

# 기계학습

---

ML프로그래밍을 위한 라이브러리

# Numpy 라이브러리

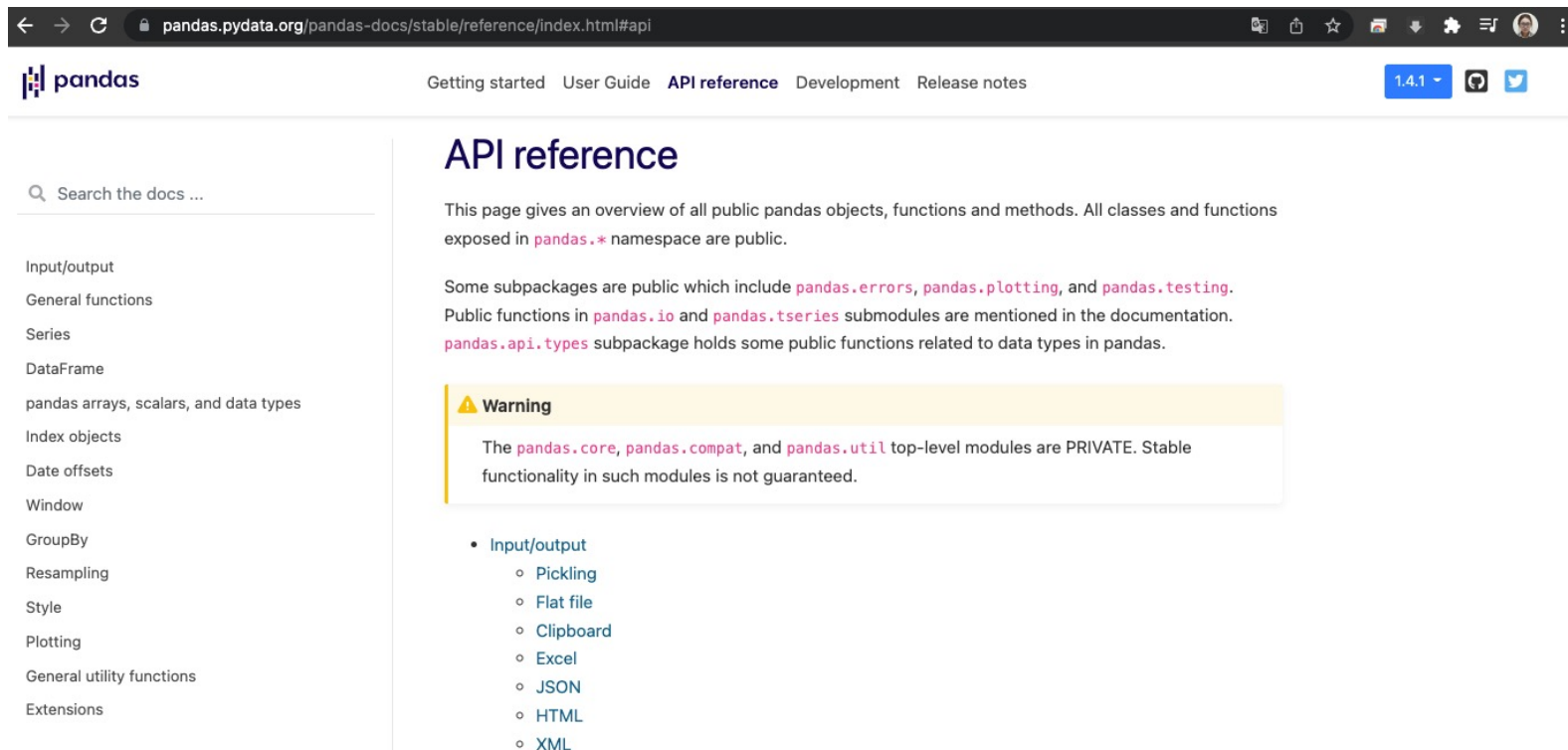
- 넘파이(numpy)
- 판다스(Pandas)
- 맷플랏립(Matplotlib)

# Numpy 라이브러리

- 넘파이(Numpy)란?
  - Numerical Python으로 수치계산을 위해 만들어진 파이썬 라이브러리
  - 넘파이 배열(ndarray)이라는 자료구조를 사용함
    - 넘파이 배열이란 다차원 배열과 행렬을 지원하고 벡터, 행렬 등의 연산을 쉽고 빠르게 수행
- 넘파이 라이브러리 불러오기
  - `import numpy as np`
  - as 뒤에 numpy라 해도 되지만 간결성을 위해 관례적으로 np를 사용함
- 넘파이 실습
  - <https://www.kaggle.com/yukyungchoi/2022-ml-numpy-cheatsheet>

# 판다스 라이브러리

- 판다스(Pandas)란?
  - 파이썬을 이용한 데이터 처리/분석 작업의 필수 라이브러리
  - 판다스 공식 문서
    - <https://pandas.pydata.org/pandas-docs/stable/>



The screenshot shows the pandas API reference page. The browser address bar displays `pandas.pydata.org/pandas-docs/stable/reference/index.html#api`. The page header includes the pandas logo, navigation links (Getting started, User Guide, API reference, Development, Release notes), and a version dropdown set to 1.4.1. A search bar on the left is labeled "Search the docs ...". A sidebar on the left lists various pandas topics: Input/output, General functions, Series, DataFrame, pandas arrays, scalars, and data types, Index objects, Date offsets, Window, GroupBy, Resampling, Style, Plotting, General utility functions, and Extensions. The main content area is titled "API reference" and contains an overview of public pandas objects, functions, and methods. It mentions that all classes and functions are exposed in the `pandas.*` namespace. It also lists some subpackages: `pandas.errors`, `pandas.plotting`, and `pandas.testing`. Public functions in `pandas.io` and `pandas.tseries` submodules are mentioned. The `pandas.api.types` subpackage holds some public functions related to data types. A yellow warning box states that the `pandas.core`, `pandas.compat`, and `pandas.util` top-level modules are PRIVATE and their stable functionality is not guaranteed. A bulleted list under "Input/output" includes: Pickling, Flat file, Clipboard, Excel, JSON, HTML, and XML.

← → ↻ 🔒 pandas.pydata.org/pandas-docs/stable/reference/index.html#api

**pandas** Getting started User Guide **API reference** Development Release notes 1.4.1

## API reference

This page gives an overview of all public pandas objects, functions and methods. All classes and functions exposed in `pandas.*` namespace are public.

Some subpackages are public which include `pandas.errors`, `pandas.plotting`, and `pandas.testing`. Public functions in `pandas.io` and `pandas.tseries` submodules are mentioned in the documentation. `pandas.api.types` subpackage holds some public functions related to data types in pandas.

**Warning**

The `pandas.core`, `pandas.compat`, and `pandas.util` top-level modules are PRIVATE. Stable functionality in such modules is not guaranteed.

- Input/output
  - Pickling
  - Flat file
  - Clipboard
  - Excel
  - JSON
  - HTML
  - XML

# 판다스 라이브러리

- 판다스 라이브러리 불러오기
  - `Import pandas as pd`
- 판다스 데이터 구조
  - 자료구조 3요소: 시리즈 (Series), 데이터프레임 (DataFrame), 패널 (Panel)
    - 데이터프레임이 가장 많이 사용됨
  - 시리즈 란?
    - 1차원 배열의 값에 각 값에 대응하는 인덱스를 부여할 수 있는 구조
  - 데이터프레임 이란?
    - 행과 열을 가지는 자료구조로, 2차원 리스트를 매개변수로 전달
- 판다스 데이터프레임 실습
  - <https://www.kaggle.com/yukyungchoi/2022-ml-pandas-cheatsheet>
- 판다스 프로파일링
  - <https://www.kaggle.com/yukyungchoi/2022-ml-pandas-profiling>

# 판다스 라이브러리

- (실습) 4주차 실습 과제1과 실습 과제2에 사용되는 데이터를 판다스 데이터프레임으로 읽고 프로파일링 해보기
  - <https://www.kaggle.com/c/2022-ml-w4p1>
  - <https://www.kaggle.com/t/e4d47e37ea3b41879d6b4670bc9f06b1>

## [기계학습][4주차][실습과제1] KNN을 이용하여 재배환경 별 작물 종류 예측 문제를 해결하라

- 캐글 리더보드: <https://www.kaggle.com/c/2022-ml-w4p1>
- 과제 제출 (1) : 캐글리더보드에 답안 제출하여 베이스라인 넘기 후 캐글 노트북 담당 조교에게 공유
- 과제 제출 (2) : KNN의 하이퍼파라미터 변경에 따른 성능결과 분석 리포트 A4 한장 이내로 제출
- 과제 제출 기한: 2022년 04월 03일 오후 11시 59분
- 제출할 곳: [admin@rcv.sejong.ac.kr](mailto:admin@rcv.sejong.ac.kr)
  - 이메일 제목 : [기계학습][4주차][실습과제1] 재배환경별 작물종류 예측 (학번\_이름)

<https://www.kaggle.com/yukyungchoi/2022-ml-w4p1-profiling>

## [기계학습][4주차][실습과제2] KNN을 이용하여 자동차 가격 예측 문제를 해결하라

- 캐글 리더보드: <https://www.kaggle.com/t/e4d47e37ea3b41879d6b4670bc9f06b1>
- 과제 제출 (1) : 캐글리더보드에 답안 제출하여 베이스라인 넘기 후 캐글 노트북 담당 조교에게 공유
- 과제 제출 (2) : KNN의 하이퍼파라미터 변경에 따른 성능결과 분석 리포트 A4 한장 이내로 제출
- 과제 제출 기한: 2022년 04월 03일 오후 11시 59분
- 제출할 곳: [admin@rcv.sejong.ac.kr](mailto:admin@rcv.sejong.ac.kr)
  - 이메일 제목 : [기계학습][4주차][실습과제2] 자동차 가격예측 (학번\_이름)

<https://www.kaggle.com/yukyungchoi/2022-ml-w4p2-profiling>

# Matplotlib 라이브러리

- Matplotlib이란?
  - 맷플롯립(Matplotlib)은 데이터를 차트나 플롯으로 시각화하는 패키지임
  - 데이터 분석에서 Matplotlib은 데이터 분석 이전에 데이터 이해를 위한 시각화나, 데이터 분석 후에 결과를 시각화하기 위해서 사용됨
  - Matplotlib을 사용할 때 주로 서브패키지인 pyplot을 사용하며, pyplot은 MATLAB의 인터페이스와 유사하게 작동할 수 있도록 MATLAB을 사용하는 사용자층이 쉽게 matplotlib으로 옮겨오도록 돕고 있음
- Matplotlib 실습
  - <https://www.kaggle.com/yukyungchoi/2022-ml-matplotlib-cheatsheet>
- 다른 시각화 툴
  - Matplotlib으로 간단한 차트나 그래프를 그리는 것은 쉬운 일이나 예쁘게 다듬고 커스터마이징 하기에는 부적합함
  - 추천할 만한 시각화 툴
    - seaborn, plotly, plotnine

# 사이킷런

---

scikit learn



# 오픈라이브러리

## ■ 기계 학습을 위한 라이브러리 #1: Scikit-Learn

- 다양한 머신러닝 알고리즘을 구현한 파이썬 라이브러리
- 심플하고 일관성 있는 API, 유용한 온라인 문서, 풍부한 예제
- 머신러닝을 위한 쉽고 효율적인 개발 라이브러리 제공
- 다양한 머신러닝 관련 알고리즘과 개발을 위한 프레임워크와 API제공
- 많은 사람들이 사용하며 다양한 환경에서 검증된 라이브러리

The screenshot shows the Scikit-Learn website at <https://scikit-learn.org/stable/>. The page has a blue header with the 'scikit-learn' logo and the tagline 'Machine Learning in Python'. Navigation links include 'Install', 'User Guide', 'API', 'Examples', and 'More'. A search bar and a 'Go' button are also present. Below the header, there are three buttons: 'Getting Started', 'Release Highlights for 0.24', and 'GitHub'. A list of features is displayed: 'Simple and efficient tools for predictive data analysis', 'Accessible to everybody, and reusable in various contexts', 'Built on NumPy, SciPy, and matplotlib', and 'Open source, commercially usable - BSD license'. The main content area is divided into three sections: 'Classification' (Identifying which category an object belongs to, with applications like spam detection and algorithms like SVM), 'Regression' (Predicting a continuous-valued attribute, with applications like drug response and algorithms like SVR), and 'Clustering' (Automatic grouping of similar objects, with applications like customer segmentation and algorithms like k-Means). Each section includes a small figure illustrating the concept.

**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/>

# 오픈라이브러리

- 기계 학습을 위한 라이브러리 #1: Scikit-Learn

Classification  
Regression  
Clustering  
Semi-Supervised Learning  
Feature Selection  
Feature Extraction  
Manifold Learning  
Dimensionality Reduction  
Kernel Approximation  
Hyperparameter Optimization  
Evaluation Metrics  
Out-of-core learning  
.....



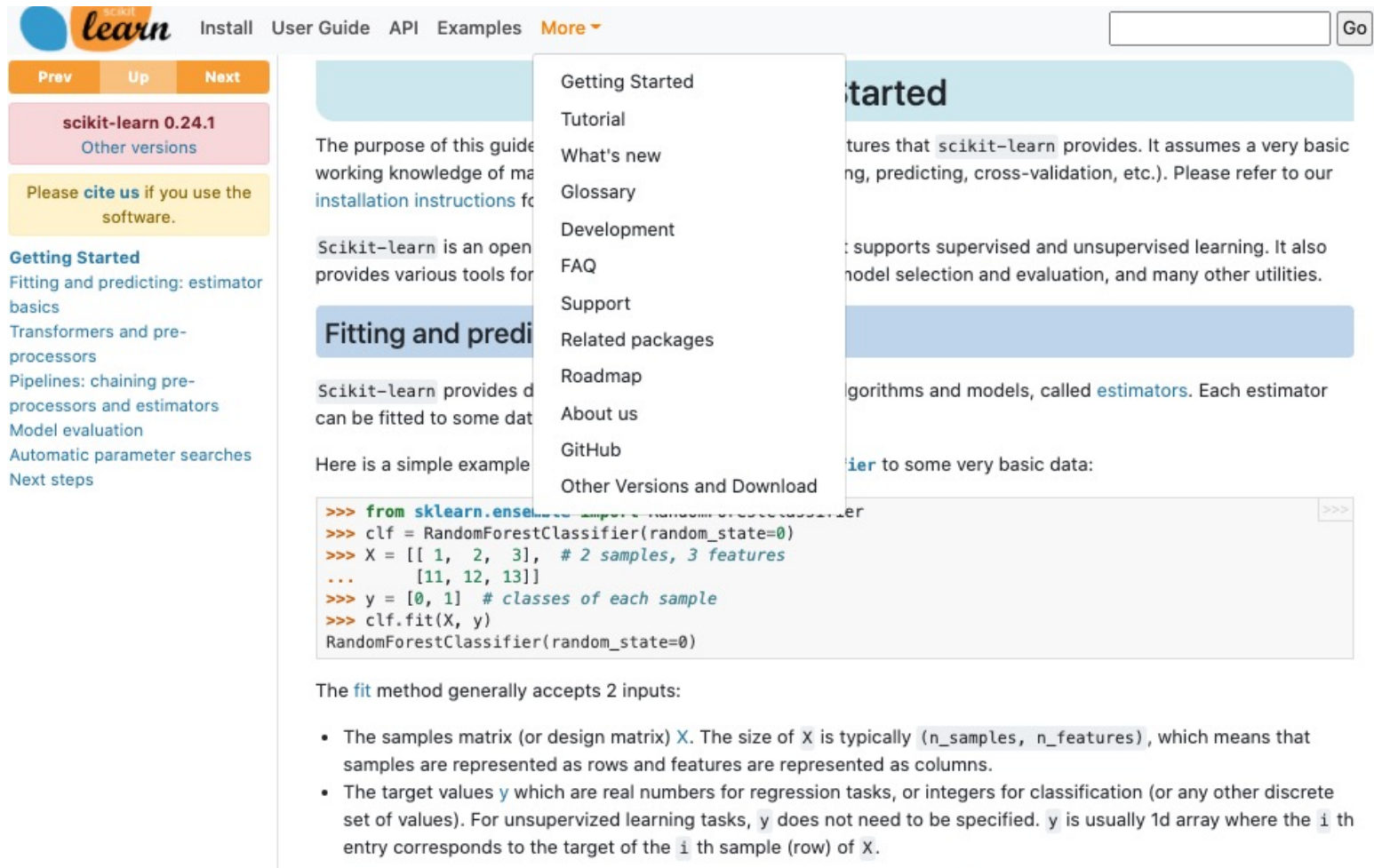
# 오픈라이브러리

## ■ 기계 학습을 위한 라이브러리 #1: Scikit-Learn

모듈	설명
<code>sklearn.datasets</code>	내장된 예제 데이터 세트
<code>sklearn.preprocessing</code>	다양한 데이터 전처리 기능 제공 (변환, 정규화, 스케일링 등)
<code>sklearn.feature_selection</code>	특징(feature)을 선택할 수 있는 기능 제공
<code>sklearn.feature_extraction</code>	특징(feature) 추출에 사용
<code>sklearn.decomposition</code>	차원 축소 관련 알고리즘 지원 (PCA, NMF, Truncated SVD 등)
<code>sklearn.model_selection</code>	교차 검증을 위해 데이터를 학습/테스트용으로 분리, 최적 파라미터를 추출하는 API 제공 (GridSearch 등)
<code>sklearn.metrics</code>	분류, 회귀, 클러스터링, Pairwise에 대한 다양한 성능 측정 방법 제공 (Accuracy, Precision, Recall, ROC-AUC, RMSE 등)
<code>sklearn.pipeline</code>	특징 처리 등의 변환과 ML 알고리즘 학습, 예측 등을 묶어서 실행할 수 있는 유틸리티 제공
<code>sklearn.linear_model</code>	선형 회귀, 릿지(Ridge), 라쏘(Lasso), 로지스틱 회귀 등 회귀 관련 알고리즘과 SGD(Stochastic Gradient Descent) 알고리즘 제공
<code>sklearn.svm</code>	서포트 벡터 머신 알고리즘 제공
<code>sklearn.neighbors</code>	최근접 이웃 알고리즘 제공 (k-NN 등)
<code>sklearn.naive_bayes</code>	나이브 베이즈 알고리즘 제공 (가우시안 NB, 다항 분포 NB 등)
<code>sklearn.tree</code>	의사 결정 트리 알고리즘 제공
<code>sklearn.ensemble</code>	앙상블 알고리즘 제공 (Random Forest, AdaBoost, GradientBoost 등)
<code>sklearn.cluster</code>	비지도 클러스터링 알고리즘 제공 (k-Means, 계층형 클러스터링, DBSCAN 등)

# 오픈라이브러리

## ■ 기계 학습을 위한 라이브러리 #1: Scikit-Learn



The screenshot shows the Scikit-Learn website. The top navigation bar includes links for 'Install', 'User Guide', 'API', 'Examples', and 'More'. A search bar is located on the right. The left sidebar contains a 'Prev', 'Up', and 'Next' navigation bar, followed by a section for 'scikit-learn 0.24.1' with a link to 'Other versions'. Below this is a 'Please cite us if you use the software.' notice. The 'Getting Started' section lists topics like 'Fitting and predicting: estimator basics', 'Transformers and pre-processors', 'Pipelines: chaining pre-processors and estimators', 'Model evaluation', 'Automatic parameter searches', and 'Next steps'. The 'Fitting and predicting' section is highlighted. The main content area features a 'Getting Started' header, a paragraph about the library's purpose, and a code example for fitting a Random Forest Classifier. A dropdown menu is open over the 'Getting Started' header, listing various links like 'Getting Started', 'Tutorial', 'What's new', 'Glossary', 'Development', 'FAQ', 'Support', 'Related packages', 'Roadmap', 'About us', 'GitHub', and 'Other Versions and Download'.

Getting Started

The purpose of this guide is to provide a working knowledge of machine learning with `scikit-learn`. It assumes a very basic understanding of machine learning (e.g., fitting, predicting, cross-validation, etc.). Please refer to our [installation instructions](#) for more details.

`Scikit-learn` is an open source machine learning library that provides various tools for machine learning.

**Fitting and predicting**

`Scikit-learn` provides a simple interface to fit a model to some data. Here is a simple example:

```
>>> from sklearn.ensemble import RandomForestClassifier
>>> clf = RandomForestClassifier(random_state=0)
>>> X = [[ 1,  2,  3], # 2 samples, 3 features
...      [11, 12, 13]]
>>> y = [0, 1] # classes of each sample
>>> clf.fit(X, y)
RandomForestClassifier(random_state=0)
```

The `fit` method generally accepts 2 inputs:

- The samples matrix (or design matrix) `X`. The size of `X` is typically `(n_samples, n_features)`, which means that samples are represented as rows and features are represented as columns.
- The target values `y` which are real numbers for regression tasks, or integers for classification (or any other discrete set of values). For unsupervised learning tasks, `y` does not need to be specified. `y` is usually 1d array where the `i`th entry corresponds to the target of the `i`th sample (row) of `X`.

# 오픈라이브러리

- Estimator API
  - 일관성
    - 모든 객체는 일관된 문서를 갖춘 제한된 메서드 집합에서 비롯된 공동 인터페이스 공유
  - 제한된 객체 계층 구조
    - 알고리즘만 파이썬 클래스에 의해 표현
    - 데이터 세트는 표준 포맷(Numpy 배열, Pandas DataFrame, Scipy 희소 행렬)으로 표현
    - 매개변수명은 표준 파이썬 문자열 사용
  - 합리적인 기본 값
    - 모델이 사용자 지정 파라미터를 필요로 할 때 라이브러리가 적절한 기본 값을 정의

# 오픈라이브러리

- API 사용 방법

- Scikit-Learn 에서 적절한 estimator 클래스를 임포트해서 모델의 클래스 선택
- 클래스를 원하는 값으로 인스턴스화해서 모델의 하이퍼 파라미터 선택
- 데이터를 특징 배열과 대상 벡터로 배치
- 모델 인스턴스의 fit() 메서드를 호출해 모델을 데이터에 적합
- 모델을 새 데이터에 대해서 적용
  - 지도 학습: 대체로 predict() 메서드를 사용해 알려지지 않은 데이터에 대한 레이블 예측
  - 비지도 학습: 대체로 transform()이나 predict() 메서드를 사용해 데이터의 속성을 변환하거나 추론