COPS Summer of Code 2025

Intelligence Guild

Club Of Programmers, IIT (BHU) Varanasi

Prerequisites

14 – 19 May 2025

Official IG Website: https://cops-iitbhu.github.io/IG-website/

All deadlines are strict. No extensions will be granted.

Introduction

COPS Summer of Code (CSOC) is a flagship initiative under the Club Of Programmers, IIT (BHU) Varanasi, with all verticals contributing through focused tracks. This document outlines the prerequisites for the Intelligence Guild vertical.

Modules will be released weekly and from time to time. **Adhere strictly to deadlines**. Submissions will be evaluated on approach, technical correctness, and clarity. The most technically accurate solution may not necessarily be the one chosen; clarity of thought and a well-reasoned approach will be valued more.

Communities

All communication for the programme will be conducted strictly via Discord. Do not reach out through other channels. Resources and updates will be posted on Github, and all notifications will be made via Discord.

Final Report

A concise report may be submitted along with your final assignment. While **not mandatory**, it may strengthen your overall evaluation. Reports must be written in LaTeX and submitted in PDF format only. We are not interested in surface-level descriptions — focus strictly on your analysis, approach, and reasoning. The report itself constitutes the final assignment. No additional files are to be submitted. Refer to the Assignment section for details. Submit your report here.

Contact Details

In case of any doubts, clarifications, or guidance, you can contact one of us.We request that you stick to Discord as the preferred mode of communication for all the questions that you have as it will also benefit others. However, you can reach out to us through other means in case we fail to respond on Discord.

- Yashashwi Singhania - 7905584242
- Vivek Kumar - 9873432572
- Tejbir Panghal - 9034705165
- Tanish Aggarwal - 9569884059
- Sakshi Kumar - 8073247266
- Manav Kumar Jalan - 9395002919

Setup

Anaconda / Miniconda Setup

For managing Python environments, both Anaconda and Miniconda are viable options:

- **Anaconda**: A comprehensive distribution that includes Python, conda, and a wide array of pre-installed packages, suitable for users seeking a ready-to-use setup.
- **Miniconda**: A minimal installer containing only Python and conda, allowing users to install only the packages they need, offering greater control over the environment.

Setting up a local Jupyter environment is strongly advised.
For official documentation and installation files, refer to the Anaconda Installation Guide and the Miniconda Installation Guide.

Google Colab / Kaggle Notebooks

Google Colab and Kaggle Notebooks are cloud-based Jupyter notebook platforms that support GPU acceleration and require no local setup. These tools are suitable alternatives if local installation is not feasible.

- Google Colab
- Kaggle Notebooks

Coding

Python

A working knowledge of Python is required, including variables, control flow, functions, lists, dictionaries, and file I/O. Review the following resources if needed:

- Python Basics (English)
- Python Full Course (Hindi)
- Kaggle Intro to Programming (Optional)

Pandas

One of the powerful Python library used for data manipulation and analysis, offering data structures like Series and DataFrame. It will be your starting point for working with data-focused libraries in Python. You can get started with pandas for ML from

- Intro to Pandas – Keith Galli
- Pandas Tutorial – Kaggle

Both will be good to go.

Numpy

NumPy is the foundational Python library for numerical computing, providing support for multi-dimensional arrays and a wide range of mathematical functions.

- NUMPY 1 HOUR RESOURCE

Matplotlib

Learn how to get insights from the data using some plots and graphs.

- Intro to Matplotlib – Keith Galli
- Matplotlib Tutorial – Kaggle

Machine Learning Essentials

Now starting with the theory part of actual ML.

This beginner-friendly course introduces core machine learning concepts like model training, validation, and prediction using real-world datasets.

- What Textbooks Don't Tell You About Curve Fitting

Linear and Logistic Regression

The course by Andrew Ng is highly recommended to start your Machine Learning journey. This course provides a great foundation and understanding of the key concepts.

Blog if you like to read.

Data Preparation

Data preparation and exploratory data analysis (EDA) are crucial steps in any machine learning pipeline. This stage helps you clean, transform, and understand your data before feeding it into models.

- A Comprehensive Guide to Data Pre-processing
- Exploratory data analysis
- Analysis of large, complex data sets
- Impute Missing Values
- Resampling Method - (5th chapter of ISLP book given in additional resources)

Additional/Optional resources

- Outlier Detection
- EDA plots
- ISLP book
- Hands on machine learning(scikit learn)
- Intermediate ML
- Lec 48-60 for linear regression codes

Assignment: Comparative Study of Multivariable Linear Regression Implementations

Objective

Implement and compare three distinct approaches to multivariable linear regression, with emphasis on convergence speed and predictive accuracy.

Data Set: Titanic Dataset Kaggle

Project Tasks

Part 1: Pure Python Implementation

- Develop a multivariable linear regression algorithm using only core Python features
- **Do not** utilize external libraries (e.g., `NumPy`, `Pandas`, `Scikit-learn`)
- **Avoid** using large language models (LLMs) such as GPT for code generation, but if you do **clearly mention in the report** where you've used it or else you will be disqualified directly.
- Prioritize simplicity and mathematical clarity
- Employ **gradient descent** as the optimization technique

Part 2: Optimized NumPy Implementation

- Re-implement the algorithm using `NumPy` to leverage vectorized operations
- Enhance performance through appropriate parallelization
- Maintain consistency with the pure Python logic to ensure a fair comparison

Part 3: Scikit-learn Implementation

- Utilize the `LinearRegression` class from the `scikit-learn` library
- Train the model on the identical dataset used in Parts 1 and 2

Evaluation Criteria

1. Convergence Time

- Measure the time taken for models in Parts 1 and 2 to converge
- For the `scikit-learn` model, record the fitting duration
- Ensure uniform initialization across all methods for a fair assessment

2. Performance Metrics

Evaluate the following regression metrics on both training and validation sets (or via cross-validation):

- Mean Absolute Error (MAE)
- Root Mean Squared Error (RMSE)
- R-squared (R^2 Score)

3. Visualization

- Plot the cost function's convergence over iterations for Parts 1 and 2
- Compare convergence speeds and final costs visually
- Present a bar or line chart contrasting the regression metrics across all three methods

4. Analysis and Discussion

- Analyze differences and similarities in convergence times and accuracies
- Elucidate causes for observed differences (e.g., vectorization effects, optimization strategies, solver types)
- Discuss scalability and efficiency trade-offs
- Comment on the influence of initial parameter values and learning rates on convergence

Submission Guidelines

- Create a GitHub repository named `<roll_number>-CSOC-IG` (e.g., `23014019-CSOC-IG`)
- Repository organization:
  - A folder named Prerequisites containing all source code implementations
  - The final report in PDF format, authored using LaTeX

  Everything must be in the github repo itself.
- Submit the repository link via the provided Google Form here

- **Note:** The report constitutes the primary assignment submission. No additional files are required
- **Deadlines are strict and will not be extended**

Final Remarks

Ensure that your submission reflects a clear understanding of the concepts and methodologies applied. Focus on the analytical aspects and the rationale behind your implementations. We look forward to your insightful contributions.

That's it, complete these resources and you will be all ready for CSOC. Hope you will give the resources and assignment your best shot—your future self will thank you!

Adios, *and keep learning!*