UECS3213 / UECS3453 Data Mining

SESSION: January, 2019

Lab 1: Introduction to Data Science Tools (Python, R)

Objectives

The objective of this lab is to introduce you to two popular tools for data science:

a) Python (programming language) and Jupyter Notebook or Spyder (Integrated Development Environment)
b) R (programming language) and RStudio (Integrated Development Environment)

Note: We will learn to use the Python data science libraries such as NumPy, Matplotlib, Pandas, scikit-learn etc. throughout the course, not R. Hence, R will only be introduced in this lab for your knowledge and you have to self-learn it if you want to.

Introduction

Two of the most popular programming tools for data science work are Python and R at the moment.

Part 1: R and RStudio

R is a language and environment for statistical computing and graphics. RStudio is a free, open source IDE (integrated development environment) for R. R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, …) and graphical techniques, and is highly extensible.

RStudio makes R easier to use. It includes a code editor, debugging & visualization tools.

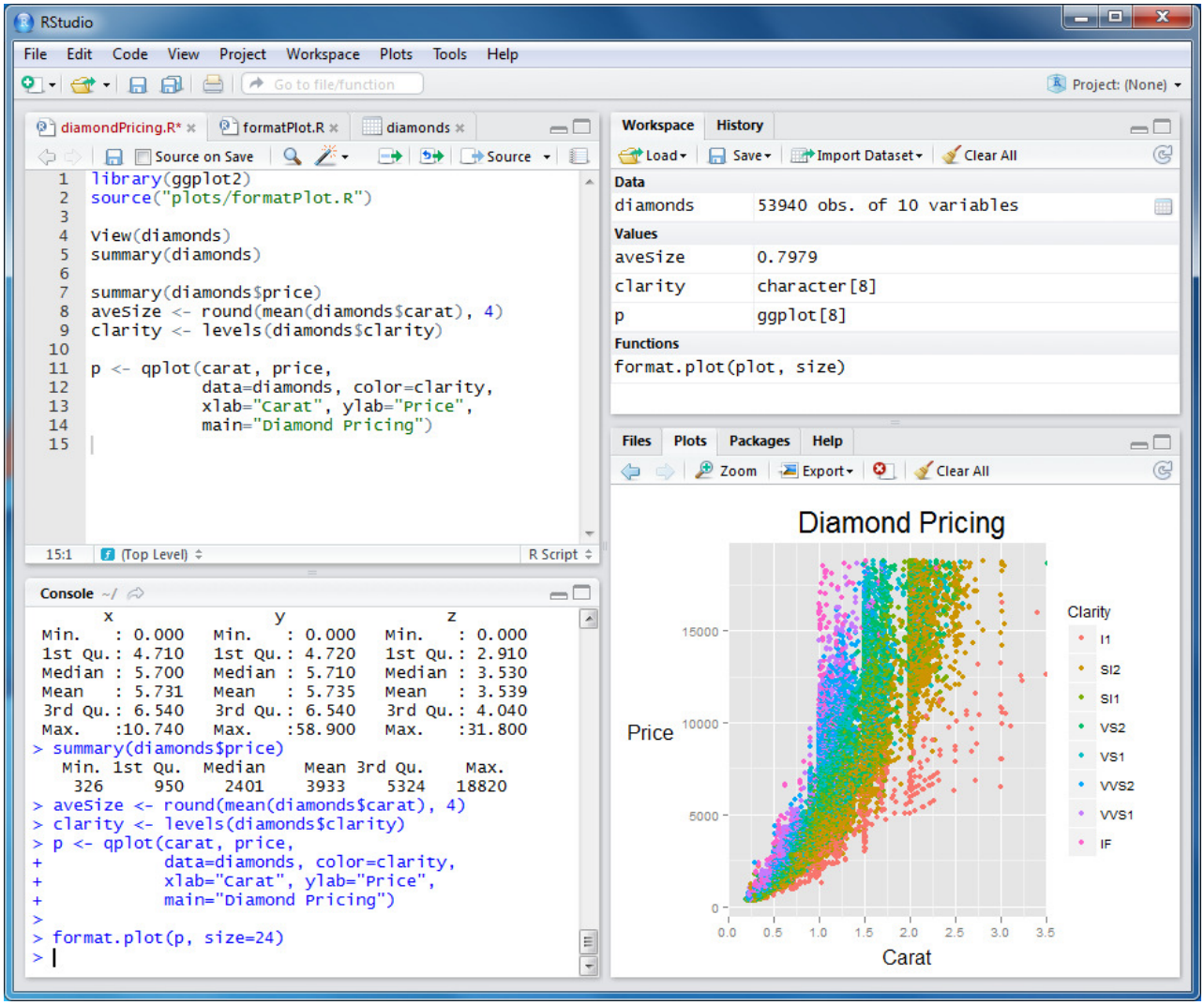

Figure 1: RStudio

- Download and Installation of R: https://cloud.r-project.org/
- Download and Installation of RStudio: https://www.rstudio.com/products/rstudio/download/

References

- https://www.r-project.org/about.html
- https://www.rstudio.com/

Part 2: Python and Jupyter Notebook / Spyder

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. The open source Anaconda Distribution is the fastest and easiest way to do Python and R data science and machine learning (https://www.anaconda.com).

Project Jupyter exists to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages. The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more. The Notebook has support for over 40 programming languages, including Python.

Apart from Jupyter Notebook, we may also use Spyder notebook as the IDE to develop Python program.

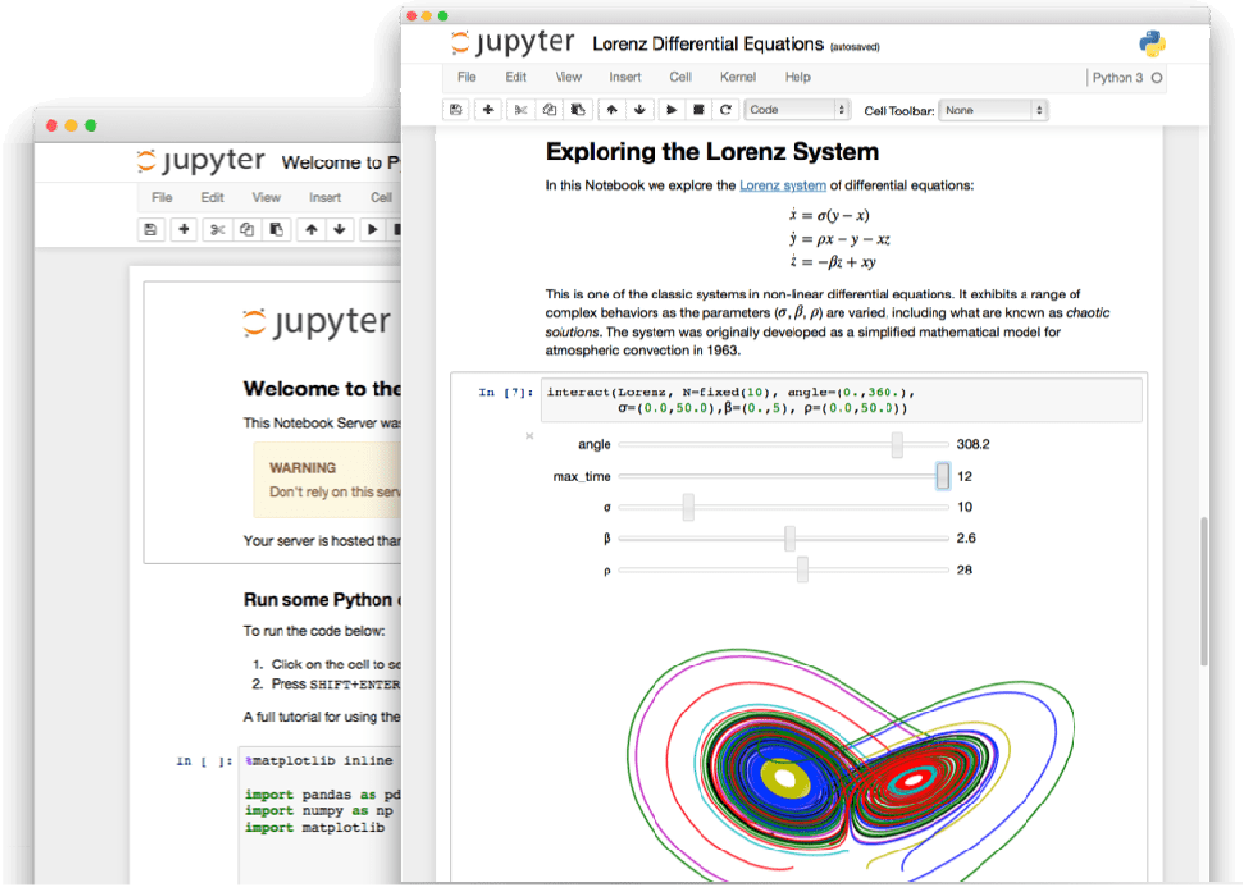

Figure 2: Jupyter Notebook

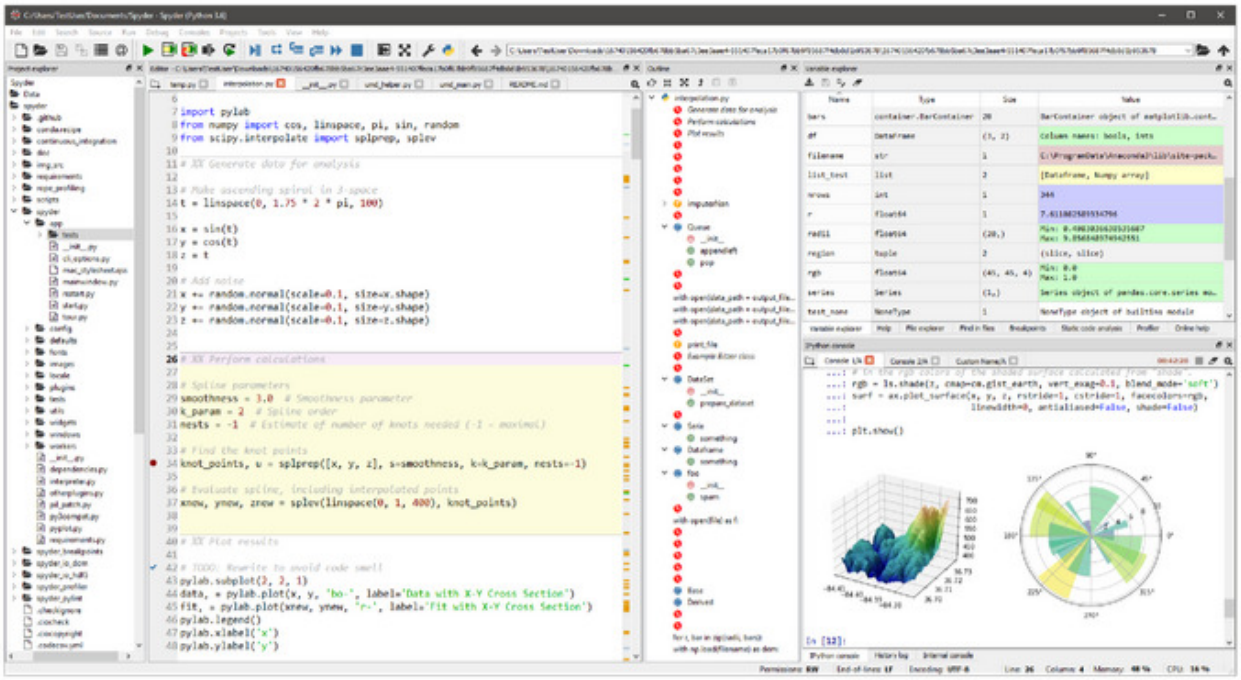

Figure 3: Spyder Notebook

Additional reading: Why Python and Jupyter Notebooks? https://unidata.github.io/online-python-training/introduction.html

Details about Python programming and its data science libraries will be described in Lab 2.

Instruction

1. Download and Installation of Python (Anaconda Distribution):
   - https://www.anaconda.com/download/
   - https://conda.io/docs/download.html
   - https://conda.io/docs/user-guide/install/windows.html

2. Check the latest packages included in the Anaconda distribution: https://docs.anaconda.com/anaconda/packages/py3.7_win-64/. Pay special attention to packages such as numpy, scipy, pandas, scikit-learn and notebook.

3. Download and Installation of Jupyter Notebook: https://jupyter.org/install.html (Skip this step if you have installed Python Anaconda distribution, Jupyter Notebook is already included), or

4. Download and Installation of Spyder Notebook: https://www.spyder-ide.org/l (Skip this step if you have installed Python Anaconda distribution, Spyder Notebook is already included)

5. Follow the Jupyter Notebook Tutorial at https://www.datacamp.com/community/tutorials/tutorial-jupyter-notebook or https://www.kaggle.com/slby9999/learn-python-challenge-day-1-exercises/edit or Spyder Notebook Tutorial at https://anaconda.org/conda-forge/spyder-notebook

References:

- https://jupyter.org/
- https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html
- https://medium.com/codingthesmartway-com-blog/getting-started-with-jupyter-notebook-for-python-4e7082bd5d46

Part 3: Python vs R

It's HARD to know whether to use Python or R for data analysis. And this is especially true if you are a newbie data analyst looking for the right language to start with. While Python is often praised for being a general-purpose language (more suited for programmers) with an easy-to-understand syntax, R's functionality is developed with statisticians in mind, thereby giving it field-specific advantages such as great features for data visualization.

Check the infographic "Data Science Wars: R vs Python": http://res.cloudinary.com/dyd911kmh/image/upload/f_auto,q_auto:best/v1523009719/main-qimg-9dcf536c501455f073dfbc4e09798a51_vpijr0.png

Additional Reading:

- https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis
- https://medium.com/@data_driven/python-vs-r-for-data-science-and-the-winner-is-3ebb1a968197

The End