Office Telephone: +91 8660382715
E-mail: info@aas-ai.tech

Anagha Agile Systems

Data Science

Audience
(Who This Course is for)

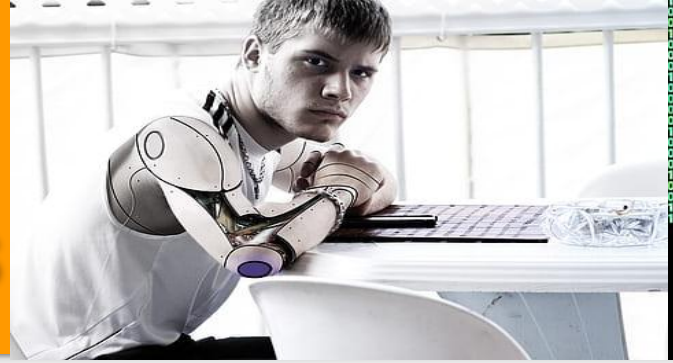
- ★ This course is for IT PROFESSIONALS who want to broaden their practical knowledge of
  - ○ Data Science,
  - ○ Scientific Computing(Image Processing, Numerical Analysis),
  - ○ Data Analytics.
  - ○ Data Visualisation.
  - ○ MatplotLib
- ★ Classroom & Hands on Labs
- ★ SciKit learn based Training

Objectives

PARTICIPANTS

- Acquire the competencies and skills necessary for working in the field of Cloud Architecture
- Can develop in their current role, or qualify for a new role
- Benefit from a selected, well-structured and quality-checked digital learning curriculum
- Learn from experts with a high practical relevance
- Exchange information on challenges and best practices in communities
- Can apply the learning content directly in their everyday professional life
- Have the opportunity to develop further by completing a certificate or university degree

Prerequisites



This is a Advanced-level learning path and so no prior knowledge of scientific computing is required, however, general knowledge of computing would be highly beneficial. To get the most of out of this course, participants should have basic proficiency with networking technologies like programming. This course is ideal for solution developers as well as experienced senior professionals interested in applying scientific computing to data science, machine learning and data engineering. Students should have beginner level Linux and intermediate level Python skills. For your project in this course, you will build a big data engineering pipeline in a SciKit platform:Jupyter Notebooks.

Class Info:
Instructor: Jayaprakash
Office: Bangalore
Office Hours: Times & Days : 4pm to 8pm on meet.google.com
Office Telephone: +91 8660382715
E-mail: info@aas-ai.tech

Prerequisite for Windows IT Professionals are:

1. Intermediate skills of Python Programming and Understanding Engineering Maths
2. Good Familiarity with Windows 10.
3. Good Familiarity with Excel
5. Good Familiarity with Github & Visual Studio.

Class Meetings

This is a fully-online class. Weekly modules will be released each Tuesday during the course. These modules will present the topics for that week in a series of live videos, screencasts, and slides. There will be short exercises within the modules to reinforce learning and give you a chance to practice what you've just learned.

Optional sections will be held during the week that are designed to help you with the implementing the work in the course as well as address more doubts on concepts taught. Sections will be held "live" via a conferencing platform, and they will also be recorded for later viewing. Training Coordinators will notify students when they will be holding sections, and will let students know the format of sections.

Class Schedule:

Section Schedule:

Monday, 10:30am - 4:30pm, hosted by Jayaprakash S T
Wednesday, 10:30am - 4:30pm, hosted by Jayaprakash S T
Friday, 10:30am - 4:30pm, hosted by Jayaprakash S T

Lab Schedule:

Tuesday, 2:30pm - 5:30pm, hosted by AAS-AI Technical team
Thursday, 2:30pm - 5:30pm, hosted by AAS-AI Technical team
Saturday, 2:30pm - 5:30pm, hosted by AAS-AI Technical team

Additional Expenses:

Texts:

Recordings, Relevant reading materials
Book on Linux Administration,
Dockers, Cloud Computing and
Python Programming

Technology:

Web server Hosting charges
Cloud Subscription and
Project Expenses

Activities, Assignments, Final Project

There are 3 types of exercises you will encounter in this course:
Practice Activities. These are smaller in-module exercises designed to give you a chance to work on what you've just learned or to answer some basic questions on what you've just learned. They are designed to be "low-stakes", but are important for the learning process. These in-module exercises will be 15% of your final grade. The completion of these activities are due before the end of the week. The activities must be completed by 11:59pm the following Wednesday.

Assignments. These are more significant exercises given each week that will be submitted and graded. The due date will be listed on each assignment and on the syllabus page. Generally, they will be due by 11:59pm the following Wednesday. These weekly exercises will be 50% of your final grade.

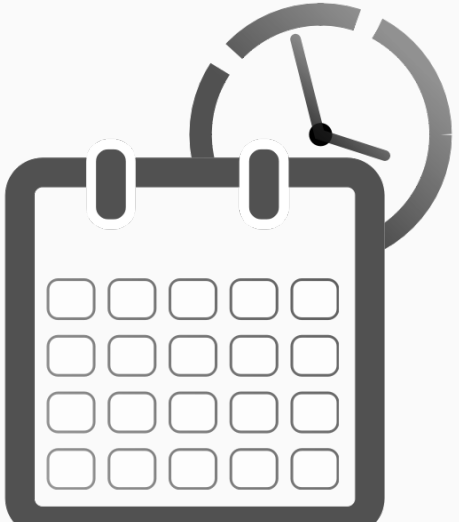
Final Project. The final project is to plan, design, build, and publish a small-scale web site. We will work on this throughout the term in stages. You will have 2 smaller assignments where you ideate and plan your site, and a final assignment where you submit your final project. As part of the work towards the final project, there will be opportunities to get feedback from your teaching assistants. The project will be 35% of your grade.

Course Duration

The Duration of the Course is 4 weeks or 240 hrs with 60% of the course is hands-on experience.

The course materials are provided are soft copies of slides and notes included.

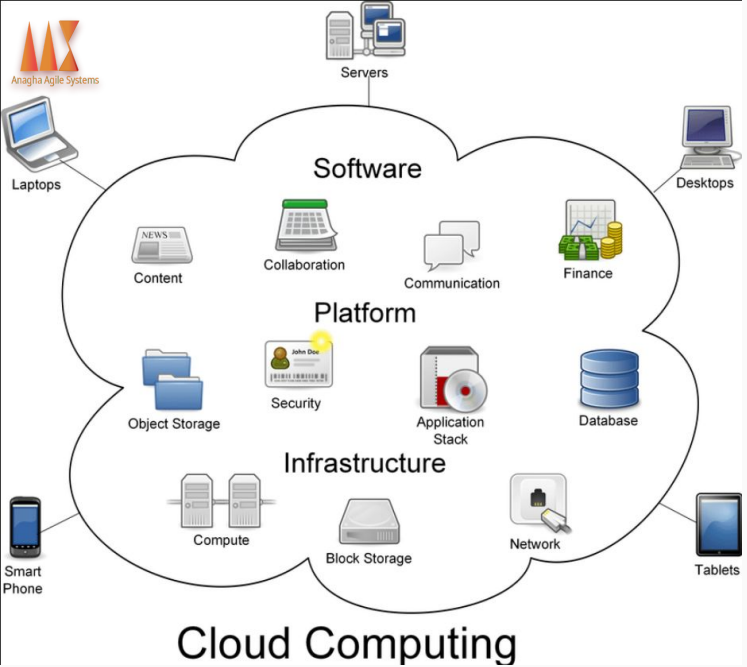
All the queries, lab assignments will be hosted on Anagha Agile Systems website & can be accessed after authentication.



Student Duties

The AAS-AI Faculty Staff recommends that you include the following in your syllabus:

- Contact information for the instructor(s) and TA(s), including email, office hours/location, etc.
- Course description that explains the course's scope and purpose, format (e.g., lecture, recitation, lab, studio), relevance to students' academic/professional goals, and major deliverables (e.g., project, presentation, paper, exhibit)
- Statement of assumed/required prior knowledge
- List of learning resources and whether they are required or supplementary (e.g., textbooks, software, etc.)
- Course-level learning objectives that articulate what students should be able to do by the end of the course
- Description of major assessments and how they contribute to the final grade
- Statements and policies that clearly communicate your expectations regarding
    - Attendance
    - Academic integrity, including collaboration and plagiarism
    - Late/make-up work
    - Accommodations for students with disabilities
- Statement encouraging student wellness
- Expected semester schedule
- Diversity statement



Scientific Computing Overview

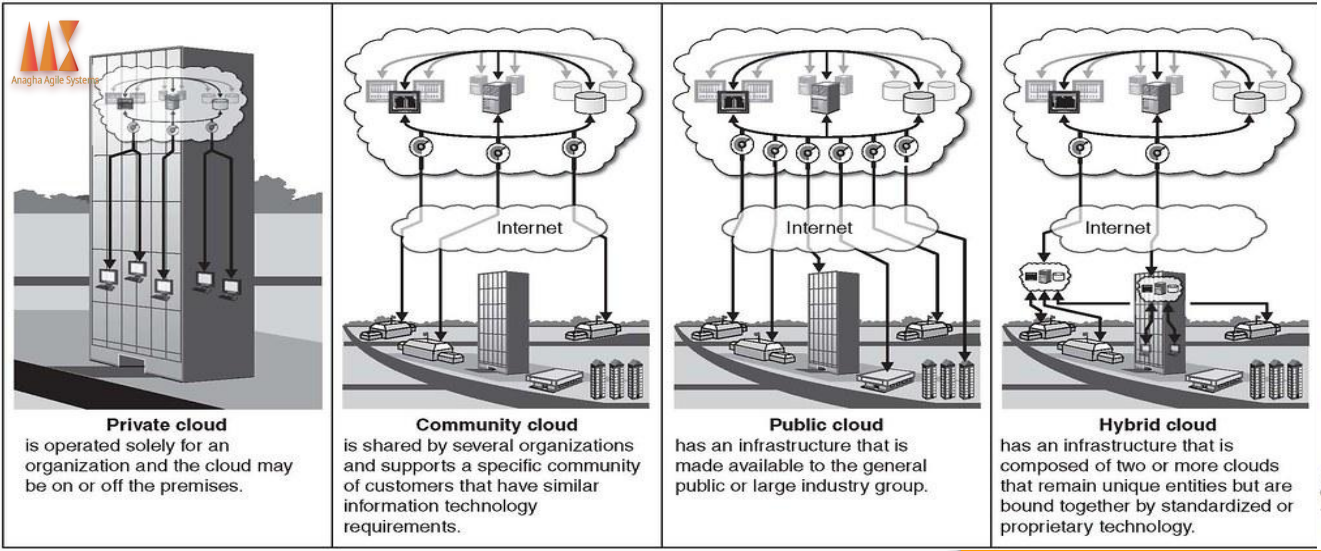
Module 01:

Introduction to Scientific Computing Virtual Classroom

01:Introduction to Python and Scientific Computing.

1.1 Overview of Python features.

1.2 Overview of SciKit-learn

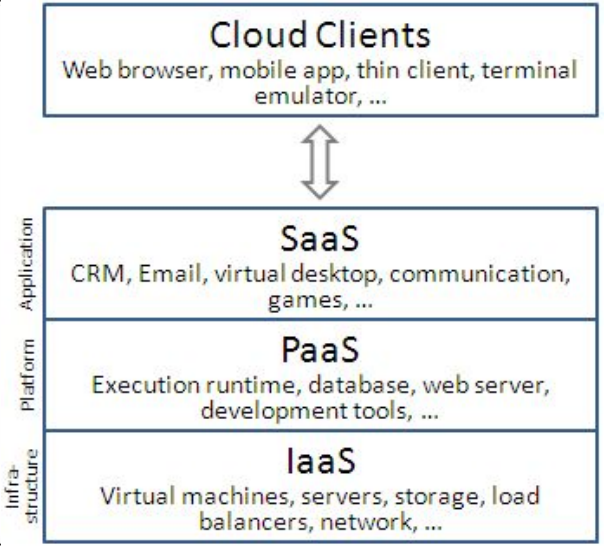


Private cloud
is operated solely for an organization and the cloud may be on or off the premises.

Community cloud
is shared by several organizations and supports a specific community of customers that have similar information technology requirements.

Public cloud
has an infrastructure that is made available to the general public or large industry group.

Hybrid cloud
has an infrastructure that is composed of two or more clouds that remain unique entities but are bound together by standardized or proprietary technology.



Numpy

Module 02:

2 NumPy arrays: tips and tricks:

2.1: NumPy Basics

2.1.1: NumPy – Working with Numerical Arrays

Introduction to Numpy

Module 02:

Anagha Agile Systems

Linear Algebra

Module 03:

Interpolation

Module 04:

Anagha Agile Systems

Optimization and fitting techniques

Module 05:

Ordinary differential equations

Module 06:

Anagha Agile Systems

Root finding

Module 07:

Anagha Agile Systems

Data Rebinning

Module 08:

8.1 Data rebinning :

Examples of rebinning data to produce smaller arrays with and without interpolation.

8.2 Rebinning FUNCTION

Anagha Agile Systems

Simple Plotting

Module 09:

Anagha Agile Systems

Convex-Hull

Module 10:

Anagha Agile Systems

Kalman filtering

Module 11:

11.1 Kalman Filtering Example from the Welch & Bishop Introduction to the Kalman Filter.

Anagha Agile Systems

Communication Theory Example of BPSK simulation.

Module 12:

12.1 Communication theory examples

12.1 These two examples illustrate simple simulation of a digital BPSK modulated communication system

[https://scipy.github.io/old-wiki/pages/Cookbook/CommTheory.html]

Anagha Agile Systems

Signal Smoothing

Module 13:

13.0 Smoothing a signal Performing smoothing of 1D and 2D signals by convolving them with a window.

13.1 Signal Processing (scipy.signal)
https://scipy-cookbook.readthedocs.io/items/SignalSmooth.html
https://docs.scipy.org/doc/scipy/reference/tutorial/signal.html

13.2 bspline(x, n)

13.3 B-spline basis function of order n.

13.4 cubic(x)

13.5 A cubic B-spline.

13.6 quadratic(x)

13.7 A quadratic B-spline.

13.8 gauss_spline(x, n)

13.9 Gaussian approximation to B-spline basis function of order n.

13.1 cspline1d(signal[, lamb])

13.11 Compute cubic spline coefficients for rank-1 array.

13.12 qspline1d(signal[, lamb])

13.13 Compute quadratic spline coefficients for rank-1 array.

13.14 cspline2d(input[, lambda, precision])

13.15 Coefficients for 2-D cubic (3rd order) B-spline.

13.16 qspline2d(input[, lambda, precision])

13.17 Coefficients for 2-D quadratic (2nd order) B-spline:

13.18 cspline1d_eval(cj, newx[, dx, x0])

13.19 Evaluate a cubic spline at the new set of points.

13.2 qspline1d_eval(cj, newx[, dx, x0])

13.21 Evaluate a quadratic spline at the new set of points.

13.22 spline_filter(Iin[, lmbda])

13.23 Smoothing spline (cubic) filtering of a rank-2 array.

Anagha Agile Systems

Nearest-neighbor queries/ KD Tree Searching

Module 14:

https://scipy-cookbook.readthedocs.io/items/KDTree_example.html
https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.KDTree.html

14.0 Spatial algorithms and data structures (scipy.spatial)
14.1 KDTree(data[, leafsize, compact_nodes, ...])
14.2 kd-tree for quick nearest-neighbor lookup.
14.3 cKDTree(data[, leafsize, compact_nodes, ...])
14.4 kd-tree for quick nearest-neighbor lookup
14.5 Rectangle(maxes, mins)
14.6 Hyperrectangle class.
https://docs.scipy.org/doc/scipy/reference/spatial.html

Anagha Agile Systems

Butterworth Bandpass Filter

Module 15:

15.1 Create and apply a Butterworth bandpass filter.

https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.buttap.html#scipy.signal.buttap
https://scipy-cookbook.readthedocs.io/items/ButterworthBandpass.html

15.2 Butterworth Bandpass: This cookbook recipe demonstrates the use of scipy.signal.butter to create a bandpass Butterworth filter.

Anagha Agile Systems

FIR Filter Design,

Module 16:

16.1 FIR Filter Design, Design a lowpass FIR filter using the window method.
https://scipy-cookbook.readthedocs.io/items/FIRFilter.html
https://scipy-cookbook.readthedocs.io/items/ApplyFIRFilter.html
https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.firwin.html
https://mpastell.com/pweave/_downloads/FIR_design_rst.html OR
https://mpastell.com/2010/01/18/fir-with-scipy/

Anagha Agile Systems

Multithreading

Module 17:

17.1 Easy multithreading for embarrassingly parallel multidimensional space using kdtrees.
https://scipy-cookbook.readthedocs.io/items/Multithreading.html

Anagha Agile Systems

KD Tree

Module 18:

18.1 KDTree Searching multidimensional space using kd-trees.
https://scipy-cookbook.readthedocs.io/items/KDTree_example.html

Anagha Agile Systems

Particle Filter

Module 19:

19.1 Particle Filter A simple particle filter algorithm for tracking objects in a video sequence.
https://scipy-cookbook.readthedocs.io/items/ParticleFilter.html

Anagha Agile Systems

Brownian Motion

Module 20:

20.1 Brownian Motion Compute Brownian motion (i.e. the Wiener process).
https://scipy-cookbook.readthedocs.io/items/BrownianMotion.html

Anagha Agile Systems

Correlated Random Samples

Module 21:

21.1 Correlated Random Samples Generate correlated normally distributed random samples.
https://scipy-cookbook.readthedocs.io/items/CorrelatedRandomSamples.html

Anagha Agile Systems

Large Markov Chains

Module 22:

22.1 Large Markov Chains Find the stationary distribution of a large Markov chain; the M/M/1 tandem queue
https://scipy-cookbook.readthedocs.io/items/Solving_Large_Markov_Chains.html

Anagha Agile Systems

Watershed algorithm

Module 23:

23.1 Watershed algorithm Apply the watershed algorithm in order to split an array into distinct components (e.g. for the segmentation of an image into objects).
https://scipy-cookbook.readthedocs.io/items/Watershed.html

Anagha Agile Systems

24.1 Linear Classification Fisher's discriminant function and Probabilistic Generative model
https://scipy-cookbook.readthedocs.io/items/LinearClassification.html

Linear Classification

Module 24:

Anagha Agile Systems

Eye Diagram

Module 25:

25.1 Plot an eye diagram using numpy and matplotlib.
https://scipy-cookbook.readthedocs.io/items/EyeDiagram.html
https://dspillustrations.com/pages/posts/misc/eye-diagram-examples.html
https://github.com/WarrenWeckesser/eyediagram
https://en.wikipedia.org/wiki/Eye_pattern

Anagha Agile Systems

Scipy

Module 26:

Anagha Agile Systems

Top level SciPy

Module 27:

	Name	Description
27.1	cluster	Clustering algorithms
27.2	constants	Physical and mathematical constants
27.3	fftpack	Fast Fourier Transform routines
27.4	integrate	Integration and ordinary differential equation solvers
27.5	interpolate	Interpolation and smoothing splines
27.6	io	Input and Output
27.7	linalg	Linear algebra
27.8	ndimage	N-dimensional image processing
27.9	odr	Orthogonal distance regression
27.1	optimize	Optimization and root-finding routines
27.11	signal	Signal processing
27.12	sparse	Sparse matrices and associated routines
27.13	spatial	Spatial data structures and algorithms
27.14	special	Special functions
27.15	stats	Statistical distributions and functions

Anagha Agile Systems

SciPy for Linear Algebra

Module 28:

Anagha Agile Systems

SciPy for Numerical Analysis

Module 29:

Anagha Agile Systems

SciPy for Signal Processing

Module 30:

Anagha Agile Systems

SciPy for Data Mining

Module 31:

Anagha Agile Systems

SciPy for Computational Geometry

Module 32:

Part II: DATA ANALYTICS

Anagha Agile Systems

Bayesian Data Science by Simulation

Module 33:

Anagha Agile Systems

Bayesian Data Science by Probabilistic Programming

Module 34:

Part III: Advanced Data Analytics

Anagha Agile Systems

Spatial Data Structures and Algorithms

Module 35:

Anagha Agile Systems

Multidimensional Image Processing Functions

Module 36:

Anagha Agile Systems

Multidimensional Image Processing Functions

Module 37:

Anagha Agile Systems

File IO

Module 38:

Anagha Agile Systems

pandas

Module 39:

Anagha Agile Systems

Scikit-Learn

Module 40:

40.1 What is Scikit-Learn?

Scikit-learn provides tools for:

- Regression, including Linear and Logistic Regression
- Classification, including K-Nearest Neighbors
- Model selection
- Clustering, including K-Means and K-Means++
- Preprocessing, including Min-Max Normalization

40.2 Getting started with Scikit-learn
40.3 Datasets and import.sklearn
40.4 Generate synthetic regression data
40.5 Data Preprocessing
40.6 Scikit-learn Linear Regression: implement an algorithm

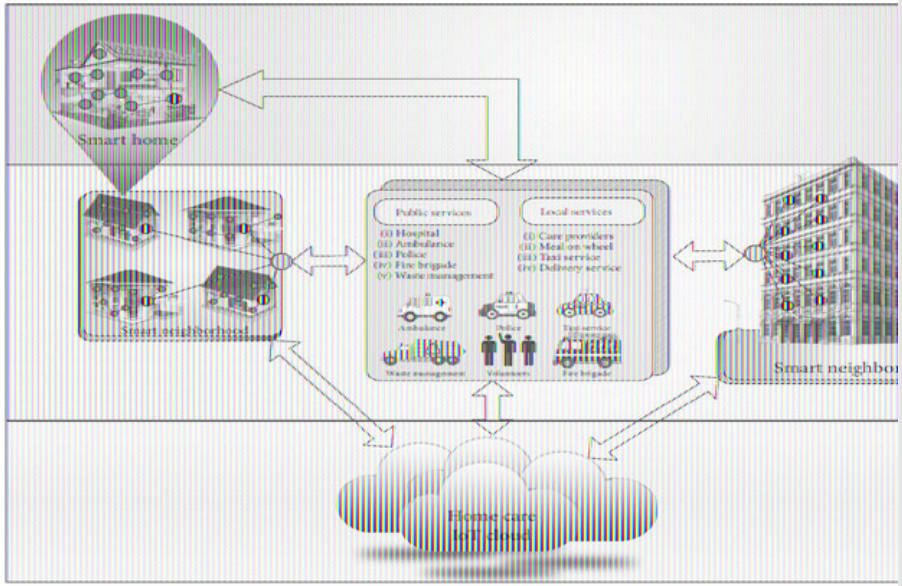
What Next?.....

You've just taken your first steps to master Scikit-Learn. Today, we covered the purpose of Sklearn, how to import or generate sample data, how to scale our data, and how to implement the popular linear regression algorithm.

As you continue your Scikit-learn journey, here are some next algorithms and topics to learn:

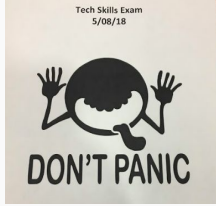
Support Vector machine
Random Forest
Naive Bayes model
Unsupervised learning
Deep learning
Logistic regression

To help you get started in Machine Learning, AAS-AI has created the course Hands-on Machine Learning & Deep Learning with Pytorch Framework. With in-depth explanations of all the Pytorch basics and popular Deep learning algorithms, this course will give you, everything you need to learn about Machine Learning & Deep Learning in one place.

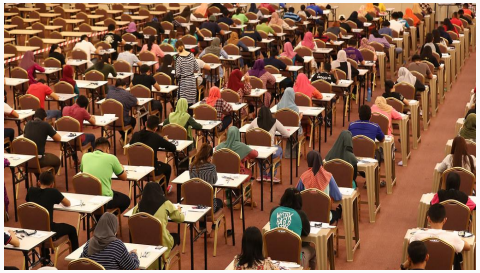


Data Science Trends

Anagha Agile Systems



1. Multi Choice Questions

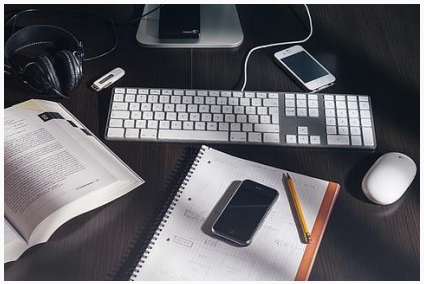


3. Oral Interview

Q & A on Project Presentation

20 mins | 180 mins | 30 mins | 30 mins | 20 mins

2. Project hands on Lab



4. Project Presentation

Conclusion

Let's wind up with different IT positions you can take up after this course.

1. **Data Analyst** With so much data in the world, and with data as such a constant in all of our lives, there is a need for professionals who can analyze that data and suss out the human element. The job of Data Analysts is to analyze this data, and determine what it tells us about the sample/population using that data.
2. **Data Engineering** In the world of big data, every job works together in order to understand the data collected and make the best use of it for the business. The role of a data engineer is to prepare the infrastructure like algorithms that data scientists and data analysts to analyze in order to enable the business to make the best decisions based on their findings.
3. **Data Product Owner** The role of Data Product Owners is to ensure value comes out of all the data projects of their companies. They need to develop a vision of what the data science team wants to achieve, together with a strategy to get there. He is someone who develops and maintains a vision and plan for that data and oversees the execution of that plan.
4. **Data Scientist** Those who practice data science are called data scientists, and they combine a range of skills to analyze data collected from the web, smartphones, customers, sensors, and other sources to derive actionable insights. " The ability to take data — to be able to understand it, to process it, to extract value from it, to visualise it, to communicate it — that's going to be a hugely important skill in the next decades. " si done by DATA SCIENTIST

Feedback

Please contact us at info@aas-ai.tech if you have any questions, or want to share your thoughts about the exam.

This course serves as a requirement to pursue other higher data science certifications.

Data Science Introduction
Evolution of Data Science
Organization Data Science Stack
Roles in Data Science
Business Analytics vs Data Science
Machine Learning in Data Science
Business Analytics Classification
Statistics – Exploratory Analysis
Statistics – Hypothesis Testing
Data Science Industry Use cases

Course Prerequiste.....