AI & Data Science Training

The definitive Master's programme taking you from foundational Python to building advanced Generative AI and Autonomous Agent systems. Become the AI-Native professional the industry is desperately seeking.

Python for AI & Data

Master Python fundamentals through advanced OOP concepts

SQL for AI & Data

Database design, querying, and optimization techniques

PowerBI for Data Analysis

Business intelligence and interactive visualizations

Maths & Stats for AI & Data

Mathematical foundations and statistical analysis

Python Libraries for AI & Data

NumPy, Pandas, Matplotlib, Seaborn, and Plotly

Machine Learning

Supervised, unsupervised, and reinforcement learning

Deep Learning & NLP

Neural networks, CNNs, RNNs, and language processing

Generative AI & Agentic AI

LLMs, RAG systems, and autonomous agent architectures

Digital Edify

India's First AI-Native Training Institute

Learn AI. Build Agents. Lead Future.

hello@digitaledify.ai

+91 6304982304

About Digital Edify

India's #1 Training Institute for the AI Era

Established: 2016 **Headquarters:** Hyderabad, Telangana **Reach:** Global (Online + Offline)

The Transformation Narrative

Digital Edify has evolved from a premium training institute in the Automation Era to an AI-first organisation leading the Agentic AI revolution. Since 2016, we've transformed over 100,000 professionals and built partnerships with more than 1,000 industry leaders. Our journey reflects the technological evolution of our time—from traditional job placement to career transformation, and now to building AI-native professionals who will shape the future of work.

1

Automation Era (2016-2023)

Premium Training Institute focused on job placement with 100K+ students trained

2

AI Revolution (2024-2025)

AI-Powered Training with industry-AI integration and career transformation focus

3

Agentic AI Leadership (2026+)

AI First Institute building AI-Native Professionals with 1 Million AI-Native Vision

> "We started in the Automation Era. We evolved through the AI Revolution. Now, we're leading the Agentic AI Future—with 100,000+ professionals already transformed and 1,000+ industry partners trusting our graduates."

Vision & Mission

Vision

"To Create 1 Million AI-Native Professionals Who Will Build the Agentic Future of Work"

Mission

"We transform learners into AI-native professionals through industry-aligned programmes that integrate Agentic AI into every discipline—from development to data science to enterprise platforms."

Course Highlights

Section 1: Python for AI & Data

Build strong Python foundations covering core syntax, data structures, OOP, file handling, and advanced programming concepts.

Section 2: SQL for AI & Data

Design, query, optimize, and automate relational databases using advanced PostgreSQL and SQL techniques.

Section 3: Power BI for Data Analysis

Create end-to-end business intelligence solutions with data modeling, DAX, dashboards, and enterprise deployment.

Section 4: Math & Stats for AI & Data

Develop mathematical, probability, and statistical foundations required for data analysis and machine learning.

Section 5: Python Libraries for AI & Data

Analyze, manipulate, and visualize data using NumPy, Pandas, and modern visualization libraries.

Section 6: Machine Learning

Build, evaluate, optimize, and deploy machine learning models across supervised, unsupervised, and reinforcement learning.

Section 7: Deep Learning & NLP

Design and train deep learning and NLP models using neural networks, CNNs, RNNs, and sequence architectures.

Section 8: Generative AI & Agentic AI

Design, deploy, and govern generative and agentic AI systems using LLMs, RAG, workflows, and production architectures.

SECTION 1

Python for AI & Data

Module 1: Python Fundamentals

Begin your journey with comprehensive Python fundamentals. This module covers everything from environment setup to control flow, providing the essential foundation for AI and data science work. You'll master Python's syntax, data types, and programming constructs through hands-on practice.

1

Environment & Syntax

- Python interpreter installation (Windows/Mac)
- IDE setup with Visual Studio Code
- 35 Python keywords and naming conventions
- Variables and memory management

2

Data Types & Operators

- Simple and complex data types
- Type conversion and casting
- Arithmetic, comparison, and logical operators

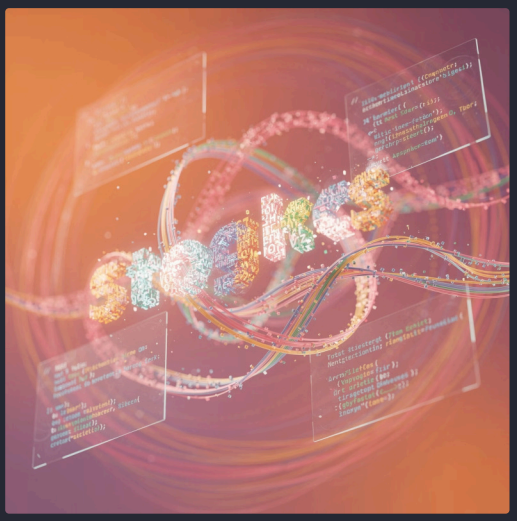
3

Control Flow

- Conditional statements (if, elif, else, match-case)
- Loops (while, for) and range() function
- Control flow (break, continue, pass)
- User input with input() function

Module 2: String Manipulation

Strings are fundamental to data processing and AI applications. This module provides comprehensive coverage of string operations, from basic indexing to advanced formatting techniques. Master the tools needed to clean, transform, and analyse textual data effectively.



Core String Operations

- String indexing (positive and negative)
- Slicing with start:end:step notation
- Concatenation and repetition operations
- String formatting (f-strings, format())
- Understanding string immutability

- ### Case Conversion

  Methods for transforming string case (upper, lower, title, capitalize)

- ### Search Methods

  Finding substrings with find, index, and count operations

- ### Validation Methods

  Checking string properties (isalpha, isdigit, isalnum, isspace)

- ### Trimming & Replacement

  Strip, lstrip, rstrip, replace, split, and join methods

- ### String Alignment

  Formatting strings with alignment methods for presentation

Module 3: Data Structures

Lists & Tuples

Lists and tuples are essential data structures for storing collections of data. This module explores their creation, manipulation, and practical applications. Understanding the differences between mutable lists and immutable tuples is crucial for effective Python programming in AI and data science contexts.

Lists: Dynamic Collections

Creation, indexing, slicing, and operations. Add elements with append, insert, extend. Remove with remove, pop, clear. Search with index and count.

List Manipulation

Sorting and reversing with sort() and reverse() methods. Powerful list comprehensions for efficient data transformation.

Tuples: Immutable Sequences

Creation, operations, and immutability benefits. Tuple packing and unpacking for elegant code. Compare lists vs tuples for optimal use.

Module 4: Data Structures

Dictionaries & Sets

Dictionaries and sets provide powerful ways to organise and manipulate data. Dictionaries offer key-value storage for structured data, whilst sets provide unique collections with mathematical operations. These structures are fundamental for data processing and algorithm implementation.

Dictionary Operations

- Creation and access patterns
- Keys, values, and items methods
- Dictionary comprehensions
- Nested dictionaries for complex data

Dictionaries

Key-value pairs for structured data storage and rapid lookups

Frozen Sets

Immutable sets for use as dictionary keys and guaranteed consistency

Sets (UUU)

Unique, unordered, unchangeable collections with mathematical operations

Set Operations

Union, intersection, difference, subset, and superset checks

Module 5: Advanced Collections & Iterators

Advanced collections and iterators unlock powerful programming patterns for efficient data processing. This module introduces specialised data structures from the collections module and explores the iterator protocol, generators, and functional programming concepts essential for writing elegant, memory-efficient code.

Collections Module

Specialised containers: namedtuple for readable tuples, Counter for counting, defaultdict for default values, and deque for efficient queue operations.

Iterators & Generators

Iteration protocol and custom iterators. Generators with yield statement for memory-efficient data processing. Generator expressions and pipelines.

Functional Programming

Lambda functions for concise operations. Higher-order functions: map, filter, and reduce. Functional programming concepts for elegant code design.

Module 6: Functions & Scope

Functions are the building blocks of modular, reusable code. This comprehensive module covers function definition, parameter types, scope rules, and advanced concepts like recursion and lambda functions. Master these concepts to write clean, maintainable code for AI and data science applications.

01

Function Basics

Definition, calling, parameters, and return statements. Multiple return values for complex operations.

02

Parameter Types

Positional, keyword, default, *args, and **kwargs for flexible function signatures.

03

Scope & Variables

Local and global scope. Global keyword usage for modifying global variables.

04

Advanced Functions

Lambda functions (IIFE), docstrings for documentation, and recursive functions.

Built-in Functions

Python's extensive library of built-in functions for common operations

User-Defined Functions

Creating custom functions tailored to specific requirements

Function Documentation

Docstrings for clear, maintainable code documentation

Module 7: Modules & Packages

Modules and packages enable code organisation and reusability at scale. Learn to create, import, and manage both built-in and external modules. Understanding package structure and the pip ecosystem is essential for leveraging Python's vast library ecosystem in AI and data science projects.

1

Module Types

- Built-in modules (math, random, datetime, os, sys)
- User-defined modules
- External packages

2

Importing Techniques

- Import statements and aliases
- From imports for specific functions
- Creating user-defined modules

3

Package Management

- Package structure and __init__.py
- Nested packages
- pip package manager
- requirements.txt management

Popular External Packages: requests for HTTP operations, pandas for data manipulation, and numpy for numerical computing form the foundation of Python's data science ecosystem.

Module 8: Working with Data Formats

Data persistence and exchange are fundamental to AI and data science workflows. This module covers file operations, directory management, and working with common data formats like CSV and JSON. Master these skills to efficiently read, write, and transform data across different formats and storage systems.

1

File Operations (CRUD)

open() function with file modes. Reading with read, readline, readlines. Writing and appending data.

2

Directory Management

File path operations with os module. Directory and folder management using os and shutil libraries.

3

CSV Files

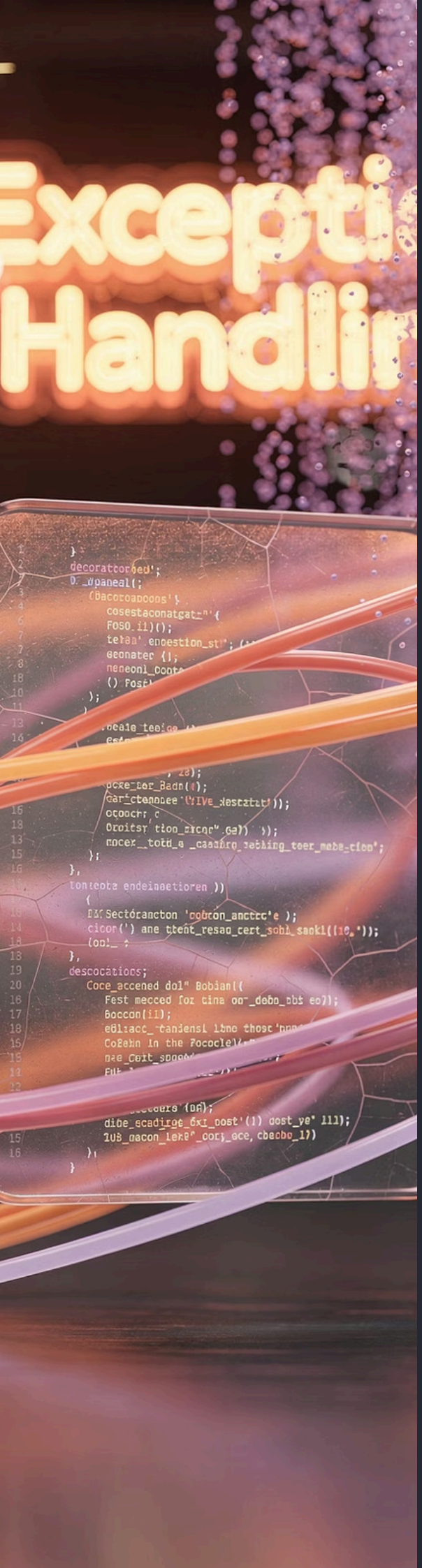
csv.reader and csv.writer for basic operations. csv.DictReader and csv.DictWriter for structured data.

4

JSON Files

JSON operations: dump, dumps, load, loads. Data serialisation and deserialisation for APIs and storage.

hello@digitaledify.ai



Module 9: Advanced Python Concepts

Advanced Python concepts elevate your code from functional to professional-grade. This module explores exception handling for robust error management, decorators for extending functionality, and context managers for resource management. These patterns are essential for building production-ready AI and data science applications.

Exception Handling

- try-except-else-finally blocks
- Catching specific exceptions
- Raising and re-raising exceptions
- Custom exception classes
- Built-in exception types

Decorators

- Function decorators fundamentals
- Decorators with arguments
- Multiple decorators
- Class decorators
- Practical decorator applications

Generators & Context Managers

- Generators deep dive
- Generator expressions
- Infinite generators
- Context managers for resources
- Custom context managers

hello@digitaledify.ai

Module 10: Object-Oriented Programming

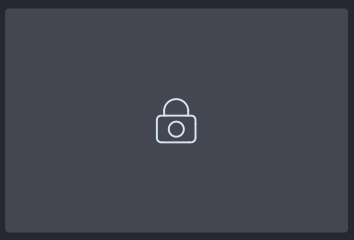
Object-Oriented Programming (OOP) is fundamental to building scalable, maintainable AI systems. This comprehensive module covers OOP philosophy, class design, and the four pillars: encapsulation, inheritance, abstraction, and polymorphism. Master these concepts to architect sophisticated data science applications and machine learning frameworks.

OOP Fundamentals

Classes, objects, attributes, __init__ constructor, and understanding self

Methods

Instance methods, class methods (@classmethod), static methods (@staticmethod)



Encapsulation

Access modifiers: public, protected, private attributes and methods

Inheritance

Single, multi-level, multiple inheritance. Method overriding and super() function

Abstraction

Abstract classes, abstract methods, ABC module for interface definition

Polymorphism

Method overriding, duck typing, special methods (__str__, __repr__, __len__)

SECTION 2

SQL for AI & Data

Module 1: Foundations of Databases & PostgreSQL

Databases are the backbone of data-driven AI systems. This module introduces relational database concepts, ACID properties, and PostgreSQL—a powerful open-source RDBMS. Learn to design schemas, create tables with proper constraints, and ensure data integrity through referential relationships essential for AI data pipelines.

Database Fundamentals

Introduction to databases, DBMS, and RDBMS concepts. ACID properties: Atomicity, Consistency, Isolation, Durability for reliable transactions.

PostgreSQL Setup

Installation on Windows, Mac, and Linux. PostgreSQL tools: psql command-line interface and pgAdmin 4 graphical interface.

Database Objects

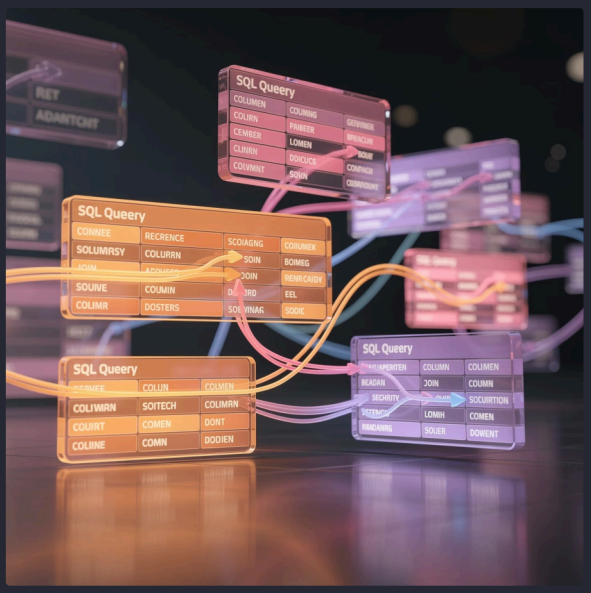
Databases, schemas, and tables structure. Data types: numeric, character, date/time, boolean, and special types for diverse data.

Constraints & Integrity

PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK, and DEFAULT constraints. Creating tables, INSERT operations, and referential integrity.

Module 2: Querying and Analysing Data

Querying is the heart of database interaction. This comprehensive module covers SELECT statements, filtering, sorting, and aggregation. Master JOIN operations to combine data from multiple tables, and leverage window functions for advanced analytics. These skills are essential for extracting insights from complex datasets in AI projects.



Core Query Operations

- SELECT with column aliases and expressions
- WHERE clause with comparison and logical operators
- BETWEEN, IN, LIKE for flexible filtering
- NULL handling (IS NULL, IS NOT NULL)
- ORDER BY, DISTINCT, LIMIT, and OFFSET

1

Functions

String, numeric, and date/time functions for data transformation

2

Aggregation

COUNT, SUM, AVG, MIN, MAX with GROUP BY and HAVING

3

Window Functions

ROW_NUMBER, RANK, LAG, LEAD for advanced analytics

4

JOINs

INNER, LEFT, RIGHT, FULL OUTER, CROSS, and SELF joins

Module 3: Advanced Queries & Data Manipulation

Advanced querying techniques unlock sophisticated data analysis capabilities. This module explores subqueries, Common Table Expressions (CTEs), set operators, and transaction management. Learn to write complex queries that combine multiple data sources, handle hierarchical data, and ensure data consistency through proper transaction control.

1 Subqueries & CTEs

Subqueries in WHERE, SELECT, FROM clauses. Correlated subqueries. EXISTS and NOT EXISTS. Common Table Expressions and recursive CTEs for hierarchical data.

2 Set Operations

UNION and UNION ALL for combining results. INTERSECT for common rows. EXCEPT for difference operations.

3 Data Modification

UPDATE statements with expressions and JOINs. DELETE statements with subqueries. TRUNCATE vs DELETE comparison.

4 Transaction Management

BEGIN, COMMIT, ROLLBACK for transaction control. Savepoints for partial rollbacks. Transaction isolation levels and concurrency control.

Module 4: Database Programming & Automation

Database programming enables automation and complex logic within the database. This module covers schema modifications, indexing for performance, views for abstraction, and PL/pgSQL for stored functions and procedures. Master triggers for automated actions and learn to build sophisticated database-driven applications.

Schema Management

ALTER TABLE operations. Adding, modifying, dropping columns. Managing constraints dynamically.

Indexes & Performance

Index types: B-tree, Hash, GIN, GiST. Creating and managing indexes. Performance optimization strategies.

Views & Abstraction

Creating views for data abstraction. Updatable views. Materialized views and refresh operations.

Functions & Procedures

PL/pgSQL programming. Stored functions with parameters. Stored procedures. Control structures and exception handling.

Triggers & Automation

BEFORE, AFTER, INSTEAD OF triggers. Trigger functions. Audit logging and data validation automation.

Module 5: Database Design & Optimisation

Effective database design is crucial for scalable AI systems. This module covers Entity-Relationship modelling, normalisation principles, and design best practices. Learn query optimisation techniques, execution plan analysis, and performance tuning strategies to build efficient, maintainable databases for data-intensive applications.

ER Modelling

Entities, attributes, relationships. Relationship types (1:1, 1:M, M:N). ER diagrams for visual design.

Normalisation

First, Second, and Third Normal Forms (1NF, 2NF, 3NF). Benefits and trade-offs. When to denormalise.

Design Best Practices

Naming conventions. Data type selection. Primary and foreign key strategies.

Query Optimisation

EXPLAIN and EXPLAIN ANALYZE. Reading execution plans. Index strategies and query rewriting.

Performance Tuning

Database statistics (ANALYZE). VACUUM maintenance. Connection pooling. Table partitioning.

SECTION 3

PowerBI for Data Analysis

Modules 1-2: Foundations & Data Connectivity

Power BI transforms raw data into actionable insights through interactive visualisations. This section introduces Business Intelligence fundamentals, Power BI architecture, and comprehensive data connectivity options. Learn to connect to files, databases, cloud sources, and web APIs whilst understanding the performance implications of different connection modes.

BI Fundamentals

Business Intelligence concepts, modern analytics approaches, Power BI components and architecture, interface navigation, and creating your first report.

Data Sources

File, database, cloud, and web source connectivity. Data source settings and credential management for secure access.

Connection Modes

Import vs DirectQuery vs Live Connection. Performance considerations and optimal mode selection for different scenarios.

hello@digitaledify.ai

Modules 3-4: Data Preparation & Modelling

Data preparation and modelling form the foundation of effective analytics. Master Power Query for data transformation, profiling, and quality assessment. Learn to design efficient data models using star and snowflake schemas, create relationships, and optimise model performance for responsive dashboards.

1

Power Query Transformations

Interface and applied steps. Data profiling and quality assessment. Essential transformations: filtering, splitting, merging. Reshaping: pivot, unpivot, grouping. Combining queries with append and merge.

2

Data Modelling

Star schema vs snowflake schema design. Creating and managing table relationships. Primary and foreign keys. Hierarchies and date dimension tables. Data model optimisation strategies.

Modules 5-6: Visualisations & DAX Fundamentals

Visual storytelling transforms data into insights. This section covers data visualisation principles, chart selection, and building interactive reports with slicers, filters, and drill-through capabilities. Introduction to DAX (Data Analysis Expressions) enables calculated columns, measures, and KPIs for sophisticated business metrics.

Visual Reports & Dashboards

Data visualisation principles and chart selection. Core visualisations: charts, tables, maps, KPIs. Interactive elements: slicers, filters, bookmarks, drill-through. Dashboard layout and mobile optimisation. Storytelling with data.

Introduction to DAX

DAX syntax and structure fundamentals. Calculated columns vs measures. Essential functions: aggregation, logical, text, date/time. CALCULATE and FILTER functions. Creating KPIs and business metrics.

Modules 7-8: Advanced DAX & Custom Analytics

Advanced DAX and custom analytics unlock sophisticated analytical capabilities. Master time intelligence for period comparisons, understand filter and row context, and leverage variables and iterator functions. Integrate R and Python for advanced analytics, and utilise AI visuals for automated insights and natural language queries.

01

Time Intelligence

YTD, MTD, QTD functions. Prior period comparisons and growth rates.

02

Advanced DAX

Filter context vs row context. Variables and iterator functions. Performance optimisation.

03

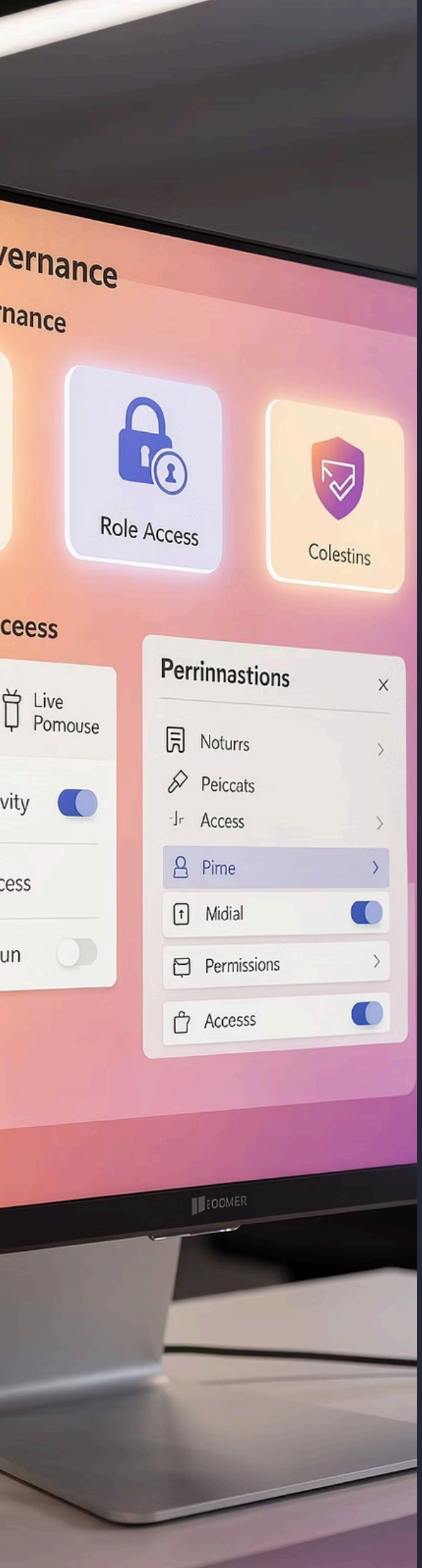
Custom Visuals

AppSource custom visuals. Advanced charts: waterfall, funnel, decomposition tree.

04

AI Analytics

R and Python integration. AI visuals: Key Influencers, Q&A, Smart Narratives.



Modules 9-10: Publishing & Enterprise Deployment

Enterprise deployment requires robust governance and collaboration capabilities. Learn to publish reports, manage workspaces, configure data refresh, and implement sharing strategies. Master Row-Level Security, deployment pipelines, and performance optimisation for scalable, secure analytics across your organisation.

Publishing & Sharing

Publishing to workspaces. Dashboards vs reports. Data refresh and gateway configuration. Sharing strategies and Power BI apps. Integration with Teams, SharePoint, Excel, PowerPoint.

Governance & Security

Admin portal and tenant settings. Row-Level Security (RLS) and Object-Level Security (OLS). Incremental refresh and aggregations.

Enterprise Features

Dataflows and deployment pipelines. Performance optimisation and capacity management. Enterprise licensing models. APIs and embedded analytics.

hello@digitaledify.ai

SECTION 4

Maths & Stats for AI & Data

Modules 1-2: Mathematical Foundations & Probability

Mathematics forms the theoretical foundation of AI and machine learning. This section covers essential linear algebra, including vectors, matrices, eigenvalues, and PCA intuition. Probability fundamentals introduce sample spaces, conditional probability, Bayes' theorem, and random variables—concepts critical for understanding machine learning algorithms.

Linear Algebra

Vectors, matrices, operations, transpose, inverse, determinant. Systems of linear equations. Eigenvalues, eigenvectors, diagonalisation. PCA intuition.

Probability Fundamentals

Sample spaces, events, probability axioms. Conditional probability and independence. Bayes' theorem. Random variables (discrete vs continuous). PMF, PDF, Monte Carlo simulations.

Modules 3-4: Distributions & Statistical Analysis

Probability distributions model real-world phenomena and form the basis of statistical inference. Master discrete distributions (Binomial, Poisson) and continuous distributions (Normal, Exponential). Learn the Central Limit Theorem, hypothesis testing, and statistical significance—essential tools for validating machine learning models and drawing conclusions from data.

Key Distributions

- Discrete: Binomial, Poisson, Geometric
- Continuous: Normal, Exponential, Uniform
- Expectation, variance, standard deviation
- Central Limit Theorem and sampling distributions

Descriptive Statistics

Mean, median, mode, standard deviation. Measures of spread. Outlier detection. Data visualisation.

Error Types

Type I and Type II errors. Power analysis and sample size determination.

Hypothesis Testing

Null and alternative hypotheses. p-values and significance. t-tests, chi-square, ANOVA. Confidence intervals.

Module 5: Applied Statistics for Machine Learning

Applied statistics bridges theory and machine learning practice. This module covers correlation analysis, linear and logistic regression, cost functions, and gradient descent—the optimisation algorithm powering most machine learning models. Understanding these concepts is essential for implementing and debugging ML algorithms effectively.

Correlation & Regression

Covariance and correlation (Pearson, Spearman). Correlation vs causation. Simple and multiple linear regression. Least squares method. R^2 and Adjusted R^2. Residual analysis.

Logistic Regression

Binary classification fundamentals. Sigmoid function and log-odds. Logistic regression for classification problems.

Optimisation

Cost functions and loss functions (MSE, Cross-entropy). Gradient descent algorithm. Learning rate and convergence. Batch vs stochastic gradient descent.

hello@digitaledify.ai

SECTION 5

Python Libraries for AI & Data

Module 1: NumPy - Numerical Computing

NumPy is the foundation of numerical computing in Python, providing efficient array operations essential for AI and data science. This module covers array creation, indexing, broadcasting, and linear algebra operations. Master NumPy to perform fast, memory-efficient computations on large datasets—a prerequisite for machine learning and deep learning.

1

Array Fundamentals

NumPy arrays vs Python lists. Array creation methods: zeros, ones, arange, linspace, random. Array shapes, reshaping, and transposing.

2

Indexing & Operations

Array indexing, slicing, and fancy indexing. Broadcasting rules and operations. Universal functions (ufuncs).

3

Linear Algebra

Dot products and matrix multiplication. Statistical operations: mean, median, std, variance, percentiles. Boolean indexing and sorting.

Module 2: Pandas - Data Manipulation & Analysis

Pandas is the premier library for data manipulation and analysis in Python. This comprehensive module covers Series and DataFrame operations, data cleaning, transformation, and aggregation. Learn to handle missing data, merge datasets, perform group operations, and conduct time series analysis—skills essential for preparing data for machine learning models.

01

DataFrames

Series and DataFrame creation. Reading CSV, Excel, JSON. Data exploration with head, tail, info, describe.

02

Data Selection

Indexing with loc, iloc, at, iat. Boolean indexing and conditional selection.

03

Data Cleaning

Handling missing data (isna, dropna, fillna). Detecting and removing duplicates. Data type conversions.

04

Transformations

Apply, map, lambda functions. GroupBy and aggregations. Pivot tables. Merging, joining, concatenating.

05

Time Series

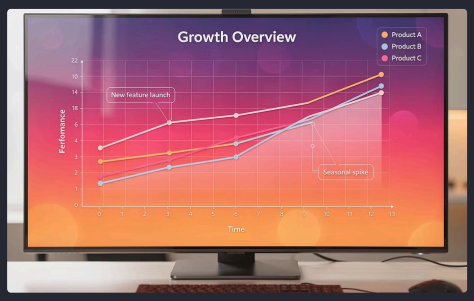
Time series analysis and date/time handling. Data input/output operations.

Module 3: Matplotlib - Static Data Visualisation

Matplotlib is Python's foundational plotting library, offering fine-grained control over visualisations. This module covers the architecture, basic plot types, customisation options, and advanced features like subplots and multiple axes. Master Matplotlib to create publication-quality figures for reports, papers, and presentations.

Basic Plots

Figure and Axes objects. Pyplot vs object-oriented approach. Line, scatter, bar, histogram, and pie charts.



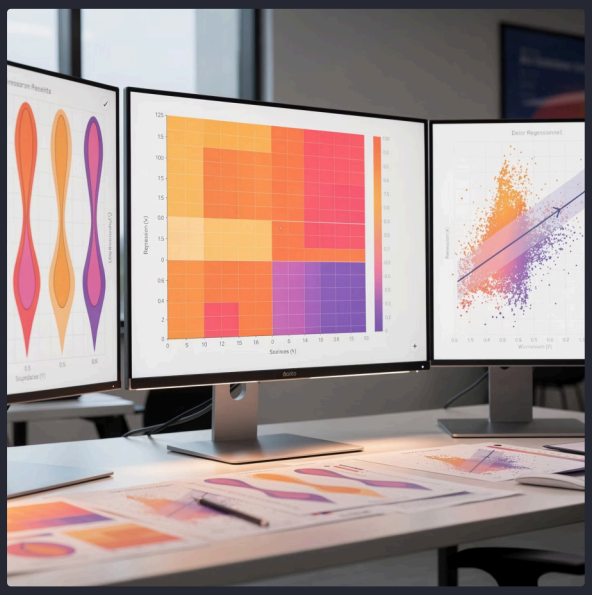
Customisation

Colors, markers, line styles. Labels, titles, legends. Ticks, grid lines, axis formatting. Annotations and text elements.

Advanced Features

Multiple subplots and grid layouts. Multiple y-axes and error bars. Box plots and violin plots. Matplotlib styles and themes.

Module 4: Seaborn - Statistical Data Visualisation



Seaborn builds on Matplotlib to provide beautiful, informative statistical visualisations with minimal code. This module covers distribution plots, categorical visualisations, relational plots, and matrix visualisations. Learn to create sophisticated multi-dimensional visualisations using facet grids and pair plots for exploratory data analysis.

Distribution Plots

histplot, kdeplot, boxplot, violinplot for visualising data distributions

Categorical Plots

barplot, countplot, pointplot for categorical data analysis

Relational Plots

scatterplot, lineplot, relplot for exploring relationships

Matrix Visualisations

heatmap, clustermap for correlation and pattern discovery

Regression & Pairs

regplot, lmplot, residplot, pairplot, jointplot for statistical analysis

Module 5: Plotly - Interactive Data Visualisation

Plotly enables interactive, web-based visualisations that engage audiences and facilitate data exploration. This module covers Plotly Express for rapid prototyping, interactive features like hover and zoom, 3D visualisations, geographic maps, and animated plots. Learn to create dashboards and export interactive visualisations for web applications and presentations.

Plotly Fundamentals

Plotly ecosystem (Express, Graph Objects, Dash). Interactive features: hover, zoom, pan, selection.

Advanced Visuals

Geographic maps and choropleths. Animated visualisations. Candlestick charts for financial data.

Chart Types

Line, scatter, bar charts. Box plots, violin plots, histograms. 3D scatter plots and surfaces.

Dashboards

Layout templates and customisation. Subplots and multi-plot dashboards. Exporting interactive plots.

SECTION 6

Machine Learning

Modules 1-2: Introduction & Mathematical Foundations

Machine Learning enables computers to learn from data without explicit programming. This section introduces ML types (supervised, unsupervised, reinforcement), learning mechanics, and the ML workflow. Mathematical foundations cover loss functions, gradient descent, and optimisation algorithms—the engines that power model training and enable machines to improve through experience.

ML Fundamentals

AI vs ML vs Deep Learning. Supervised, unsupervised, reinforcement learning. Models, parameters, hyperparameters.

Optimisation

Loss functions (MSE, MAE, Cross-Entropy). Gradient descent fundamentals. Learning rate and convergence.

Advanced Optimisers

Batch vs Stochastic vs Mini-batch. Adam, RMSprop, AdaGrad, Momentum. Local vs global minima.

Modules 3-4: ML Life Cycle & Regression

The ML life cycle provides a structured approach to building effective models. Learn problem definition, data collection, EDA, feature engineering, and preprocessing. Master linear regression—the foundation of predictive modelling—including cost functions, evaluation metrics, and diagnosing model performance through residual analysis.

Problem Definition

CRISP-DM workflow. Problem definition and success metrics. Data collection and quality assessment.

Data Preparation

EDA and statistical analysis. Handling missing data and outliers. Feature engineering and selection. Encoding categorical variables.

Data Splitting

Train-test-validation split strategies. Cross-validation techniques for robust evaluation.

Linear Regression

Simple and multiple regression. OLS and gradient descent. Evaluation metrics: R^2, RMSE, MAE. Residual analysis.

Modules 5-6: Regularisation & Classification I

Regularisation prevents overfitting by constraining model complexity. Learn polynomial regression, the bias-variance tradeoff, and regularisation techniques (Ridge, Lasso, Elastic Net). Classification introduces logistic regression and Naive Bayes—probabilistic methods for predicting categorical outcomes essential for many real-world applications.

Polynomial Regression & Regularisation

Polynomial regression and feature transformation. Bias-Variance Tradeoff. Overfitting vs Underfitting. Ridge (L2), Lasso (L1), and Elastic Net regularisation. Hyperparameter tuning with GridSearchCV and RandomizedSearchCV.

Classification Algorithms I

Binary vs Multi-class classification. Logistic Regression and sigmoid function. Log Loss (Binary Cross-Entropy). Decision boundaries. Multi-class strategies (One-vs-Rest, One-vs-One, Softmax). Naive Bayes Classifier and Bayes' Theorem. Types: Gaussian, Multinomial, Bernoulli.

Modules 7-8: Classification II & Model Evaluation

Tree-based methods offer interpretable, powerful classification. Learn Decision Trees, Random Forests, and Gradient Boosting (XGBoost, LightGBM, CatBoost). Support Vector Machines provide robust classification through margin maximisation. Master evaluation metrics (confusion matrix, precision, recall, ROC-AUC) and techniques for handling imbalanced datasets.

Tree-Based Methods

- Decision Trees: Entropy, Information Gain, Gini Impurity
- Tree construction (ID3, C4.5, CART) and pruning
- Random Forest and Bagging
- Gradient Boosting (XGBoost, LightGBM, CatBoost)

Support Vector Machines

SVM fundamentals. Margin and support vectors. Hard vs Soft margin. Kernel Trick: Linear, RBF, Polynomial, Sigmoid kernels.

Model Evaluation

Confusion Matrix. Accuracy, Precision, Recall, F1-Score. ROC curves and AUC-ROC. Handling imbalanced datasets with SMOTE.

Modules 9-10: Unsupervised Learning & Deployment

Unsupervised learning discovers hidden patterns in unlabelled data. Master clustering algorithms (K-Means, Hierarchical, DBSCAN, GMM) and dimensionality reduction (PCA, t-SNE). Reinforcement Learning introduces agent-environment interaction. Learn model deployment, creating APIs, monitoring, and responsible AI principles for production systems.

1

Clustering

K-Means, Hierarchical, DBSCAN, GMM. Clustering evaluation metrics.

2

Dimensionality Reduction

PCA and t-SNE. Curse of dimensionality. Feature extraction vs selection.

3

Reinforcement Learning

Agent, Environment, State, Action, Reward. MDP and Q-Learning. Exploration vs Exploitation.

4

Model Deployment

Model persistence (pickle, joblib, ONNX). APIs (Flask/FastAPI). Monitoring and drift detection. Responsible AI.

SECTION 7

Deep Learning & NLP

Modules 1-2: Neural Networks & Frameworks

Deep Learning extends machine learning with neural networks capable of learning hierarchical representations. This section covers neural network fundamentals, activation functions, backpropagation, and optimisation. Master PyTorch and TensorFlow—the industry-standard frameworks—for building, training, and deploying deep learning models at scale.

Neural Network Fundamentals

Artificial neural networks and perceptrons. Activation functions: sigmoid, tanh, ReLU, and variants. Forward propagation and backpropagation algorithms.

Optimisation & Regularisation

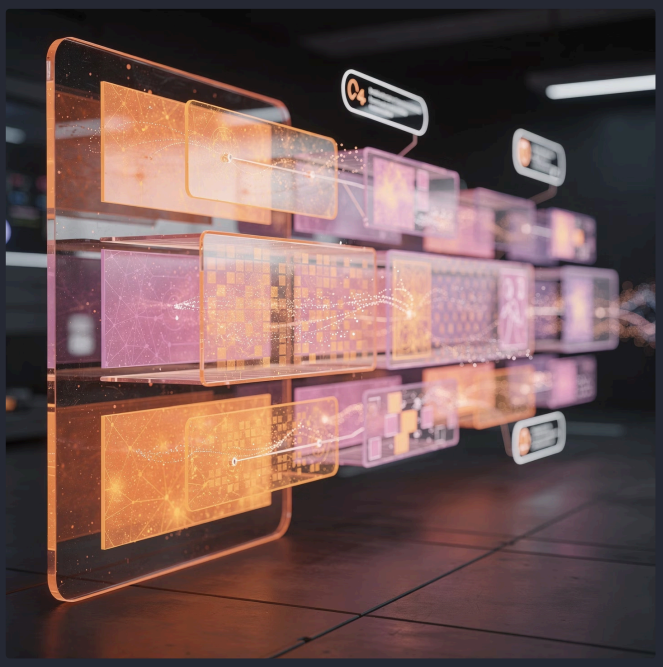
Gradient descent and optimisation techniques (SGD, Adam, RMSprop). Loss functions and performance metrics. Overfitting, underfitting, and regularisation.

Deep Learning Frameworks

PyTorch and TensorFlow introduction. Tensors, computational graphs, automatic differentiation. Building neural networks with PyTorch nn.Module. Data loading, batching, augmentation. Training loops, validation, checkpointing. GPU acceleration.

Modules 3-4: Convolutional Neural Networks

Convolutional Neural Networks revolutionised computer vision by automatically learning spatial hierarchies of features. This section covers convolution operations, pooling layers, and classic architectures (LeNet, AlexNet, VGG). Advance to modern architectures (ResNet, Inception, DenseNet), transfer learning, and object detection fundamentals for real-world vision applications.



CNN Fundamentals

Convolution operation (filters, strides, padding). Pooling layers and feature extraction. Classic architectures: LeNet, AlexNet, VGG. Building CNNs from scratch. Visualising CNN features and receptive fields.

Advanced CNNs

Advanced architectures: ResNet, Inception, DenseNet. Batch normalisation and dropout. Transfer learning and fine-tuning. Data augmentation techniques. Object detection basics (R-CNN concepts). Image segmentation fundamentals. Model optimisation and compression.

Modules 5-6: Recurrent Neural Networks

Recurrent Neural Networks process sequential data by maintaining internal state. This section covers vanilla RNNs, the vanishing gradient problem, and solutions through LSTM and GRU architectures. Learn bidirectional and stacked RNNs, sequence-to-sequence models, and encoder-decoder architectures for tasks like machine translation and time series prediction.

1 RNN Fundamentals

Sequential data and temporal dependencies. Vanilla RNNs and vanishing gradient problem. Backpropagation Through Time (BPTT).

2 LSTM & GRU

Long Short-Term Memory (LSTM) networks. Gated Recurrent Units (GRU). LSTM gates and cell states. Training strategies for RNNs.

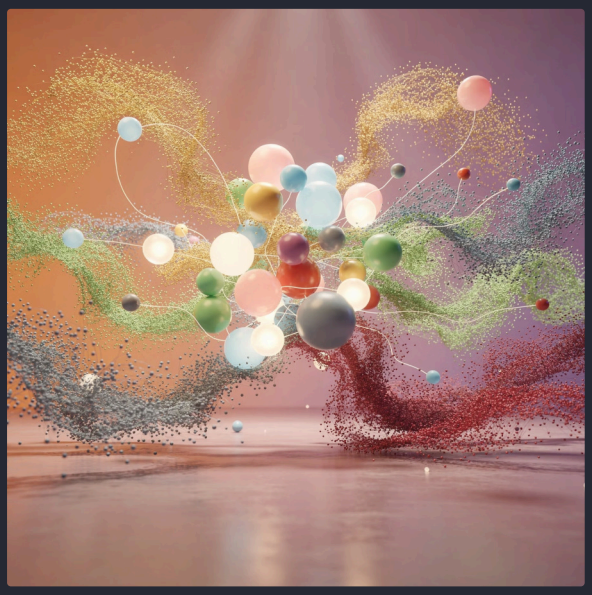
3 Advanced RNNs

Bidirectional RNNs. Stacked and deep RNN architectures. Sequence-to-sequence models. Encoder-decoder architectures.

4 Training Strategies

Teacher forcing and inference strategies. Handling variable-length sequences. Advanced training techniques for RNNs.

Modules 7-8: Natural Language Processing



Natural Language Processing enables machines to understand and generate human language. This section covers text preprocessing, word embeddings (Word2Vec, GloVe, FastText), and language modelling. Master text classification and sentiment analysis using CNNs, RNNs, and LSTMs—essential skills for building chatbots, recommendation systems, and content analysis tools.

NLP Foundations

Text preprocessing and tokenisation. Bag of Words and TF-IDF. Word embeddings: Word2Vec, GloVe, FastText. Training custom embeddings. Language modelling and N-gram models.

Text Classification

Text classification pipeline design. Feature extraction for text. CNN and RNN for text classification. Sentiment analysis techniques. Multi-class and multi-label classification. Handling imbalanced text datasets.

Modules 9-10: Sequence Models & Information Extraction

Sequence-to-sequence models enable tasks like machine translation and text summarisation. This section covers encoder-decoder architectures, beam search, and evaluation metrics (BLEU, ROUGE). Named Entity Recognition and information extraction enable structured data extraction from unstructured text—critical for building knowledge bases and intelligent search systems.

01

Seq2Seq Models

Sequence-to-sequence architecture deep dive. Machine translation fundamentals. Encoder-decoder models.

02

Decoding Strategies

Handling long sequences. Beam search and decoding strategies. Evaluation metrics: BLEU, ROUGE.

03

Named Entity Recognition

NER fundamentals. Part-of-speech tagging. Sequence labelling with RNNs and LSTMs. BiLSTM-CRF for NER.

04

Information Extraction

Information extraction techniques. Relation extraction. Building domain-specific NER systems. Evaluation metrics.

SECTION 8

Generative AI & Agentic AI

Modules 1-2: Foundations & Prompt Engineering

Generative AI represents the frontier of artificial intelligence, enabling machines to create human-like text, images, and code. This section introduces Large Language Models (LLMs), transformer architecture, and major models (GPT, Claude, Gemini, DeepSeek). Master prompt engineering and context design to unlock LLM capabilities whilst minimising hallucinations and optimising costs.

LLM Fundamentals

Large Language Models and transformer architecture. Comparing GPT, Claude, Gemini, DeepSeek. Evolution from GPT-1 to 2026 frontier models.

Multimodal Prompting

Text, image, and audio prompting. Domain-specific prompt design for specialised applications.

Model Selection

LLM architecture and tokenisation. Model selection for different use cases. Cost optimisation strategies.

Prompt Engineering

Advanced prompt engineering techniques. Context engineering and design. Reasoning mode optimisation. Reducing hallucinations. Zero-shot, few-shot, chain-of-thought prompting.

Modules 3-4: LLM APIs & RAG Systems

LangChain 1.0 provides a unified framework for building LLM applications across multiple providers. This section covers API integration, create_agent abstraction, and middleware systems. Retrieval-Augmented Generation (RAG) combines LLMs with external knowledge bases, enabling accurate, up-to-date responses. Master vector databases and production RAG pipelines for enterprise applications.

LLM APIs & LangChain 1.0

OpenAI, Anthropic, Google, and DeepSeek APIs. LangChain 1.0 fundamentals and create_agent abstraction. Middleware systems for customisation. Multi-provider integration. Streaming and batching. Function calling and structured outputs. Cost-optimised pipelines.

RAG & Vector Databases

Vector databases: ChromaDB, Pinecone, Qdrant. Building production RAG pipelines. Agentic RAG and self-improving retrieval. MCP-Enhanced RAG. Embedding strategies. Hybrid search (semantic and keyword). Document processing at scale. Hallucination reduction techniques.

Modules 5-6: Production Deployment & Agentic AI

Production deployment requires robust interfaces, security, and governance. Learn Streamlit and Gradio for rapid prototyping, LangGraph Platform for scalable deployment, and EU AI Act compliance. Agentic AI introduces autonomous systems that plan, reason, and act. Master LangChain 1.0 Agents, Model Context Protocol (MCP), and tool integration for building intelligent assistants.

Production Deployment

Streamlit and Gradio interfaces. LangGraph Platform deployment. Cost optimisation strategies. AI governance and EU AI Act compliance. API security and rate limiting. Monitoring and observability. Scaling strategies. Enterprise tool integration.

Introduction to Agentic AI

Agentic AI fundamentals: plan, reason, act. LangChain 1.0 Agents with middleware. Model Context Protocol (MCP). Tool integration patterns. Enterprise adoption and use cases. Agent architectures and design patterns.

Modules 7-8: LangGraph & Advanced Workflows

LangGraph 1.0 enables sophisticated AI workflows through graph-based state management. This section covers architecture, node caching, pre/post hooks for guardrails, and building production workflows. Master advanced patterns including parallel execution, conditional routing, iterative refinement, and type-safe streaming for complex multi-stage AI systems.

LangGraph 1.0 Fundamentals

- LangGraph 1.0 architecture and design
- State management and graph-based logic
- Node caching for development efficiency
- Pre/Post hooks for guardrails and validation
- Building AI workflows for production use cases

1

Parallel Execution

Parallel execution with deferred nodes for efficient processing

2

Conditional Routing

Conditional routing and decision trees for dynamic workflows

3

Iterative Refinement

Iterative refinement loops for quality improvement

4

Type-Safe Streaming

Type-safe streaming for real-time applications

5

Production Workflows

Essay evaluation, customer feedback routing, multi-stage approval, quality-gated content generation

Modules 9-10: Persistence & Production Systems

Production agentic systems require durable state management, human oversight, and robust security. This final section covers persistence with PostgreSQL and Redis, human-in-the-loop implementations, and multi-day workflow support. Master LangGraph Platform deployment, multi-agent system design, Google A2A Protocol, LangSmith observability, MCP security, and compliance for building enterprise-grade autonomous AI systems.

Persistence & HITL

- Durable state management
- Built-in persistence (PostgreSQL, Redis)
- Human-in-the-loop (HITL) implementations
- Multi-day workflow support
- Enterprise compliance and audit trails
- Restart and failure recovery

Production Agentic Systems

- LangGraph Platform deployment
- Multi-agent system design
- Google A2A Protocol for agent-to-agent communication
- LangSmith observability and monitoring
- MCP security model
- Prompt injection prevention
- Compliance and audit trails
- Agent guardrails and safety

Congratulations! You've completed the comprehensive AI & Data Science Master's programme. You now possess the skills to build sophisticated AI systems—from foundational Python to advanced Generative AI and Autonomous Agents. The industry awaits your expertise.