CS 412 Introduction to Machine Learning

Regression - Code Tutorial

Instructor: Wei Tang
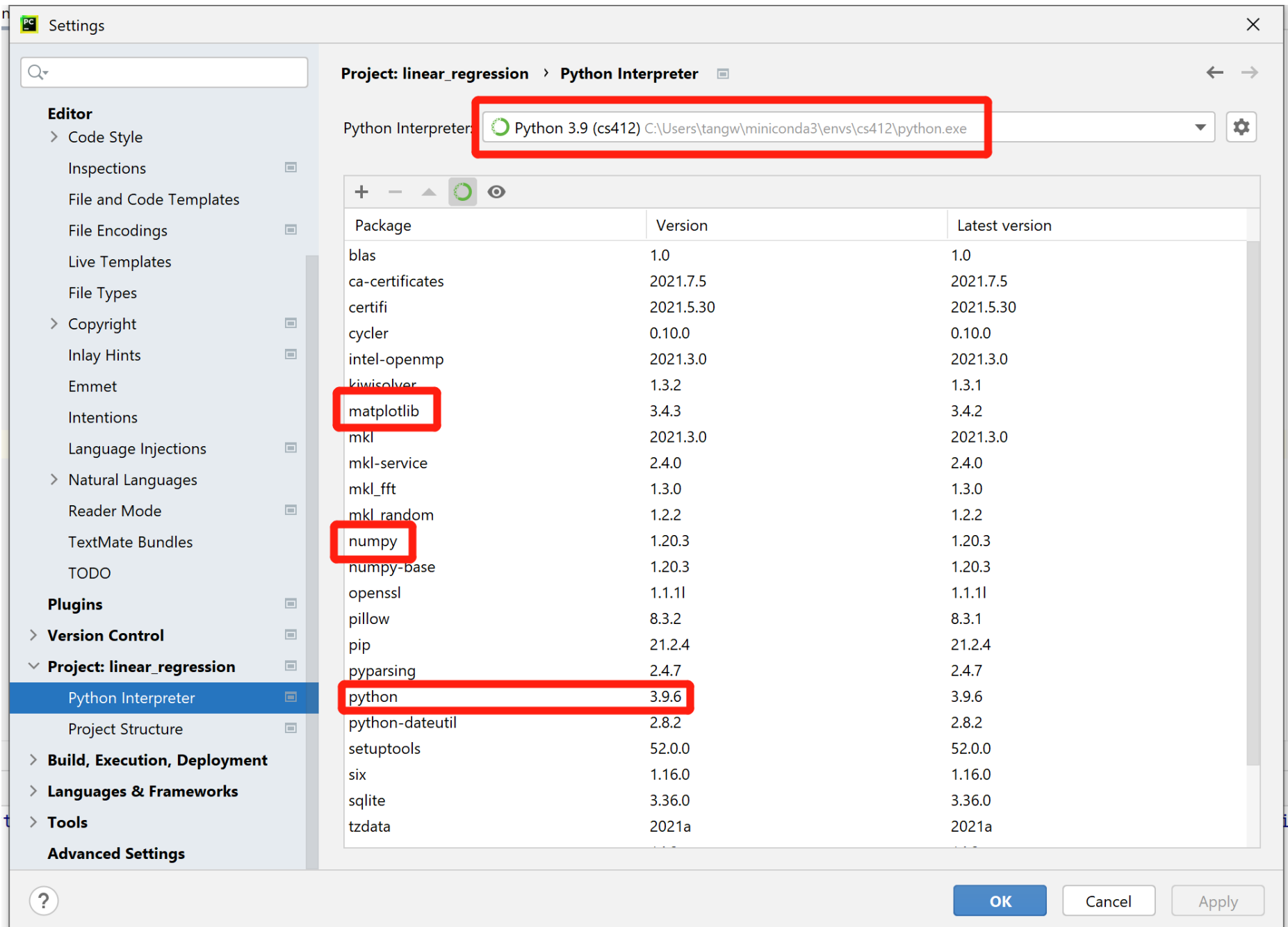
Department of Computer Science

University of Illinois at Chicago

Chicago IL 60607

https://tangw.people.uic.edu
tangw@uic.edu

Slides credit: Shawon Ashraf

Instructor's Environment

- OS: Windows 10
  - Linux, e.g., Ubuntu, and Mac are fine.

- Python 3.9
  - <u>Do not</u> use Python 2

- Package management: Conda (miniconda3)
  - Recommended
  - Alternatives: pip, virtualenv

- IDE: PyCharm (free community version)
  - VS code, vim + terminal, etc are fine.
  - Configure a Conda virtual environment in PyCharm:
   https://www.jetbrains.com/help/pycharm/conda-support-creating-conda-virtual-environment.html

Settings

Package	Version	Latest version
blas	1.0	1.0
ca-certificates	2021.7.5	2021.7.5
certifi	2021.5.30	2021.5.30
cycler	0.10.0	0.10.0
intel-openmp	2021.3.0	2021.3.0
kiwisolver	1.3.2	1.3.1
matplotlib	3.4.3	3.4.2
mkl	2021.3.0	2021.3.0
mkl-service	2.4.0	2.4.0
mkl_fft	1.3.0	1.3.0
mkl_random	1.2.2	1.2.2
numpy	1.20.3	1.20.3
numpy-base	1.20.3	1.20.3
openssl	1.1.1l	1.1.1l
pillow	8.3.2	8.3.1
pip	21.2.4	21.2.4
pyparsing	2.4.7	2.4.7
python	3.9.6	3.9.6
python-dateutil	2.8.2	2.8.2
setuptools	52.0.0	52.0.0
six	1.16.0	1.16.0
sqlite	3.36.0	3.36.0
tzdata	2021a	2021a

Code tutorial

1. Run the code to see the results
2. Study code by modules
3. Run step by step and check what is going on

```python
import numpy as np
import matplotlib.pyplot as plt
```

Generate data

```python
def generate_data():
    X = np.random.uniform(0, 1, (100,1))
    y = 3 * X + 2 + np.random.randn(100, 1)*0.3

    return X, y
```
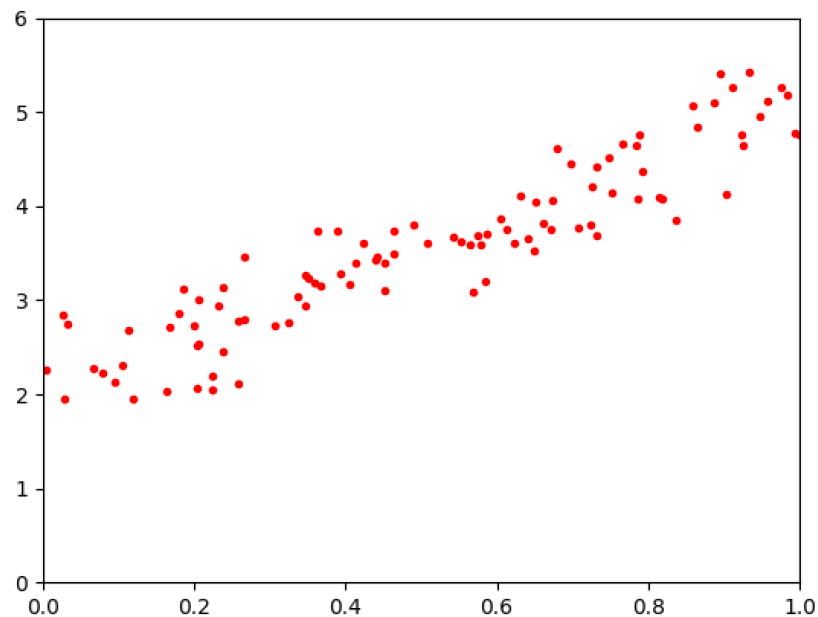
Learning (training)

```python
def learning(X, y):
    X_transpose = X.T
    estimated_params = np.linalg.inv(X_transpose.dot(X)).dot(X_transpose).dot(y)
```

$$\hat{\beta} = \left(X^{\mathsf{T}}X\right)^{-1}X^{\mathsf{T}}Y$$

```python
X_b = np.concatenate((np.ones((100, 1)), X), axis=1)
theta = learning(X_b, y)
```

Inference (testing)

```python
test_X = np.array([[0], [1]])
test_X_b = np.concatenate((np.ones((2, 1)), test_X), axis=1)

prediction = test_X_b.dot(theta)
```