



AI Champ Syllabus

Note: *The syllabus is mentioned in the course handbook. However, we have reiterated the syllabus according to the sections. Some detailed information for mini-projects is removed to provide flexibility to the course.*

Note: *The Beginners can also attend the sessions done for intermediate and master. However, we suggest sticking to basics and hence the differentiation. The same goes for Intermediate.*

Beginners

1. Linux Basics
2. Command Line Interface
3. Integrated Development Environments
4. Virtual Environments
5. Understanding Path(s)
6. Basics of Python
7. 20+ Practice Questions on Python
8. Github Basics and Practice
9. Reading and Understanding Errors
10. Tips on Debugging
11. Machine Learning Introduction
12. Classification, Regression, and Clustering
13. Some Traditional ML Algorithms Overview
14. Understanding Problems in ML
15. Numpy Practice
16. Pandas Practice
17. Scikit Learn Practice

18. Mini Projects (3)
19. Final Project (1)

Intermediate

1. Beginners Course Modules (Do only what is necessary)
2. Deep Learning Basics
3. Computer Vision Basics
4. Natural Language Processing Basics
5. Mini Projects (5)
6. Case Study Project (1)

Master

1. Beginners Course Modules (Do only what is necessary)
2. Intermediate Course Modules (Do only what is necessary)
3. Mini Projects (5 - Extra)
4. Optional Chance to Work on Live Project from Codevector Labs (as a paid intern)

More Details

Deep Learning Basics

1. Deep Learning history and inspiration from the human brain.
2. An overview of mathematics will help you understand the language used in deep learning.
3. ANN and its components - Perceptron, Layers, Weight, Bias, Loss, Backpropagation, Activation Functions.
4. Understanding the general problems and some tips to train the models - Imbalanced Datasets, Overfitting, Underfitting, Regularization, Augmentation, Small Datasets, Transfer Learning, Vanishing Gradients, Exploding Gradients, Normalization.

Computer Vision Basics

1. Computer Vision history and inspiration from the human vision.

2. An overview of mathematics will help you understand the language used in computer vision. Mostly contains matrix operations.
3. Understanding what an image is to a computer, the coordinate system of the image, overview of some operations that can be performed on the image/videos.
4. CNN and its components - Similarity with ANN, Convolution operation, Kernels, Activation Maps, Pooling, Padding, Flattening, FCN.
5. Understanding the general problems and some tips to train the model - Same as deep learning but more specific to CNNs. eg. Different Augmentation techniques, Using different architectures of CNN models for your use case, BatchNormalization, etc.

Natural Language Processing Basics

1. Can machines understand the natural language as humans do?
2. Do computers only understand numbers? How can computers represent natural language in numbers?
3. NLP applications, overview, and pipeline - Sentence Segmentation, Word Tokenization, POS Tagging, Lemmatization, Stemming, Stop Words, NER.
4. Using popular NLP libraries - NLTK, SpaCy, Gensim, Transformers.
5. Overview of deep learning methods for NLP tasks.

Case Studies

1. 10+ real-world case studies - select one for the main capstone project.
2. Case-study specific curated learning content - Blogs, Videos, Slides, Research Papers, etc.
3. Tips on how to find current state-of-the-art (SOTA) methods for a given task.
4. Tips on using open-source libraries and codes for the project.
5. Datasets - Publicly available, exclusively created datasets.
6. Tips on creating your own datasets - How to reduce efforts while creating a dataset?
7. Overview of some software and platforms used to annotate the datasets.
8. My model works decently on the test dataset but doesn't work very well on real-world, uncontrolled data. What to do now?
9. What are REST APIs?
10. Creating WEB APIs using python frameworks such as Flask, Django, FastAPI.
11. Deploy the AI/ML models to the cloud.
12. Understanding the cost required to deploy the AI/ML models.
13. Perform load testing on your deployed solution - How much load it can handle?
14. Can we optimize to reduce latency and cost?

15. How will we know if our server crashed? Do we have any backup plans? How to fix it ASAP?
16. Our two cents on - What to do next? How do I become an expert in the domain I like?

