Our Guide to Open Source in Data Science

The Open Source Revolution

Over the past few decades, digital technologies have completely transformed our way of life. From how we communicate to the way we conduct business, software has disrupted how value is generated today. This third industrial revolution is where information technology and digital solutions have become widely used to automate production and improve productivity (World Economic Forum).

Arguably, the most significant catalyst to the adoption and development of digital technologies is open source software, which has led to many of the most exciting innovations of the 21st century (zdnet). At a simple level, open-source software is a type of software where the source code is released with flexible licensing so that it can be accessed, used, distributed, and modified by other developers. As such, open source software has introduced a paradigm shift: Organizations can now build secure, high-quality software that gives them more flexibility while hiring and retaining the best talent there is. The advantages of open-source software are many, which is why now more than ever, organizations across all industries are adopting open-source technologies (Red Hat).

Security

While open-source software is often dismissed as being insecure, the truth is that open-source software is one of the most secure on the market. An open community-based approach incentivizes hundreds, if not thousands of developers to monitor for vulnerabilities in a program (Bluespark). This means that vulnerabilities and security flaws are found and fixed much faster than in proprietary software (PCWorld). Moreover, open-source software allows organizations to perform security audits on any software they plan to onboard in their processes, as opposed to “black box” proprietary software (inc).

Quality

According to Red Hat, the leading cause for organizations to adopt open-source software is because it’s considered to be higher quality than proprietary solutions. Its quality is driven by the nature of open source collaboration, where practitioners have input in how the tools they use are designed. This “built for the people by the people” approach encourages better alignment between open source software designers, and their end users. Moreover, this means that teams across the organization can adopt, build, and customize software with the same open-source tools. This helps teams avoid falling into the trap of silos where teams use different sets of proprietary software. Finally, the free and open nature of open-source software means that practitioners have a faster speed to value as they can easily repurpose existing code-bases for their specific needs (Xorlogics).

Cost

One of the most obvious benefits of open source adoption is total reduced costs. Open-source software streamlines costs for organization across the software adoption flow. For starters, organizations can quickly adopt a solution and experiment with it, avoiding time spent on vendor-led proof of concepts and requests for proposals. Moreover, open-source software minimizes the amount of licensing and maintenance costs, as open-source libraries are upgraded for free. Most importantly, open-source software is extensible and customizable, whereas proprietary software locks organizations in with vendors even when it ceases to meet the demands of their use cases (O'Reilly).

Flexibility

One of the key differentiators between proprietary and open-source software is flexibility and customization. Ultimately, proprietary software is controlled and managed by its developers, whereas open source software has much more flexible licensing. This enables organizations to customize software for the workflows and provides them more control over the tools and solutions they develop (inc). Moreover, open-source software is interoperable, meaning that it can work with a variety of data formats, and is designed for cloud and cloud-native technologies. Finally, open source software enables organizations to avoid vendor lock-in and allows them to test and try software before committing to a solution (InfoWorld).

Talent and Skills

Arguably one of the most important aspects of open-source software is how it intersects with talent acquisition and retention. In short, organizations that contribute to and work with open source projects attract and retain better talent. For starters, open source tools and programming languages have become the standard in academia and industry alike, facilitating skill sharing and development.

"It means we build better software, write better code, our engineers are able to work with more pride, and we're able to retain the world's best engineers because they know they can open source their work."

James Pearce
Engineering Director at Facebook (VentureBeat)

Most importantly, open source contribution enables organizations to attract, and hire the best talent. According to Wipro, 80% of organizations who work with and contribute to open source projects got into it specifically to attract and retain talent. It's no surprise then that 86% of information technology leaders believe that the most innovative companies are adopting and investing in open source software today (Red Hat).

Open-Source Software in Data Science

Now, we stand on the cusp of a fourth industrial revolution (Salesforce) that will be defined by the intersection of data-driven and data-generating technologies such as artificial intelligence, machine learning, the internet of things, and more. Open-source technologies will continue to empower organizations to make the most of their data and create transformative solutions, processes, and products with machine learning and data science.

The two most commonly used open source programming languages in data science are R and Python. There's a lot of discussion on the **difference between the two languages** as they provide thousands of open source data science and machine learning packages. At DataCamp, we've built our entire data science curriculum around empowering people and organizations to become data fluent by teaching the most popular and powerful open source frameworks for both languages.

While not every member of the organization is required to learn Python or R to become data-driven (The Data Leader's Guide to Upskilling), these technologies underpin the growth and adoption of data science in an organization. This guide will demystify the most popular data science and machine learning packages and tools in R and Python and uncover their use cases throughout an organization.

Data Manipulation

Python

`pandas` pandas is one of the most popular Python packages in data science for working with tabular data due to its ease of use, ability to work with large quantities of data, built-in plotting and aggregation tools, and more. It supports reading and writing a variety of data types, from CSV and Excel files to SQL and more.

`geopandas` GeoPandas is built on top of pandas and extends pandas capabilities to easily work, process, manipulate, and visualize geo-spatial data in Python.

`numpy` NumPy is one of the most elemental packages in Python, as many other packages are built on top of it, including pandas and SciPy. It allows the formation, transformation, and manipulation of arrays, among other operations.

`scipy` SciPy stands for scientific Python, and contains a set of scientific tools and techniques for statistics, linear algebra, data processing, and more.

Our content

Courses

Introduction to Python | Pandas Foundations | Manipulating DataFrames with pandas | Working with Geospatial Data in Python | Analyzing US Census Data in Python | Exploratory Data Analysis in Python

Tracks

Data Manipulation with Python (4 courses)

R

`dplyr` `tidyr` `readr` `tibble` Considered as the core packages of the tidyverse for data manipulation, these open source packages offer a host of tools and functions to read, manipulate, and tidy data. The readr package allows practitioners read in a variety of data types, whereas the tidyr, dplyr, and tibble packages offer a suite of tools to manipulate, clean, tidy, and work with data efficiently.

`data.table` The data.table package is used for working with tabular data in R and is widely known for its speed of execution on larger datasets and its intuitive syntax.

`xts` xts is one of the most popular packages for working with time series data in R. It allows a host of functions for working with time series data, such as indexing, resampling, handling missing data, and more.

Our content

Courses

Introduction to the Tidyverse | Data Manipulation with dplyr | Exploratory Data Analysis in R | Manipulating Time Series Data with xts and zoo in R | Data Manipulation with data.table in R

Tracks

Data Manipulation with R (5 courses)

Use Cases

Automate legacy Excel workflows

Conduct time-series analysis on sales data

Analyze traffic rates for city planning

Conduct Covid-19 contact tracing analysis

Optimize business processes with various constraints

Data Visualization

Python

`matplotlib` Matplotlib is the most popular data visualization package on Python, enabling comprehensive creation and customization of different types of data visualizations in Python.

`seaborn` Seaborn is a data visualization package built on top of Matplotlib that allows for the easy creation of highly aesthetic plots in Python

`bokeh` `plotly` Bokeh and Plotly are interactive visualization libraries that allow for the creation and customization of interactive plots and widgets that can be published in web pages

`follium` Follium is built on top of Javascript's Leaflet package, which provides the ability to easily visualize geospatial data in Python with robust styling capabilities

Our content

Courses

Introduction to Data Visualization with Matplotlib | Introduction to Data Visualization with Seaborn | Interactive Data Visualization with Bokeh | Visualizing Time Series Data in Python | Visualizing Geospatial Data in Python | Introduction to Data Visualization with Plotly in Python

Tracks

Data Visualization with Python (5 courses)

R

`ggplot2` The most popular data visualization package for R, this tidyverse package allows the creation and customization of a range of data visualizations. It also offers a range of extensions to visualize unique data structures like network data, quickly develop themes, animate plots, and more.

`leaflet` Originally a Javascript package, the Leaflet package provides the ability to easily visualize geospatial data in R with robust styling capabilities.

`rbokeh` `plotly` Rbokeh and Plotly are interactive visualization libraries that allow for the creation and customization of interactive plots and widgets that can be published in web pages.

Our content

Courses

Introduction to Data Visualization with ggplot2 | Interactive Data Visualization with rbokeh | Interactive Data Visualization with plotly in R | Interactive Maps with leaflet in R

Tracks

Data Visualization with R (3 courses)

Use Cases

Visually compare multiple columns using subplots

Create presentation ready-plots with three lines of code

Build free interactive dashboards to track key performance indicators hosted on web-pages

Visualize Covid-19 cases across the world

Data Cleaning

Python

`recordlinkage`	Built on top of pandas, the Record Linkage library allows the linking and merging of two or more data sources. It helps to match and deduplicate records that are believed to be the same entity.
`missingno`	Missingno allows the quick visualization and inspection of missing data, enabling data scientists to determine the root cause of missingness.

Our content

Courses

Cleaning Data in Python | Dealing with Missing Data in Python

Tracks

Importing and Cleaning Data with Python (5 courses)

R

`reclin`	reclin is an R library that allows linking and merging between of two or more data sources. It helps to match and deduplicate records that are believed to be the same entity.
`forcats`	The forcats package is a tidyverse package that enables practitioners to quickly solve common problems when working with categorical data, such as re-ordering, collapsing, and reordering categories.
`naniar` `vim`	The naniar and VIM packages allow the quick visualization and inspection of missing data, enabling data scientists to determine the root cause of missingness.

Our content

Courses

Cleaning Data in R | Dealing with Missing Data in R | Handling Missnig Data with Imputations in R | Working with Data in the Tidyverse

Tracks

Importing and Cleaning Data with R (4 courses)

Use Cases

- Consolidate and deduplicate disparate organizational data and establish trust in data quality
- Determine the root cause of missing data in a database
- Clean the results of a survey

Probability and Statistics

Python

`pymc3` PyMC3 is one of the most popular Python packages for probabilistic programming. It provides a host of tools to work with probabilistic programming in Python, including modeling, simulation, transformations and more.

`statsmodels` Statsmodels is a Python library that provides a host of statistical functions and capabilities, including regression models, time series analysis, experiment design, and more.

`arch` The arch package contains a set of functions for forecasting highly volatile time series data. Often used in finance, the arch package enables practitioners to model, evaluate, and work with GARCH models in Python, which are popular for forecasting volatile time series data.

Our content

Courses

Introduction to Regression with statsmodels in Python | Statistical Thinking in Python | ARIMA Models in Python | GARCH Models in Python | Customer Analytics and A/B Testing in Python

Tracks

Time Series with Python (5 courses) | Statistics Fundamentals with Python (5 courses)

R

`mass` MASS is an R library that provides a host of datasets and functionalities for statistical analysis, including regression models, statistical tests, and more.

`stats` stats is an R package that provides a comprehensive set of functions and capabilities, including regression models, plotting functionality, time series analysis, experiment design, and more.

`fable` Part of the tidyverts set of packages for time series forcasting, this package offers a range of tools and functions for easily performing and evaluating common time series forecasting models.

`powerMediation` The powerMediation package provides a robust set of tools for designing, running, and evaluating statistical experiments in R.

Our content

Courses

Introduction to Statistical Modeling in R | Correlation and Regression in R | Foundations of Inference | Foundations of Probability in R | Experimental Design in R | Survival Analysis in R

Tracks

Time Series with R (6 courses) | Statistics Fundamentals with R (5 courses)

Use Cases

Determine the best performing webpage enhancement with an A/B test

Forecast demand with supply chain planning

Measure the volatility of a stock portfolio

Evaluate the results of a clinical trial in Pharmaceuticals

Machine Learning

Python

`scikit-learn` scikit-learn is the most popular and versatile machine learning framework across any programming language. It includes a host of tools, functions, and techniques, covering the entirety of the machine learning pipeline from exploratory analysis, to data pre-processing, modeling and training, and accuracy evaluation.

`catboost` `lightGBM` `xgboost` CatBoost, LightGBM, and XGBoost are machine learning libraries for gradient boosting on decision trees. While there are differences between them when it comes to training speed, how they handle missing values, feature importance methods, and other key technical components; gradient boosted trees have become widely popular for machine learning on tabluar data.

`tensorflow` TensorFlow is an end-to-end deep learning framework developed by Google that provides a comprehensive set of tools for deep learning model building, evaluation, and deployment.

`keras` Keras is a library built on top of TensorFlow, meant to reduce the barrier to working with deep learning by simplifying model building, evaluation, and deployment.

`pytorch` PyTorch is another popular deep learning framework developed by Facebook. It is widely used in research and provides a comprehensive toolset for deploying deep learning models in production.

Our content

Courses

Supervised Learning with scikit-learn | Unsupervised Learning in Python | Cluster Analysis in Python | Introduction to Tensorflow in Python | Extreme Gradient Boosting with XGBoost | Introduction to Deep Learning with Keras | Introduction to Deep Learning with PyTorch | Machine Learning for Marketing in Python

Tracks

Machine Learning Fundamentals with Python (5 courses)
Machine Learning Scientist with Python (23 courses)

R

`tidymodels` Similar to the tidyverse, tidymodels is a collection of R packages designed for the machine learning workflow. It contains a range of packages such as `rsample` for better data splitting and sampling, efficient modelling with `parsnip` , hyperparameter tuning with the tune package, and more.

`caret` Caret is one of the most popular machine learning packages in R. It includes a host of tools, functions, and techniques, that cover the entirety of the machine learning pipeline from exploratory analysis, to data pre-processing, modeling and training, and accuracy evaluation.

`xgboost` Just like its Python counterpart, XGBoost is a machine learning library for gradient boosting on decision trees, a widely popular technique for machine learning on tabluar data.

`metrics` Metrics is one of the most popular R packages for evaluating the performance of a range of machine learning predictions, from classification to regression, time series forecasting, and more.

`rpart` rpart is a popular package for working with tree-based models in R. It allows predicting categorical and continuous outcomes by developing easy to visualize decision rules.

Our content

Courses

Supervised learning in R: Classification | Supervised learning in R: Regression | Unsupervised Learning in R | Machine Learning with caret in R | Machine Learning for Marketing Analytics in R | Tree-based models in R

Tracks

Machine Learning Fundamentals in R (4 courses)
Machine Learning Scientist with R (15 courses)

Use Cases

Predict customer churn with classification models
Predict housing prices with regression models
Detect customer segments with unsupervized learning
Develop an image recognition system to digitize documents

Natural Language Processing

Python

`gensim` Gensim is a fast and efficient Python library for topic modeling, document comparison, topic identification, and more on large text datasets.

`spacy` spaCy is an open source library for Natural Language Processing that performs a range of NLP tasks from tokenization, part-of-speech tagging, lemmatization, text classifciation, and more.

`nltk` NLTK is an open source Python library that provides a host of NLP tools for data preprocessing, classification, parsing text, sentiment analysis, and more.

Our content

Courses

Introduction to Natural Language Processing in Python | Sentiment Analysis in Python | Advanced NLP with spaCy | Feature Engineering fo NLP in Python | Machine Translation in Python

Tracks

Natural Language Processing in Python (6 courses)

R

`tidytext` tidytext provides a suite of NLP functions to make text mining tasks easier, more effective, and consistent with the tidyverse toolset. It allows practitioners to efficiently perform tasks like tokenization, sentiment analysis, remove stotpwords, and more.

`topicmodels` The topicmodels package provides a host of topic modeling functions aimed at identifying and summarizing text and categorizing documents.

`stringr` Stringr is one of the most popular packages in R for working with text data. Part of the tidyverse packages, it allows a host of operations on text data such as string detection, string subsetting, joining and splitting strings, and more.

Our content

Courses

Introduction to Natural Language Processing in R | Introduction to text analysis in R | Intermediate Regular Expressions in R | String Manipulation with stringr in R | Text Mining with Bag of Words in R

Tracks

Text Mining with R (4 courses)

Use Cases

Categorize documents based on topic

Pre-process text data for deep learning models

Perform sentiment analysis on customer tweets

Application Specific Packages

Python

`networkx` NetworkX is one of the most popular Python packages for creating, manipulating, and studying network structures in Python.

`tweepy` Tweepy is an easy to use Python library for accessing and manipulating twitter

`pypfopt` PyPorfolioOpt is a popular Python package for portfolio analysis, optimization, and quantiative risk management in Python.

`skimage` Scikit-image is an open source library containing a collection of image processing algorithms such as feature dettection, filtering, segmentation, and more.

`opencv` OpenCV is one of the most popular computer vision libraries on Python that contains a wide range of tools for working with and processing image data.

Our content

Courses

Introduction to Network Analysis in Python | Intermediate Network Analysis in Python | Analyzing Social Media Data in Python | Image Processing in Python | Quantitative Risk Management in Python

Tracks

Image Processing with Python (3 courses)
Applied Finance in Python (4 courses)

R

`igraph` igraph is one of the most popular packages for creating, manipulating, visualizing and studying network structures in R.

`rtweet` rtweet is an easy-to-use R library for accessing and manipulating Twitter data.

`qrm` QRM is a popular package for portfolio analysis, optimization, and quantitative risk management in R.

`magick` Built on top of ImageMagick STL, a popular open source library for working with image data, magick provides a comprehensive set of functionalities to work with and process image data in R.

Our content

Courses

Network Analysis in R | Case Studies: Network Analysis in R | Network Analysis in the Tidyverse | Quantiative Risk Management in R | Analyzing Social Media Data in R

Tracks

Marketing Analytics with R (6 courses)
Applied Finance in R (7 courses)

Use Cases

Optimize supply chain flows with network analytics

Analyze the popularity of a service in a given geographical location

Automatically optimize a stock porfolio

Perform optical character recognition for document digitization

Reporting and Communicating Data

Python

`dash`

Dash is a highly robust framework for building rich, interactive, and customizable data visualization apps that can be rendered and shared easily on a web browser.

`streamlit`

Streamlit is another highly popular framework for quickly building and sharing data apps. While it is highly useful for sharing data insights on a variety of use cases, it is especially used for sharing machine learning model results and analysis.

`jupyter notebooks`

Arguably the most popular tool in Python for data science, Jupyter Notebooks are the IDE of choice for 74% of data scientists (Kaggle). Jupyter Notebooks are an open source web application that allows creating and sharing documents containing live code, visualizations, and narrative text. They've completely revolutionized how data scientists share their work, and will continue to lower the barrier for data democratization (DataCamp).

Our content

Courses

Building Dashboards with Dash and plotly (coming soon)

Projects

Comparing Search Interest wiith Google Trends | Exploring the evolution of lego
Bad passwords and the NIST guidelines | Analyzing TV Data

R

`R Markdown`

One of the most popular tools in the R data science stack, the R Markdown Notebook is similar to a Jupyter Notebook in Python. It allows practioners to analyze, describe, share, and reproduce their analyis in a friendly notebook interface.

`shiny`

Shiny is one of the most popular packages in data science and in the R data science stack. It provides the ability to create highly robust dashboards and web apps that can be rendered and easily shared on a web browser.

`shinydashboards`

shinydashboard is a library built on top of shiny that makes it easy to develop data visualization dashboards with shiny.

`flexdashboards`

flexdashboard is an open source R library that makes it easy to develop dashboards with RMarkdown.

Our content

Courses

Building Web Applications with Shiny in R | Building Dashboards with shinydashboard |
Case Studies: Building Web Applications with Shiny in R | Reporting with R Markdown |
Building Dashboards with flexdashboard

Tracks

Shiny Fundamentals (4 courses)

Use Cases

- Live tracking of team or company OKRs with a web-based dashboard
- Sharing machine learning experiment results with business stakeholders
- Automating legacy Excel workflows
- Posting and sharing data analysis results to business stakeholders
- Onboarding new hires on data processess with notebook tutorials

Big Data

Python

`pyspark` Apache Spark is an open-source distributed data processing framework that can perform data processing tasks on very large datasets. PySpark provides a Python API for working with Spark.

`dask` Dask is a flexible library for parallel computing in Python. It provides parallelized NumPy and pandas DataFrame objects which provide faster performance while working with big data in Python.

Our content

Courses

Introduction to PySpark | Cleaning Data in PySpark | Machine Learning with PySpark
Building Recommendation Engines with PySpark | Parallel Programming with Dask in Python

Tracks

Big Data with PySpark (6 courses)

R

`fst` fst provides a fast and flexible way to serialize data frames. It allows for faster read and write times, and enables practitioners to work more quickly with big data in R.

`sparklyr` Spark is an open-source distributed data processing framework that can perform data processing tasks on very large datasets. sparklyr provides an R API for working with Spark.

Our content

Courses

Introduction to Spark with sparklyr in R | Scalable Data Processing in R |
Parallel Programming in R

Tracks

Big Data with R (5 courses)

Use Cases

Perform market basket analysis on millions of customer e-commerce transactions

Quickly analyze millions of Covid-19 infections

Develop a recommendation engine on large movie streaming datasets

Data Engineering

Python

`airflow` Developed by Airbnb, Apache Airflow is an open-source tool for data workflow automation. It is highly scalable and extensible, and works well with a variety of common tools like cloud providers, databases, Salesforce, and more.

`sqlalchemy` SQLAlchemy is a comprehensive SQL toolkit for Python that enables mapping SQL tables to user-defined Python objects, making it easy to create tables, map relations between them, and ingest data all through Python.

`sqlite3` SQLite3 provides a SQL interface in Python that allows practitioners to connect to a SQL database and execute SQL code within Python.

Our content

Courses

Introduction to Data Engineering | Introduction to Airflow in Python | Introduction to Databases in Python | Streamlined data ingestion with pandas | Introduction to Importing Data in Python

Tracks

Data Engineer with Python (25 courses)

R

`jsonlite` `xml2` While Python is more known for data engineering, jsonlite and xml2 are R packages that provide a host of tools for working with, processing, and transforming JSON and XML files in R. They allow practitioners to easily work with web-data, and are optimized for building pipelines with R.

`odbc` The odbc package provides a wide range of functionality for connecting to, and working with databases in R. It provides support for various types of databases, from MySQL, PostgreSQL, SQL Server, SQLite, BigQuery, Redshift and more.

`dbi` DBI provides a SQL interface in R that allows practitioners to connect to a SQL database and execute SQL code within R. There are many packages built on top of DBI that make it even easier to connect to databases in R, such as `RPostgreSQL`, `RMySQL`, and `ROracle`.

Our content

Courses

Intermediate Importing Data in R | Working with web data in R | Introduction to Relational Databases in SQL

Tracks

R Programmer (12 courses)

Use Cases

Scheduling a daily data analysis workflow

Extract, transform, and load data into a database

Scrape web pages and load its contents into a database

How open source is driving data fluency

Just as the open source revolution catalyzed the software revolution, it is also paving the way toward data democratization and organization-wide data fluency. This is especially accelerated by the open, collaborative nature of open source data science (Anaconda) and the speed of innovation it allows (TechRepublic).

Data-fluent teams around the world are using open source data science tools and technologies to democratize data by providing better access to data, streamlining data processes, creating time-saving tools, and upskilling their people. This ultimately results in equipping stakeholders across an organization with the tools to make data-driven decisions.

For example, Airbnb and Spotify open sourced their proprietary tools **Airflow** and **Luigi**, enabling organizations to easily and scalably build data pipelines and provide better, more resilient access to data. Lyft's **Amundsen** allows organizations to discover, update, and understand the changes that occur to their data, building trust for data-driven teams. Netflix embraced the Jupyter Notebook (**Netflix**), using it as a central tool within many of its processes through the use of notebook templates. This allows data-driven teams comprising business analysts to data engineers to easily work with data.

Open-source software also allows data scientists to create highly flexible tools and frameworks that are tailor-made for their organizations' workflows. This enables organizations to simplify complex data processes, allowing anyone with basic coding skills **to work with data**. For example, DataCamp's data science team has open sourced R and Python packages, `dbconnectR` and `dbconnect-python` , that simplify connecting to databases, enabling data consumers to access data with limited **R** or **Python** skills. Airbnb developed an R package named `rbnb` , which allows teams to easily access and move data within Airbnb's data infrastructure, easily create branded visualizations, access different RMarkdown report templates, and access custom functions to **optimize specific Airbnb data workflows**.

→ Discover **Open Source at DataCamp**

Upskilling is a key component of open source data science

While the benefits of streamlining data processes and developing time-saving tools cannot be understated, these tools require the necessary skills across teams. This is why upskilling is a key component of open source driven data democratization. For example, Airbnb launched a **Data University** aimed at providing thousands of its employees the necessary skills to work with open source software for data science. **Bloomberg uses DataCamp** as part of a blended learning environment to teach data analysis with Python and empower employees of all skill levels to write data-driven financial news stories. DataCamp partnered with a major global retail bank to transition their risk analytics department from SAS to Python, reducing dependence on licensed legacy software and focusing on future-proof open source Python packages like pandas and scikit-learn. As organizations look to scale their data science with better open source tooling, closing the skills gap will need to go hand in hand with these efforts.

"We've trialed a number of other online learning solutions, but only DataCamp provides the interactive experience that reinforces learning. Just as you wouldn't trust a surgeon who had watched some videos about surgery, you couldn't trust a developer who has watched some videos about programming. There's a great depth of content on the site. It's great for absolute beginners, but there is very advanced content for users with more experience."

Sarah Schlobohm
Senior Analytics Manager, Global Risk Analytics, HSBC

DataCamp's proven learning methodology for learning open source data science

Beyond courses and tracks, DataCamp's proven learning methodology provides a cyclical process for learning and retention. This learning methodology enables learners across the data fluency spectrum to assess their skills and identify gaps, develop a learning plan based on these gaps, practice skills, and apply them in a real-world setting. Experienced data scientists can upskill on new open source packages and tools in their target domain, and domain experts can learn the fundamentals of data literacy and data science to get started with open source technologies.

Assess

Effective learning starts with understanding skill gaps and strengths. With DataCamp Signal™, learners can understand specific skill gaps they have across various topics and tools. From data literacy assessments like understanding and interpreting data to programming and machine learning assessments in R or Python, our 10-minute adaptive evaluations provide learners with personalized skill gaps and learning paths to address their skill gaps.

Learn

DataCamp's growing course library houses more than 350 expert-led, hands-on courses across various technologies and domains for all data skills and levels. Learners can hit the ground running with our learn-by-doing-approach—our bite-sized videos and interactive coding exercises allow them to start working with their preferred tool and topic right in the browser.

Practice

The next step in DataCamp's proven learning methodology is to practice all the information retained in courses. Using practice mode, learners can practice what they've learned with short challenges to test critical concepts. With over 3,400 practice questions, learners can practice their skills across various technologies and topics. Our mobile app is the perfect way to practice and learn on the go.

Apply

Once skills have been assessed, cultivated through courses, and sharpened through practice, learners are ready to apply their skills in a project-based environment. With DataCamp projects, learners can solve a variety of real-world R and Python data science projects. Learners can opt for guided projects, where they can follow step-by-step tasks and receive helpful feedback as they apply their newfound skills. They can also opt for unguided projects, which are open-ended and offer a variety of possible solutions along with a live-code-along video to follow how an expert data scientist would approach a solution.

DataCamp's entire learning experience is easy to implement and manage for teams of any size, with an administrator dashboard that allows custom learning paths based on roles and departments, advanced analytics and insights to measure the impact of online learning, and seamless SSO and LMS integrations. Teams benefit from our Customer Success Managers, who partner with organizations to accelerate learning adoption and provide valuable recommendations to help achieve organization-wide data fluency. We have more than 7 million learners around the world—and we're just getting started. Close the talent gap. Visit datacamp.com.

datacamp
clear as data