Experiment No. 1

Aim: Introduction to Data analytics libraries in Python and R.

Objective- Understand the use of Python and R, To effectively use libraries for data science.

Description:

Why Choose Python?

Python is a general-purpose, open-source programming language used in various software domains, including data science, web development, and gaming.

Launched in 1991, Python is one of the most popular programming languages in the world, occupying the top position in several programming language popularity indices, such as the TIOBE Index and the PYPL Index.

One of the reasons for the worldwide popularity of Python is its community of users. Python is backed by a vast community of users and developers who ensure the smooth growth and improvement of the language, as well as the continuous release of new libraries designed for all kinds of purposes.

Python is an easy language to read and write due to its high similarity with human language. In fact, high readability and interpretability are at the heart of the design of Python. For these reasons, Python is often cited as a go-to programming language for newcomers with no coding experience.

Over time, Python has been gaining popularity in the field of data science thanks to its simplicity and the endless possibilities provided by the hundreds of specialized libraries and packages that support any kind of data science task, such as data visualization, machine learning, and deep learning.

Why Choose R?

R is an open-source programming language specifically created for statistical computing and graphics.

Since its first launch in 1992, R has been widely adopted in scientific research and academia. Today, it remains one of the most popular analytics tools used in both traditional data analytics and the rapidly-evolving field of business analytics. It ranks 11th and 7th position in the **TIOBE** Index and the **PYPL** Index, respectively.

Designed with statisticians in mind, with R, you can use complex functions within a few lines of code. All kinds of statistical tests and models are readily available and easily used, such as linear modeling, non-linear modeling, classifications, and clustering.

The extensive possibilities R offers are mostly due to its huge community. It has developed one of the richest collections of data-science-related packages. All of them are available via the Comprehensive R Archive Network (**CRAN**).

Another feature that makes R particularly remarkable is the power to generate quality reports with support for data visualization and its available frameworks to create interactive web applications. In this sense, R is widely considered the best tool for making beautiful graphs and visualizations

R vs Python: Key Differences

Purpose

While Python and R were created with different purposes –Python as a general-purpose programming language and R for statistical analysis–nowadays, both are suitable for any data science task. However, Python is considered a more versatile programming language than R, as it's also extremely popular in other software domains, such as software development, web development, and gaming.

Type of Users

As a general-purpose programming language, Python is the standard go-to choice for software developers breaking into data science. Plus, Python's focus on productivity makes it a more suitable tool to build complex applications.

By contrast, R is widely used in academia and certain sectors, such as finance and pharmaceuticals. It is the perfect language for statisticians and researchers with limited programming skills.

Learning curve

Python's intuitive syntax is considered one of the closest programming languages to English. This makes it a very good language for new programmers, with a smooth and linear learning curve. Although R is designed to run basic data analysis easily and within minutes, things get harder with complex tasks, and it takes more time for R users to master the language.

Overall, Python is considered a good language for beginner programmers. R is easier to learn when you start out, but the intricacies of advanced functionalities make it more difficult to develop expertise.

Popularity

Although new programming languages, like **Julia**, are recently gaining momentum in data science, Python and R remain the absolute kings in the discipline.

However, in terms of popularity –always a very slippery concept– the differences are striking. Python has consistently outranked R, especially in recent years. Python ranks first in several programming language popularity indexes. This is due to the widespread use of Python in multiple software domains, including data science. By contrast, R is mostly employed in data science, academia, and certain sectors.

Common Libraries

Both Python and R have robust and extensive ecosystems of packages and libraries specifically designed for data science. Most packages in Python are hosted in the Python Package Index (**PyPi**), whereas **R packages** are normally stored in the Comprehensive R Archive Network (**CRAN**).

Below you can find a list of some of the most popular data science libraries in R and Python.

R packages:

- **dplyr**: It is a data manipulation library for R.
- **tidyr**: a great package that will help you get your data clean and tidy.
- **ggplot2**: the perfect library for visualizing data.
- **Shiny**: It is the ideal tool for creating interactive web apps directly from R.
- **Caret**: one of the most important libraries for machine learning in R.

Python packages:

- **NumPy**: provides a large collection of functions for scientific computing.
- **Pandas**: perfect for data manipulation.
- **Matplotlib**: the standard library for data visualization.
- **Scikit-learn**: is a library in Python that provides many machine learning algorithms.
- **TensorFlow**: a widely used framework for deep learning.

Common IDEs

An IDE, or Integrated Development Environment, enables programmers to consolidate the different aspects of writing a computer program. They are powerful interfaces with integrated capabilities that allow developers to write code more efficiently.

In Python, the most popular IDEs in data science are Jupyter Notebooks and its modern version, JupyterLab, as well as Spyder.

As for R, the most commonly used IDE is RStudio. Its interface is organized so that the user can view graphs, data tables, R code, and output all at the same time.

Python vs R: A Comparison

	R	**Python**
Purpose	Very popular in academia and research, finance and data science	Well-suited for many programming domains, including data science, web development, software development, and gaming
First Release	1993	1991
Type of Language	General-purpose programming language	General-purpose programming language
Open Source?	Yes	Yes

Ecosystem	Nearly 19,000 packages available in the Comprehensive R Archive Network (**CRAN**)	+300,000 available packages in the Python Package Index (**PyPi**)
Ease of Learning	R is easier to learn when you start out, but gets more difficult when using advanced functionalities.	Python is a beginner-friendly language with English-like syntax.
IDE	RStudio. Its interface is organized so that the user can view graphs, data tables, R code, and output all at the same time.	Jupyter Notebooks and its modern version, JupyterLab, and Spyder.
Advantages	· Widely considered the best tool for making beautiful graphs and visualizations. · Has many functionalities for data analysis. · Great for statistical analysis.	· General-purpose programming languages are useful beyond just data analysis. · Has gained popularity for its code readability, speed, and many functionalities. . · Has high ease of deployment and reproducibility.

Disadvantages	· More difficult to learn for people with no software development background. · Limited user community compared to Python · R is considered a computationally slower language compared to Python, especially if the code is written poorly. · Finding the right library for your task can be tricky, given the high number of packages available in CRAN	· Weak performance with huge amounts of data · Poor memory efficiency · Python does not have as many libraries for data science as R. · Python requires rigorous testing as errors show up in runtime. · Visualizations are more convoluted in Python than in R, and results are not as eye pleasing or informative.
Trends	11th in TIOBE and 7th in PYPL (December 2022)	1th in TIOBE and 1th in PYPL (December 2022)

Attach Libraries you searched in Lab session-

Python:

1. NumPy:

- NumPy is a fundamental library for numerical computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with mathematical functions to operate on these arrays.

2. Pandas:

- Pandas is a data manipulation and analysis library. It provides data structures like DataFrame for efficient data manipulation with built-in methods for reshaping, merging, and aggregating data.

3. Matplotlib:

- Matplotlib is a popular plotting library in Python. It allows the creation of a wide variety of static, animated, and interactive plots, charts, and graphs.

4. Scikit-learn:

- Scikit-learn is a machine learning library for Python. It provides simple and efficient tools for data mining and data analysis, including various algorithms for classification, regression, clustering, and more.

5. TensorFlow:

- TensorFlow is an open-source machine learning library developed by Google. It is widely used for building and training deep learning models, especially neural networks.

6. Keras:

- Keras is a high-level neural networks API that runs on top of TensorFlow. It simplifies the process of building and experimenting with deep learning models, making it accessible to a broader audience.

7. Beautiful Soup:

- Beautiful Soup is a web scraping library that simplifies the extraction of data from HTML and XML files. It provides Pythonic idioms for navigating, searching, and modifying a parse tree.

8. Requests:

- Requests is a simple and elegant HTTP library for Python. It allows users to send HTTP requests and handle responses easily, making it a preferred choice for interacting with web APIs.

9. NLTK (Natural Language Toolkit):

- NLTK is a powerful library for natural language processing. It provides tools and resources for tasks such as tokenization, stemming, tagging, parsing, and more.

10. OpenCV:

- OpenCV is a computer vision library that offers a wide range of tools for image and video processing. It is widely used in computer vision applications and projects.

R Libraries:

1. dplyr:

- dplyr is a popular data manipulation library in R. It provides a set of functions for efficiently manipulating data frames, including filtering, sorting, summarizing, and joining data.

2. ggplot2:

- ggplot2 is a versatile plotting library in R. It follows the grammar of graphics, allowing users to create complex, customizable visualizations with a high level of abstraction.

3. caret:

- caret (Classification And REgression Training) is a comprehensive package for machine learning in R. It provides a consistent interface for various modeling techniques, along with tools for data pre-processing and model evaluation.

4. tidyr:

- tidyr is another useful data manipulation library in R. It helps with data cleaning and reshaping by providing functions for tidying messy datasets.

5. randomForest:

- randomForest is an R package for building random forests, an ensemble learning method. It is widely used for both classification and regression tasks, providing a robust and flexible modeling approach.

6. Shiny:

- Shiny is an R package for building interactive web applications directly from R. It allows users to create web-based dashboards and visualizations with minimal coding effort.

7. CaretEnsemble:

- CaretEnsemble is an extension of the caret package, allowing users to combine and ensemble different models seamlessly. It enhances the modeling capabilities provided by caret.

8. ROCR:

- ROCR is a package for evaluating and visualizing the performance of classification algorithms. It provides tools for creating ROC curves and calculating performance metrics.

9. rvest:

- rvest is an R package designed for web scraping. It simplifies the process of extracting information from HTML pages, making it a valuable tool for collecting data from the web.

10. glmnet:

- glmnet is a package for fitting generalized linear models with regularization. It is particularly useful for tasks such as regression and classification with high-dimensional data.

Conclusion:

In summary, Python's NumPy, Pandas

, Matplotlib, and TensorFlow excel in numerical computing, data manipulation, visualization, and deep learning. R stands out with dplyr, tidyr, ggplot2, caret, and randomForest, emphasizing efficient data manipulation, visualization, and machine learning. Both languages offer unique strengths, with Python being versatile and widely used, while R provides specialized tools for statistical analysis and visualization. The choice depends on project requirements and individual preferences. Together, they form a comprehensive toolkit for diverse data science tasks.