# 

# Program Outline

# **Data Science & Machine Learning**

# **Career Path**

# Batch 02

Course Instructors

**Md. Aksadur Rahman**

Adjunct Lecturer,

United International University (UIU)

Data Science Consultant,

Bdjobs.com Ltd

Lead Trainer of EDGE Projects,

ICT Division, BCC

**Sabbir Hossain Rossi**

Senior Business Intelligence Analyst,

German-based Retail Company  
&  
Former Data Scientist Team Lead,

Celloscope & DOER Services Limited

**Araf Mustavi**

Analyst, IDT Operations

BAT Bangladesh

Ex - Machine Learning Engineer,

ACI Limited

**Tools & Technology**

**Database Fundamentals & SQL (BigQuery) for Data Scientists,**

**Python Fundamentals,**

**Statistics with Python for Data Scientists,**

**Machine Learning with Python,**

**MLOps,**

**Deep Learning with Python**

**\*\*\***

**How to Effectively Apply for a Data Scientist//ML Engineer Position?**

**GitHub & PortfolioDevelopment**

**Interview Skill Development**

**Freelancing Guidance**

**Python Fundamentals**

**Week 01**

* **Class 1: Basics of Python Programming Language**
* Introduction to Python Programming Language.
* Introduction to Google Colab & Jupyter Notebook for Writing and Executing Python Codes.
* Getting used to the User Interface (UI) of Jupyter Notebook/Google Colab.
* How to use AI for writing Python Codes in Colab?
* Python Data Types: Numeric, Strings, Boolean, List, Dictionary, Tuple, Set, None.
* Iterables & Mutability; List vs. Tuples.
* **Class 2: Variables, Type Conversion, Operators & Methods**
* Assigning & Overwriting Variables; Naming Convention for Variables.
* Type Conversion.
* Operators: Arithmetic & Logical/Comparison.
* Numeric & String (Text) Methods.
* Difference between Methods and Functions
* String Indexing & Slicing.
* String Methods.
* F String.
* **Class 3: Conditional Statement, User Defined Functions & Iteration**
* IF & ELIF Statements.
* How to Define/Create a Function in Python?
* How to Create a Function with Parameter/Argument(s)? Function within a Function.
* Palindrome.
* Use Conditional Statements and Functions Together.
* Using FOR & WHILE Loops.
* Combining Conditional Statements & Loops.
* Combining Conditional Statements, Functions & Loops.
* How to Iterate over a Dictionaries.
* Error Handling in Python.
* Use of Libraries in Python.

**\*\*\*** By the end of the Python Fundamentals module, students will have built a solid foundation in core Python programming concepts and practices. This is critical for anyone pursuing a career in data science or as a machine learning (ML) engineer, as Python is the primary language used in these fields.  
  
**Outcomes:**

* Mastery of Data Types.
* Understanding Python Basics.
* Variables and Type Operations.
* String Handling and Methods.
* Control Flow and Logic.
* Functions and Modular Programming.
* Iterative Processing.

**Database Fundamentals & SQL for Data Scientists**

**Week 02**

* **Class 4: Fundamentals of DBMS-I**
* What is a Relational Database Management System?
* ACID Property.
* How data is stored in a relational database? Concept of Normalization.
* What is a NoSQL Database and BASE Property?
* Difference between Relational and NoSQL Database.
* Which one should you choose in which case?
* For Data Science Workflow or, BI Solutions what should you choose to store your data?
* What is BigQuery and what advantage does it provide?
* Introduction to Google BigQuery and Query using BigQuery SQL.
* Accessing Google BigQuery Public Dataset.
* What is Dataset, Table & Schema in BigQuery?
* Getting Familiar with SQL Queries in Cloud Environment.
* Database vs. Data Warehouse vs. Data Lake/Lakehouse.
* OLTP vs. OLAP.
* **Class 5: Fundamentals of DBMS-II**
* Installing MySQL Locally.
* MySQL Workbench UI Tour.
* Concepts of Database, Schema, Table and Fields.
* Types of SQL Commands: DDL, DML, DCL, TCL, DQL.
* What are the Commands under each of these categories and what does each of these commands do?
* As a Data Analyst & Data Scientist which Commands do you only need to Master?
* **Constraints:** Primary Key, Foreign Key, Not Null, Unique, Check, Default.
* **Data Types:** DATE, DATETIME, VARCHAR, INT, NUMBER, FLOAT, Auto Increment.
* **Class 6: Designing a Relational Database in MySQL & Bulk Insertion**
* Designing and Creating Database, Schema, Tables.
* Masterclass on DDL, DML, TCL & DQL Commands/Statements.
* Bulk Insert of Data in MySQL.
* Error Handling during Bulk Insertion.
* Concepts of Indexing & Partitioning to make a Database Faster.

**Week 03**

* **Class 7: Exploratory Data Analysis using SQL**
* Basic Queries: SELECT, FROM, WHERE, LIKE, ILIKE, IN, DISTINCT, BETWEEN, GROUP BY, ORDER BY, LIMIT, OFFSET, ALIAS.
* Aggregate Functions: COUNT, SUM, AVG, MIN, MAX.
* Difference between WHERE and HAVING Clause.
* Some built-in Functions: EXTRACT, DATE\_PART, TO\_DATE, TO\_CHAR.
* CASTING, SUBSTRING, POSITION, COALESCE, NULLIF.
* **Class 8: JOINING, UNION & CASE WHEN**
* SQL Join: Left, Right, Inner & Full Join.
* Be aware of **Cross Join!**
* UNION.
* **SQL Code Order of Execution.**
* Wide use of **CASE WHEN** Statement during data Cleaning, Analysis & Feature Engineering.
* **Class 9: Subqueries, CTEs, View & RFM Segmentation**
* Use of Subqueries & CTEs in long SQL Codes.
* Segmenting Customers based on their Recency, Frequency & Monetary value using SQL.
* Use Case of Views; How to Create Views?
* Between Subqueries & CTEs one is more efficient?

**Week 04**

* **Class 10: Use of Window Functions**
* Window Functions, the most widely used SQL Commands used by Data Analysts & Data Scientists.
* Window Functions: RANK, DENSE\_RANK, ROW\_NUMBER, LEAD, LAG, FIRST VALUE, AGGREGATE WINDOW FUNCTION, FRAME SPECIFICATION, WINDOW CHAINING.
* Measuring Rolling/Moving Average, MoM % Change; Removing Duplicates and Identifying Top Performers using Window Functions.
* **Class 11: Cohort Analysis & Colab (Python) to BiqQuery Integration**
* Cohort Analysis using Retail Data.
* Measuring Customer Lifetime Value, Retention, and Churn Rate.
* Stored Procedure.
* Google Colab to BigQuery Integration.

**\*\*\*** By the end of this module, Students would be able to confidently perform any sort of Data Analysis and Reporting for different Departments in any Organisations using SQL. They would be well-equipped to understand any complex queries; validating data and help any Business by generating important KPIs.

They would also be able to help Businesses by Performing Complex Analyses like Cohort and RFM Segmentation. They would understand the importance of Cloud Computing, VIEW, Stored Procedures which would help them when they would be working on Python/Power BI/Tableau, etc. with Database/Data Warehouse as Data Sources.

**Data Analysis using NumPy, Pandas**

**Week 04**

**Class 12: Data Analysis using NumPy**

* A brief introduction
* Jupyter notebook Installation & exploring.
* Google Colab
* NumPy arrays, arrange, linspace
* Array methods and attributes.
* Indexing, slicing
* Broadcasting
* Boolean masking
* Arithmetic Operations
* Universal Functions

**Week 05**

**Class 13: Data Analysis using Pandas -1**

* Pandas Introduction.
* Pandas Data Structures - Series
* Pandas Data Structures – DataFrame
* Creating DataFrame
* Grab data (column wise)
* Grab data (raw wise)
* Grabbing an element or a sub-set of the dataframe
* Adding new column
* Deleting the column
* Boolean mask
* reset\_index(), set\_index(), head(), tail(), info(), describe()

**Class 14: Data Analysis using Pandas -2**

* Handling Missing Data
* Data Wrangling
* Combining, Merging, Joining, Group by
* Useful Methods and Operations
* unique(), nunique(), value\_counts(), count(), sort\_value()
* Different Ways of Creating DataFrame

**Class 15: Exploratory Data Analysis (EDA) using Pandas**

**Week 06**

**Class 16: Project on Data Analysis using Pandas**

\*\*\* NumPy and Pandas are two fundamental Python libraries that form the cornerstone of data analysis and manipulation. Learning these libraries equips you with the tools to efficiently work with large datasets, extract meaningful insights, and make data-driven decisions.

Key Purposes: Data Cleaning and Preparation:

* Handling Missing Data: Imputing missing values, removing outliers, and handling inconsistencies.
* Data Formatting: Converting data types, standardising formats, and cleaning text data.
* Data Integration: Merging and joining datasets from various sources.
* Efficient Analysis: Structured and organised data facilitates efficient analysis and visualisation.

**Data Visualization using the most widely used Python library**

**Week 06**

**Class 17: Data Visualization using Matplotlib**

* Basic Plotting
* Creating multiple plots on the same canvas.
* Matplotlib "Object Oriented" approach
* Creating a figure and a set of subplots
* Saving figures
* Decorating the figures

**Class 18: Data Visualisation Using Pandas**

* Histograms
* Style Sheets
* Area plot
* Bar plot
* Line plot
* Hexbin
* Scatter plot
* Box Plot
* Pie Plot

**Week 07**

**Class 19: Data Visualization using Seaborn**

* [Distribution Plots](http://seaborn.pydata.org/tutorial/distributions.html#distribution-tutorial): Visualising the distribution of a dataset
* Joint plot, Pair plot, KDE plot
* Plotting with categorical data
* Swarm plot and Strip plot
* Box plot and Violin plot
* Bar plot and Point plot
* Matrix Plots, Heatmap

**Class 20: Data Visualization using Plotly**

* Histogram
* Bar Chart
* Stacked Bar Chart
* Bubble Chart
* Donut Chart
* Scatter Chart
* Line Chart
* Spider Chart
* Sunburst Chart
* Waterfall Chart

**Class 21: Exploratory Data Analysis (EDA) with data Visualisations**

\*\*\*Data visualisation is a powerful technique that helps us understand and communicate complex information through visual representations. Matplotlib, a versatile Python library, is a key tool in this process.

Key Purposes of Data Visualization:

Exploratory Data Analysis (EDA):

* Identifying Patterns: Spotting trends, seasonality, and outliers.
* Understanding Distributions: Visualising the shape and spread of data.
* Detecting Anomalies: Identifying unusual data points.

Communicating Insights:

* Storytelling: Creating compelling narratives from data.
* Simplifying Complex Ideas: Making complex data accessible to a wider audience.
* Highlighting Key Findings: Emphasising important conclusions.

**Statistics with Python for Data Science**

**Week 08**

**Class 22: Descriptive Statistics with Measures of Central Tendency & Dispersion**

* Measures of Central Tendency [Mean, Median, Mode], and Location [Quartile, Decile, Percentile].
* Which Measure is the Best one in which situation?
* Measures of Dispersion [Range, Variance, Standard Deviation, Coefficient of Variation].

**Class 23: Shape Characteristics & Causality**

* Measures of Shape Characteristics: Skewness, Kurtosis.
* Correlation Analysis.
* Concepts of Regression Analysis.
* What is causality and what is the difference between Correlation and Regression Analysis?

**Class 24: Sampling Methods and Probability Concepts**

* What is Sampling Technique and what are the different Sampling Methods?
* When to Choose which Sampling Method?
* What is Probability?
* Random Variables and Introduction to Probability Distributions.

**Week 09**

**Class 25: Inferential Statistics**

* Normal Distribution & Standard Normal Distribution.
* Central Limit Theorem and where this is applied?
* Estimators and Estimates.
* Confidence Interval and Margin of Error.
* Hypothesis Test Concepts: Null & Alternative Hypothesis, Type 1 & Type 2 Error, Test Statistics, Level of Significance & p-value.

**Class 26: Parametric Hypothesis Test**

* Assumptions for Choosing a Parametric Hypothesis Test.
* T-tests, F-test, Chi-squared Test.
* Executive the Hypothesis Testing in Python and Interpreting the Results.
* When to choose which one?

**Class 27: A/B Testing & Non-parametric Hypothesis Tests**

* What is A/B Testing and why is it so popular among the Data Scientists?
* Performing A/B Testing in a Dynamic way in Python.
* When to pick Non-parametric Hypothesis Tests?
* Mann-Whitney U Test (Wilcoxon Rank-Sum Test), Wilcoxon Signed-Rank Test, Kruskal-Wallis Test, Kolmogorov-Smirnov Test (for Normality Checking).

**Week 10**

**Class 28: Introduction to the Basics of Time Series Analysis**

* Introduction to Time Series Analysis
* Moving Averages
* Exponential Smoothing
* Decomposition

**Class 29: Time Series Analysis with Statistical Modeling**

* Different Approach to TRAIN-TEST Split for Time Series Forecasting.
* ARIMA
* SARIMAX
* AUTO ARIMA

**Class 30: Time Series Analysis with ML Approach Prophet library**

* Forecasting Future Data using Facebook Prophet Library.
* Comparing the Results between the Statistical Approach and ML Approach.

**\*\*\*** By Completing the Statistics with Python for Data Science module, students will gain a robust understanding of essential statistical concepts and how to apply them to real-world Business Decision-making and data science problems using Python. This module aims to bridge statistical theory and practical implementation, which is critical for data analysis and predictive modelling.

**Overall Achievement:**   
By the end of this module, students will have the ability to:

* Analyse and Interpret data using descriptive and inferential statistical methods.
* Apply Python to execute statistical analyses, enhancing their data science skills.
* Select and implement the appropriate hypothesis tests for various data types and scenarios to make effective Business Decisions.
* Understand Trends and Forecast/Predict Future Values for a Business.
* Understand the role of statistical analysis in data-driven decision-making processes, preparing them for roles in data science and machine learning.

**Machine Learning**

**Data Processing, Transforming and Feature Engineering**

**Week 11**

**Class 31: Introduction to Machine Learning**

* Introduction to ML - What, Why
* Machine Learning Applications
* Supervised Learning
* Unsupervised Learning
* What is a Machine Learning Model?
* Training data and Test data
* Splitting Data, Train set & Test set
* Underfitting and Overfitting
* KFold Cross Validation

**Class 32: Data processing, Transforming, Extractions**

* Feature Scaling Theory
* Feature Scaling - Hands-on
* Feature Extractions
* Image to Pixel using CV2
* Resize, bitwise not and Pixel to Photo
* How to prepare a dataset using photos
* Linear Discriminant Analysis (LDA)

**Class 33: Feature Selection, Outlier Detection and Removal**

* Feature Importance, Feature Selection
* Label Encoding
* Ordinal Encoding
* One Hot Encoding
* Outlier Detection and Removal Using IQR
* Outlier Detection and Removal using Standard Deviation
* Outlier Detection and Removal using Z-Score

\*\*\*By the end of this module, students will have the ability to

* Features Scaling: Scale features as per model requirements (e.g., Normalizations Standardization and Absolute ).
* Selecting Relevant Features: Identifying the most important features for a specific task.
* Transforming Features: Applying transformations to improve model performance (e.g., log transformation, square root transformation).
* Encoding Categorical Variables: Converting categorical variables into numerical representations (e.g., one-hot encoding, label encoding).
* Outlier Detection & Removal: Finding and removing outliers

Why is Data Processing Important?

* Improved Model Performance: Well-prepared data leads to more accurate and reliable models.
* Enhanced Insights: Clean and transformed data allows for deeper understanding of the underlying patterns.
* Efficient Analysis: Structured and organised data facilitates efficient analysis and visualisation.
* Better Decision Making: Data-driven decisions based on accurate and reliable insights.

**Supervised Learning (Regression)**

**Week 12**

**Class 34: Regression using Statsmodels**

* Simple vs. Multiple Linear Regression Model.
* Assumptions of Linear Regression
* Multiple Linear Regression Model in Python using Statsmodels (Statistical Approach for Estimating the Causality/Dependency).
* Interpreting the Regression Model Outputs.
* R-squared and Adjusted R-squared.

**Class 35: Regression using Scikit-learn**

* Splitting Data into Train-Test Set
* Simple Linear Regression Model in Python using Scikit-learn (Machine Learning Approach for Prediction Purpose)
* Multiple Linear Regression Model in Python using Scikit-learn
* Model Evaluation Metrics: R-squared, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), etc.
* Residual Plots.
* Polynomial Regression

**Class 36: Regression (Regularized Regression)**

* Bias and Variance, Bias Variance Trade off
* Introduction to Regularized Regression Methods
* Ridge Regression Concept
* LASSO Regression Concept
* Elastic Net Regression Concept
* Which one to choose?

**Week 13**

**Class 37: Project on Regression Learning (**[**Used Car Price Prediction**](http://localhost:8888/tree/All%20Projects%20Collections/Used%20Car%20Price%20Prediction)**)**

**Supervised Learning (Classifications)**

**Class 38: Binary Logistic Regression Model**

* Class Imbalance Problem
* Class Imbalance Problem Handling
* Confusion Matrix (Precision, Recall, f1 score)
* Binary Logistic Regression Theory
* Binary Logistic Regression Algorithm
* Binary Logistic Regression Pen & Paper Exercise
* **Hands-on:** Model development using Binary Logistic Regression
* How to save and load your trained Machine Learning Model

**Class 39: K Nearest Neighbours**

* K Nearest Neighbors Theory
* K Nearest Neighbors Algorithm
* K Nearest Neighbors Pen & Paper Exercise
* Model development using K Nearest Neighbors - Hands-on
* **Hands-on: Handwritten digits recognition using KNN**

**Week 14**

**Class 40: Decision Tree**

* Decision Tree Theory
* Decision Tree Algorithm
* Decision Tree Pen & Paper Exercise
* **Hands-on:** Model development using [Decision Tree](https://www.udemy.com/tutorial/data-science-and-machine-learning-using-python-bootcamp-qazi/scikit-learn-decision-tree-and-random-forests-part-1/)

**Class 41: Support Vector Machines (SVMs)**

* Support Vector Machines Theory
* Support Vector Machines Algorithm
* Support Vector Machines (SVMs)
* [Hyper-parameter tuning](http://localhost:8888/tree/34_Hyper-parameter%20tuning)
* GridsearchCV
* **Hands-on:** Model development using Support Vector Machines

**Class 42: Naïve Bayse Classifications**

* Bag of words, vectorization
* Bag of N-grams
* Naïve Bayse Classifications Theory
* Naïve Bayse Classifications Algorithm
* Naïve Bayse Classifications Pen & Paper Exercise
* Naïve Bayse Classifications
* **Hands-on: Ham Spam Detection using Naïve Bayse Classification**

**Week 15**

**Class 43: Project on Supervised Learning (Classification)**

**Unsupervised Learning (Clustering)**

**Class 44: K Means Clustering**

* K Means Clustering Theory
* K Means Clustering Algorithm
* [K Means Clustering, Elbow method](https://www.udemy.com/tutorial/data-science-and-machine-learning-using-python-bootcamp-qazi/k-means-clustering-theory-lecture/)
* K Means Clustering - Hands-on
* **Hands-on: RFM Based** [**Customer Segmentation**](http://localhost:8888/tree/37_Real%20Projects/K-means/Kmeans_Customer%20Segmentation) **using K Means Clustering**

**Class 45: Principal Component Analysis (PCA)**

* [What is Principal Component Analysis?](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-what-is-principal-component-analysis)
* [PCA vs LDA Analysis](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-pca-vs-lda-vs-factor-analysis)
* [What are Principal Components?](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-what-are-principal-components)
* [Principal Component Analysis (PCA) Examples](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-principal-component-analysis-pca-examples)
* **Hands-on:** [Implement PCA in Python](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-implement-pca-in-r-amp-python-with-interpretation)
* [Predictive Modelling with PCA Components](https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/#h-predictive-modeling-with-pca-components)

**Week 16**

**Class 46: Project on Unsupervised Learning (Clustering)**

**Ensemble Learning**

**Class 47: Ensemble Learning Models - Random Forest**

* Ensemble Learning
* What is Bagging
* What is Bagged Tree
* Random Forest
* Random Forest Theory
* Random Forest Algorithm
* **Hands-on:** Random Forest

**Class 48: Ensemble Learning Models – XGBoost**

* Boosting
* XGBoost Theory
* XGBoost Algorithm
* **Hands-on:** XGBoost

**Deep Learning (7 Classes + 1 Project)**

**Week 17**

**Class 49: Deep Learning Fundamentals and Model Training**

* Introduction to deep learning, neural network basics, and model architecture.
* Fundamentals of training neural networks, including loss functions and optimization.

**Hands-on:** Training a basic neural network on a simple dataset.

**Class 50: Convolutional Neural Networks (CNNs) and Convolution**

* Introduction to CNNs, the concept of convolution, and layers used in CNNs.
* Applications of CNNs in image classification and object detection.

**Practical:** Building and training a CNN model on an image classification task.

**Class 51: Hyper-parameter Tuning and Optimization**

* Techniques for tuning deep learning models and managing overfitting.

**Hands-on:** Optimizing a CNN or OCR model with hyperparameter adjustments.

**Week 18**

**Class 52: Model Evaluation and Validation Techniques**

* Evaluation metrics and validation techniques for image classification and OCR models.
* Practical: Evaluating the performance of the OCR pipeline and CNN model.

**Class 53: Introduction to Image Processing**

* Overview of Computer Vision and its role in ML and deep learning.

**Practical:** Preprocessing images including resizing, filtering, and edge detection.

**Class 54: Object Detection and Recognition**

* Introduction to Object detection and principles.

**Week 19**

**Class 55: Real-World Applications**

* Advanced image processing for region detection.

**Practical:** Detecting objects in complex images and extracting information.

**Class 56: Project Presentation and Discussion**

* Students work on a project combining learnings
* Project presentations, peer feedback, and instructor critique.

**MLOps (4 Classes)**

**Class 57: Introduction to MLOps Concepts**

* Overview of MLOps and the ML model lifecycle.
* Importance of MLOps in production environments and introduction to CI/CD in ML.

**Week 20**

**Class 58: Building and Managing ML Pipelines**

* Creating an end-to-end ML pipeline, including preprocessing, training, and inference.

**Practical:** Using python to build a preprocessing pipeline within a full ML pipeline.

**Class 59: Cloud Deployment of ML Models**

* Deploying a deep learning model to the cloud and creating an API endpoint.

**Hands-on:** Deploying a model for real-time access.

**Class 60: Monitoring and Maintenance of Deployed Models**

* Techniques for monitoring model performance and handling model drift.

**Practical:** Setting up monitoring for the deployed model.

**Job Preparation**

**Extra Week taken by the Support Instructor:**

* **Class 1:**

\* Introduction to GitHub

\* How to keep your projects on GitHub Repository?

* **Class 2:**  
  \* How to Open a LinkedIn Account and Use it effectively for Networking and Job Search?
* **Class 3:**

\* CV Making & How to Write a Cover Letter?

\* Portfolio Building.

* **Class 4:**  
  \* Guidance onFreelancing Career

\* Interview Skill Development & How to Effectively Apply for a Position.

\* Roadmap for Future Ahead.