Programming for Data Analysis

A. M. Sadeghzadeh, Ph.D.

Sharif University of Technology
Computer Engineering Department (CE)
Machine Learning MicroMaster

July 11, 2023

Today's Agenda

Course logistics

Course information

- Course Name: Programming for Data Analysis
  - Time: Sat-Mon 18:00-19:30
  - Room: https://vc.sharif.edu/ch/amsadeghzadeh

Course information

- Course Name: Programming for Data Analysis
  - Time: Sat-Mon 18:00-19:30
  - Room: https://vc.sharif.edu/ch/amsadeghzadeh

- Instructor
  - Amir Mahdi Sadeghzadeh (amsadeghzadeh@gmail.com)
    - Office: CE-501
    - Office hours: by appointment and through email

Course information

- Course Name: Programming for Data Analysis
  - Time: Sat-Mon 18:00-19:30
  - Room: https://vc.sharif.edu/ch/amsadeghzadeh

- Instructor
  - Amir Mahdi Sadeghzadeh (amsadeghzadeh@gmail.com)
    - Office: CE-501
    - Office hours: by appointment and through email
    - Telegram id: @amirmahdii70
    - Bale id: @amirmahdii70

Course information

- Course Name: Programming for Data Analysis
  - Time: Sat-Mon 18:00-19:30
  - Room: https://vc.sharif.edu/ch/amsadeghzadeh
- Instructor
  - Amir Mahdi Sadeghzadeh (amsadeghzadeh@gmail.com)
    - Office: CE-501
    - Office hours: by appointment and through email
    - Telegram id: @amirmahdii70
    - Bale id: @amirmahdii70
- Quera: https://quera.org/course/add_to_course/course/14462/
  - Notebooks and Lecture slides
  - Discussions and HWs

Course information

- Course Name: Programming for Data Analysis
  - Time: Sat-Mon 18:00-19:30
  - Room: https://vc.sharif.edu/ch/amsadeghzadeh
- Instructor
  - Amir Mahdi Sadeghzadeh (amsadeghzadeh@gmail.com)
    - Office: CE-501
    - Office hours: by appointment and through email
    - Telegram id: @amirmahdii70
    - Bale id: @amirmahdii70
- Quera: https://quera.org/course/add_to_course/course/14462/
  - Notebooks and Lecture slides
  - Discussions and HWs
- TAs
  - TBA

References

1. Wes McKinney, Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, 3rd edition, 2022.
2. ”Python Programming for Data Science” By Tomas Beuzen
3. Jake VanderPlas, Python Data Science Handbook, 2016.

Course objective

Course objective

- Gain knowledge on the main principles of programming in the Data science context
- Develop ability to handle and visualise data
- Apply computational thinking in various applications domains

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation
- Getting Started with pandas

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation
- Getting Started with pandas
- Data Loading, Storage, and File Formats
  - Relational and non-relational Databases
  - Data Warehouse
  - Data Wrangling: Join, Combine, and Reshape

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation
- Getting Started with pandas
- Data Loading, Storage, and File Formats
  - Relational and non-relational Databases
  - Data Warehouse
  - Data Wrangling: Join, Combine, and Reshape
- Data Cleaning and Preparation
  - Formatting, Normalizing, and Binning Data
  - Missing Values

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation
- Getting Started with pandas
- Data Loading, Storage, and File Formats
  - Relational and non-relational Databases
  - Data Warehouse
  - Data Wrangling: Join, Combine, and Reshape
- Data Cleaning and Preparation
  - Formatting, Normalizing, and Binning Data
  - Missing Values
- Data Analysis
  - Data Distribution
  - Data Pipeline
  - Data analysis with scipy and numpy

Course outline

- Python Basics
  - Basic Python Data Types
  - Lists and Tuples
  - Dictionaries
  - Control Flow
  - Conditions
  - Functions and Loops
  - Random Sample Generation
- NumPy Basics: Arrays and Vectorized Computation
- Getting Started with pandas
- Data Loading, Storage, and File Formats
  - Relational and non-relational Databases
  - Data Warehouse
  - Data Wrangling: Join, Combine, and Reshape
- Data Cleaning and Preparation
  - Formatting, Normalizing, and Binning Data
  - Missing Values
- Data Analysis
  - Data Distribution
  - Data Pipeline
  - Data analysis with scipy and numpy
- Data Visualization
  - Exploratory Data Analysis
  - Visualizing Data with Matplotlib, Seaborn, and plotly

Grading Policy

- Homework (30%)
- Mini-Exams (10%)
- Final (60%).

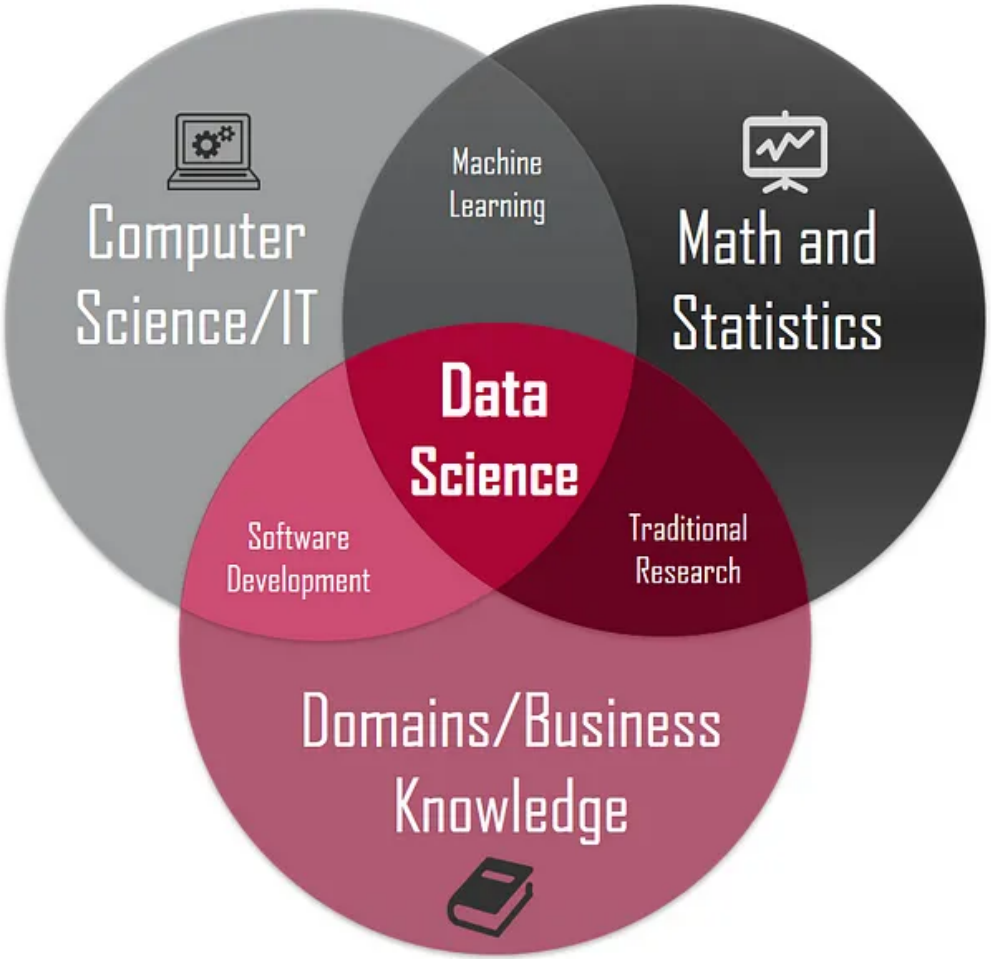
!

Enjoy the course :)

Introduction

What is Data Science?

Data science is an interconnected field that involves the use of **statistical and computational methods** to extract **insightful information and knowledge from data**.



Applications of Data Science

Data science is used in every domain.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.
- **Advertising**: Data science helps to find correct user to show a particular banner or advertisement.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.
- **Advertising**: Data science helps to find correct user to show a particular banner or advertisement.
- **Business Intelligence**: Data Science is widely used in business intelligence to help companies make data-driven decisions.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.
- **Advertising**: Data science helps to find correct user to show a particular banner or advertisement.
- **Business Intelligence**: Data Science is widely used in business intelligence to help companies make data-driven decisions.
- **Fraud Detectio**n: Data Science is used extensively in the finance industry to detect fraudulent activities.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.
- **Advertising**: Data science helps to find correct user to show a particular banner or advertisement.
- **Business Intelligence**: Data Science is widely used in business intelligence to help companies make data-driven decisions.
- **Fraud Detectio**n: Data Science is used extensively in the finance industry to detect fraudulent activities.
- **Natural Language Processing**: Natural Language Processing (NLP) is a branch of Data Science that involves teaching computers to understand human language.

Applications of Data Science

Data science is used in every domain.

- **Healthcare**: healthcare industries uses the data science to make instruments to detect and cure disease.
- **Image Recognition**: The popular application is identifying pattern in images and finds objects in image.
- **Advertising**: Data science helps to find correct user to show a particular banner or advertisement.
- **Business Intelligence**: Data Science is widely used in business intelligence to help companies make data-driven decisions.
- **Fraud Detectio**n: Data Science is used extensively in the finance industry to detect fraudulent activities.
- **Natural Language Processing**: Natural Language Processing (NLP) is a branch of Data Science that involves teaching computers to understand human language.
- **Recommendation Systems**: Data Science is used in recommendation systems to suggest products,

What Kinds of Data?

When I say “data,” what am I referring to exactly?

What Kinds of Data?

When I say “data,” what am I referring to exactly?

- The primary focus is on **structured data**, a deliberately vague term that encompasses many different common forms of data, such as:

What Kinds of Data?

When I say "data," what am I referring to exactly?

- The primary focus is on **structured data**, a deliberately vague term that encompasses many different common forms of data, such as:
  1. **Tabular** or spreadsheet-like data in which each column may be a different type (string, numeric, date, or otherwise). This includes most kinds of data commonly stored in relational databases or tab- or comma-delimited text files.
  2. **Multidimensional** arrays (matrices).
  3. **Multiple tables** of data interrelated by key columns (what would be primary or foreign keys for a SQL user).
  4. Evenly or unevenly spaced **time series**.

What Kinds of Data?

When I say "data," what am I referring to exactly?

- The primary focus is on **structured data**, a deliberately vague term that encompasses many different common forms of data, such as:
  1. **Tabular** or spreadsheet-like data in which each column may be a different type (string, numeric, date, or otherwise). This includes most kinds of data commonly stored in relational databases or tab- or comma-delimited text files.
  2. **Multidimensional** arrays (matrices).
  3. **Multiple tables** of data interrelated by key columns (what would be primary or foreign keys for a SQL user).
  4. Evenly or unevenly spaced **time series**.

This is by no means a complete list. Even though it may not always be obvious, **a large percentage of datasets can be transformed into a structured form** that is more suitable for analysis and modeling.

Why Python for Data Analysis?

Among interpreted languages, for various historical and cultural reasons, Python has developed a **large and active scientific computing and data analysis community**.

Why Python for Data Analysis?

Among interpreted languages, for various historical and cultural reasons, Python has developed a **large and active scientific computing and data analysis community**.

In recent years, Python's improved **open source libraries (such as pandas and scikit-learn)** have made it a popular choice for data analysis tasks.

Why Python for Data Analysis?

Among interpreted languages, for various historical and cultural reasons, Python has developed a **large and active scientific computing and data analysis community**.

In recent years, Python's improved **open source libraries (such as pandas and scikit-learn)** have made it a popular choice for data analysis tasks.

What people are increasingly finding is that Python is a suitable language **not only for doing research and prototyping but also for building the production systems**.

Essential Python Libraries

Essential Python Libraries

- numpy
- pandas
- matplotlib
- scipy
- scikit-learn

NumPy

NumPy, short for Numerical Python, has long been a cornerstone of **numerical computing in Python**. It provides the data structures, algorithms, and library glue needed for most scientific applications involving numerical data in Python.

NumPy

NumPy, short for Numerical Python, has long been a cornerstone of **numerical computing in Python**. It provides the data structures, algorithms, and library glue needed for most scientific applications involving numerical data in Python.

NumPy contains, among other things:

- A fast and efficient **multidimensional array object ndarray**
- Functions for performing element-wise **computations with arrays or mathematical operations between arrays**
- Tools for **reading and writing array-based datasets** to disk
- **Linear algebra** operations, Fourier transform, and **random number generation**

Pandas

Pandas provides high-level data structures and functions designed to make **working with structured or tabular data** intuitive and flexible.

Pandas

Pandas provides high-level data structures and functions designed to make **working with structured or tabular data** intuitive and flexible.

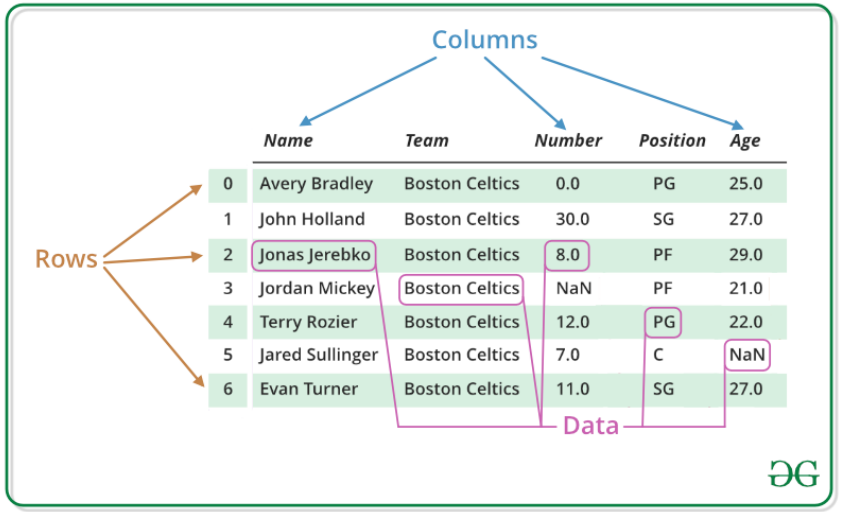
The primary objects in pandas that will be used in this course are the **DataFrame**, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object.

Pandas

Pandas provides high-level data structures and functions designed to make **working with structured or tabular data** intuitive and flexible.

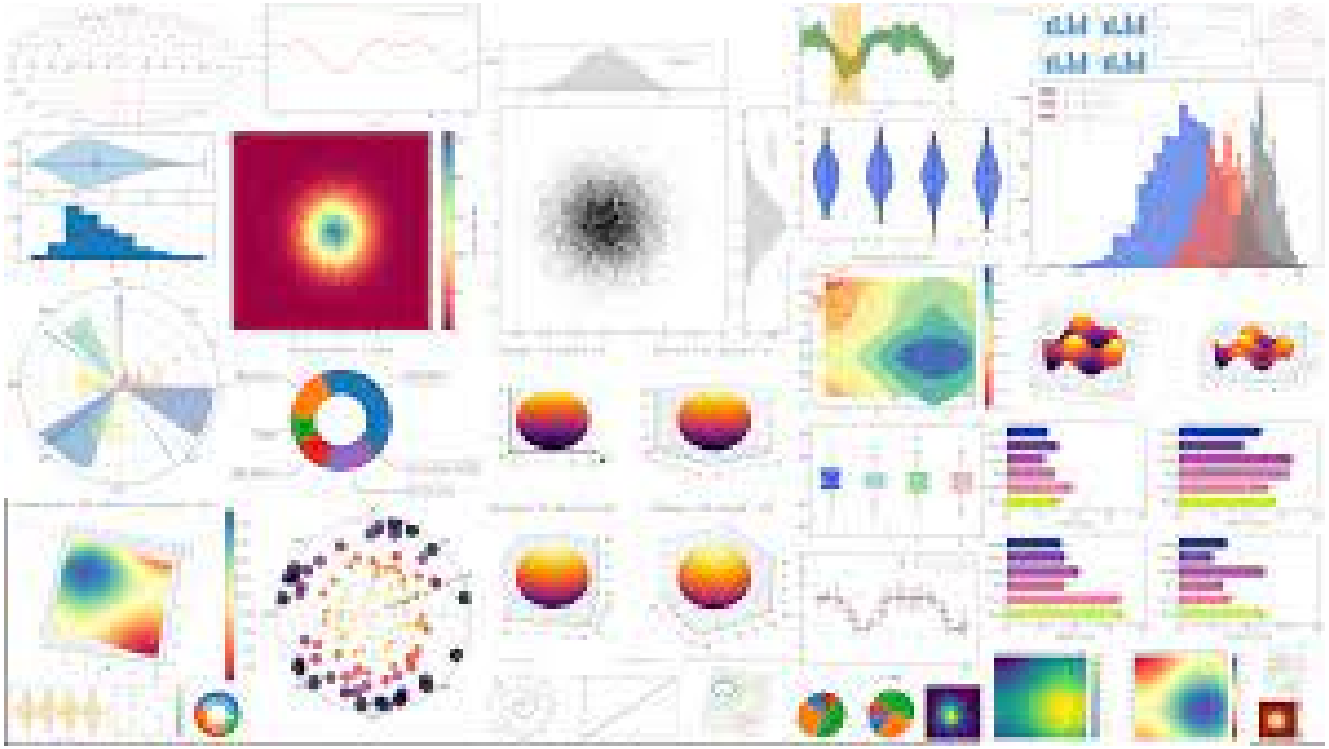
The primary objects in pandas that will be used in this course are the **DataFrame**, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object.

- pandas blends the array-computing ideas of **NumPy** with the kinds of data manipulation capabilities found in spreadsheets and **relational databases** (such as SQL).
- Since data manipulation, preparation, and cleaning are such important skills in data analysis, **pandas is one of the primary focuses of this course**.



Matplotlib

Matplotlib is the most popular Python library for producing plots and other two-dimensional data visualizations.



SciPy

SciPy is a collection of packages addressing a number of foundational problems in scientific computing

- scipy.integrate
  - Numerical integration routines and differential equation solvers
- scipy.linalg
  - Linear algebra routines and matrix decompositions extending beyond those provided in numpy.linalg
- scipy.optimize
  - Function optimizers (minimizers) and root finding algorithms
- scipy.signal
  - Signal processing tools

Scikit-learn

Scikit-learn is the premier general-purpose machine learning toolkit for Python programmers.

Scikit-learn

Scikit-learn is the premier general-purpose machine learning toolkit for Python programmers.

It includes submodules for such models as:

- Classification: SVM, nearest neighbors, random forest, logistic regression, etc.
- Regression: Lasso, ridge regression, etc.
- Clustering: k-means, spectral clustering, etc.
- Dimensionality reduction: PCA, feature selection, matrix factorization, etc.
- Model selection: Grid search, cross-validation, metrics
- Preprocessing: Feature extraction, normalization

Python

The Python Interpreter

The Python interpreter runs a program by executing one statement at a time.

```
$ python
Python 3.10.4 | packaged by conda-forge | (main, Mar 24 2022, 17:38:57)
[GCC 10.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 5
>>> print(a)
5
```

The >>> you see is the prompt after which you'll type code expressions.

To exit the Python interpreter, you can either type exit()

Running Program from Python File

Running Python programs is as simple as calling python with a .py file as its first argument. Suppose we had created hello_world.py with these contents:

hello_world.py

```
print("Hello world")
```

You can run it by executing the following command (the hello_world.py file must be in your current working terminal directory):

```
$ python hello_world.py
Hello world
```

Running Program from Python File

Running Python programs is as simple as calling python with a .py file as its first argument. Suppose we had created hello_world.py with these contents:

hello_world.py

```
print("Hello world")
```

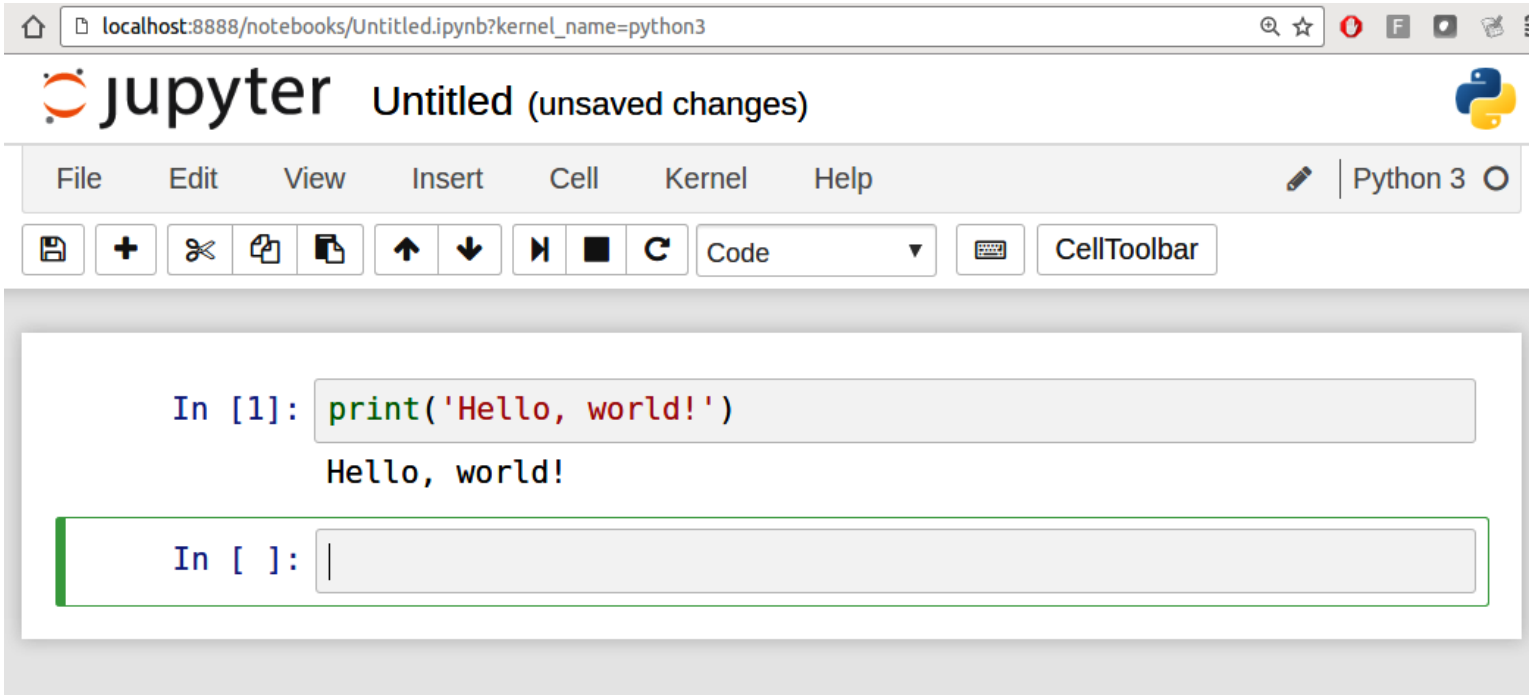
You can run it by executing the following command (the hello_world.py file must be in your current working terminal directory):

```
$ python hello_world.py
Hello world
```

While some Python programmers execute all of their Python code in this way, those doing **data analysis or scientific computing make use of IPython**, an enhanced Python interpreter, or **Jupyter notebooks, web-based code notebooks** originally created within the IPython project.

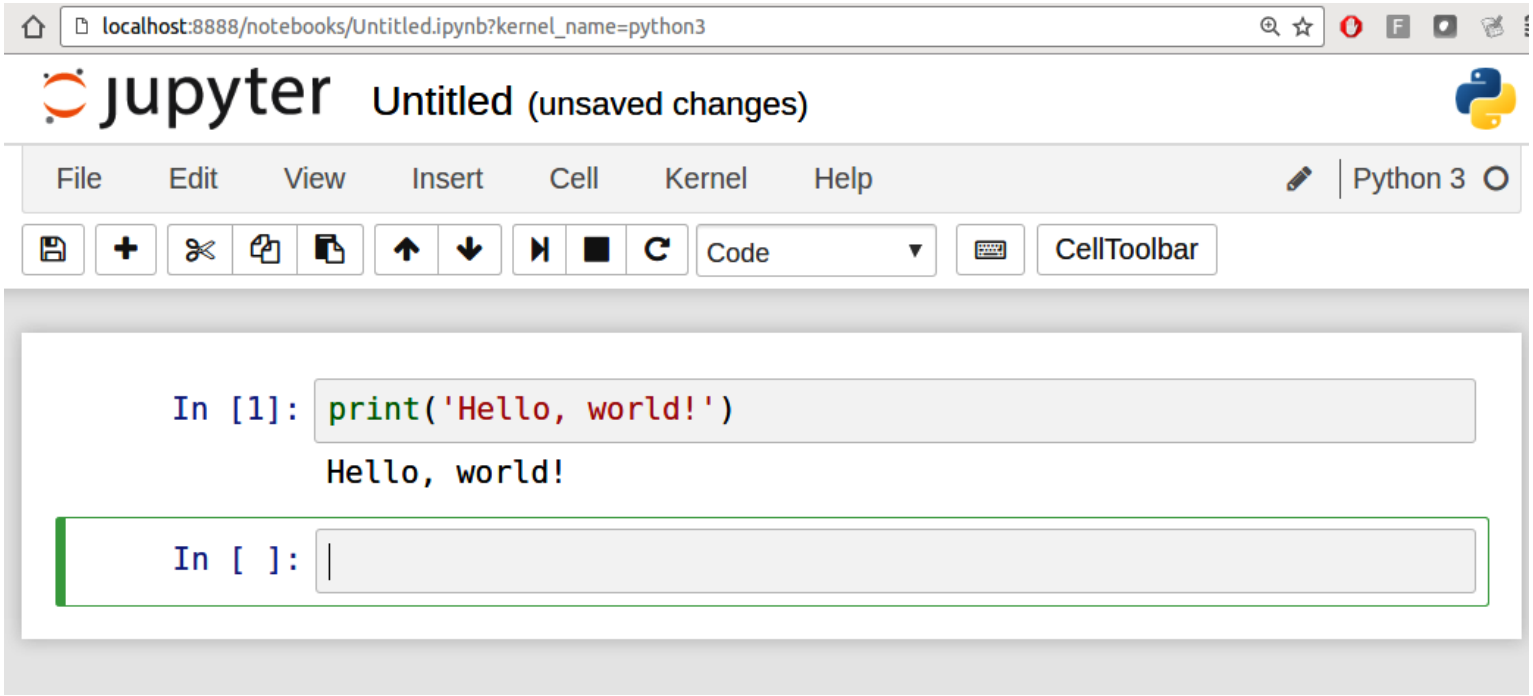
Running the Jupyter Notebook

One of the major components of the Jupyter project is the notebook, a type of interactive document for code, text (including Markdown), data visualizations, and other output. The Python Jupyter kernel uses the IPython system for its underlying behavior.



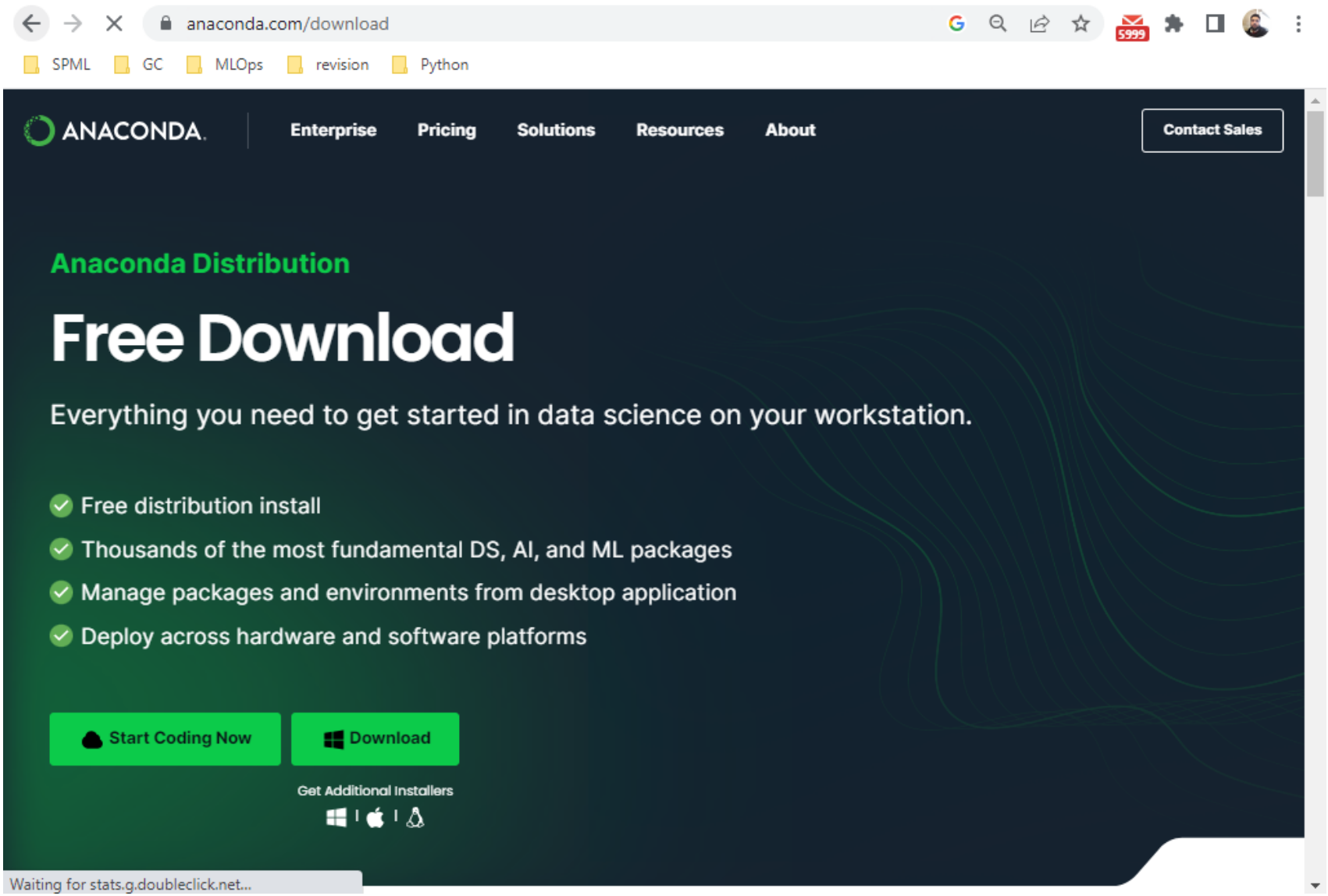
Running the Jupyter Notebook

One of the major components of the Jupyter project is the notebook, a type of interactive document for code, text (including Markdown), data visualizations, and other output. The Python Jupyter kernel uses the IPython system for its underlying behavior.

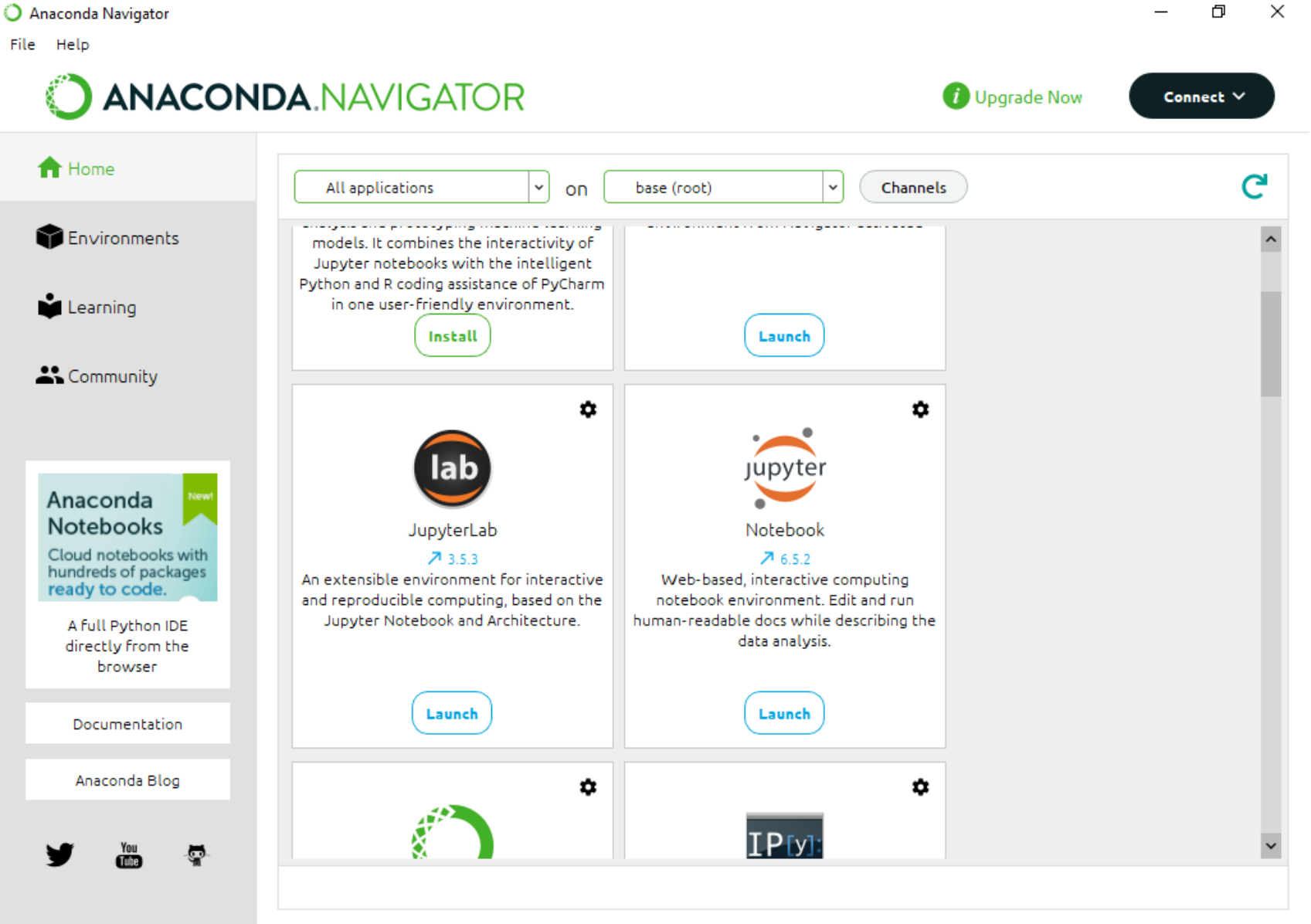


When you **save the notebook** (see ”Save and Checkpoint” under the notebook File menu), it creates a file with the extension **.ipynb**.

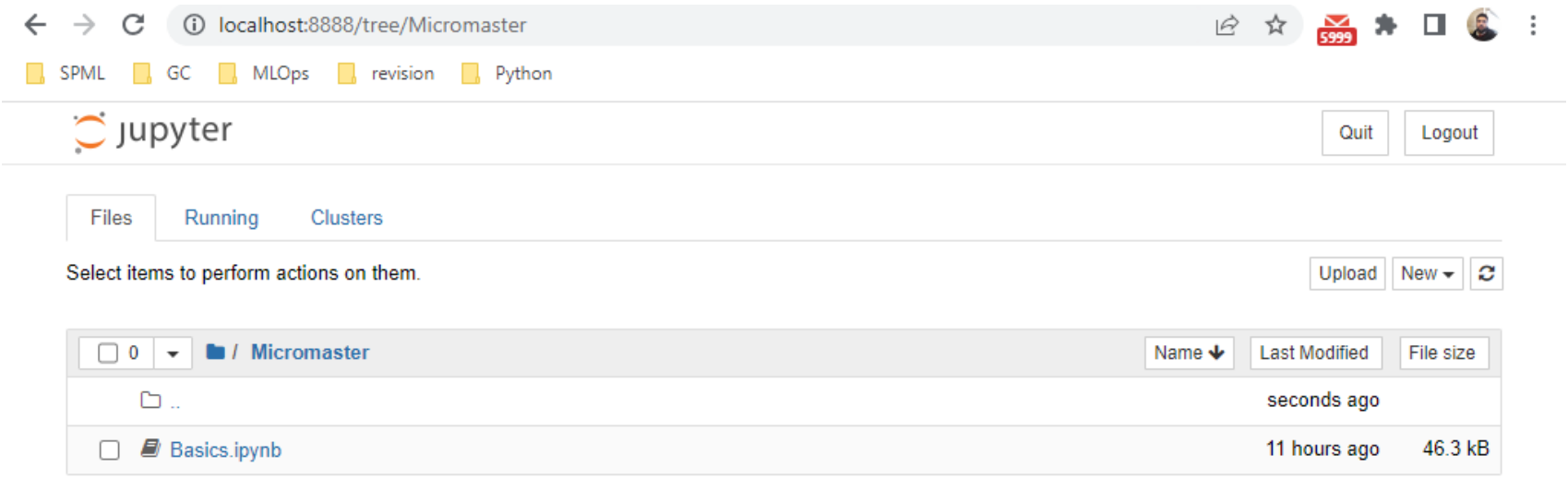
Installation and Setup - ANACONDA



ANACONDA Navigator



Jupyter Notebook



Jupyter Notebook

localhost:8888/notebooks/Micromaster/Basics.ipynb

SPML GC MLOps revision Python

jupyter Basics Last Checkpoint: 11 hours ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Markdown

none	NoneType	Null Object	represents no value	None

Numeric data types

There are three distinct numeric types: `integers`, `floating point numbers`, and `complex numbers` (not covered here). We can determine the type of an object in Python using `type()`. We can print the value of the object using `print()`.

In [ ]:
```
x = 42
```

In [ ]:
```
type(x)
```

In [ ]:
```
print(x)
```

In Jupyter/IPython (an interactive version of Python), the last line of a cell will automatically be printed to screen so we don't actually need to explicitly call `print()`.

In [ ]:
```
x  # Anything after the pound/hash symbol is a comment and will not be run
```

In [ ]:
```
pi = 3.14159
pi
```

In [ ]:
```
type(pi)
```