Data Science Tools and Ecosystem

In this notebook, Data Science Tools and Ecosystem are summarized.

Some of the popular languages that Data Scientists use are:

1. Python
2. R
3. SQL

🧰 Common Data Science Libraries

1. **Pandas** – For data manipulation and analysis (tables, CSVs, cleaning, merging). → 🐍 *Python*
2. **NumPy** – For fast mathematical and array operations. → 🐍 *Python*
3. **Matplotlib** – For creating line charts, bar charts, and visualizations. → 🐍 *Python*
4. **Seaborn** – For beautiful and easy-to-read statistical plots (built on Matplotlib). → 🐍 *Python*
5. **Scikit-learn** – For machine learning: regression, classification, clustering, etc. → 🐍 *Python*
6. **TensorFlow** – For deep learning and neural networks (developed by Google). → 🐍 *Python*
7. **PyTorch** – For deep learning and AI research (developed by Meta). → 🐍 *Python*
8. **Keras** – For building and training neural networks easily (runs on TensorFlow). → 🐍 *Python*
9. **Statsmodels** – For statistical modeling, regression analysis, and hypothesis testing. → 🐍 *Python*
10. **SciPy** – For scientific computing (integration, optimization, and linear algebra). → 🐍 *Python*
11. **PySpark** – For distributed computing and big data processing using Apache Spark. → 🐍 *Python*
12. **SparkR** – For working with Apache Spark from R. → 🐍 *R*
13. **XGBoost** – For high-performance gradient boosting in ML models. → 🐍 *Python* / 🐍 *R*
14. **LightGBM** – For fast and scalable gradient boosting (from Microsoft). → 🐍 *Python* / 🐍 *R*
15. **Plotly** – For interactive visualizations and dashboards. → 🐍 *Python* / 🐍 *R*
16. **Dash** – For building web-based interactive dashboards (built on Plotly + Flask). → 🐍 *Python*

17. **ggplot2** – For elegant and layered visualizations using the Grammar of Graphics. → 🐍 *R*
18. **dplyr** – For data manipulation and transformation (similar to Pandas). → 🐍 *R*
19. **tidyr** – For data cleaning and reshaping in tidy format. → 🐍 *R*
20. **shiny** – For creating interactive web applications for data analysis. → 🐍 *R*
21. **OpenCV** – For computer vision and image processing. → 🐍 *Python*
22. **NLTK** – For natural language processing (tokenization, stemming, text analysis). → 🐍 *Python*
23. **spaCy** – For advanced NLP and named entity recognition. → 🐍 *Python*
24. **BeautifulSoup** – For web scraping and extracting data from HTML pages. → 🐍 *Python*
25. **Requests** – For sending HTTP requests and working with APIs. → 🐍 *Python*
26. **Joblib** – For saving/loading machine learning models and parallel processing. → 🐍 *Python*
27. **Caret** – For training and evaluating machine learning models. → 🐍 *R*
28. **mlr3** – For modern machine learning workflows in R. → 🐍 *R*
29. **data.table** – For fast data manipulation and aggregation (like Pandas). → 🐍 *R*
30. **Altair** – For declarative and statistical data visualizations. → 🐍 *Python*

🧰 Data Science Tools

Data Science Tools	Description / Use
Jupyter Notebook	Interactive environment for writing and running Python code, visualizations, and documentation.
RStudio	Integrated development environment (IDE) for R, used for statistical analysis and visualization.
Apache Zeppelin	Web-based notebook for interactive data analysis and visualization with multiple languages (Python, Scala, SQL, etc.).
Visual Studio Code	Lightweight open-source code editor with extensions for Python, R, and data science workflows.
Spyder	Scientific Python IDE designed for data analysis, debugging, and visualization.
Google Colab	Cloud-based Jupyter Notebook that runs Python code on Google’s servers (no setup needed).
KNIME Analytics Platform	Visual workflow tool for data mining, ETL (extract, transform, load), and machine learning.
Orange Data Mining	Open-source visual programming tool for data analysis and machine learning.
IBM Watson Studio	Cloud-based collaborative platform for data science, AI model building, and deployment.

Data Science Tools	Description / Use
Apache Spark (PySpark)	Distributed computing framework for large-scale data processing and analytics.

Below are a few examples of evaluating arithmetic expressions in Python.

In [18]:

```
(3*4)+5
# This is a simple arithmetic expression to multiply then add integers
```

Out[18]:

```
17
```

In [19]:

```
# This will convert 200 minutes to hours by dividing by 60
200 / 60
```

Out[19]:

```
3.3333333333333335
```

Objectives:

- List popular languages for Data Science
- Identify common data science libraries and tools
- Demonstrate basic arithmetic operations in Python
- Understand the use of Jupyter Notebooks for data analysis

Author

Gabriel Gamy