Coincent
Learning Drives Excellence

Coincent 3 Year Program Curriculum
Data Science Domain

Partnered by

Microsoft

WORLD RISK & GOVERNANCE

Empowering Learners,
Accelerating Careers.

coincent.ai

ABOUT COINCENT

Coincent offers a 3-Year Program that is a well-structured, career-focused initiative designed to equip students with practical skills, real-world experience, and strong placement support. The program is tailored to ensure progressive learning and career readiness across three year phases.

Why It's Unique

- Only one batch per year with limited seats (150 students) per Domain to maintain quality.
- Prepares students step-by-step to become job-ready by graduation.

DETAILED ABOUT COINCENT 3 YEAR DATA SCIENCE PROGRAM

"Data Science Program at Coincent – Learn by Doing"

Data science is the field of extracting meaningful insights from data using techniques from statistics, computer science, and machine learning. It involves collecting, cleaning, analyzing, and visualizing data to help businesses and organizations make data-driven decisions. Data

scientists use tools like Python, R, SQL, and data visualization platforms to uncover trends and solve real-world problems.

Key Points:

1. **Data Analysis & Visualization** – Understanding data through charts, dashboards, and statistics.
2. **Machine Learning** – Building predictive models using algorithms.
3. **Data Cleaning & Preparation** – Making raw data usable and accurate.
4. **Decision Making** – Turning insights into strategies and actions.

3-Year Program Structure Breakdown

TABLE OF CONTENTS

Year 1 :- Industrial Training

Chapter 1 – Introduction to Data Science

1.1 Introduction to Data Science

- What is Data Science?
- Applications across industries (healthcare, finance, marketing, etc.)
- Difference between Data Science, AI, ML, and Big Data

Key Benefits:

- Builds a strong foundation in understanding how to turn raw data into valuable insights.
- Equips you with a modern, high-demand tech skill used across industries.
- Shows the real-world impact of data science — from predicting diseases to detecting fraud.
- Opens diverse career paths by applying skills to multiple sectors.
- Clarifies roles and technologies, helping you choose a specialized learning or career track.
- Helps you understand how these fields connect and complement each other in solving complex problems.

1.2 Data Science Pipeline

- Problem definition
- Data collection

- Data cleaning & preprocessing
- Exploratory Data Analysis (EDA)
- Feature Engineering
- Model Building
- Evaluation & Deployment

1.3 Data Science Career

- Roles: Data Analyst, Data Engineer, Data Scientist, ML Engineer
- Tools: Python, R, SQL, Tableau, Spark
- Career paths and skill roadmap

Key Benefits and Output:

Concept	Key Benefits	Outcomes
Problem Definition	Aligns technical work with goals Clarifies project direction	Clear problem statement Defined success metrics
Data Collection	Raw dataset gathered Awareness of legal/data issues	Raw dataset gathered Awareness of legal/data issues
Data Cleaning & Preprocessing	Improves data quality Makes data usable for modeling	Cleaned, formatted data Handled missing/outlier values
Exploratory Data Analysis (EDA)	Reveals patterns & anomalies Builds data intuition	Graphs, insights, statistics Feature/model selection guidance
Feature Engineering	Boosts model performance Applies domain knowledge	New/transformed features Better model inputs
Model Building	Teaches algorithm use Hands-on with ML libraries	Trained predictive model Knowledge of tuning and optimization
Evaluation & Deployment	Assesses real-world value Prepares model for use	Metrics reports (accuracy, etc.)Model deployed as API or app

Key Benefits and Output:

Aspect	Key Benefits
Career Path Clarity	Choose a role aligned with your interests and strengths
Skill Roadmap	Progress from beginner to expert in a structured way
Industry Alignment	Learn in-demand skills used in real-world jobs
Cross-role Understanding	Know how different data roles collaborate in teams
Lifelong Learning	Identify what to learn next to stay relevant in the data field

Chapter 2 – Introduction to Python for Data Science

2.1 Introduction to Python

- Why Python for Data Science?
- Python installation (Anaconda/Jupyter/VSCode)

2.2 First Program

- print(), understanding code structure

2.3 Comments

- Single-line, multi-line, docstrings

2.4 Data Types

- 2.4.1: Numbers, Strings
- 2.4.2: Lists, Tuples, Sets, Dictionaries

2.5 Operators

- 2.5.1: Arithmetic, Comparison
- 2.5.2: Logical, Bitwise, Membership

2.6 Variables

- 2.6.1: Variable declaration, rules
- 2.6.2: Type casting

2.7 Syntax

- Python indentation, blocks, line continuation

2.8 Working with Numbers

- math module, rounding, random numbers

2.9 Working with Strings

- 2.9.1: Basic operations, slicing
- 2.9.2: String methods, formatting

2.10 Date & Time

- Working with datetime module

2.11 Conditional Statements

- 2.11.1: if, else
- 2.11.2: elif, nested conditions

2.12 For Loop

- 2.12.1: Basic loop, range()
- 2.12.2: Nested loops, break, continue

2.13 While Loop

- Loop conditions and control

2.14 Lists, Tuples, Sets

- 2.14.1: List operations
- 2.14.2: Tuple properties
- 2.14.3: Set operations

2.15 Dictionaries

- 2.15.1: Keys, values
- 2.15.2: Methods, nesting

2.16 Functions

- 2.16.1: Defining functions, parameters, return, scope

2.17 NumPy

- 2.17.1: Arrays, creation, indexing
- 2.17.2: Array operations, broadcasting
- 2.17.3: Statistical & mathematical functions

2.18 Matplotlib

- 2.18.1: Line & bar charts
- 2.18.2: Histograms, pie charts, customization

2.19 Pandas

- 2.19.1: Series, DataFrame creation, indexing
- 2.19.2: Data cleaning, merging, groupby, filtering

Key Benefits and Output:

Point	Benefit	Output / Skill Gained
1. Strong Foundation in Python	Learn syntax, variables, data types, operators, and code structure	Ability to write clean, readable, and error-free Python scripts
2. Problem Solving with Conditions & Loops	Master if, else, for, and while loops	Build logic-based applications with flow control and automation
3. Working with Core Data Structures	Understand lists, tuples, sets, dictionaries	Store, access, and manipulate structured data effectively
4. Function Design & Code Reuse	Learn function creation, scope, parameters, return values	Write modular, reusable, and maintainable code
5. Numerical Computing with NumPy	Use arrays, indexing, and statistical functions	Perform fast numerical operations and matrix computations
6. Data Visualization with Matplotlib	Create line, bar, pie charts, and customize plots	Visually analyze trends, comparisons, and distributions
7. Data Manipulation with Pandas	Work with Series/DataFrames, filtering, groupby, merging	Clean, transform, and analyze large datasets efficiently
8. Real-World Programming Tools	Use datetime, math, random, and data formatting methods	Build feature-rich, real-world-ready programs and scripts

Chapter 3 – Mathematics for Data Science

3.1 Overview of Math Requirements

- Algebra, statistics, probability, linear algebra

3.2 Introduction to Statistics

- Mean, median, mode

3.3 Measure of Spread

- Variance, standard deviation, IQR, range

3.4 Probability

- Classical, conditional, joint and marginal probability

3.5 Conditional Probability

- Bayes Theorem, independence

3.6 Data Scientist vs Statistician

- Skills, tools, responsibilities

3.7 Data Preprocessing

- Normalization, standardization, handling missing values

Key Benefits and Output:

Point	Benefit	Output / Skill Gained
1. Math & Stats Foundation	Learn algebra, statistics, variance, and probability concepts	Gain analytical skills to interpret data and support ML algorithms
2. Probability & Decision Making	Understand conditional probability and Bayes Theorem	Make data-driven predictions and handle uncertainty in real-world scenarios
3. Data Readiness Skills	Learn data preprocessing techniques like normalization and handling missing values	Clean and prepare raw datasets for accurate and efficient analysis

Chapter 4 – Introduction to Machine Learning

4.1 Introduction to ML

- What is ML, differences with AI & DL

4.1.1 Types of ML Algorithms

- Supervised, Unsupervised, Reinforcement

4.2 Steps in Building ML Model

- Define problem → Collect Data → Clean Data → Train → Test → Evaluate

4.3 Linear Regression

- Concept, simple & multiple regression, evaluation metrics

4.4 Logistic Regression

- Binary classification, sigmoid function

4.5 K-Nearest Neighbors (KNN)

- Distance metric, choosing K, overfitting/underfitting

4.6 Naive Bayes

- Bayes theorem, spam classification

4.7 Clustering Overview

- Types and applications

4.8 K-Means Clustering

- Centroid selection, elbow method

4.9 Hierarchical Clustering

- Dendrograms, Agglomerative vs Divisive

4.10 Dimensionality Reduction

- Purpose and techniques

4.10.1 Principal Component Analysis (PCA)

- Variance, eigenvalues, dimensional reduction

4.10.2 Linear Discriminant Analysis (LDA)

4.11 Supervised vs Unsupervised Learning

- Characteristics and examples

4.12 Semi-Supervised Learning

- Real-world examples, pseudo-labeling

4.13 Reinforcement Learning

4.13.1: Agents, environment, reward system

4.13.2: Q-learning, exploration vs exploitation

4.14 Data Science vs Machine Learning

- Comparison table and use cases

Key Benefits and Output

Point	Benefit	Output / Skill Gained
1. Clear Understanding of ML Foundations	Learn the core concepts of Machine Learning, how it differs from AI & DL, and types of ML algorithms	Ability to distinguish between supervised, unsupervised, and reinforcement learning with real-world use case awareness
2. End-to-End ML Model Development	Explore the entire workflow: problem definition, data collection, cleaning, model training, testing & evaluation	Gain hands-on experience in building complete ML pipelines using structured approaches
3. Proficiency in Core ML Algorithms	Learn and implement key algorithms such as Linear & Logistic Regression, KNN, Naive Bayes, and clustering methods	Apply the right algorithm based on data type and problem domain (e.g., classification, prediction, grouping)
4. Dimensionality & Advanced Techniques	Understand the role and implementation of PCA, LDA, and Reinforcement Learning (Q-learning)	Develop the ability to handle high-dimensional data and create learning agents using reward-based systems
5. Strategic Thinking & Career Relevance	Compare ML vs Data Science and understand learning types, pseudo-labeling, and algorithm characteristics	Build decision-making skills on model selection, and prepare for roles in data science, ML engineering, or applied AI

Chapter 5 – Deep Learning and Data Mining

5.1 Deep Learning

- Neural networks, perceptrons, backpropagation

5.2 TensorFlow and Keras

- Building a neural network, training, evaluating

5.3 Data Mining

- Patterns, knowledge discovery, CRISP-DM

5.3.1 Preprocessing in Data Mining

- Feature selection, discretization, transformation

5.4 Natural Language Processing

- Tokenization, stemming, lemmatization

5.4.1 Preprocessing in NLP

- Stopword removal, word embeddings (TF-IDF, Word2Vec)

Key Benefits and Output:

Point	Benefit	Output / Skill Gained
1. Comprehensive ML Foundations	Understand what Machine Learning is and how it differs from AI and DL	Build clarity on core ML concepts, types (supervised, unsupervised,reinforcement), and real-world relevance
2. End-to-End Model Building	Learn the complete ML workflow from problem definition to evaluation	Gain the ability to build, train, test, and improve ML models using structured steps
3. Master Core ML Algorithms	Deep dive into Linear & Logistic Regression, KNN, Naive Bayes, and Clustering methods	Implement predictive and classification models with practical use cases like spam filtering or customer segmentation
4. Explore Advanced ML Concepts	Learn dimensionality reduction techniques like PCA & LDA, and fundamentals of Reinforcement Learning	Solve high-dimensional problems and build intelligent agent-based systems using Q-learning principles
5. Practical Comparisons & Application Thinking	Compare Supervised, Unsupervised, Semi-Supervised Learning and ML vs Data Science	Make informed decisions on choosing the right ML approach based on dataset type and business problem

Year 2 – Application & Project Phase

– Year 2 is full of hands-on-experience on 8 live projects –

Hierarchical Clustering
Hierarchical Clustering is an unsupervised learning technique used to group similar data points into clusters based on distance metrics. It builds a tree-like structure (dendrogram) to show how clusters are merged or split. The algorithm does not require the number of clusters to be specified in advance. It is useful in market segmentation, social network analysis, and gene classification. Tools like **Scikit-learn** and **SciPy** are commonly used.

1. **ChatBot**
   A ChatBot is an AI-powered tool designed to simulate human-like conversations using natural language processing (NLP). It can be rule-based or machine learning-based, handling tasks like customer support or appointment scheduling. ChatBots learn from data and improve over time, making them increasingly accurate. Libraries like **NLTK**, **spaCy**, and frameworks like **Rasa** or **Dialogflow** are widely used. It enhances user engagement and automation.

2. **Linear Discriminant Analysis (LDA)**
   LDA is a supervised dimensionality reduction technique that maximizes class separability by projecting data into lower dimensions. It is commonly used in classification tasks to improve performance and interpretability. LDA assumes normally distributed classes and equal class covariances. It's effective in fields like face recognition and medical diagnosis. Tools like **Scikit-learn** offer built-in LDA support.

3. **Hate Speech Detection**
   Hate Speech Detection is a text classification task that identifies and filters offensive or harmful content online. It involves preprocessing text, feature extraction (TF-IDF or embeddings), and training classification models like Logistic Regression or LSTM. It's vital for maintaining safe online platforms and social media moderation. Datasets like Twitter Hate Speech and tools like TensorFlow, scikit-learn, and BERT are commonly used.

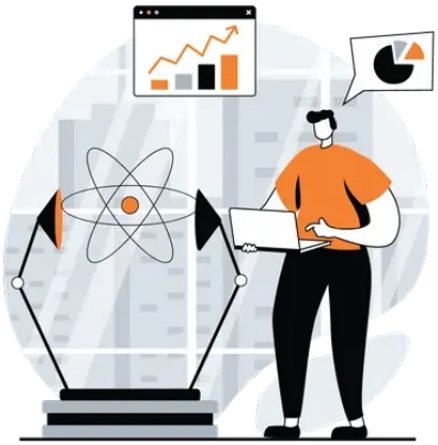
4. **SMS Spam Classification**
   SMS Spam Classification is a binary text classification project that identifies whether a message is spam or ham (not spam). It involves preprocessing the text (removing stop words, tokenization), feature extraction using TF-IDF orCountVectorizer, and training models like Naive Bayes or Logistic Regression. This project helps automate spam detection in messaging systems. Evaluation metrics like accuracy, precision, and recall are used to assess performance.
   Tools used include **Scikit-learn, NLTK,** and **Pandas**.

5. **Surprise Housing Case Study**
   The Surprise Housing Case Study aims to build a regression model to predict house prices based on features like location, size, and amenities. It involves exploratory data analysis (EDA), feature engineering, multicollinearity checks, and model building using linear regression. The goal is to provide accurate price predictions to aid decision-making in real estate. The project also focuses on model interpretability and performance evaluation.

   Tools used include **Pandas, Matplotlib, Seaborn, and Scikit-learn**.

6. Credit EDA (Exploratory Data Analysis)

Credit EDA involves analyzing credit-related datasets to uncover patterns, detect anomalies, and understand borrower behavior. It includes univariate and bivariate analysis, missing value treatment, and correlation study. Visualizations like histograms, heatmaps, and box plots help reveal relationships between features and credit risk. The insights guide the development of credit scoring



Year 3 – Placement & Internship Phase:

1. Guaranteed Internship Phase

- In Year 3, Coincent guarantees an internship with partner companies. The internship includes a formal Internship Offer Letter and a Completion Certificate upon successful completion.
- This is part of their "Industrial Training + Internship" model — It covers live classes, mentorship, and project work, but the internship phase itself is completely complimentary

2. Structured Placement Preparation

- Coincent supports students in portfolio-building with multiple completed projects (typically around 8) and Microsoft-aligned certifications .
- Coincent provides mock interviews, resume reviews, and training for HR and technical rounds — all aimed at preparing you for real-world hiring.

3. Final Take

- Coincent's 3rd year transforms theory into practical experience through a guaranteed internship, builds a robust credentials portfolio, and equips you with placement-ready skills via mock interviews and resume prep. If you're in your 4th year, this phase sets you on a clear trajectory from "training" to "hired."

Step Into Top Tech Roles

The leading and high-demand roles in the Data Science Field along with a brief description of each:

1. Data Scientist

- **Focus:** Analyzing data, building predictive models, extracting insights.
- **Skills:** Python/R, statistics, machine learning, data visualization.
- **Tools:** Pandas, Scikit-learn, TensorFlow, SQL, Jupyter, Tableau.

2. Data Analyst

- **Focus:** Data querying, reporting, and visualization to support business decisions.
- **Skills:** SQL, Excel, data storytelling, basic statistics.
- **Tools:** Power BI, Tableau, Python (Pandas/Matplotlib), Google Data Studio.

3. Machine Learning Engineer

- **Focus:** Designing, training, and deploying ML models in production.
- **Skills:** Deep learning, model optimization, software engineering.
- **Tools:** TensorFlow, PyTorch, MLflow, Docker, Kubernetes.

4. Data Engineer

- **Focus:** Building and maintaining data pipelines and architecture.
- **Skills:** ETL processes, big data frameworks, database design.
- **Tools:** Apache Spark, Hadoop, SQL/NoSQL, Airflow, AWS/GCP.

5. AI/Deep Learning Engineer

- **Focus:** Developing neural network models for vision, NLP, etc.
- **Skills:** Deep learning, computer vision, natural language processing.
- **Tools:** PyTorch, TensorFlow, Keras, Hugging Face, OpenCV.

6. Business Intelligence (BI) Developer

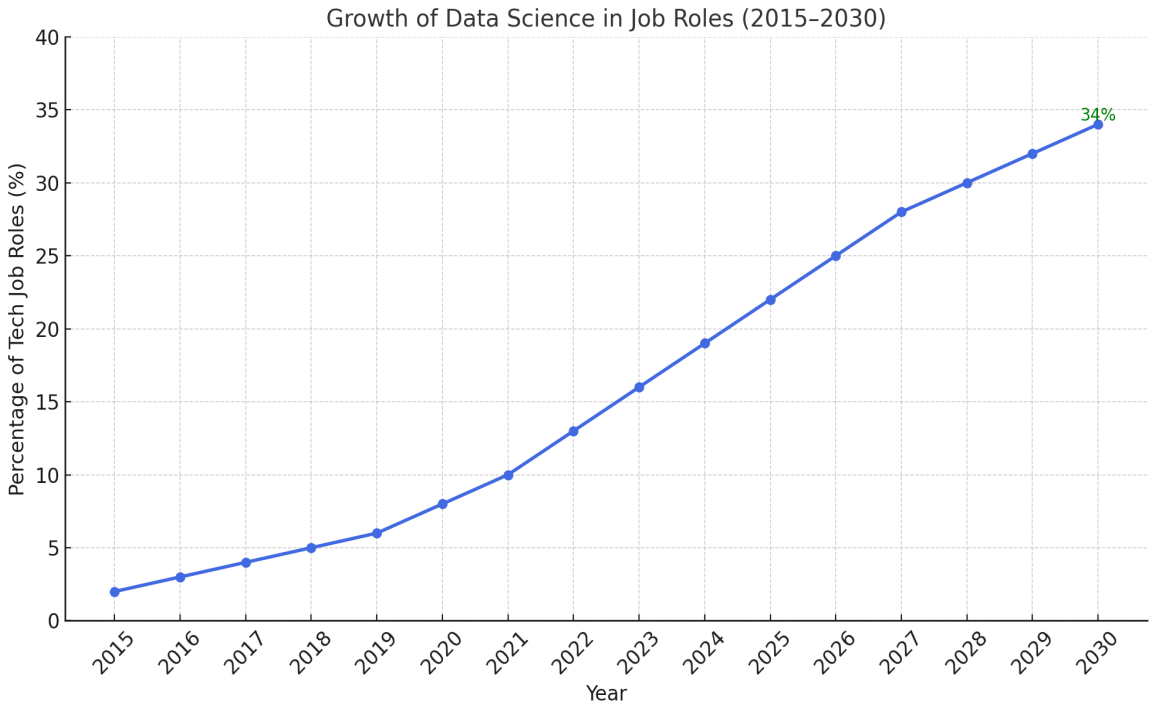
- **Focus:** Transforming raw data into actionable insights using dashboards.
- **Skills:** Data modeling, data visualization, database querying.
- **Tools:** Power BI, Tableau, Looker, DAX, SQL.

7. Data Architect

- **Focus:** Designing the overall structure and integration of data systems.
- **Skills:** Data modeling, systems architecture, data governance.
- **Tools:** ER/Studio, AWS Redshift, Azure Synapse, Snowflake.

8. MLOps Engineer

- **Focus:** Automating ML workflows and managing model lifecycle in production.
- **Skills:** DevOps + ML, model deployment, CI/CD pipelines.
- **Tools:** Docker, Kubernetes, MLflow, TensorFlow Serving, Git.



The growth of Data Science roles as a percentage of overall tech job roles from 2015 to 2030 (projected). As you can see:

- In 2015, data science roles made up around 2% of tech jobs.
- By 2025, the share is expected to reach 22%, and by 2030, it's projected to be around 34% of all tech roles.