Learning From Data
Review Session: Scientific Programming in Python

Feng Zhao zhaof17@mails.tsinghua.edu.cn

9/18/2020

Overview

- Environment choices
- Popular packages in Python
  - `numpy`
  - `scipy`
  - `matplotlib`
- GitHub classroom

Scientific Programming Tools

- Operating systems, containers and clusters
- Programming language
  - interpreted language: Python
  - compiled language: C, C++
- Package manager for Python
  - pip: `https://pypi.org`
  - conda: `https://anaconda.org`

In this course, conda is recommended.

Tips for using conda

- Download: `https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/`
- Setup Mirror: `https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/`
- Install packages: `conda install scipy matplotlib`
- Check your install: `python -c "import numpy; print(numpy.__version__)"`

Numpy

Numpy: n-dimensional array manipulation

Code snippet

create a vector of length 3 and compute its ℓ_2 norm

```python
import numpy as np
a = np.array([1, 2, 3])
print(np.linalg.norm(a))
```

compute the eigenvalues of a square matrix:

```python
A = np.array([[1, 2], [3, 4]])
print(np.linalg.eig(A)[0])
```

compute the summation of each row for a matrix

```python
A = np.array([[1, 2], [3, 4], [5, 6]])
print(np.sum(A, axis=1))
```

matrix product

```python
print(A @ np.array([1, 1]))
```

Scipy

Scipy: algorithms of applied mathematics

Code snippet

the pdf of normal distribution

```python
import scipy.stats
x = np.linspace(-3, 3)
y = scipy.stats.norm.pdf(x)
print(x, y)
```

Matplotlib

Matplotlib – plotting experiment results

Code snippet

sample data from Gaussian and draw histogram

```python
    import matplotlib.pyplot as plt
    c = np.random.normal(size=1000)
    plt.hist(c, density=True)
    plt.plot(x, y)
    plt.show()
```

Summary

- numpy
- scipy
- matplotlib

Further reference:
`https://cs231n.github.io/python-numpy-tutorial/`

GitHub Classroom

Places to submit your programming assignments

Steps

1. Register an account for GitHub
2. Use Invitation URL to get the starting code
3. Upload your modification to your own workspace
4. Check the Autograding; Should be ✓; No X mark

Have a try

Linear regression

Consider the linear observation model

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{c}$$

where the X is a 10000×10 matrix, and $\boldsymbol{w}, \boldsymbol{c}$ are column vectors with length 10 and 10000. Use programming to find the a that minimizes the loss $\frac{1}{2}\|X\boldsymbol{w} - y\|_2^2$. See details in the **linear_regression.py**.

- Invitation URL: `https://classroom.github.com/a/ylEoHU6G`
- Hint: use the formula: $\boldsymbol{w} = (X^TX)^{-1}X^T\boldsymbol{y}$.