

“나는 데이터로 축구를 한다



Football Analytics

Day 0 / 사전교육

© 2024 Winning Twelve / v0.1.1 / Last Modified 2024-07-10

목 차

1 소개.....	2
2 질의응답 및 지원.....	2
3 Python.....	2
3.1 개념 잡기.....	2
3.2 실습.....	3
4 Conda.....	3
4.1 Conda.....	3
4.2 Miniconda vs. Anaconda.....	3
4.2.1 Anaconda.....	3
4.2.2 Miniconda.....	3
4.3 개발 환경 만들기.....	4
5 Jupyter Notebook.....	4
5.1 대화형 코드 실행.....	4
5.2 Markdown.....	4
5.3 Jupyter Notebook vs. Jupyter Lab.....	4
5.3.1 사용자 인터페이스.....	5
5.3.2 기능 및 확장성.....	5
5.3.3 사용 경험.....	5
5.3.4 호환성.....	5
6 Pandas.....	5
6.1 DataFrame.....	5
6.2 Indexing & Slicing.....	5
6.3 Sorting.....	5
6.4 Boolean Indexing.....	6
6.5 Index & Columns.....	6
7 Matplotlib.....	6

1 소개

이 사전 교육은 축구 데이터 분석에 기초가 되는 중요한 내용들을 포함하고 있으며, 참가자들이 캠프 동안 본격적으로 실습에 참여하기 위해 필요로 되는 가장 기본적인 지식과 기술을 습득할 수 있도록 돕습니다.

- **Python**: 데이터 과학에서 가장 널리 사용되는 프로그래밍 언어로, 기본 문법과 데이터 처리 방법을 학습합니다.
- **Conda**: Python 프로젝트를 위한 환경 관리와 패키지 관리를 용이하게 하는 도구로, 효율적인 개발 환경 설정 방법을 익힙니다.
- **Jupyter Notebook**: 대화형 코드 실행과 데이터 시각화가 가능한 웹 기반 인터페이스로, 실시간 코드 테스트 및 결과 확인 방법을 배웁니다.
- **Pandas**: 데이터 분석을 위한 핵심 라이브러리로, 데이터 구조화, 처리, 분석 기술을 배웁니다.
- **Matplotlib**: 데이터를 시각화하는 강력한 도구로, 그래프와 차트를 생성하여 데이터를 직관적으로 표현하는 방법을 배웁니다.
- **Git과 GitHub**: 소스 코드 버전 관리 및 협업을 위한 필수 도구로, 프로젝트의 변경 사항을 추적하고 다른 참가자들과 효율적으로 협력하는 방법을 익힙니다.

(*) 캠프에서 사용될 내용에만 맞추어져 있기 때문에,
예를 들어 Python이나 시각화 라이브러리 등의 모든 내용을 다루지는 않습니다.

2 질의응답 및 지원

- 이번 캠프를 위한 전용 Discord 채널이 있습니다. 질문이 있으신 분들은 Discord에 질문을 주시면 최대한 도와드리도록 하겠습니다!
- #facamp-summer24: <https://discord.gg/vstpAVhw>

3 Python

Python(파이썬)이라는 컴퓨터 언어가 처음이신 분들은 꼭 시청하시기 바랍니다.

3.1 개념 잡기

아래 강의는 비전공자가 Python 코딩에 대한 개념을 빠르고 쉽게 잡을 수 있도록 도와주는 강의입니다.

- <https://www.youtube.com/watch?v=T6z-0dpXPvU>
- 처음부터 1:17:08까지만 시청하시면 되겠습니다.
- 소요시간: 약 77분

3.2 실습

아래 강의는 조금 더 실습 위주의 강의입니다.

- <https://youtu.be/ftQZo7XaTOA?si=sS2hT3fjTSD0-ayW&t=3110>
- 00:51:50부터 05:28:28까지만 시청하시면 되겠습니다.
- 소요시간: 약 4시간 반

(*) 위 두 강의에서는 <http://python.org>에서 제공하는 Python과 Microsoft사의 Visual Studio Code (VSCode)를 사용해서 개발 환경을 만들지만, 캠프에서는 Miniconda와 Jupyter Lab을 사용할 예정입니다.

4 Conda

4.1 Conda

Conda는 파이썬 및 기타 프로그래밍 언어로 작성된 패키지, 라이브러리, 환경을 관리하기 위한 오픈 소스 패키지 관리 시스템 및 환경 관리 시스템입니다. Conda는 데이터 과학 및 머신러닝 작업을 위한 도구로 널리 사용되며, 다양한 운영 체제에서 동작합니다. 주요 기능은 다음과 같습니다:

1. **패키지 관리:** Conda는 패키지 설치, 업데이트, 삭제, 종속성 관리를 쉽게 할 수 있게 해줍니다. Conda 저장소에는 파이썬뿐만 아니라 다양한 언어의 패키지가 포함되어 있어, 필요한 패키지를 쉽게 설치할 수 있습니다.
2. **환경 관리:** Conda는 서로 다른 프로젝트나 작업을 위한 격리된 환경을 생성하고 관리할 수 있습니다. 이를 통해 각 환경별로 다른 패키지 버전이나 종속성을 유지할 수 있습니다.
3. **운영 체제 간 호환성:** Conda는 Windows, macOS, Linux 등 여러 운영 체제에서 동일하게 사용할 수 있어, 다양한 플랫폼에서 일관된 작업 환경을 제공합니다.

4.2 Miniconda vs. Anaconda

Conda는 Anaconda와 Miniconda의 핵심 구성 요소로, 이 두 배포판은 모두 Conda를 포함하고 있습니다. Anaconda와 Miniconda는 Conda를 기반으로 하여 Python 패키지 관리와 환경 관리 기능을 제공합니다. Anaconda는 Conda와 함께 많은 수의 패키지와 도구들을 포함한 배포판으로, 처음부터 다양한 데이터 과학 도구들을 사용할 수 있습니다. 반면, Miniconda는 Conda와 최소한의 패키지만 포함한 경량 배포판으로, 필요한 패키지만 개별적으로 설치하여 사용할 수 있습니다. 이들의 관계를 자세히 살펴보면 다음과 같습니다:

4.2.1 Anaconda

- **포함 내용:** Anaconda는 Conda와 함께 1500개 이상의 과학 계산, 데이터 과학, 머신러닝 등에 필요한 패키지들을 포함하고 있는 배포판입니다.
- **목적:** 데이터 과학 작업에 필요한 모든 도구와 라이브러리를 한 번에 제공하여, 사용자들이 설치 과정에서 겪는 불편함을 최소화합니다.
- **크기:** 약 3GB의 대용량 설치 파일로, 많은 패키지와 도구들을 미리 포함하고 있습니다.

4.2.2 Miniconda

- **포함 내용:** Miniconda는 Conda와 최소한의 필수 패키지만 포함된 경량 배포판입니다.

- 목적: 사용자가 필요한 패키지와 도구만 선택적으로 설치할 수 있도록 하여, 더 작은 설치 파일과 유연한 환경 구성을 제공합니다.
- 크기: 약 50MB의 소용량 설치 파일로, 빠르고 간편하게 설치할 수 있습니다.

4.3 개발 환경 만들기

캠프에서는 Miniconda를 설치해서 사용할 예정이고, Miniconda 설치할 때 Python이 자동으로 설치됩니다. 그리고 나서 Jupyter Lab을 설치할 예정입니다.

- https://drive.google.com/file/d/1Pt4xQ5EohKkfGvgH6t9HpdakCd26MNo/view?usp=drive_link
- 소요시간: 약 10분

5 Jupyter Notebook

5.1 대화형 코드 실행

코드를 쭉 길게 작성한 후, “실행해줘”라고 하던 기존의 방식과는 약간 다르게, 코드를 셀(cell) 단위로 쪼개서 실행하고 결과를 바로 바로 확인할 수 있는, 꼭 대화를 주고 받는 듯한 형태로 코드 작성을 도와주는 개발 환경입니다.

Python은 어느 정도 아는데, Jupyter Notebook은 처음 접하시는 분들은 꼭 시청하시기 바랍니다.

- https://youtu.be/dJfq-eCi7KI?si=1_hehXTCFo38HuG9&t=430
- 00:07:10부터 끝까지 시청하시면 됩니다 (앞부분은 Anaconda 설치 부분이라 건너뛰기)
- 소요시간: 1.25배속으로 약 30분

(*) 위 강의에서는 Anaconda를 사용하지만, 저희는 Miniconda를 사용해서 만든 환경을 이용하게 됩니다.

5.2 Markdown

뒤에 “Notebook(노트북)”이라는 이름이 붙은 이유는, 단순히 코드 작성만 하도록 도와주는 개발 환경이 아니라, 코드에 곁들여서 서식 있는 문서 넣기, 수학기공식 넣기, 시각화 넣기, 동영상 넣기 등등이 가능하기 때문에, 실험실에서 실험하면서 노트를 적어나가는 듯한 형태로 작업을 할 수 있기 때문입니다. 이런 문서 작성을 돕기 위해 Markdown(MD)이라는 언어를 사용하는데, 웹페이지를 만들때 사용하는 HTML과 비슷하지만, 훨씬 문법이 쉽고 간결/간단한 것이 특징입니다.

- 기본: <https://gist.github.com/ihoneymon/652be052a0727ad59601>
- 고급(영어): <https://www.markdownguide.org/basic-syntax/>
- HTML을 아시는 분들: <https://www.heropy.dev/p/B74sNE>
- 소요시간: 약 30분

5.3 Jupyter Notebook vs. Jupyter Lab

Jupyter Notebook은 단순하고 직관적인 인터페이스로 빠르게 시작할 수 있는 반면, Jupyter Lab은 더 많은 기능과 확장성을 제공하여 복잡한 작업을 효율적으로 처리할 수 있도록 돕습니다.

5.3.1 사용자 인터페이스

- **Jupyter Notebook:** 간단한 웹 기반 인터페이스로, 노트북 형태로 코드를 작성하고 실행할 수 있습니다. 주로 코드 셀(Code Cell), 마크다운 셀(Markdown Cell), 출력 셀(Output Cell)로 구성되어 있습니다.
- **Jupyter Lab:** Jupyter Notebook의 대부분의 기능을 그대로 가지고 있되, 더 발전된 인터페이스를 제공하며, 여러 파일을 탭이나 패널로 동시에 열 수 있습니다. 다양한 작업 공간을 지원하여 더 복잡한 워크플로우를 관리하기에 적합합니다.

5.3.2 기능 및 확장성

- **Jupyter Notebook:** 기본적인 기능에 초점을 맞추고 있으며, 주로 단일 노트북에서 작업을 수행합니다.
- **Jupyter Lab:** 다양한 플러그인과 확장 기능을 통해 기능을 확장할 수 있습니다. 터미널(Terminal), 텍스트 편집기(Text Editor), 데이터 파일 뷰어(Data File Viewer) 등 여러 도구를 통합하여 사용할 수 있습니다.

5.3.3 사용 경험

- **Jupyter Notebook:** 초보자에게 더 직관적이며, 빠르게 노트북을 작성하고 실행하기에 좋습니다.
- **Jupyter Lab:** 더 고급 사용자에게 적합하며, 복잡한 데이터 분석 및 프로젝트 관리를 위한 다양한 기능을 제공합니다.

5.3.4 호환성

- **Jupyter Notebook:** Jupyter Notebook 포맷(.ipynb)을 기본으로 사용합니다.
- **Jupyter Lab:** Jupyter Notebook과 완전히 호환되며, 기존의 노트북 파일을 그대로 사용할 수 있습니다.

6 Pandas

데이터 분석을 위한 핵심 라이브러리로, 데이터 구조화, 처리, 분석 기술을 배웁니다.

6.1 DataFrame

- https://youtu.be/SVjKsvvhWlQ?si=0f_Hxt7eHWEbI0Ex
- 소요시간: 약12분

6.2 Indexing & Slicing

- <https://youtu.be/W2KYhekdtdfU?si=xUYfFwCpsU-AWry0>
- 소요시간: 약9분

6.3 Sorting

- <https://youtu.be/63eJ2izsFus?si=ROXYnzE0OeoVg9g2>

- 소요시간: 약4분

6.4 Boolean Indexing

- <https://youtu.be/poaOpqec8ho?si=8slg8wPR561OeG99>
- 소요시간: 약9분

6.5 Index & Columns

- <https://youtu.be/sKtMAVnpHHs?si=Zv558LUN2YGy-3d4>
- 소요시간: 약15분

7 Matplotlib

데이터를 시각화하는 강력한 도구로, 그래프와 차트를 생성하여 데이터를 직관적으로 표현하는 방법을 배웁니다.

- <https://www.youtube.com/playlist?list=PLc937W-QYjyjsi3SCeHbpoa6vrPZ0LVc0>
- 소요시간: 약1시간