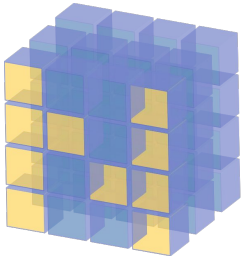
The Array API

Thomas Moreau

https://data-apis.org/array-api/

https://data-apis.org/dataframe-api/

Many array manipulation libraries…



NumPy

DASK

CuPy

PyTorch

… not really compatible!

Array API - layer to create compatible libraries

- Minimal set to implement libraries on top of these arrays.
- Based on numpy.
- Focus on pure-python code (no C-extension).
- Some limitations compared to classical numpy:
  - No inplace operations!
  - Functions only work with arrays.

Demo usage

```python
import array_api_compat

def least_square(X, y, lr=1e-3, max_iter=100):
    # Get the array namespace
    xp = array_api_compat.array_namespace(X, y)

    # Write standard algorithm
    w = xp.zeros(X.shape[1:], dtype=X.dtype, device=getattr(X, 'device', None))
    for i in range(max_iter):
        w = w - lr * xp.matrix_transpose(X) @ ( X @ w - y)
    return w
```

```python
import jax.numpy as jnp
least_square(jnp.array(X), jnp.array(y))
```

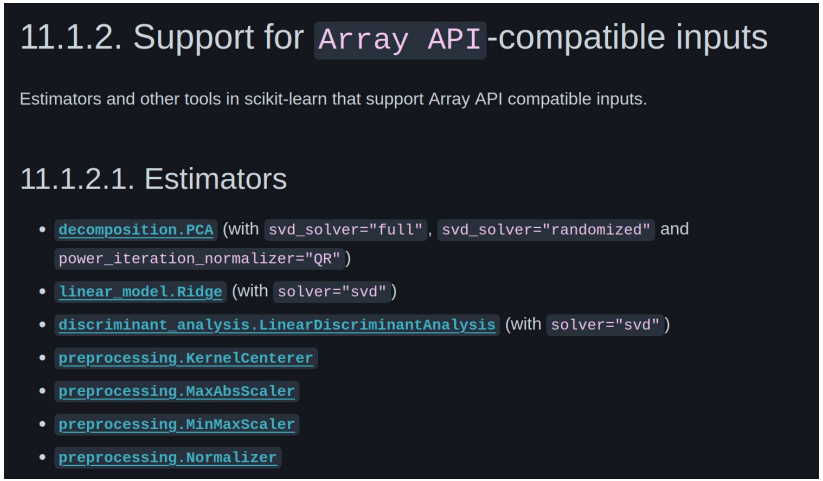
```python
import jax
least_square_jitted = jax.jit(
    least_square, static_argnames=['lr','max_iter']
)
least_square_jitted(jnp.array(X), jnp.array(y))
```

```python
import numpy as np

X = np.random.randn(1000, 10)
beta = np.random.randn(10)
y = X @ beta + np.random.randn(1000)

least_square(X, y)
```

```python
import torch
least_square(torch.tensor(X), torch.tensor(y))
```

```python
least_square(
    torch.tensor(X, device='cuda:3'),
    torch.tensor(y, device='cuda:3')
)
```

sklearn is making some algorithms compatible with the API:



Array API

https://data-apis.org/array-api

DataFrame API

https://data-apis.org/dataframe-api/

Good idea to think of it when developing novel algorithms that only require python calls!