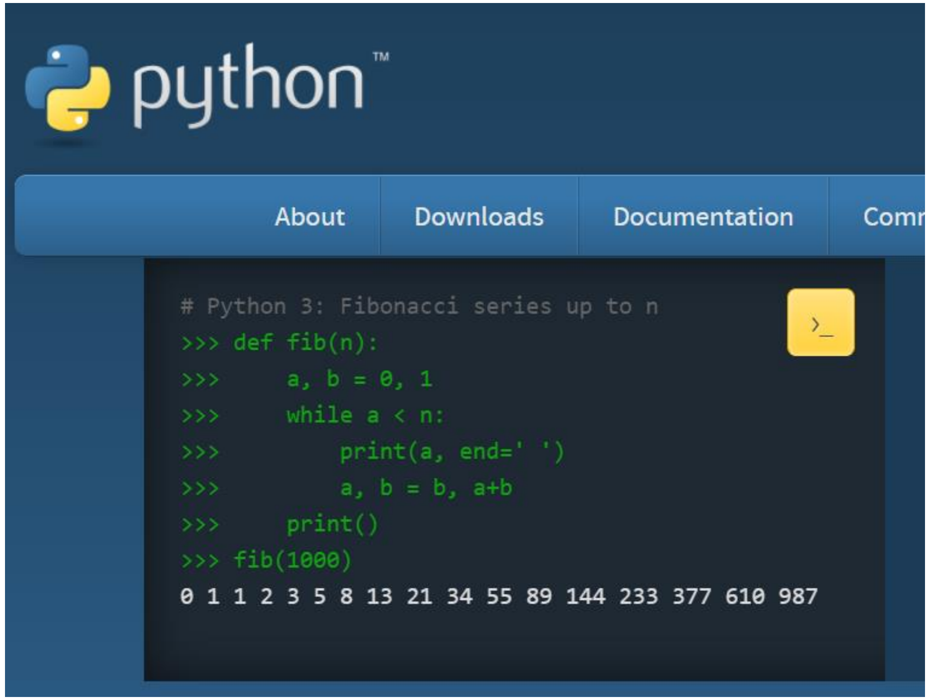
✧ Python，读作[ˈpaɪθɑn]，翻译成汉语是蟒蛇的意思，并且Python的logo也是两条缠绕在一起的蟒蛇的样子。



https://www.tiobe.com/tiobe-index

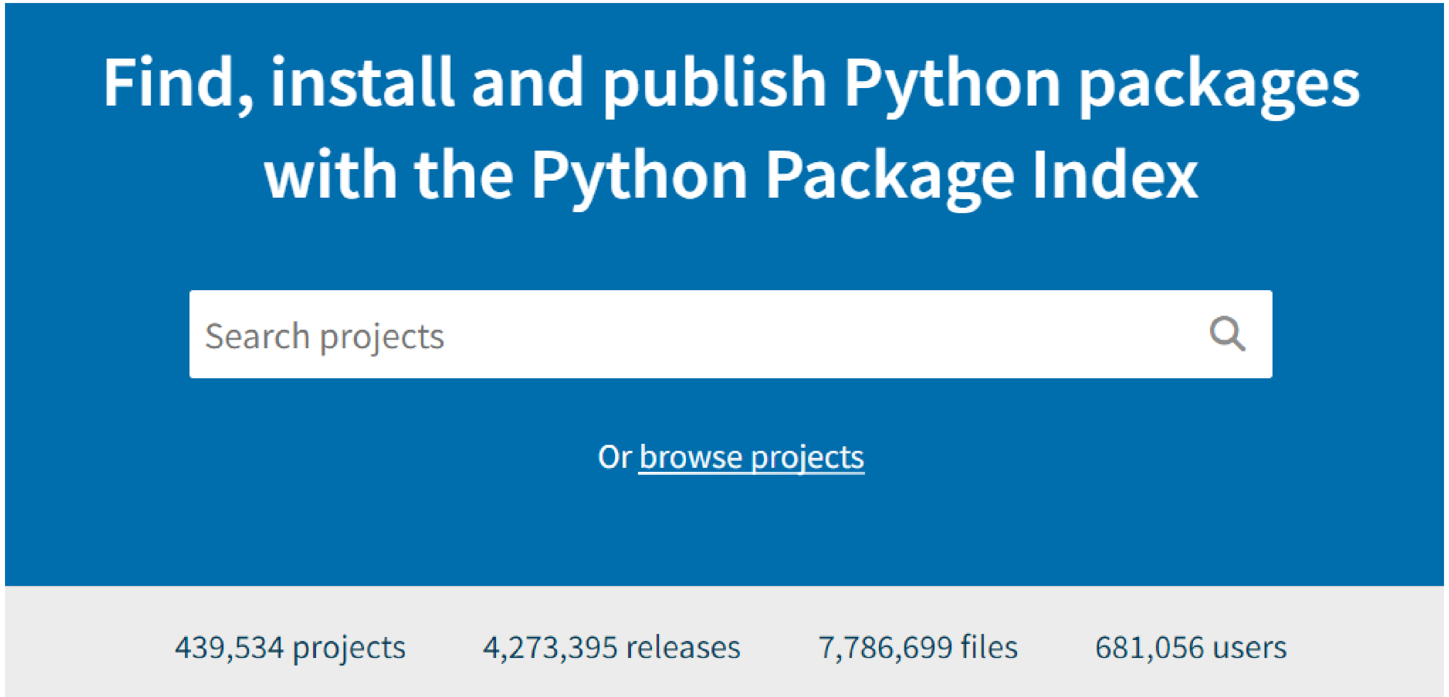
Python语言的特点

1. 简单易学、明确优雅、开发速度快；2. 跨平台、可移植、可扩展、交互式、解释型、面向对象的动态语言；3. 大量的标准库和第三方库；4. 社区活跃，贡献者多，互帮互助；
5. 开源语言，发展动力巨大



```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

https://www.python.org


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

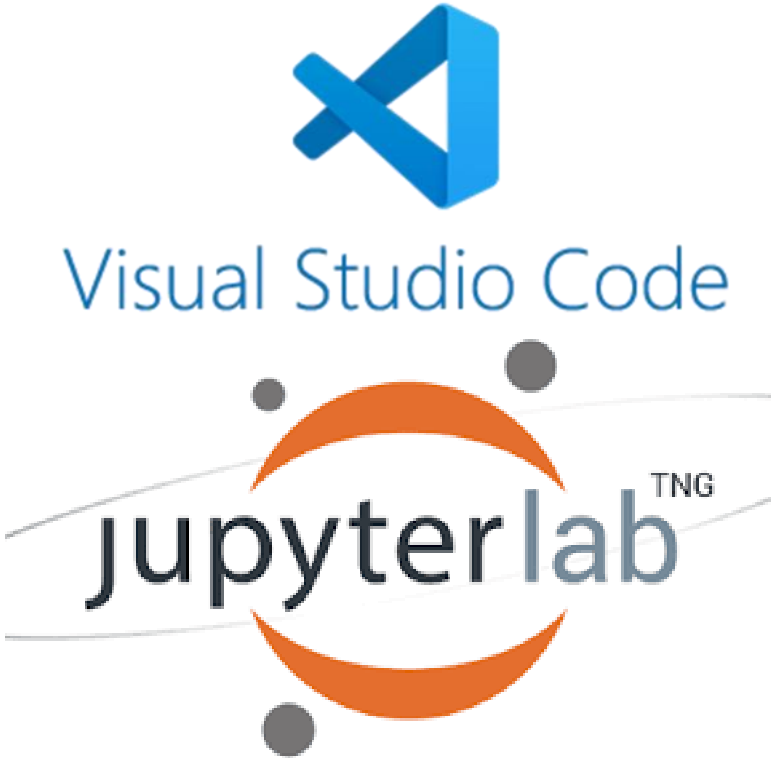
Or browse projects

439,534 projects 4,273,395 releases 7,786,699 files 681,056 users

https://pypi.org

开始使用Python

集成了Python编译器

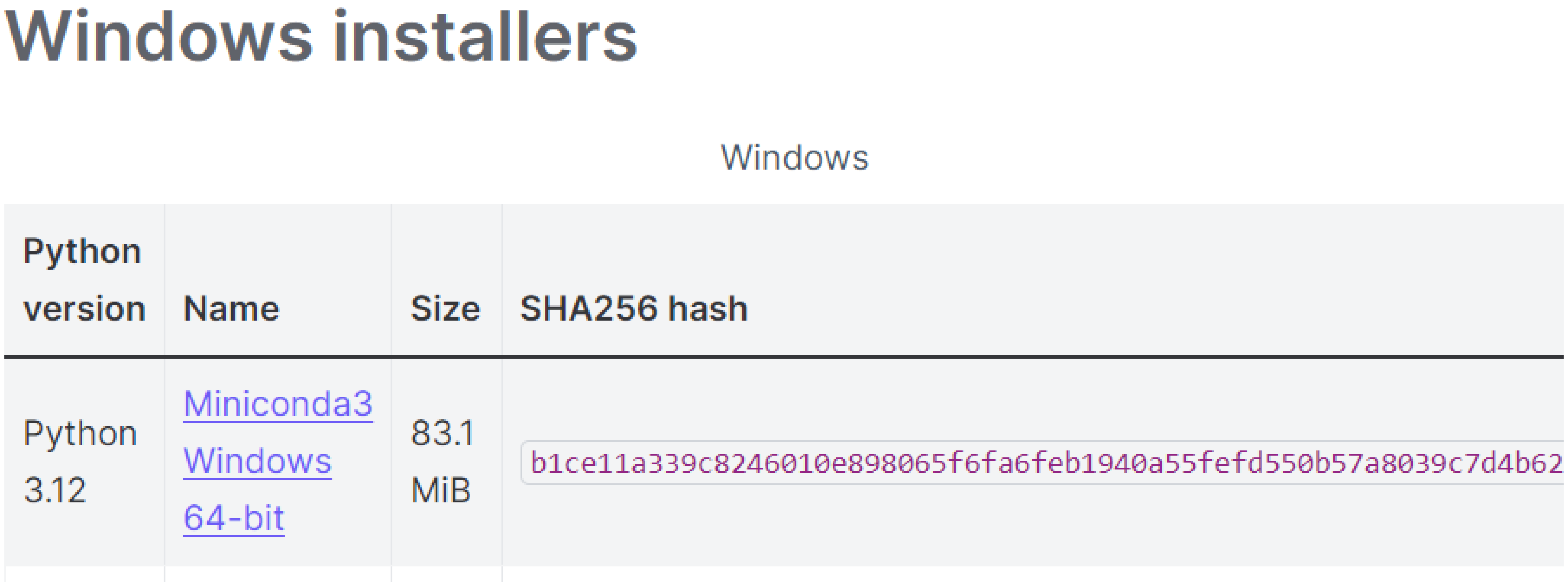


Python编写、学习平台

Miniconda下载

Miniconda是一款小巧的python环境管理工具，安装包大约只有50M多点，其安装程序中包含conda软件包管理器和Python。一旦安装了Miniconda，就可以使用conda命令安装任何其他软件工具包并创建环境等。

Windows版本：**conda install -c msys2 m2-base** (可以处理shell相关的东西)



Windows installers

Windows

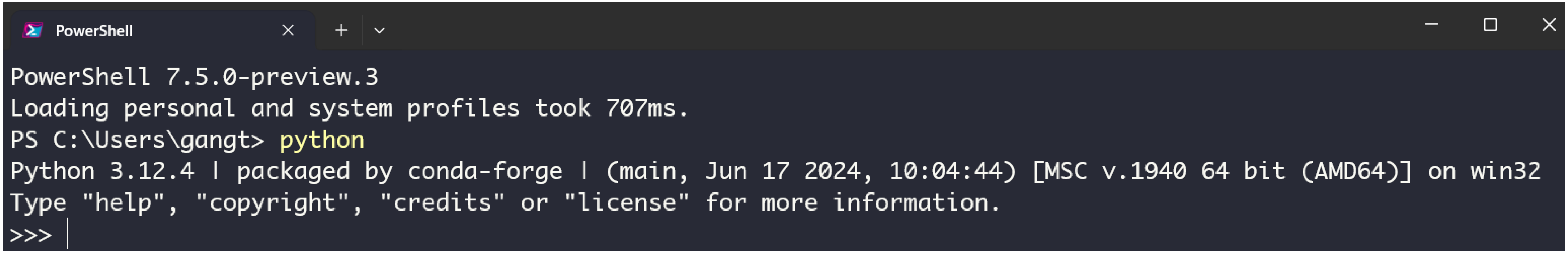
Python version	Name	Size	SHA256 hash
Python 3.12	Miniconda3 Windows 64-bit	83.1 MiB	`b1ce11a339c8246010e898065f6fa6feb1940a55fefd550b57a8039c7d4b62`

https://docs.conda.io/en/latest/miniconda.html

Miniconda安装

Windows版本：

双击“Miniconda3-py312_24.5.0-0-Windows-x86_64.exe”，根据提示安装

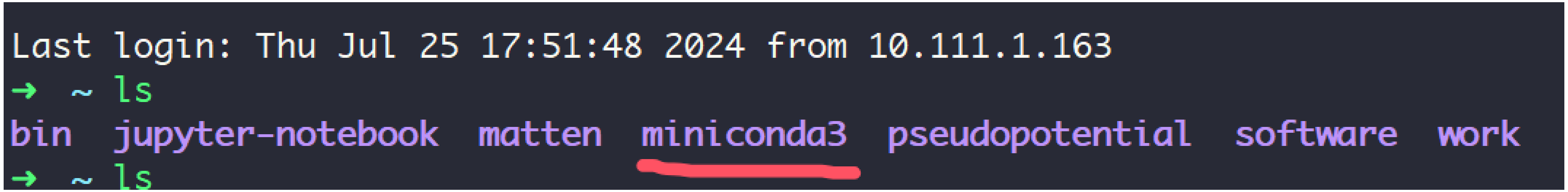


```
PowerShell 7.5.0-preview.3
Loading personal and system profiles took 707ms.
PS C:\Users\gangt> python
Python 3.12.4 | packaged by conda-forge | (main, Jun 17 2024, 10:04:44) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

https://github.com/PowerShell/PowerShell/releases

Linux版本：

sh Miniconda3-py312_24.5.0-0-Linux-x86_64.sh，然后根据提示安装



```
Last login: Thu Jul 25 17:51:48 2024 from 10.111.1.163
➜  ~ ls
bin  jupyter-notebook  matten  miniconda3  pseudopotential  software  work
➜  ~ ls
```

各种Python工具包安装

- conda是一个通用的包管理器，意思是什么语言的包都可以用它进行管理，自然也包python，它很像一个跨平台版本的apt或者yum，而且conda是开源的(github链接：https://github.com/conda/conda)

- pip同conda一样，也是一个包管理器，但它只能管理python包，并且它是python官方认可的包管理器，其中pip的含义是Pip Installs Packages，最常用于安装在PyPI (Python Package Index https://pypi.python.org/pypi)上发布的包，在通过conda list命令查看当前环境下已安装的package时，通过pip安装的package在Channel那一列会显示pypi。

各种Python工具包安装

```
#Conda相关命令：
>> conda update conda or conda update --all
>> conda clean -p
>> conda clean -t

#Pymatgen安装命令：
>> conda install -c conda-forge pymatgen
#Links: https://pymatgen.org/installation.html

#Jupyter notebook安装命令：
>> pip install jupyterlab
#Links:https://jupyter.org/install
```

https://mp.weixin.qq.com/s/TwQg6d52mGCMYprzdOS5FA

conda/pip命令安装机器学习相关软件包

```
#Matminer安装命令：
>> pip install matminer
#conda install -c conda-forge matminer
#Links: https://hackingmaterials.lbl.gov/matminer/installation.html

#scikit-learn安装命令：
>> pip install -U scikit-learn
#conda install -c conda-forge scikit-learn
#Links: https://scikit-learn.org/stable/install.html
```

matminer

matminer is a Python library for data mining the properties of materials.

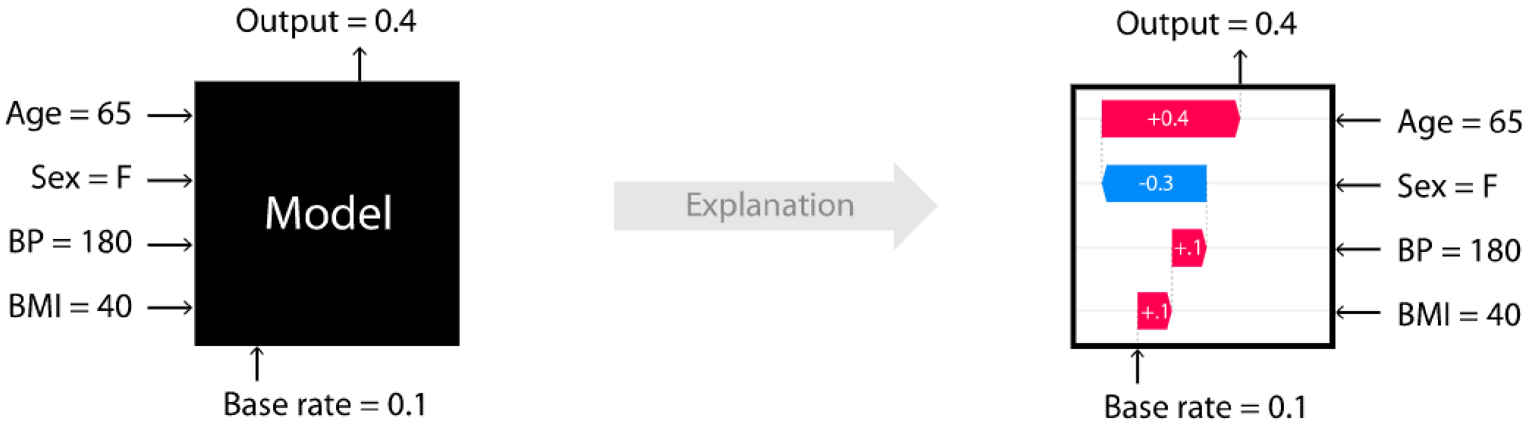
scikit-learn

Machine Learning in Python

conda/pip命令安装机器学习相关软件包

```
#SHAP安装命令：
>> conda install -c conda-forge shap
#pip install shap
#Links: https://github.com/slundberg/shap

#XGBoost机器学习算法安装命令：
>> conda install -c conda-forge py-xgboost
#pip install xgboost -i https://pypi.tuna.tsinghua.edu.cn/simple
#Links: https://xgboost.readthedocs.io/en/stable/install.html
```

SHAP



dmlc XGBoost

conda/pip换源方法 (针对下载慢等解决方法)

Conda换源

```
>> vi ~/.condarc
```

```
channels:
  - http://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/win-64
  - http://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/win-64
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge/

show_channel_urls: true
ssl_verify: true
changeps1: False
```

#https://zhuanlan.zhihu.com/p/87123943

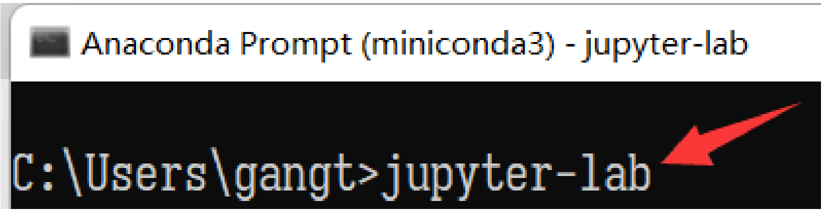
Pip换源

```
>> pip install 软件包名 -i https://pypi.tuna.tsinghua.edu.cn/simple
```

#https://zhuanlan.zhihu.com/p/109939711

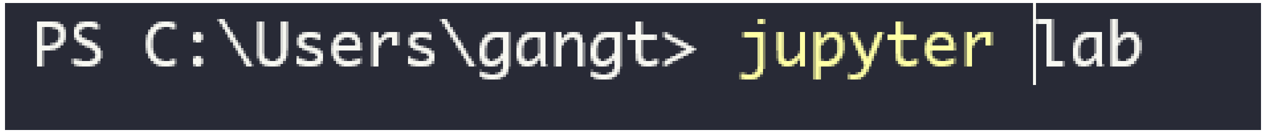
Jupyterlab基本使用

JupyterLab是广受欢迎的Jupyter Notebook「新」界面。它是一个交互式的开发环境，可用于notebook、代码或数据，因此它的扩展性非常强。用户可以使用它编写notebook、操作终端、编辑 markdown 文本、打开交互模式、查看csv文件及图片等。除此以外，JupyterLab还具有灵活而强大的用户界面。

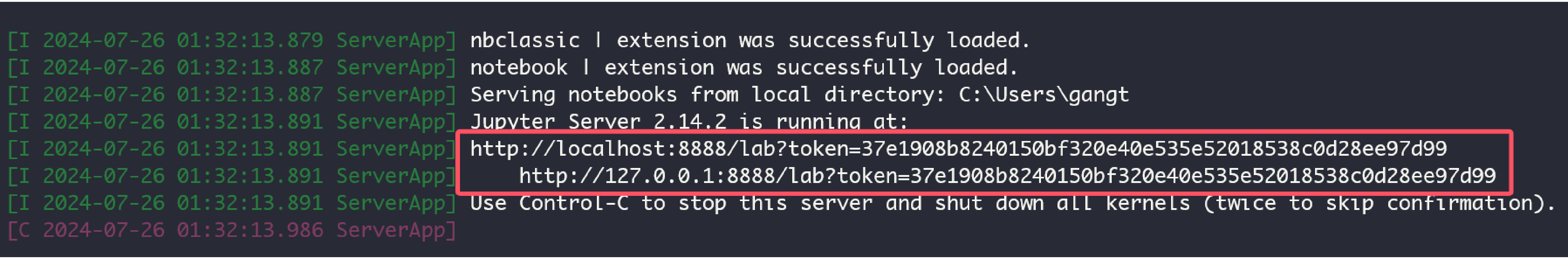


Anaconda Prompt (miniconda3) - jupyter-lab

```
C:\Users\gangt>jupyter-lab
```

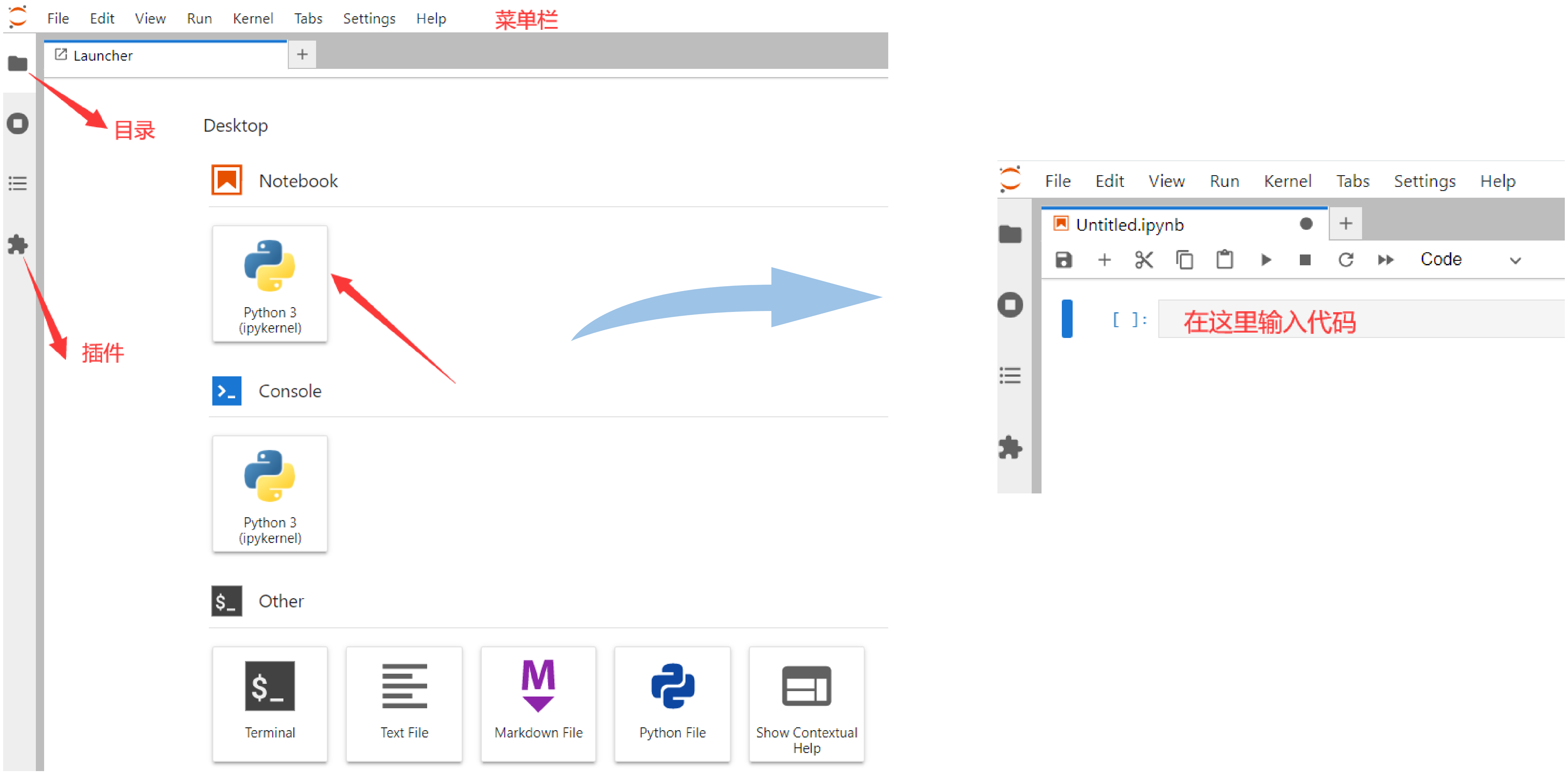


```
PS C:\Users\gangt> jupyter lab
```



```
[I 2024-07-26 01:32:13.879 ServerApp] nbclassic | extension was successfully loaded.
[I 2024-07-26 01:32:13.887 ServerApp] notebook | extension was successfully loaded.
[I 2024-07-26 01:32:13.887 ServerApp] Serving notebooks from local directory: C:\Users\gangt
[I 2024-07-26 01:32:13.891 ServerApp] Jupyter Server 2.14.2 is running at:
[I 2024-07-26 01:32:13.891 ServerApp] http://localhost:8888/lab?token=37e1908b8240150bf320e40e535e52018538c0d28ee97d99
[I 2024-07-26 01:32:13.891 ServerApp]     http://127.0.0.1:8888/lab?token=37e1908b8240150bf320e40e535e52018538c0d28ee97d99
[I 2024-07-26 01:32:13.891 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 2024-07-26 01:32:13.986 ServerApp]
```

Jupyterlab基本使用



菜单栏

目录

插件

在这里输入代码

Jupyterlab基本使用

命令模式 (按键 Esc 开启)

编辑模式(按键 Enter 切换)

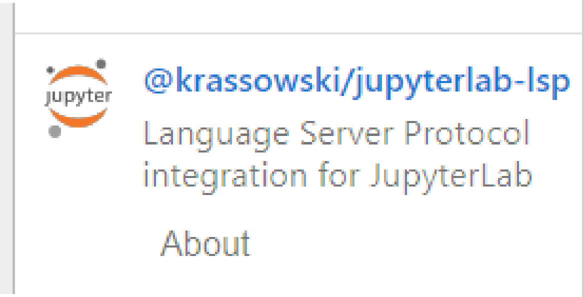
Enter: 切换到编辑模式
A: 在代码块前插入空白代码块
B: 在代码块后插入空白代码块
DD: 删除代码块
X: 剪切当前代码块
C: 复制当前代码块
V: 粘贴当前代码块
Z: 取消删除代码块

ESC: 切换到命令模式
Tab: 代码补全或缩进
Shift+Enter: 运行当前代码块并选定下一代码块
Ctrl+Enter: 运行当前代码块
Alt+Enter: 运行当前代码块并在后面插入新代码块

https://cloud.tencent.com/developer/article/1971992

Jupyterlab基本插件安装—jupyterlab-lsp

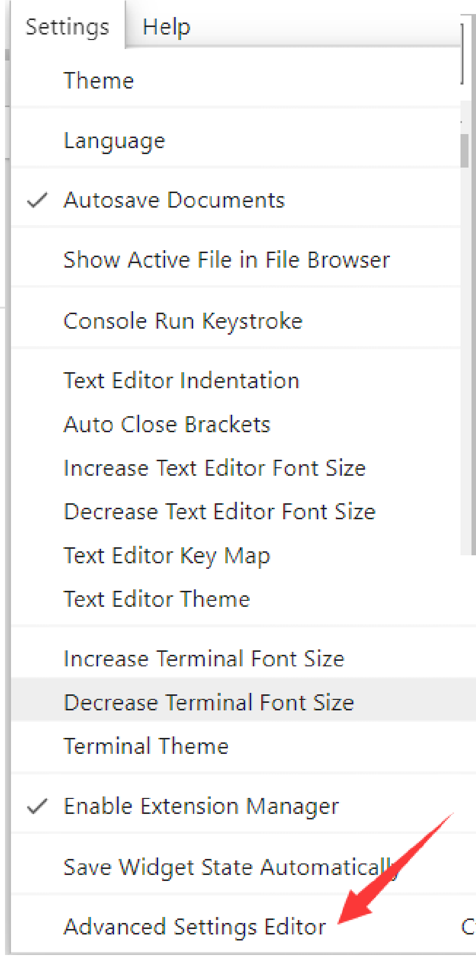
```
>> pip install jupyter-lsp     # https://github.com/jupyter-lsp/jupyterlab-lsp
>> pip install python-lsp-server[all]
>> jupyter labextension install @krassowski/jupyterlab-lsp    #安装插件
```



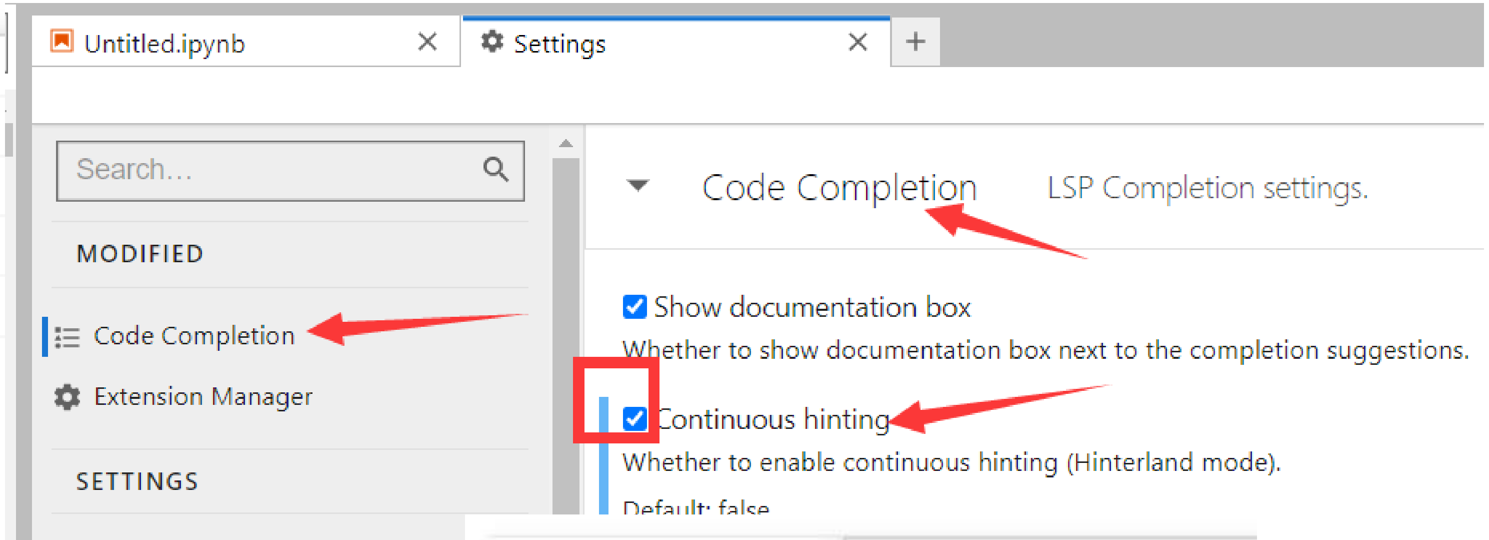
@krassowski/jupyterlab-lsp

Language Server Protocol integration for JupyterLab

About



Settings
Help
Theme
Language
✓ Autosave Documents
Show Active File in File Browser
Console Run Keystroke
Text Editor Indentation
Auto Close Brackets
Increase Text Editor Font Size
Decrease Text Editor Font Size
Text Editor Key Map
Text Editor Theme
Increase Terminal Font Size
Decrease Terminal Font Size
Terminal Theme
✓ Enable Extension Manager
Save Widget State Automatically
Advanced Settings Editor Ctrl+,



Code Completion LSP Completion settings.

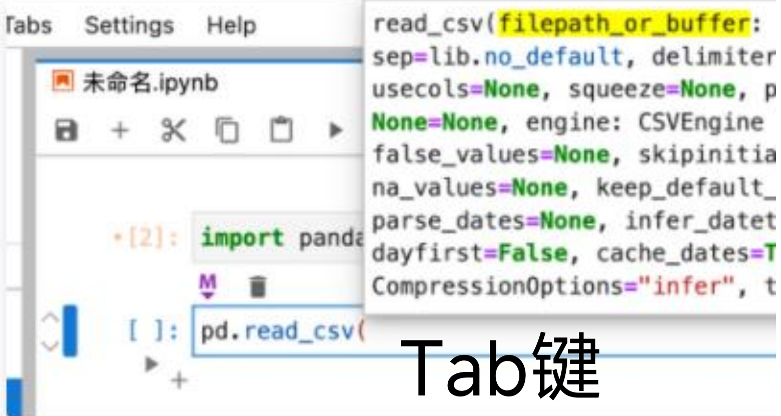
☑ Show documentation box

Whether to show documentation box next to the completion suggestions.

☑ Continuous hinting

Whether to enable continuous hinting (Hinterland mode).

Default: false



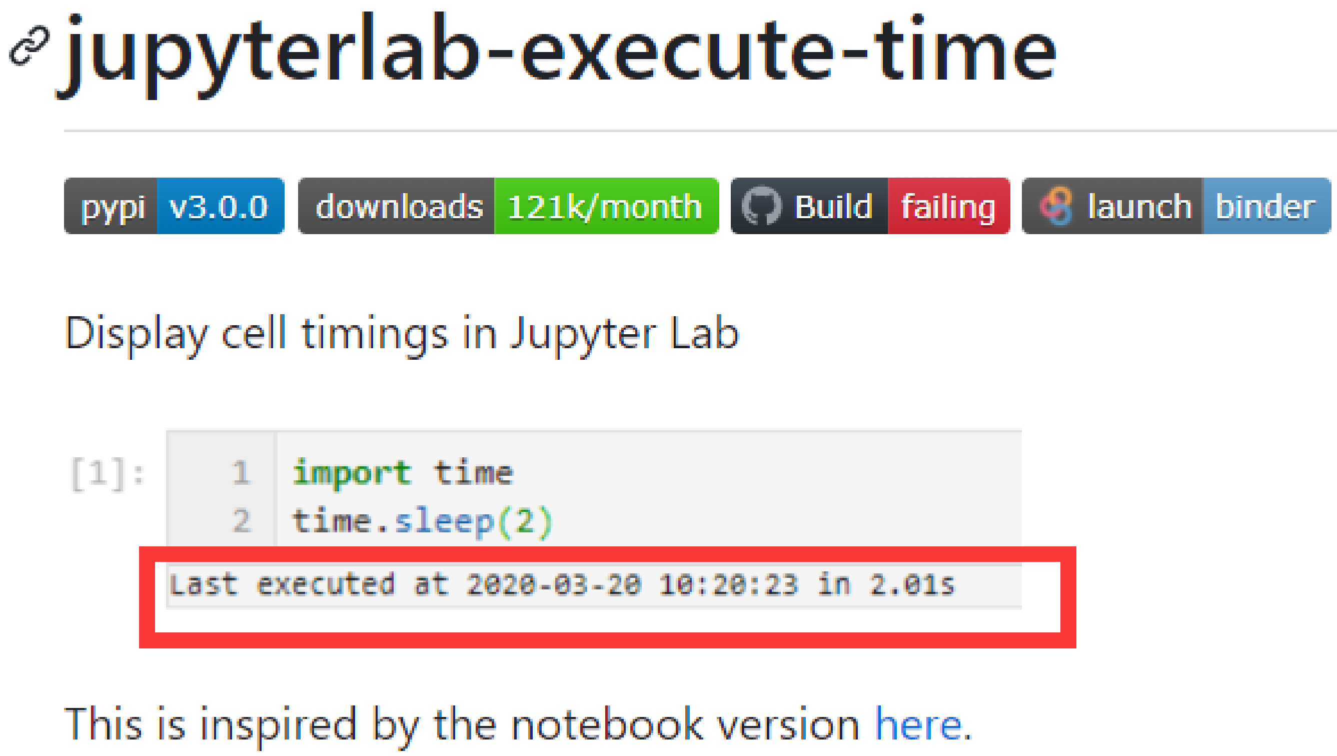
```
[2]: import panda
[ ]: pd.read_csv(
```

```
read_csv(filepath_or_buffer:
sep=lib.no_default, delimiter
usecols=None, squeeze=None, p
None=None, engine: CSVEngine
false_values=None, skipinitia
na_values=None, keep_default_
parse_dates=None, infer_datet
dayfirst=False, cache_dates=T
CompressionOptions="infer", t
```

Tab键

Jupyterlab基本插件安装—jupyterlab-execute-time

```
>> pip install jupyterlab_execute_time
   or
>> conda install -c conda-forge jupyterlab_execute_time
```



jupyterlab-execute-time

Display cell timings in Jupyter Lab

```
[1]: import time
     time.sleep(2)
Last executed at 2020-03-20 10:20:23 in 2.01s
```

This is inspired by the notebook version here.

https://github.com/deshaw/jupyterlab-execute-time

目前提供免费在线Jupyterlab notebook的资源

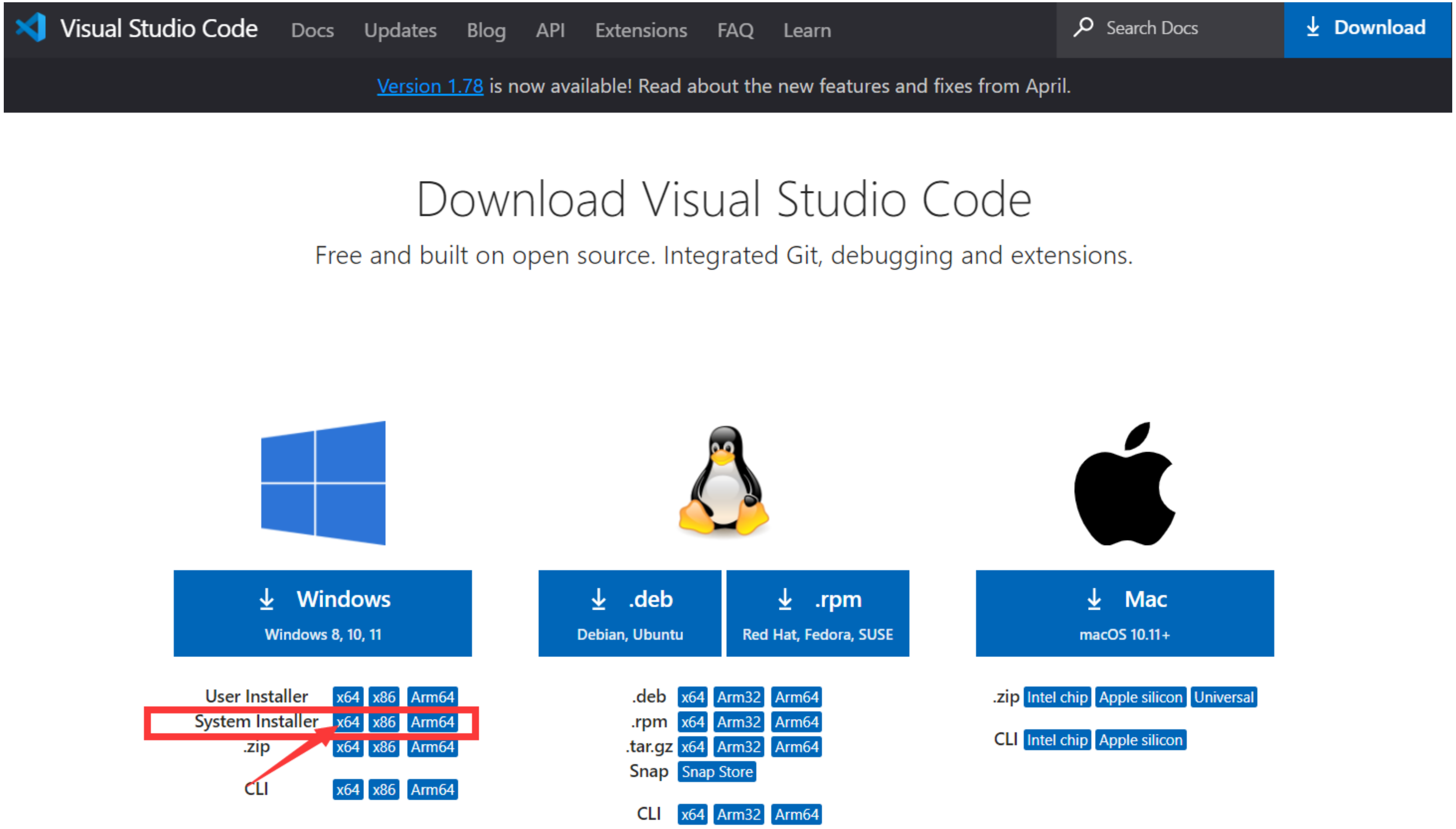
Bohrium
A Cloud Platform Designed for Micro-scale Scientific Computing

https://bohrium-doc.dp.tech/docs/userguide/Notebook/

https://www.kaggle.com/docs/efficient-gpu-usage

https://colab.research.google.com/#

Visual Studio Code下载安装



Version 1.78 is now available! Read about the new features and fixes from April.

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.

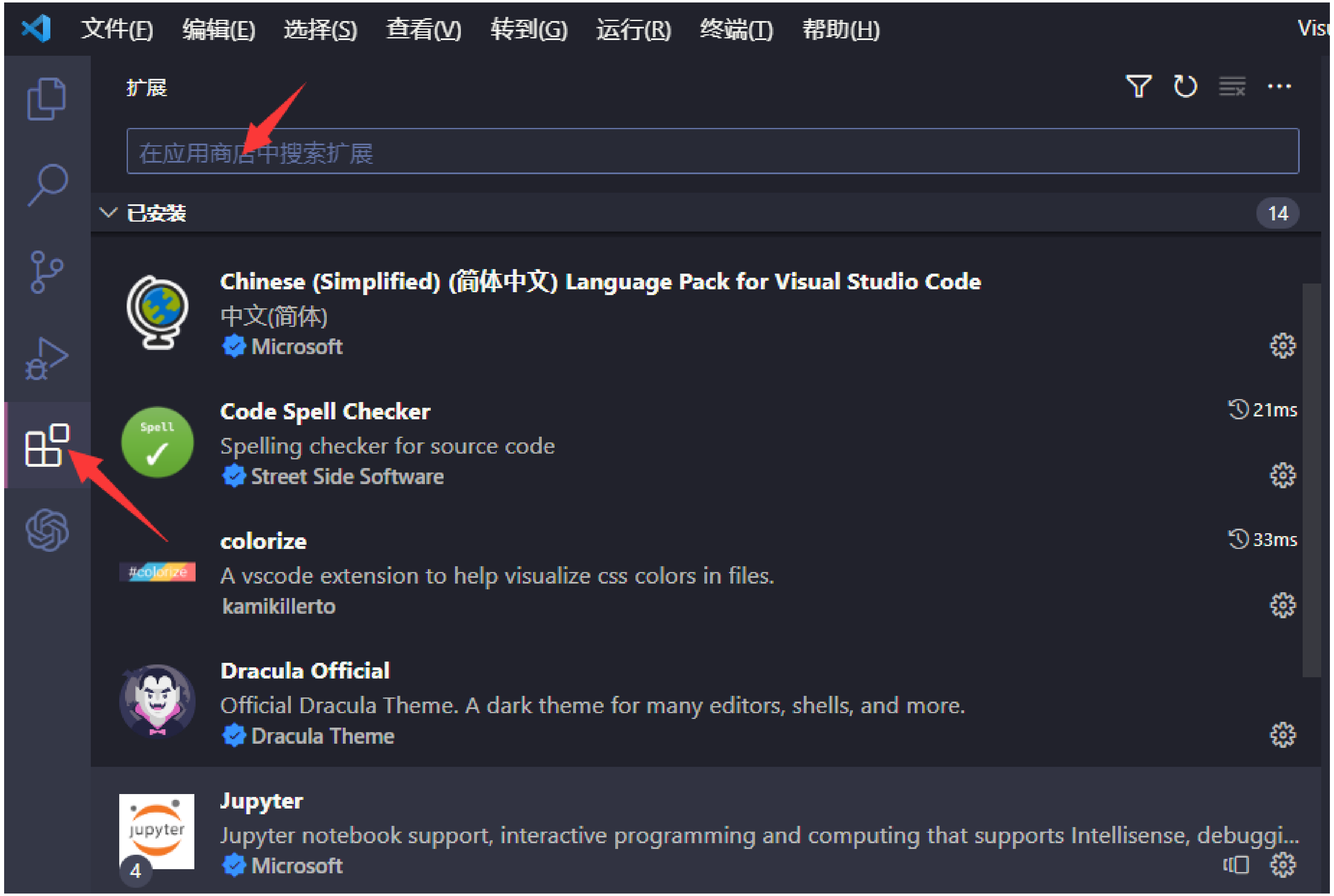
User Installer x64 x86 Arm64
System Installer x64 x86 Arm64
.zip x64 x86 Arm64
CLI x64 x86 Arm64

.deb x64 Arm32 Arm64
.rpm x64 Arm32 Arm64
.tar.gz x64 Arm32 Arm64
Snap Snap Store
CLI x64 Arm32 Arm64

.zip Intel chip Apple silicon Universal
CLI Intel chip Apple silicon

https://code.visualstudio.com/Download

Visual Studio Code基本插件安装



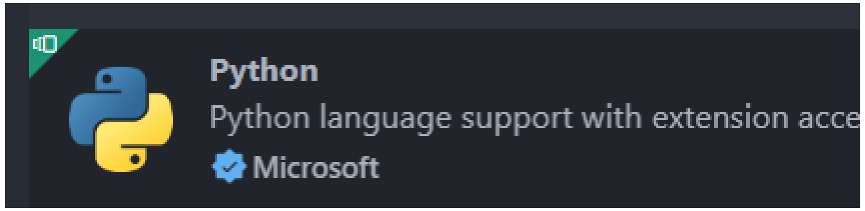
Chinese (Simplified) (简体中文) Language Pack for Visual Studio Code
中文(简体)
Microsoft

Code Spell Checker 21ms
Spelling checker for source code
Street Side Software

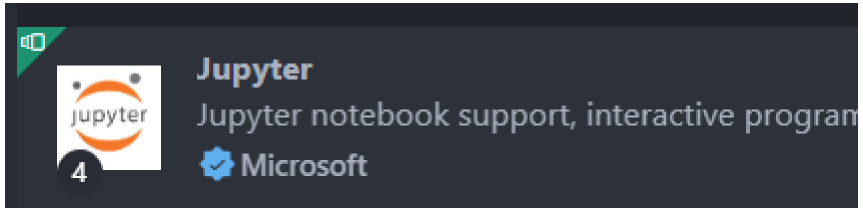
colorize 33ms
A vscode extension to help visualize css colors in files.
kamikillerto

Dracula Official
Official Dracula Theme. A dark theme for many editors, shells, and more.
Dracula Theme

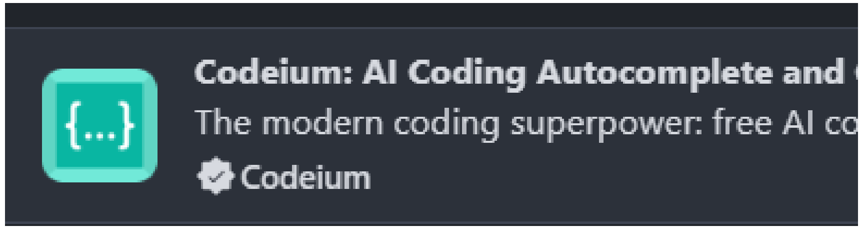
Jupyter
Jupyter notebook support, interactive programming and computing that supports Intellisense, debuggi...
Microsoft



Python
Python language support with extension acce
Microsoft



Jupyter
Jupyter notebook support, interactive program
Microsoft



Codeium: AI Coding Autocomplete and
The modern coding superpower: free AI co
Codeium

https://cloud.tencent.com/developer/article/2344052

Python基本语法

知乎 bilibili GitHub

https://kimi.moonshot.cn

Python基础库

库是完成一定功能的代码的集合

scikit learn
NumPy
pandas
tkinter
SciPy
matplotlib
Version 3.4.3
PyTorch
pymatgen
matminer

重要链接：https://github.com/ShowMeAI-Hub/awesome-AI-cheatsheets/tree/main

Python库的导入

1. import 语句
使用方法：import A (导入A模块，例如导入numpy模块，import numpy)

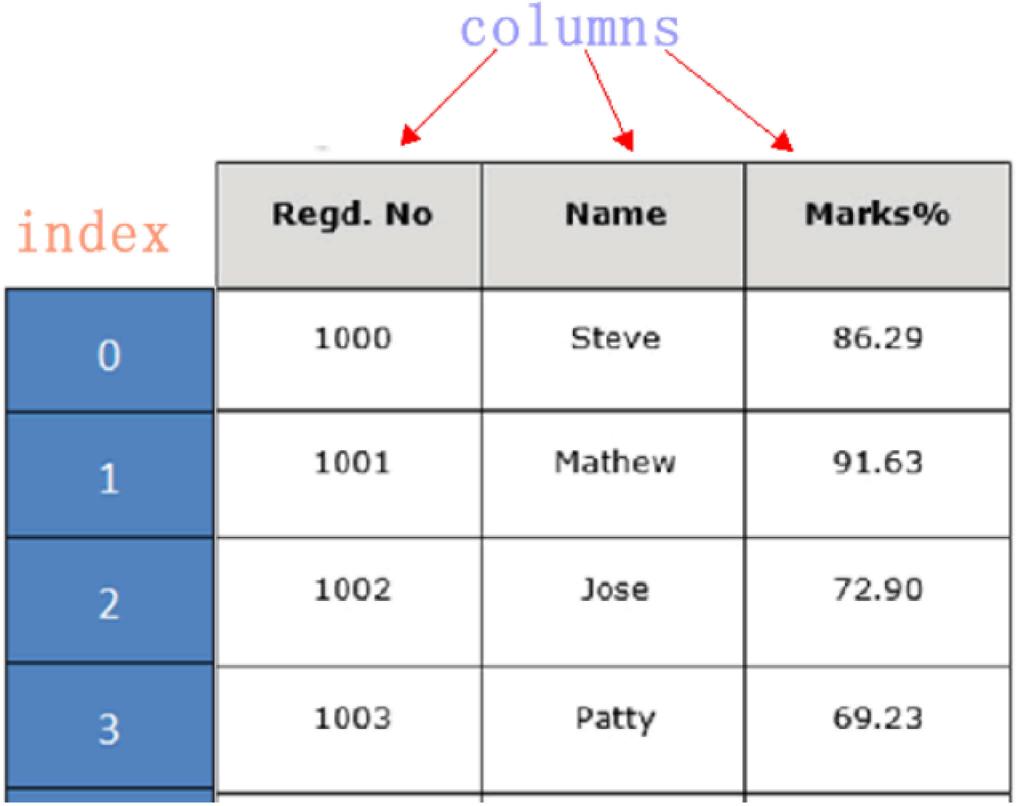
可添加别名，例如 import numpy as np，程序中则可使用np代表numpy

2. from … import … 语句
使用方法：from A import a1 (在内存中创建并加载A模块中a1工具的副本，例如导入numpy模块中的zeros函数，from numpy import zeros)

与import A.a1的区别，前者可直接调用，后者只能使用全名

Pandas DataFrame基本使用—创建

DataFrame是Pandas的重要数据结构之一，也是在使用Pandas进行数据分析过程中最常用的结构之一。**DataFrame一个表格型的数据结构**，既有**行标签(index)**，又有**列标签(columns)**，它也被称异构数据表，所谓异构，指的是表格中每列的数据类型可以不同，比如可以是**字符串、整型或者浮点型**等。



columns
index

	Regd. No	Name	Marks%
0	1000	Steve	86.29
1	1001	Mathew	91.63
2	1002	Jose	72.90
3	1003	Patty	69.23

```
#创建空的DataFrame对象
import pandas as pd
df = pd.DataFrame()
print(df)

#列表创建DataFame对象
data = [1,2,3,4,5]
df = pd.DataFrame(data)
```

DataFrame创建：https://blog.csdn.net/ccc369639963/article/details/124192330

Pandas DataFrame基本使用—操作和索引

```
df.index            # DataFrame行索引
df.columns          # DataFrame列索引
df.head(n)          # DataFrame的前n行
df.tail(n)          # DataFrame的最后n行
df.shape            # 行数和列数
df.info()           # 索引，数据类型和内存信息
df.describe()       # 数值列的摘要统计信息
df.isnull()         # 空值检查，返回Boolean Arrray
df.notnull()        # 与pd.isnull() 相反
df.dropna()         # 删除所有包含空值的行
df.dropna(axis=1)   # 删除所有包含空值的列
```

https://mp.weixin.qq.com/s/c4ADwRDEpsBn7y6KLHPv1A
https://mp.weixin.qq.com/s/qADQzUoAby1SmThkOwYn5A
https://mp.weixin.qq.com/s/FIK78HxHcxZLOBJarpmZhg

Pandas DataFrame基本使用—操作和索引

```
df.loc[a]                #选取一行
df.loc[3:6]              #选取多行
df[(df[‘column’] >= t1) & (df[‘column’] <= t2)] #选取某个区间的多行

df[‘name’]               #选取一列
df[[‘column_name1’, ‘column_name2’]]        #选取多列
df.iloc[:, 0:5]          #按位置取某几列

df.loc[2][3]             #取指定第2行第3列的元素
```

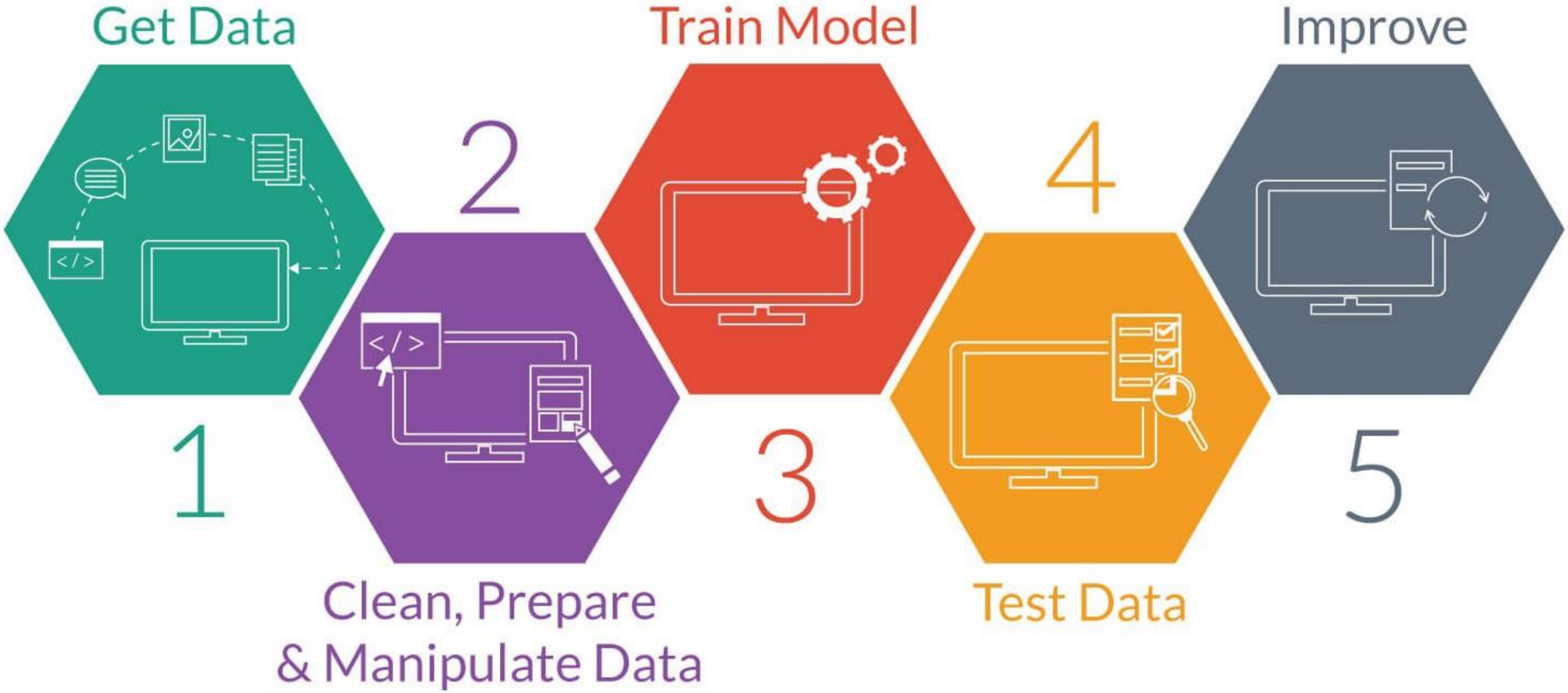
df.loc[行索引,列名]:
loc函数是基于行索引index和列名进行索引的

df.iloc[行位置,列位置]:
iloc函数是基于行和列的位置进行索引的，索引值从0开始，并且得到的结果不包括最后一个位置的值

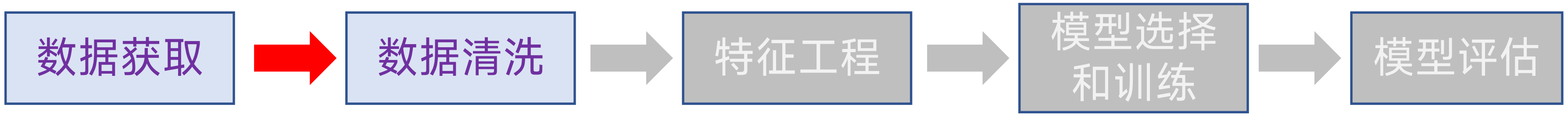
https://blog.csdn.net/qq_40326787/article/details/107013767

机器学习基本流程

数据获取 → 数据清洗 → 特征工程 → 模型选择和训练 → 模型评估



https://www.zhihu.com/question/58339949



数据获取

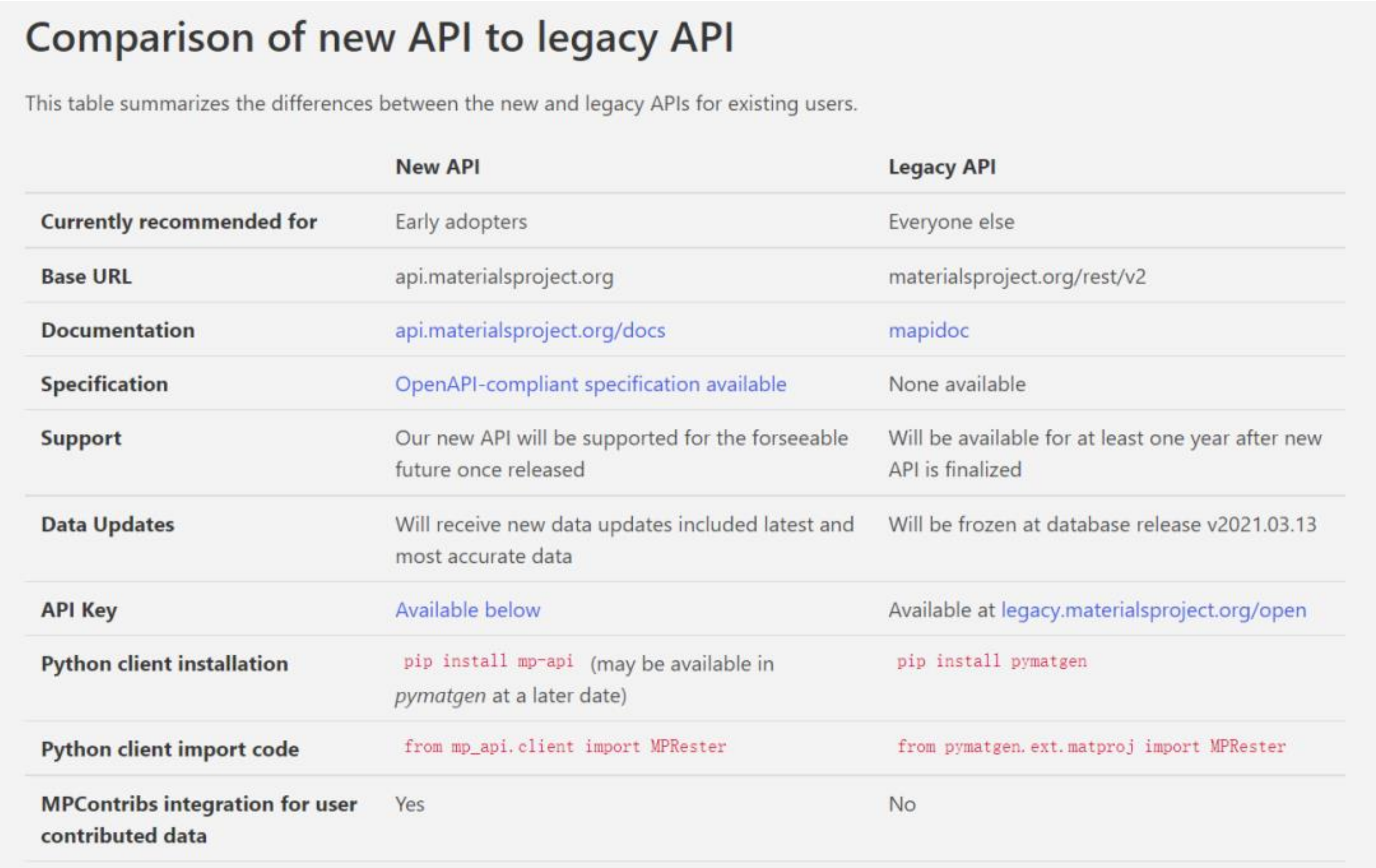
数据清洗

特征工程

模型选择和训练

模型评估

新Materials Project批量筛选数据



Comparison of new API to legacy API

This table summarizes the differences between the new and legacy APIs for existing users.

	New API	Legacy API
Currently recommended for	Early adopters	Everyone else
Base URL	api.materialsproject.org	materialsproject.org/rest/v2
Documentation	api.materialsproject.org/docs	mapidoc
Specification	OpenAPI-compliant specification available	None available
Support	Our new API will be supported for the forseeable future once released	Will be available for at least one year after new API is finalized
Data Updates	Will receive new data updates included latest and most accurate data	Will be frozen at database release v2021.03.13
API Key	Available below	Available at legacy.materialsproject.org/open
Python client installation	`pip install mp-api` (may be available in *pymatgen* at a later date)	`pip install pymatgen`
Python client import code	`from mp_api.client import MPRester`	`from pymatgen.ext.matproj import MPRester`
MPContribs integration for user contributed data	Yes	No

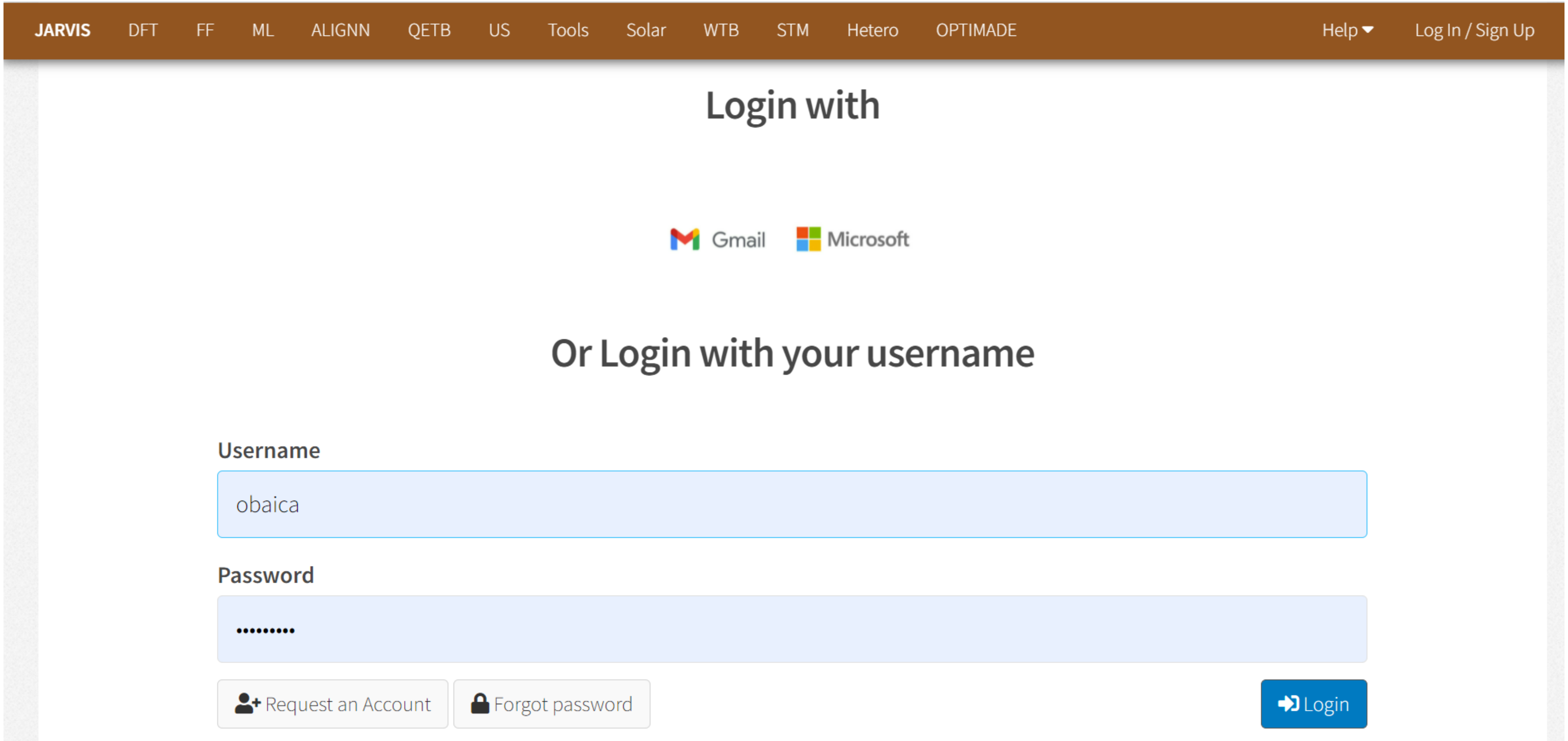
https://matsci.org/t/new-mp-api-fails-in-jupyter/42967/4
https://docs.materialsproject.org/downloading-data/using-the-api/querying-data
https://docs.materialsproject.org/downloading-data/using-the-api/examples

新Materials Project批量筛选数据

```
#新的api_key需要安装mp-api:
>> pip install mp-api

from mp_api.client import MPRester

with MPRester("your_api_key_here") as mpr:
    docs = mpr.summary.search(material_ids=["mp-149", "mp-13", "mp-22526"])

example_doc = docs[0]
mpid = example_doc.material_id
formula = example_doc.formula_pretty
print(mpid)
print(formula)
```

https://mp.weixin.qq.com/s/T8dkL7bRX64hVqjiksJ2_w

JARVIS-DFT数据库网站访问



Login with

Or Login with your username

Username: obaica

Password: •••••••••

https://jarvis.nist.gov/jarvisdft

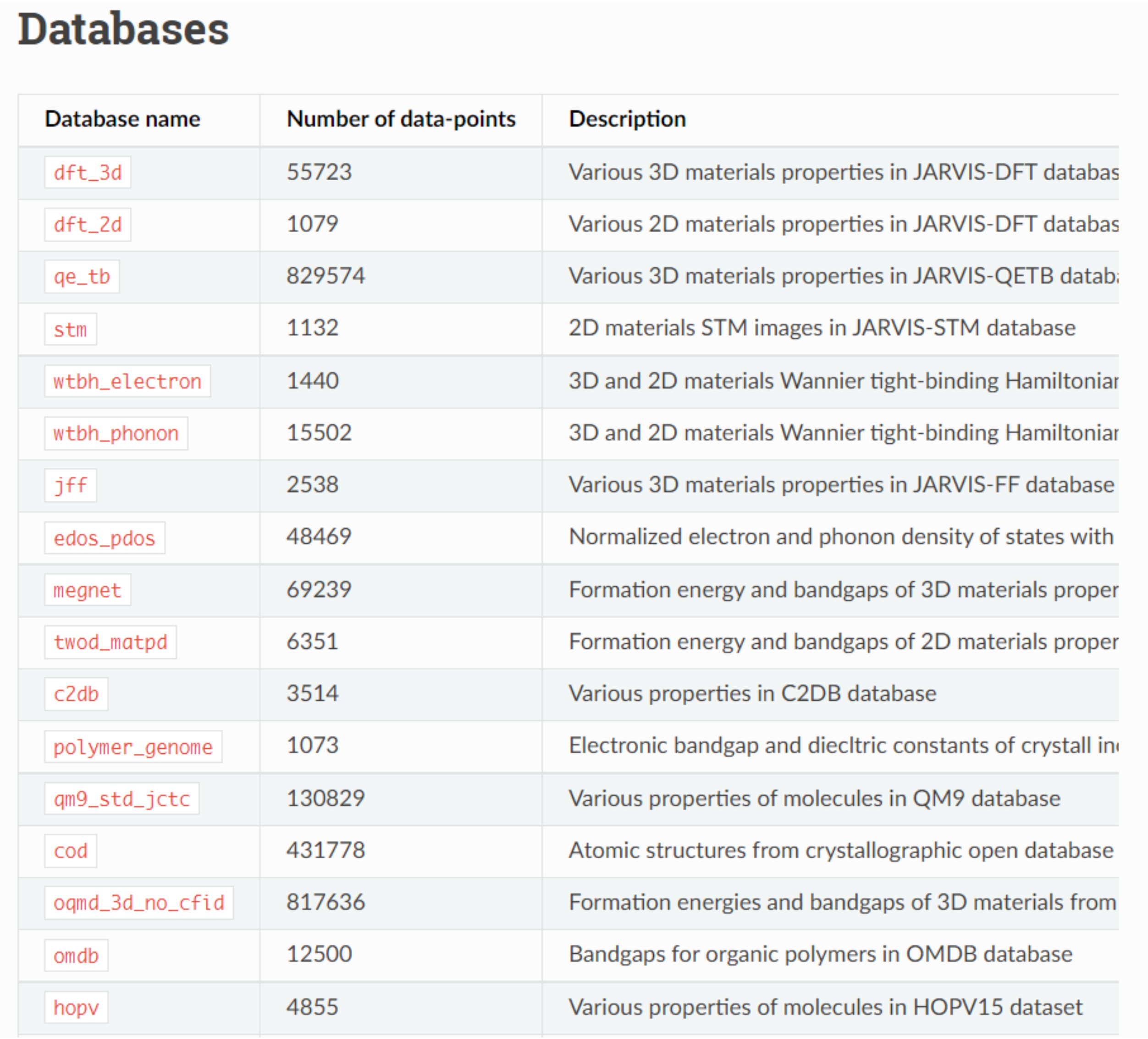
JARVIS-DFT数据库网站访问

```
pip install -U jarvis-tools
```

```
from jarvis.db.figshare import data
d = data('dft_3d') #choose a name of dataset from above
# See available keys
print (d[0].keys())
# Dataset size
print(len(d))

# If pandas framework needed
import pandas as pd
df = pd.DataFrame(d)
print(df)
```

https://jarvis-tools.readthedocs.io/en/master/databases.html

JARVIS-DFT数据库网站访问



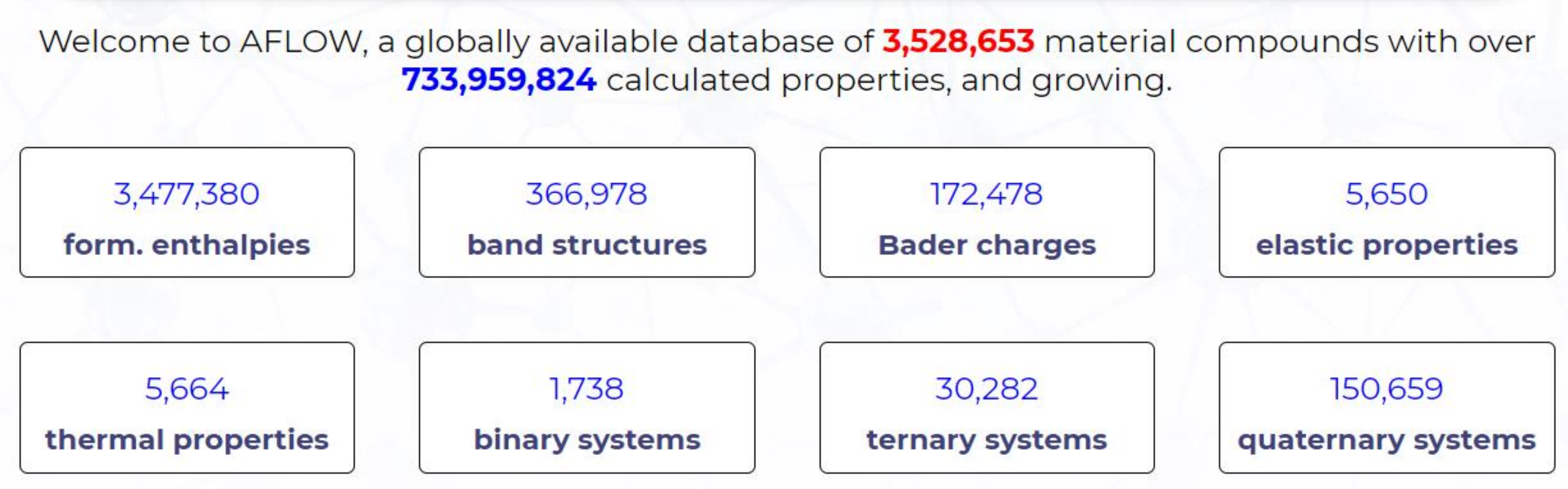
Databases

Database name	Number of data-points	Description
dft_3d	55723	Various 3D materials properties in JARVIS-DFT databas
dft_2d	1079	Various 2D materials properties in JARVIS-DFT databas
qe_tb	829574	Various 3D materials properties in JARVIS-QETB datab
stm	1132	2D materials STM images in JARVIS-STM database
wtbh_electron	1440	3D and 2D materials Wannier tight-binding Hamiltoniar
wtbh_phonon	15502	3D and 2D materials Wannier tight-binding Hamiltoniar
jff	2538	Various 3D materials properties in JARVIS-FF database
edos_pdos	48469	Normalized electron and phonon density of states with
megnet	69239	Formation energy and bandgaps of 3D materials proper
twod_matpd	6351	Formation energy and bandgaps of 2D materials proper
c2db	3514	Various properties in C2DB database
polymer_genome	1073	Electronic bandgap and diecltric constants of crystall in
qm9_std_jctc	130829	Various properties of molecules in QM9 database
cod	431778	Atomic structures from crystallographic open database
oqmd_3d_no_cfid	817636	Formation energies and bandgaps of 3D materials from
omdb	12500	Bandgaps for organic polymers in OMDB database
hopv	4855	Various properties of molecules in HOPV15 dataset

https://jarvis-tools.readthedocs.io/en/master/databases.html

AFLOW数据库网站访问

AFLOW
Automatic-FLOW for Materials Discovery

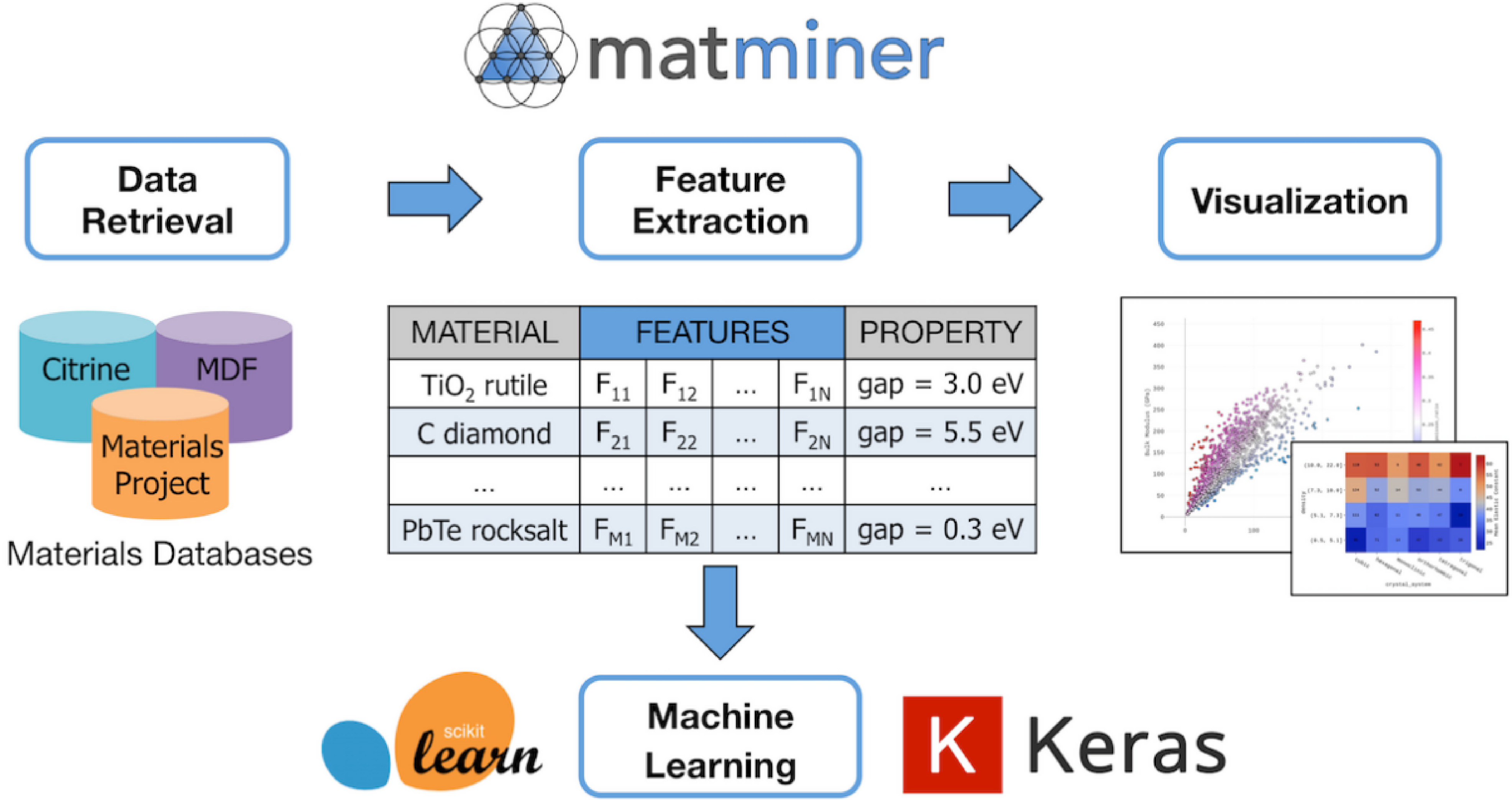


Welcome to AFLOW, a globally available database of **3,528,653** material compounds with over **733,959,824** calculated properties, and growing.

3,477,380 **form. enthalpies**	366,978 **band structures**	172,478 **Bader charges**	5,650 **elastic properties**
5,664 **thermal properties**	1,738 **binary systems**	30,282 **ternary systems**	150,659 **quaternary systems**

https://aflowlib.org/documentation

Matminer数据库数据访问



MATERIAL	FEATURES				PROPERTY
TiO_2 rutile	F_{11}	F_{12}	…	F_{1N}	gap = 3.0 eV
C diamond	F_{21}	F_{22}	…	F_{2N}	gap = 5.5 eV
…	…	…	…	…	…
PbTe rocksalt	F_{M1}	F_{M2}	…	F_{MN}	gap = 0.3 eV

https://hackingmaterials.lbl.gov/matminer

Matminer数据库数据访问

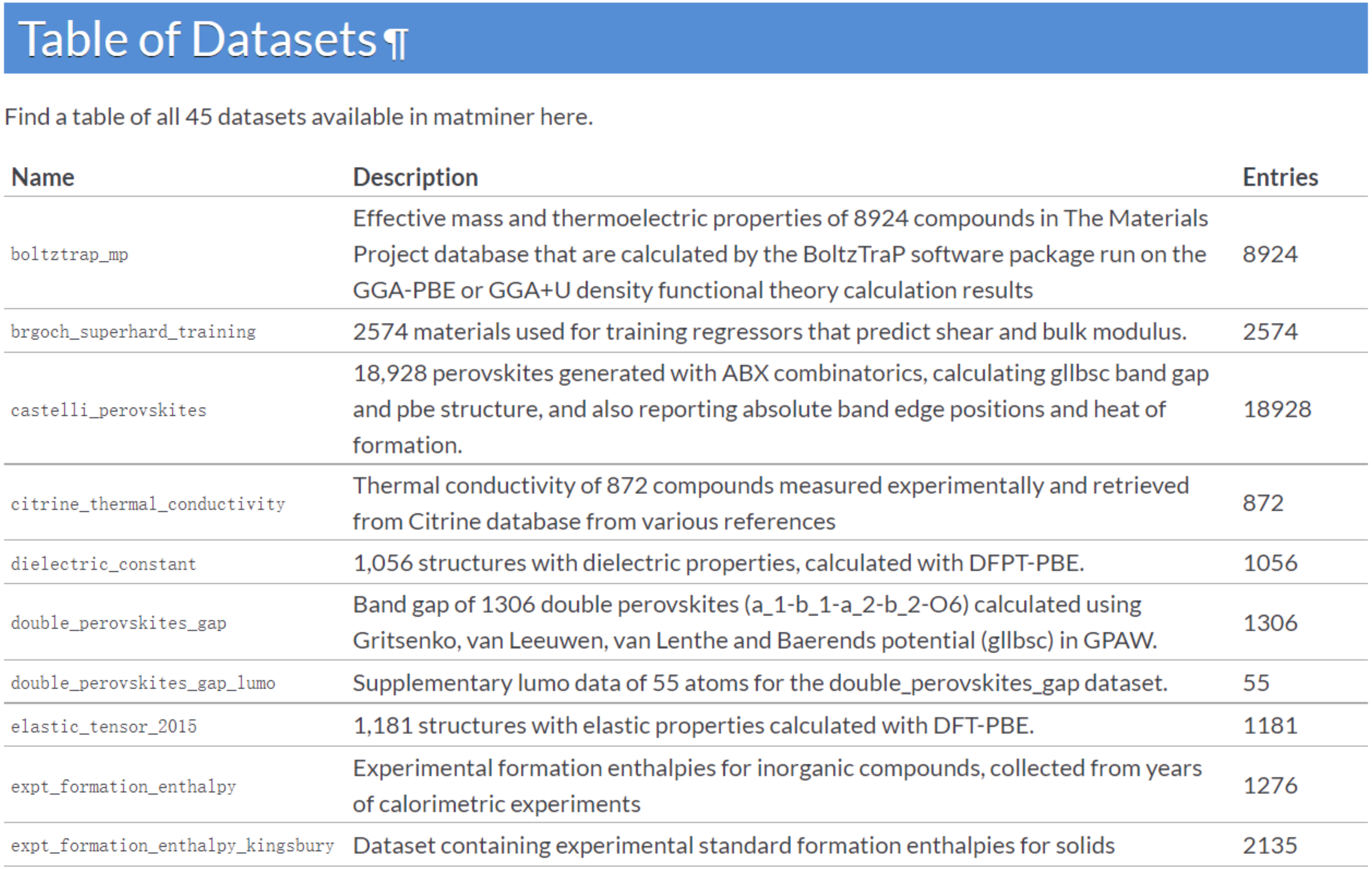


Table of Datasets¶

Find a table of all 45 datasets available in matminer here.

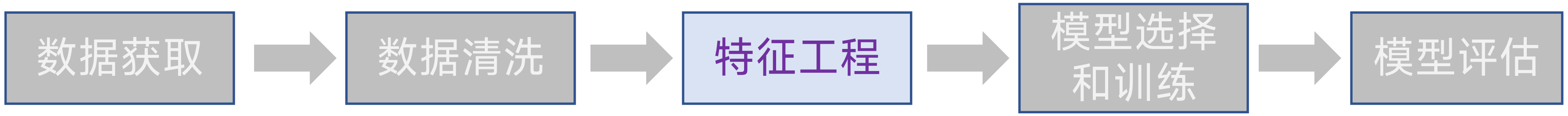
Name	Description	Entries
boltztrap_mp	Effective mass and thermoelectric properties of 8924 compounds in The Materials Project database that are calculated by the BoltzTraP software package run on the GGA-PBE or GGA+U density functional theory calculation results	8924
brgoch_superhard_training	2574 materials used for training regressors that predict shear and bulk modulus.	2574
castelli_perovskites	18,928 perovskites generated with ABX combinatorics, calculating gllbsc band gap and pbe structure, and also reporting absolute band edge positions and heat of formation.	18928
citrine_thermal_conductivity	Thermal conductivity of 872 compounds measured experimentally and retrieved from Citrine database from various references	872
dielectric_constant	1,056 structures with dielectric properties, calculated with DFPT-PBE.	1056
double_perovskites_gap	Band gap of 1306 double perovskites (a_1-b_1-a_2-b_2-O6) calculated using Gritsenko, van Leeuwen, van Lenthe and Baerends potential (gllbsc) in GPAW.	1306
double_perovskites_gap_lumo	Supplementary lumo data of 55 atoms for the double_perovskites_gap dataset.	55
elastic_tensor_2015	1,181 structures with elastic properties calculated with DFT-PBE.	1181
expt_formation_enthalpy	Experimental formation enthalpies for inorganic compounds, collected from years of calorimetric experiments	1276
expt_formation_enthalpy_kingsbury	Dataset containing experimental standard formation enthalpies for solids	2135

https://hackingmaterials.lbl.gov/matminer/dataset_summary.html

Matminer数据库数据访问

```
from matminer.datasets.convenience_loaders import load_elastic_tensor
df = load_elastic_tensor()  # loads dataset in a pandas DataFrame object

from matminer.datasets.convenience_loaders import load_dielectric_constant
df = load_dielectric_constant()

from matminer.datasets.convenience_loaders import load_jarvis_dft_2d
df = load_jarvis_dft_2d()

from matminer.datasets.convenience_loaders import load_jarvis_dft_3d
df = load_jarvis_dft_3d()
```

https://hackingmaterials.lbl.gov/matminer/matminer.datasets.html#module-matminer.datasets.convenience_loaders



数据获取

数据清洗

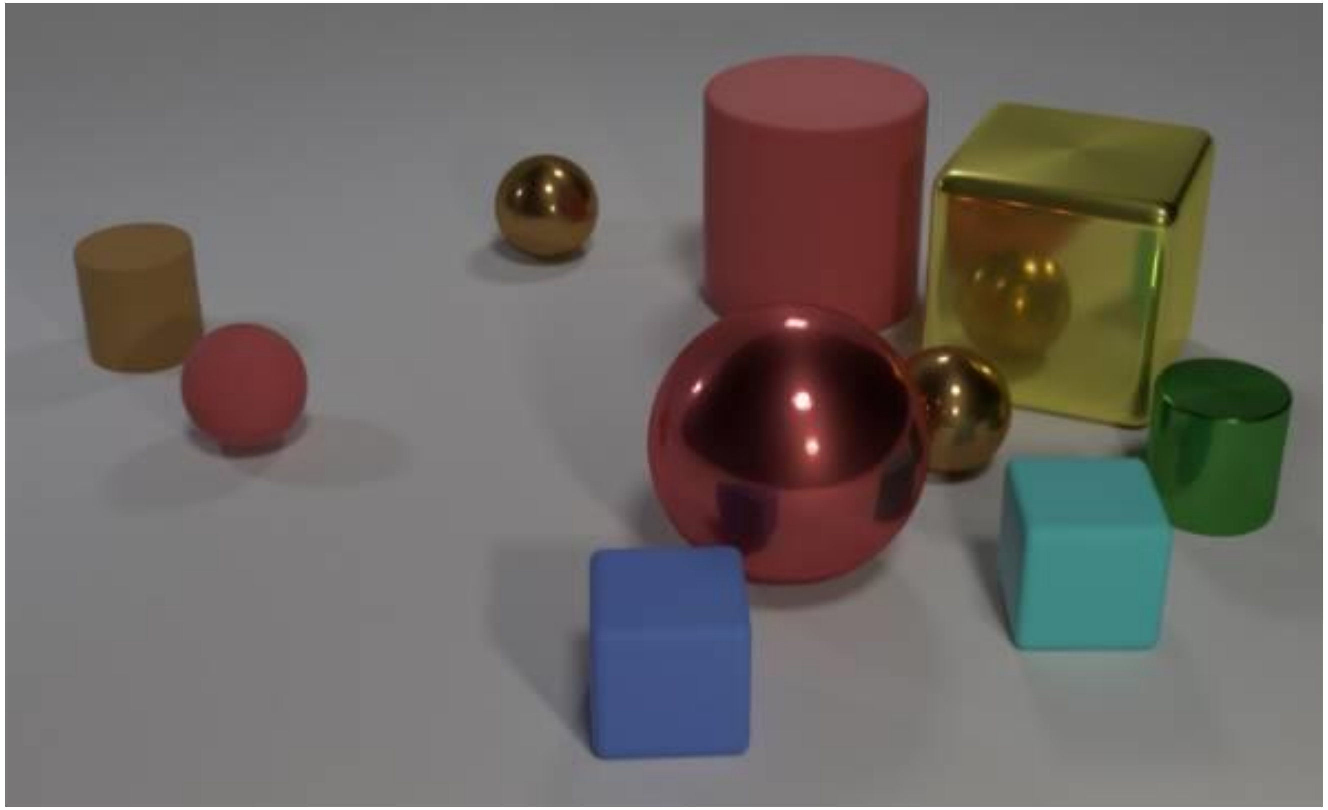
特征工程

模型选择和训练

模型评估

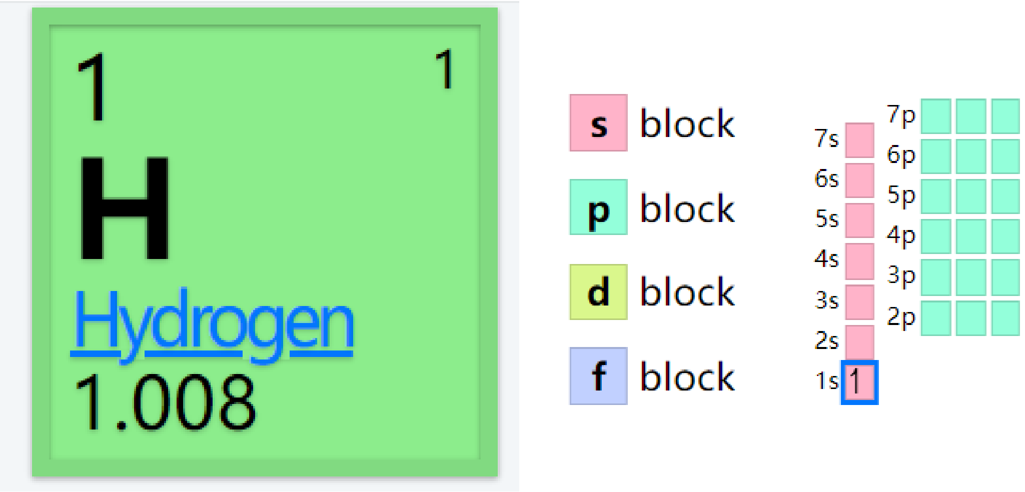
机器学习中的特征/描述符的定义

在机器学习中，特征是被观测对象的一个独立可观测的属性或者特点。于己而言，特征是某些突出性质的表现，于他而言，特征是区分事物的关键。

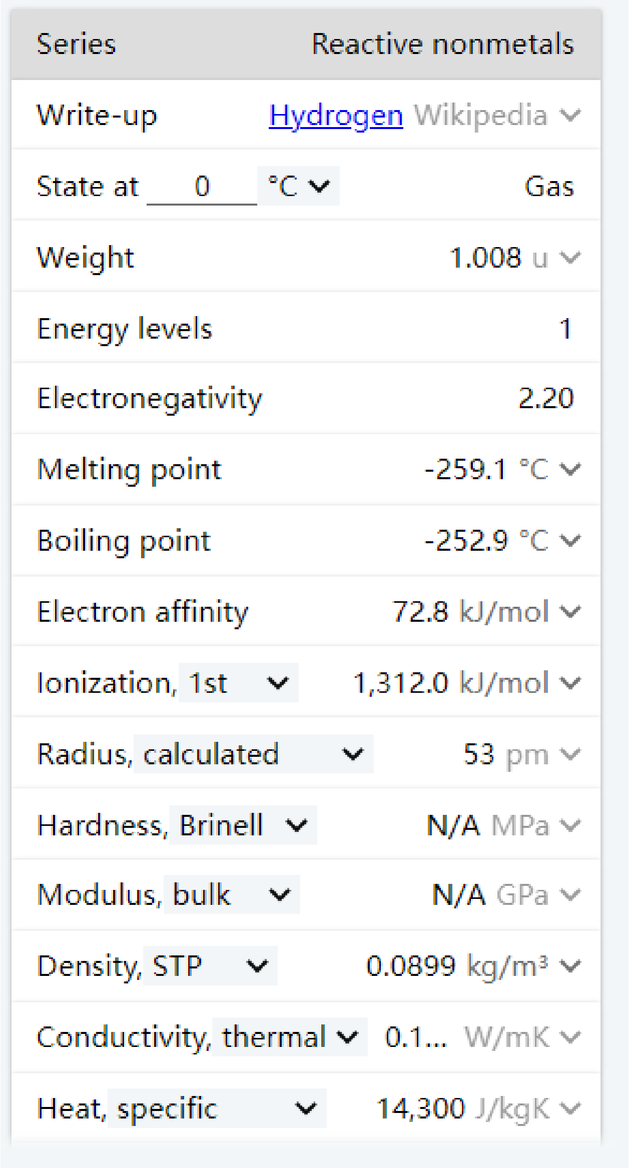


(根据定义这幅图中的特征有哪些？)

原子的基本特征/描述符

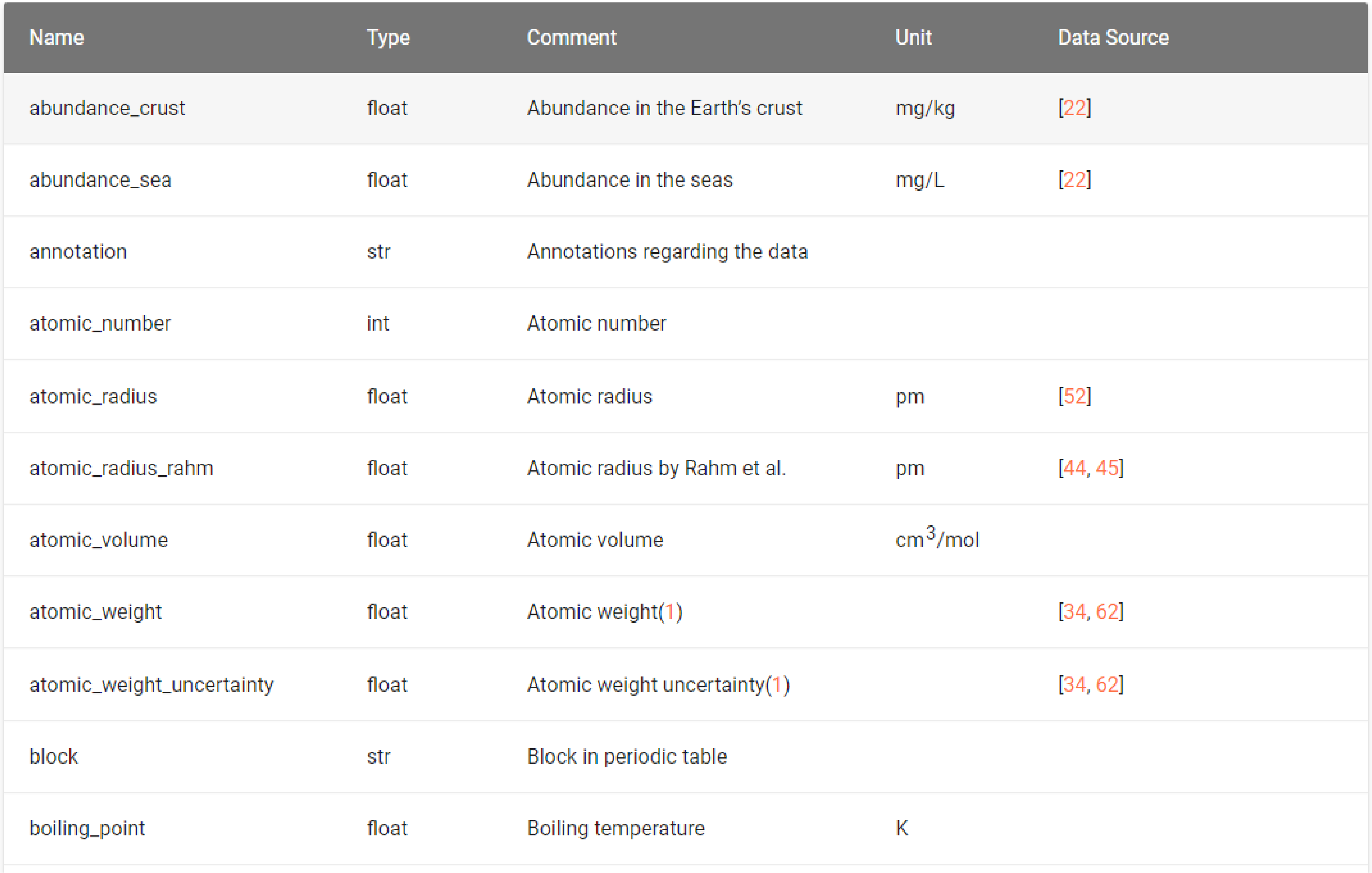


https://ptable.com



Series	Reactive nonmetals
Write-up	Hydrogen Wikipedia
State at 0 °C	Gas
Weight	1.008 u
Energy levels	1
Electronegativity	2.20
Melting point	-259.1 °C
Boiling point	-252.9 °C
Electron affinity	72.8 kJ/mol
Ionization, 1st	1,312.0 kJ/mol
Radius, calculated	53 pm
Hardness, Brinell	N/A MPa
Modulus, bulk	N/A GPa
Density, STP	0.0899 kg/m³
Conductivity, thermal	0.1... W/mK
Heat, specific	14,300 J/kgK

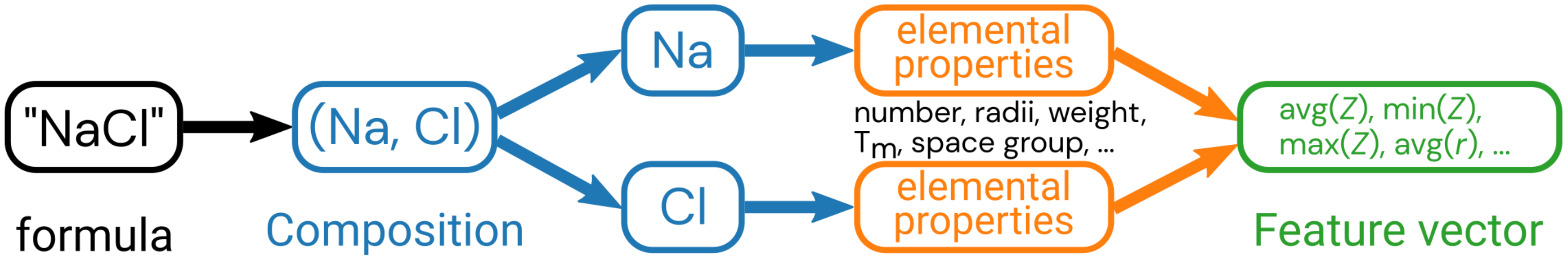
元素描述符的获取



Name	Type	Comment	Unit	Data Source
abundance_crust	float	Abundance in the Earth's crust	mg/kg	[22]
abundance_sea	float	Abundance in the seas	mg/L	[22]
annotation	str	Annotations regarding the data		
atomic_number	int	Atomic number		
atomic_radius	float	Atomic radius	pm	[52]
atomic_radius_rahm	float	Atomic radius by Rahm et al.	pm	[44, 45]
atomic_volume	float	Atomic volume	cm^3/mol	
atomic_weight	float	Atomic weight(1)		[34, 62]
atomic_weight_uncertainty	float	Atomic weight uncertainty(1)		[34, 62]
block	str	Block in periodic table		
boiling_point	float	Boiling temperature	K	

https://mendeleev.readthedocs.io/en/stable/data.html

Matminer中特征工程的主要原理



"NaCl"

formula

(Na, Cl)

Composition

Na

Cl

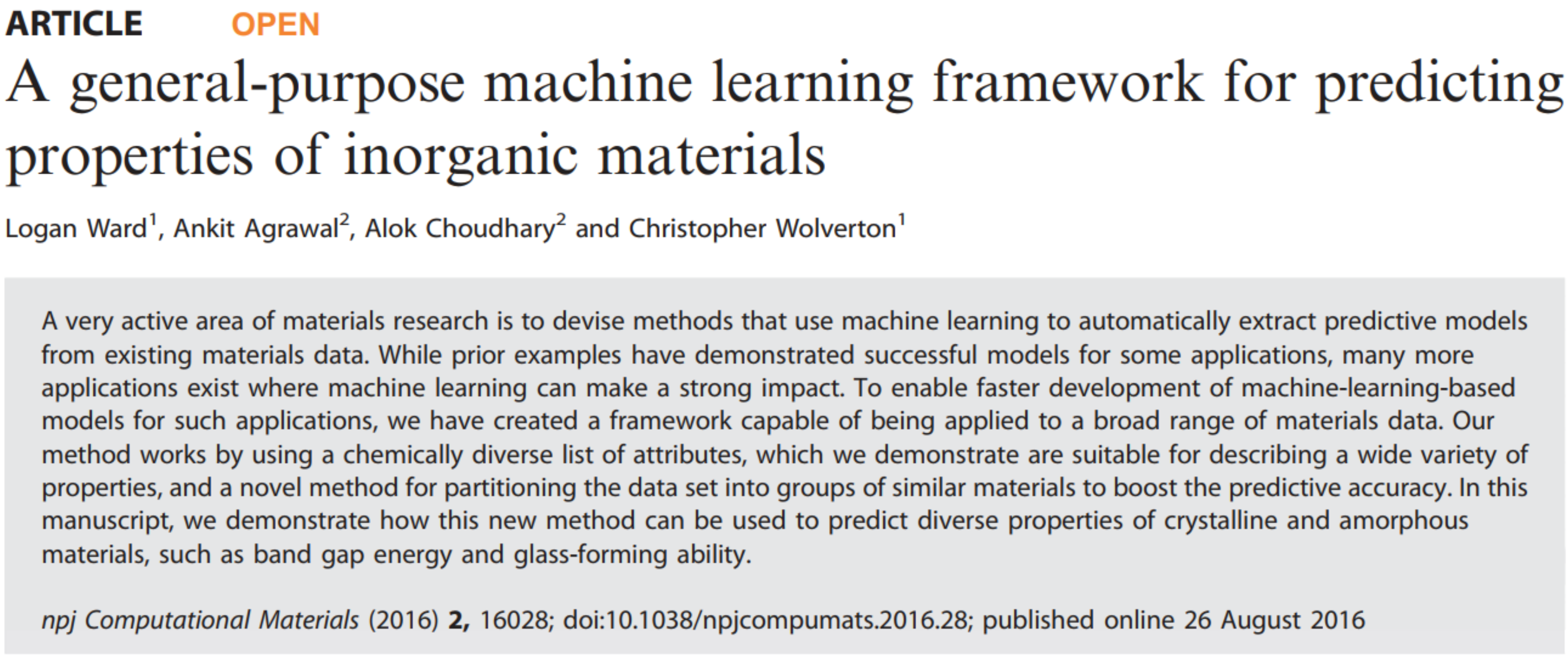
elemental properties

number, radii, weight, T_m, space group, ...

elemental properties

avg(Z), min(Z), max(Z), avg(r), ...

Feature vector



ARTICLE OPEN

A general-purpose machine learning framework for predicting properties of inorganic materials

Logan Ward[1], Ankit Agrawal[2], Alok Choudhary[2] and Christopher Wolverton[1]

A very active area of materials research is to devise methods that use machine learning to automatically extract predictive models from existing materials data. While prior examples have demonstrated successful models for some applications, many more applications exist where machine learning can make a strong impact. To enable faster development of machine-learning-based models for such applications, we have created a framework capable of being applied to a broad range of materials data. Our method works by using a chemically diverse list of attributes, which we demonstrate are suitable for describing a wide variety of properties, and a novel method for partitioning the data set into groups of similar materials to boost the predictive accuracy. In this manuscript, we demonstrate how this new method can be used to predict diverse properties of crystalline and amorphous materials, such as band gap energy and glass-forming ability.

npj Computational Materials (2016) **2,** 16028; doi:10.1038/npjcompumats.2016.28; published online 26 August 2016

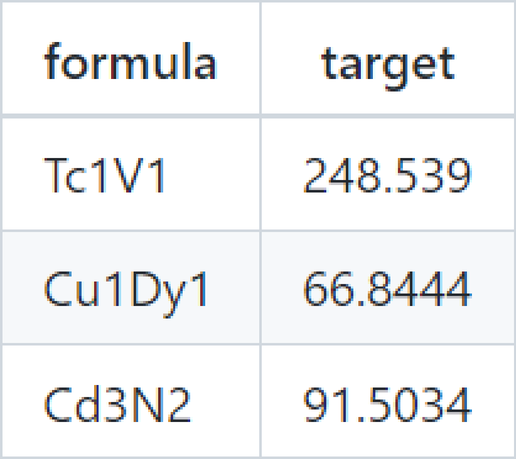
https://www.nature.com/articles/npjcompumats201628

Matminer软件的基本使用—特征化

```
from matminer.featurizers.conversions import StrToComposition
df = StrToComposition().featurize_dataframe(df, "formula")

from matminer.featurizers.composition import ElementProperty
ep_feat = ElementProperty.from_preset(preset_name="magpie")
df = ep_feat.featurize_dataframe(df, col_id="composition")

from matminer.featurizers.conversions import CompositionToOxidComposition
df = CompositionToOxidComposition().featurize_dataframe(df, "composition")

from matminer.featurizers.composition import OxidationStates
os_feat = OxidationStates()
df = os_feat.featurize_dataframe(df, "composition_oxid")
```

https://hackingmaterials.lbl.gov/matminer/featurizer_summary.html
https://mp.weixin.qq.com/s/U99hAXOsNob1sgAehlED3A

CBFV软件的基本使用

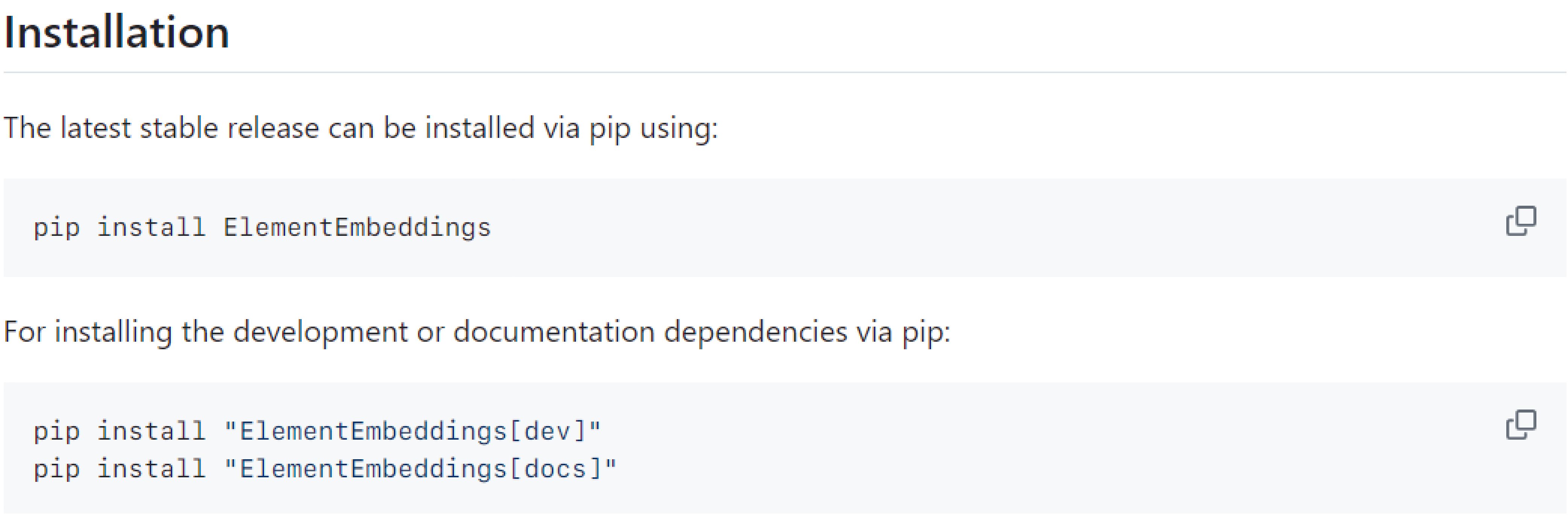


formula	target
Tc1V1	248.539
Cu1Dy1	66.8444
Cd3N2	91.5034

```
>> pip install CBFV

>> from CBFV import composition
>> X, y, formulae, skipped = composition.generate_features(df)
```

https://github.com/kaaiian/CBFV

常用的元素描述符生成软件总结



Installation

The latest stable release can be installed via pip using:

```
pip install ElementEmbeddings
```

For installing the development or documentation dependencies via pip:

```
pip install "ElementEmbeddings[dev]"
pip install "ElementEmbeddings[docs]"
```

https://github.com/WMD-group/ElementEmbeddings

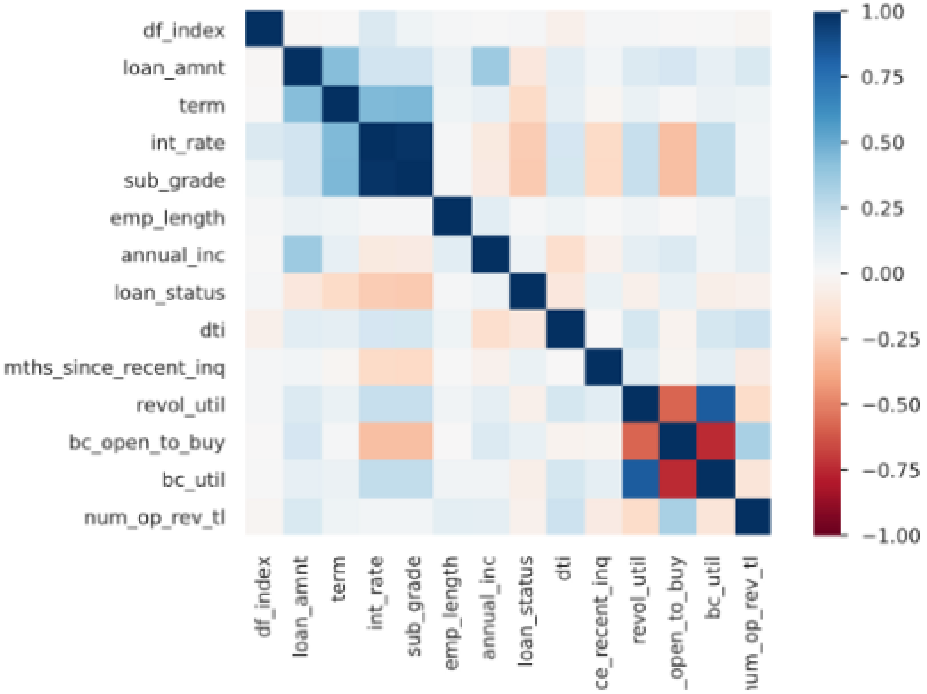
特征之间的相关性判断

当数据集的特征之间具有高度的正相关或负相关时，机器学习模型可能会受到多重共线性的影响。高度相关的特征可能提供相同的信息。在这种情况下可能会导致扭曲或误导的结果，为了解决这个问题，我们可以只保留一个特征，删除多余的特征，这样是不丢失任何信息的。

比如月薪和年薪；虽然它们可能不一样，但它们可能有相同的模式。像逻辑回归和线性回归这样的模型对这个问题很敏感，如果用这样的冗余特征训练模型，可能会产生误导的结果。因此我们应该以消除其中一个为目标。

https://mp.weixin.qq.com/s/cnf1HBuV3shYr2P_Mi68Kg
https://mp.weixin.qq.com/s/rgWyrd53LpBNdZZVQJ97LQ

皮尔逊相关系数(Pearson correlation coefficient)



$$r = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}}.$$

在统计学中，皮尔逊相关系数(Pearson correlation coefficient)，又称皮尔逊积矩相关系数(Pearson product-moment correlation coefficient，简称 PPMCC或PCCs)。用于衡量两个变量X和Y之间的线性相关相关关系，值域在-1与1之间。

皮尔逊相关系数的解释

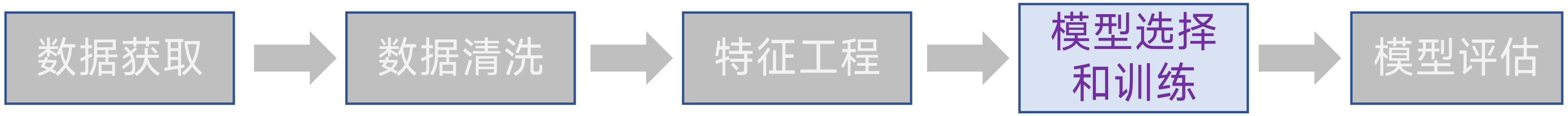
如何理解相关矩阵：相关性范围从+1到-1，其中:

零相关表示变量之间没有关系;

相关性为-1表示完全负相关，这意味着当一个变量上升时，另一个变量下降;

相关性为+1表示完全正相关，这意味着两个变量一起朝同一个方向移动。

https://blog.csdn.net/qq_41721951/article/details/109645921
https://blog.csdn.net/weixin_41744624/article/details/109266940



数据获取

数据清洗

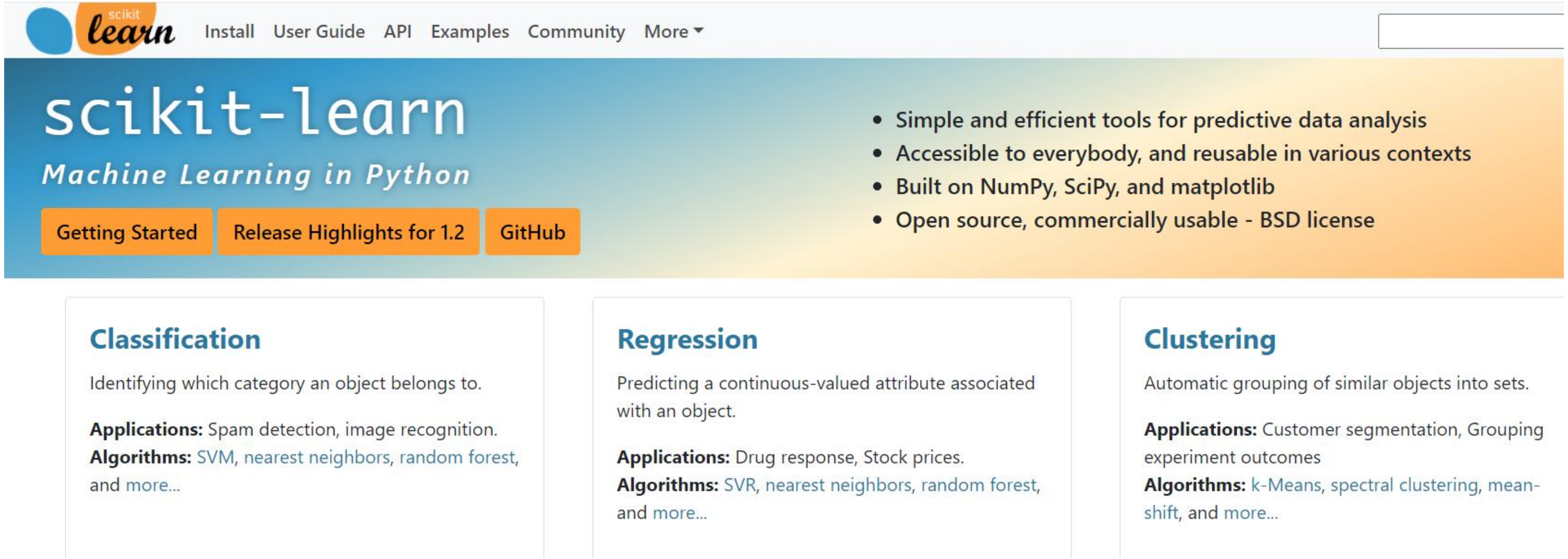
特征工程

模型选择和训练

模型评估

Scikit-learn基本介绍

sklearn是一个Python第三方提供的非常强力的机器学习库，它包含了六大任务模块：分别是分类、回归、聚类、降维、模型选择和预处理。



scikit-learn

Machine Learning in Python

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

https://scikit-learn.org/stable

Scikit-learn机器学习算法

```
### 决策树回归 ###
from sklearn import tree
model_DecisionTreeRegressor = tree.DecisionTreeRegressor()

### 线性回归 ###
from sklearn import linear_model
model_LinearRegression = linear_model.LinearRegression()

### SVM回归 ###
from sklearn import svm
model_SVR = svm.SVR()

### KNN回归 ###
from sklearn import neighbors
model_KNeighborsRegressor = neighbors.KNeighborsRegressor()

### 随机森林回归 ###
from sklearn import ensemble
model_RandomForestRegressor =
ensemble.RandomForestRegressor(n_estimators=20) #用20个决策树
```

https://zhuanlan.zhihu.com/p/368380116

xgboost机器学习算法

xgboost的全称是eXtreme Gradient Boosting，由华盛顿大学的陈天奇博士提出，在Kaggle的希格斯子信号识别竞赛中使用，因其出众的效率与较高的预测准确度而引起了广泛的关注。

```
import xgboost as xgb
from xgboost import plot_importance
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split

model = xgb.XGBRegressor()
```

https://xgboost.readthedocs.io/en/stable/get_started.html
https://blog.csdn.net/weixin_42462804/article/details/104352985
https://zhuanlan.zhihu.com/p/142115015 (XGBoost+Boosting原理简介)
https://zhuanlan.zhihu.com/p/31182879 (史上最详细的XGBoost实战)
https://blog.csdn.net/hocfkey/article/details/124577750 (LightGBM)

LightGBM机器学习算法

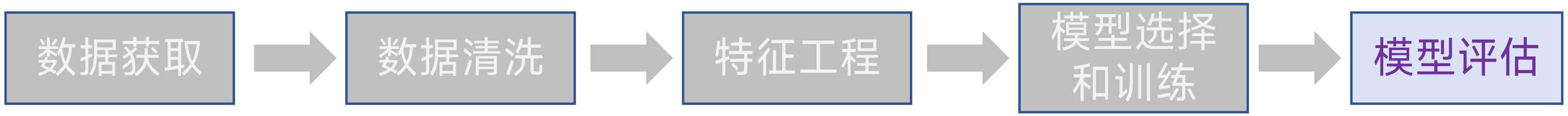
```
#安装命令
pip install lightgbm

#调用命令
import lightgbm as lgbm
model = lgbm.LGBMRegressor()
```

https://lightgbm.readthedocs.io/en/v3.3.2
https://zhuanlan.zhihu.com/p/99069186
https://www.showmeai.tech/article-detail/195
https://blog.csdn.net/hocfkey/article/details/124577750
https://blog.csdn.net/weixin_42813521/article/details/119054445

AdaBoost机器学习算法

```
#调用命令
from sklearn.ensemble import AdaBoostRegressor
model = AdaBoostRegressor()
```

https://zhuanlan.zhihu.com/p/39972832
https://zhuanlan.zhihu.com/p/68770891



数据获取

数据清洗

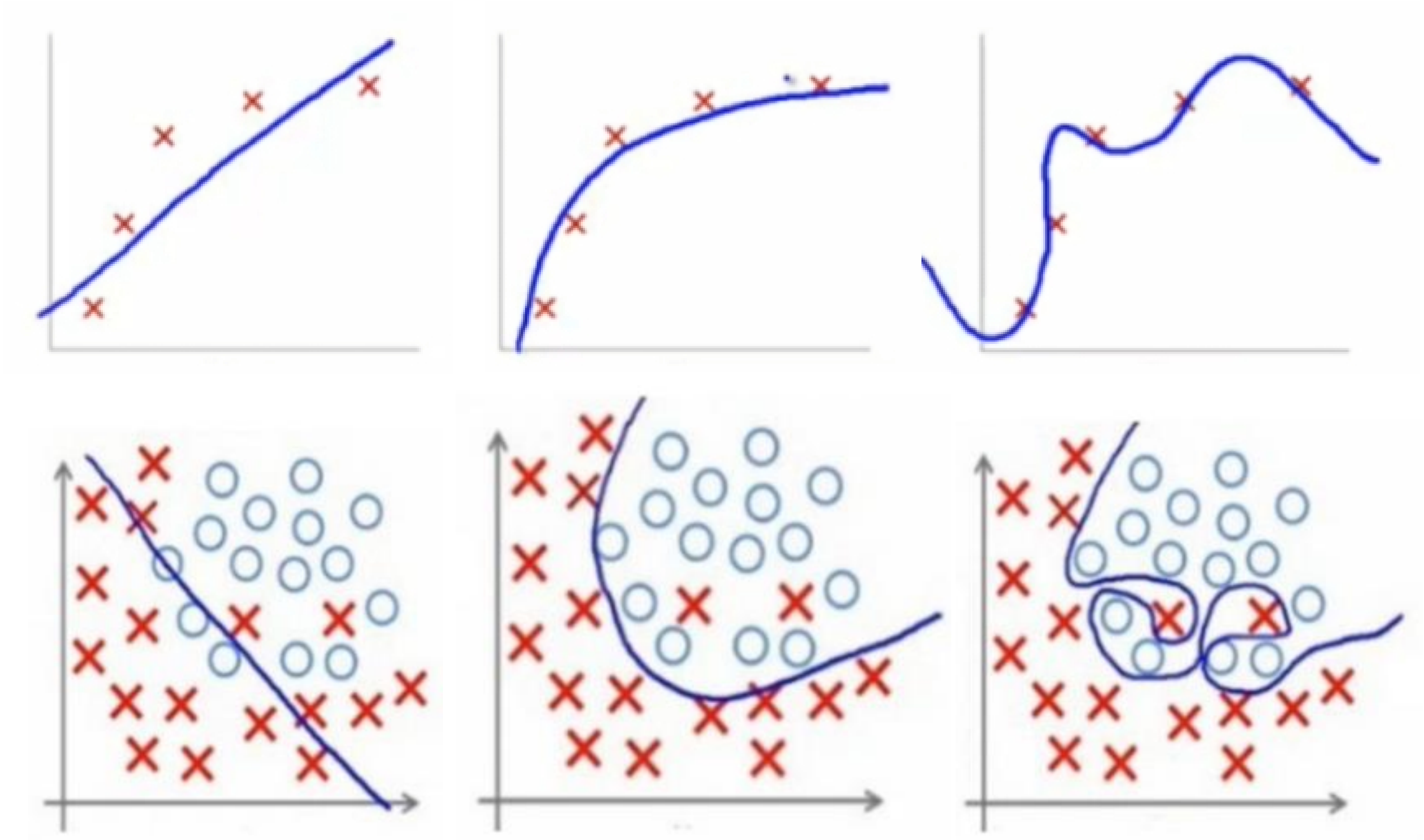
特征工程

模型选择和训练

模型评估

机器学习模型评估

欠拟合　　好的拟合　　过拟合



https://www.cvmart.net/community/detail/6083

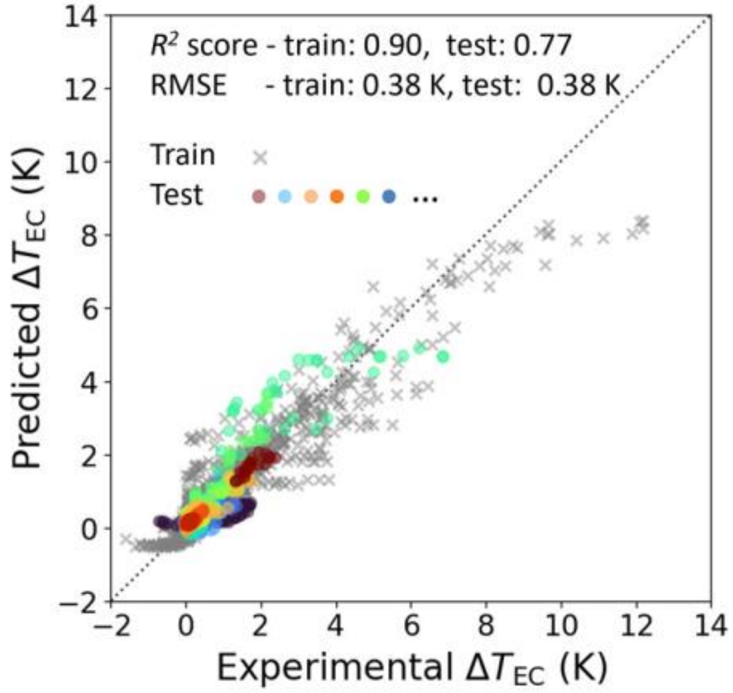
机器学习模型评估—回归模型

1、平均绝对误差（MAE）
2、均方误差（MSE）
3、均方根误差（RMSE）
4、归一化均方根误差（NRMSE）
5、决定系数（R2）

$$\operatorname{RMSD}(\hat{\theta}) = \sqrt{\operatorname{MSE}(\hat{\theta})} = \sqrt{\operatorname{E}((\hat{\theta}-\theta)^2)}.$$

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=0}^{n_{\text{samples}}-1}(y_i-\hat{y}_i)^2}{\sum_{i=0}^{n_{\text{samples}}-1}(y_i-\bar{y})^2}$$

知乎 @Jerry



npj Computational Materials (2022) 8:140

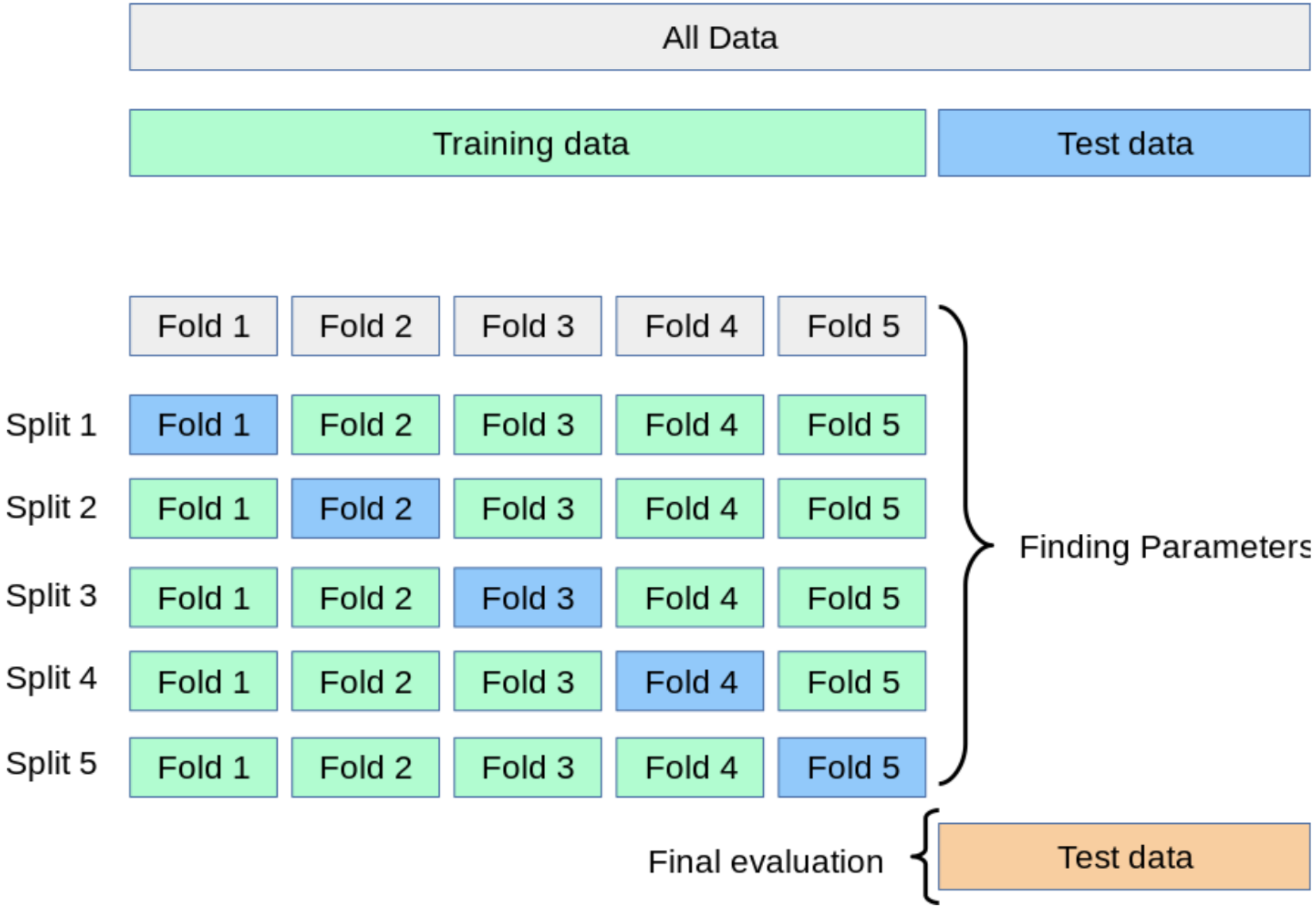
R2是多元回归中的回归平方和占总平方和的比例，它是度量多元回归方程中拟合程度的一个统计量，反映了在因变量y的变差中被估计的回归方程所解释的比例。

R2越接近1，表明回归平方和占总平方和的比例越大，回归线与各观测点越接近,用x的变化来解释y值变差的部分就越多，回归的拟合程度就越好。

https://zhuanlan.zhihu.com/p/86120987

机器学习模型评估—交叉验证法/k折交叉验证

- ✓ 概念：先将数据集D划分为k个大小相似的互斥子集。每一次用k-1个子集的并集作为训练集，剩下的一个子集作为测试集；这样就可以获得k组训练/测试集，从而可进行k次训练和测试，最终返回的是这k个测试结果的均值。
- ✓ 每一个子集Di都尽可能保持数据分布的一致性，即从D中通过分层采样得到。
- ✓ k折交叉验证通常要随机使用不同的划分重复p次，最终的评估结果是这p次k折交叉验证结果的均值。目的是减小因为样本不同而引入的差别。



All Data

Training data | Test data

Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Split 1: Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Split 2: Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Split 3: Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Split 4: Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Split 5: Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5

Finding Parameters

Final evaluation: Test data

scikit-learn中文社区：https://scikit-learn.org.cn/view/6.html
https://www.cnblogs.com/jyroy/p/13547118.html

机器学习模型评估—留出法

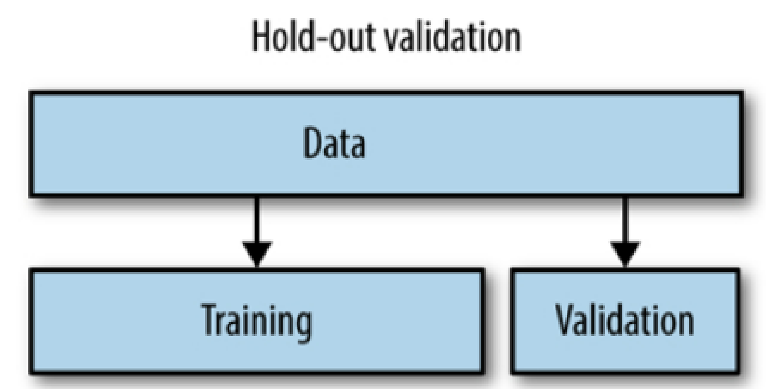
- ✓ 概念：将数据集D划分为两个互斥的集合，其中一个集合为训练集S，另一个为测试集T，在S上训练出模型后，用T来评估其测试误差，作为对泛化误差的估计。

- ✓ 训练/测试集的划分要尽可能保持数据分布的一致性（即类别比例相似），避免因数据划分过程中引入的额外的偏差而对最终结果产生影响。如果从采样的角度来看待数据集的划分过程，则保留类别比例的采样方式通常称为分层采样。

- ✓ 在使用留出法的时候，一般要采用若干次随即划分、重复进行实验评估后取平均值作为留出法的结果。

- ✓ 一般来说，大约2/3~4/5的样本用于训练，其余用于测试。

https://www.cnblogs.com/jyroy/p/13547118.html

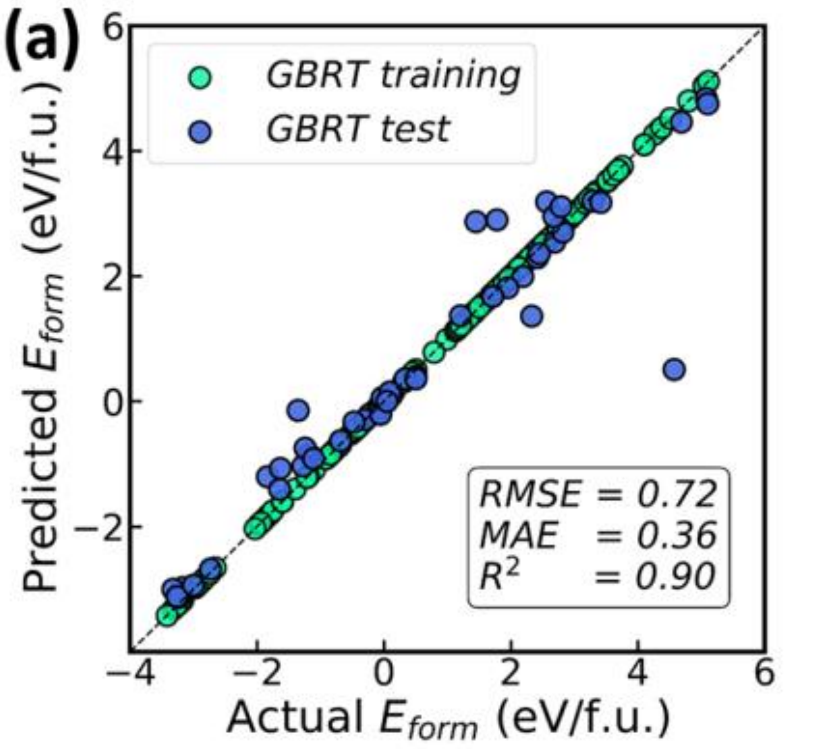


Hold-out validation

Data

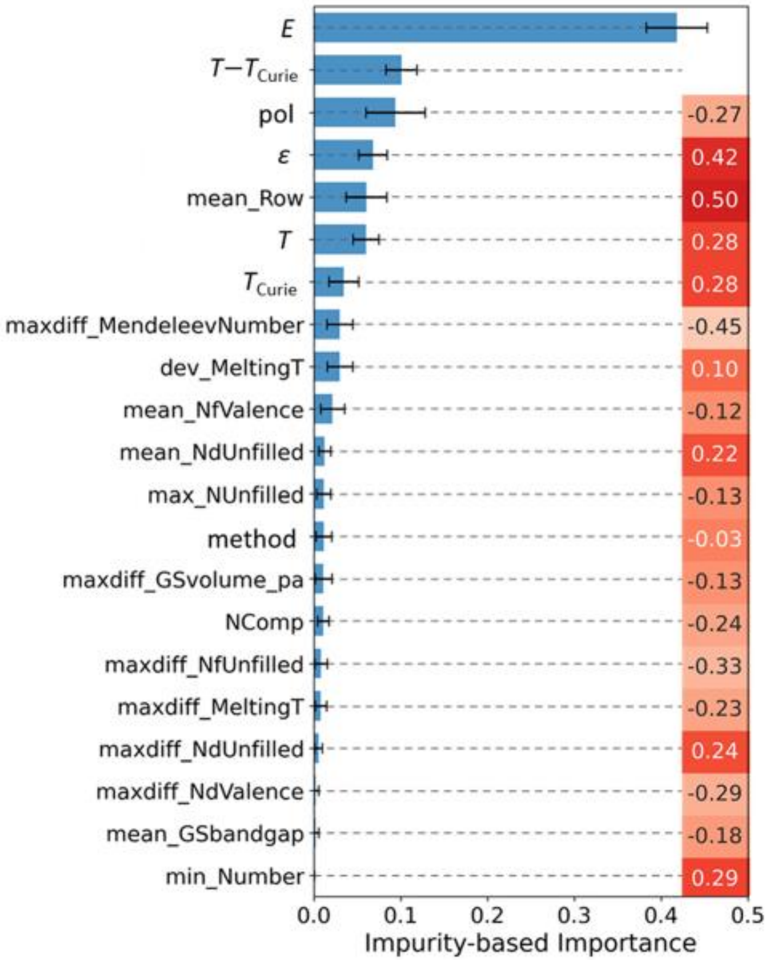
Training

Validation



https://doi.org/10.1021/jacs.2c07434

机器学习模型解释—Feature Importance

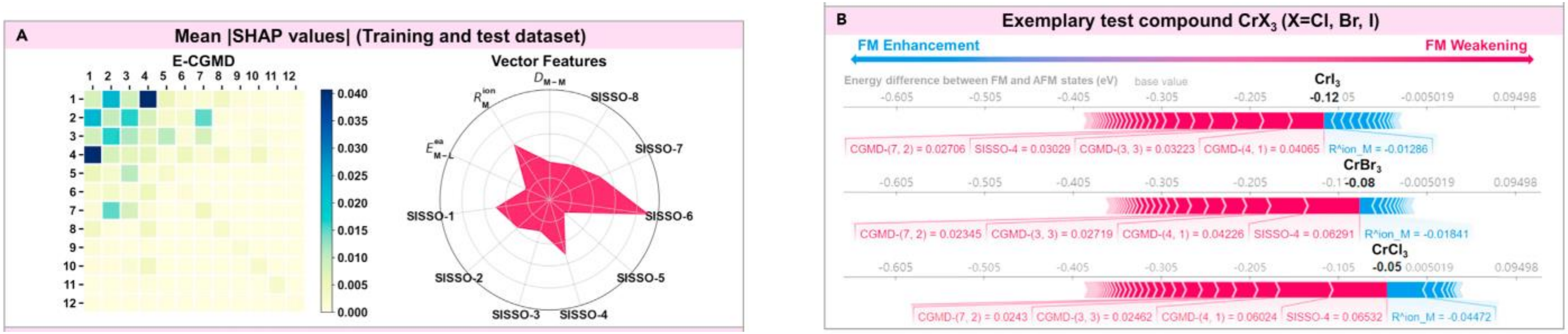


npj Computational Materials (2022) 8:140

https://blog.csdn.net/weixin_44803791/article/details/109776357

机器学习模型解释—SHAP

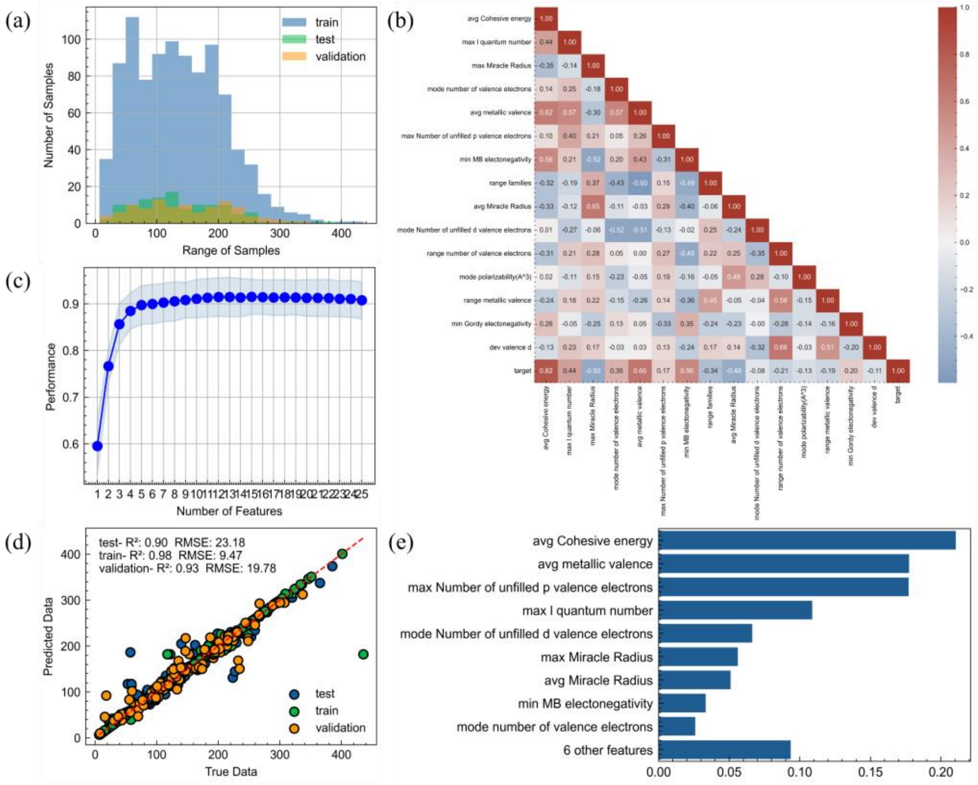
SHAP是Python开发的一个"模型解释"包，可以解释任何机器学习模型的输出。其名称来源于SHapley Additive exPlanation，在合作博弈论的启发下SHAP构建一个加性的解释模型，所有的特征都视为“贡献者”。对于每个预测样本，模型都产生一个预测值，SHAP value就是该样本中每个特征所分配到的数值。



https://doi.org/10.1016/j.chempr.2021.11.009

https://blog.csdn.net/weixin_44803791/article/details/109776357
https://zhuanlan.zhihu.com/p/83412330

机器学习软件推荐



https://github.com/NianSan-H/mlrap

https://github.com/uw-cmg/MAST-ML
https://github.com/ppdebreuck/modnet