

Introductory Data Analysis Course: Conda Environment Setup Instructions

In preparation for our upcoming exercise for the Introductory Data Analysis course, please ensure that your laptops are fully set up to work with Python and that you bring the laptops to our sessions.

Follow these steps according to your operating system to set up Python, Conda, and essential libraries for this module.

1. Installing Miniconda (alternatively use Anaconda):

Anaconda is a package manager, environment manager, and distribution of Python, with many data science libraries included. It simplifies the installation and management of Python packages and their dependencies. To download and install Conda:

1. go to [[Miniconda installation page](#), [Anaconda Installation page](#)] and download the installer for your operating system.
2. Install Miniconda/Anaconda: Run the installer and follow the instructions. Follow the prompts to complete the installation.
3. During installation, you may be prompted to add Conda to PATH. This allows you to call `conda` from any terminal window.
4. You may need to restart your terminal
5. Open

2. Installing Required Python Libraries:

open the “Anaconda Powershell Prompt” (on windows) or terminal (Linux or Mac OS), then create a virtual environment (recommended for isolating dependencies):

```
```bash:
conda create -n IDA python=3.9
```
```

Activate the created environment:

```
```bash:
conda activate IDA
```
```

Now you are working in your IDA environment. Next, install the packages:

```
```bash
conda install numpy pandas matplotlib seaborn scikit-learn jupyter scipy
statsmodels
```
```

3. Testing the Installation

After setting up everything, verify your installation by opening a Notebook. To do so:

1. Open the terminal or Anaconda Prompt. Make sure you have activated conda environment IDA by typing.

```
```bash
conda activate IDA
```
```

2. Create a new directory for the exercise and Move to to it.

```
```bash
Mkdir ~/Desktop/IDA_exercises
cd ~/Desktop/IDA_exercises
```
```

3. Start Jupyter by typing:

```
```bash
jupyter notebook .
```
```

4. In the notebook interface, try importing the libraries to confirm installation:

```
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
from scipy import stats
import statsmodels.api as sm
```
```

4. Explanation of Key Components:

- NumPy: Provides support for large, multi-dimensional arrays and matrices, along with mathematical functions.
- Pandas: Used for data manipulation and analysis, with data structures and operations for handling datasets.
- Matplotlib: A plotting library that helps visualize data with charts and graphs.
- Seaborn: Based on Matplotlib, used for statistical data visualization and aesthetic data representation.

- SciKit-Learn: A comprehensive machine learning library with tools for model training, evaluation, and data preprocessing.
- Jupyter: An interactive notebook that allows running code, visualizations, and documentation in one place.
- SciPy: Core statistical functions such as Hypothesis testing (t-test, ANOVA, chi-square, etc.), Probability distributions, Correlations and density functions...etc.
- Statsmodels: Deep statistical modeling (more formal than SciPy)

5. Additional Tips

- Stay Updated: Periodically update Conda and libraries using ``conda update --all`` to ensure you have the latest features and security patches.
- Resource Management: Close Jupyter Notebooks when not in use to free up memory, especially if working with large datasets (everything you load in the notebook will end up in the memory)
- Practice: Familiarize yourself with each tool by practicing small projects or tutorials to gain confidence in using them. The internet is full of resources that you could use to learn python and data science.