

2401\_PTDS

# Unsupervised Learning Kick-Off

October 2024

- 01. Announcements**
- 02. Unsupervised Learning**
- 03. Recommender Systems**
- 04. The Final Boss – Project Kick-off 🕶️**

- Welcome to the **Final Sprint!!**
- No assessments due this week :)
- Please remember to share your thoughts in the **NLP & Classification End of Sprint Feedback**
- For Everyone-> [Please submit this Project Form by Wednesday, 23 October, 11:59 PM.](#)

***Should we not receive your response, we will by default assign you to a TEAM.***

## • Week 6 and Beyond:

- Exam Deadline: 02 December, 2024
- Consolidation Week: Monday, 2 December 2024 – Sunday, 9 December 2024
- Internal Exams: Monday, 9 December 2024 – Monday, 16 December 2024

01. Announcements
02. Unsupervised Learning
03. Recommender Systems
04. The Final Boss – Project Kick-off 🕶️

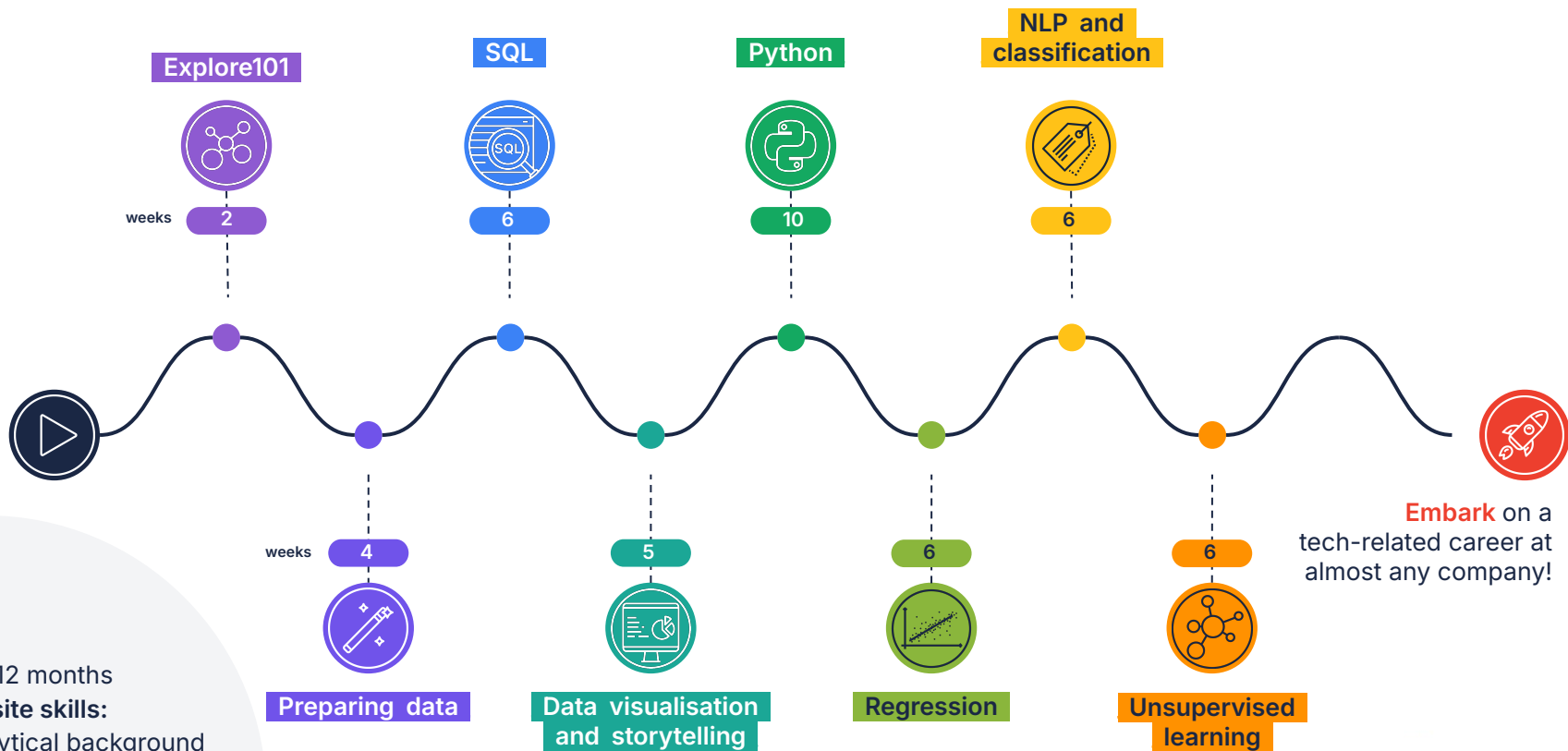
slido



**How confident are you in  
your Data Science Skills at  
this point - One word?**

① Click **Present with Slido** or install our [Chrome extension](#) to activate this poll while presenting.

# Data Science with EXPLORE AI ACADEMY

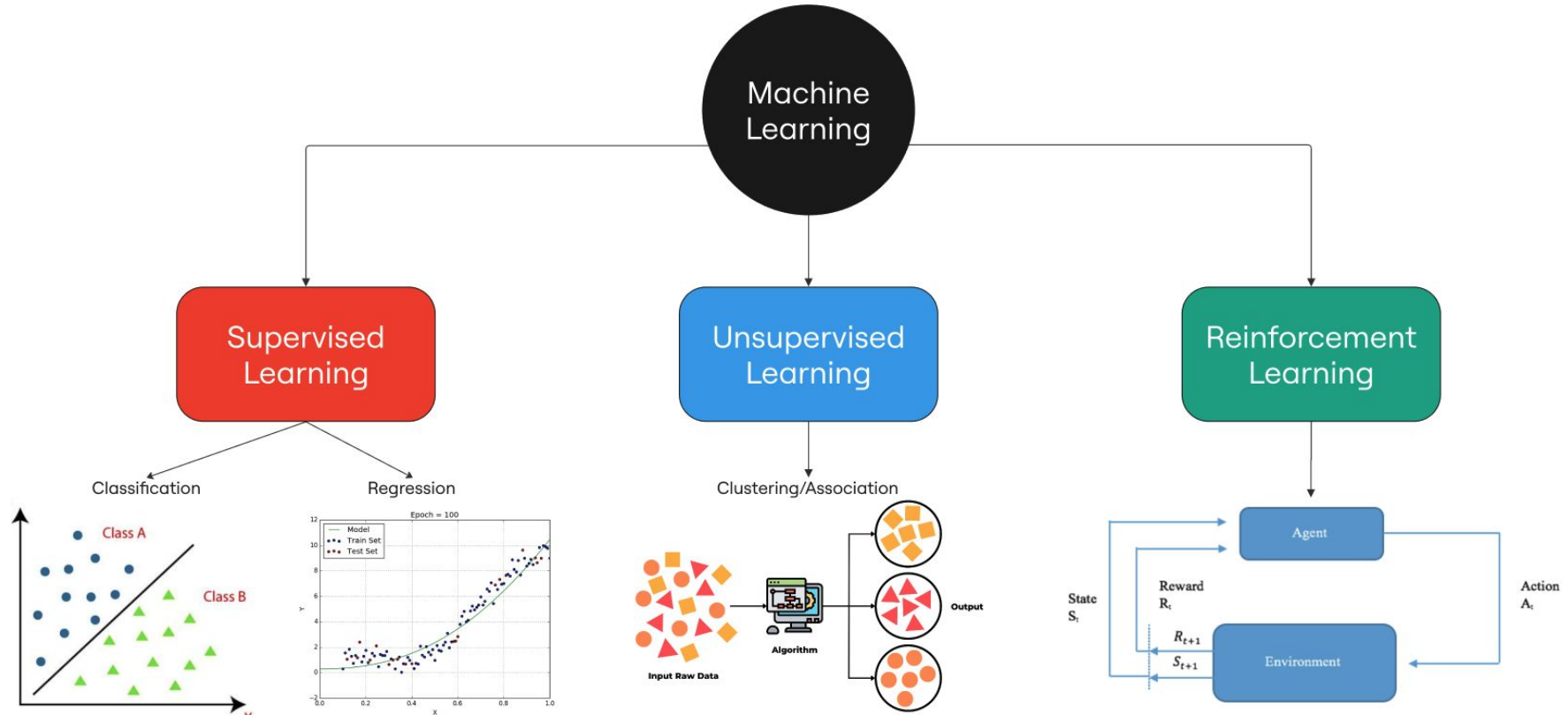


**Duration:** 12 months

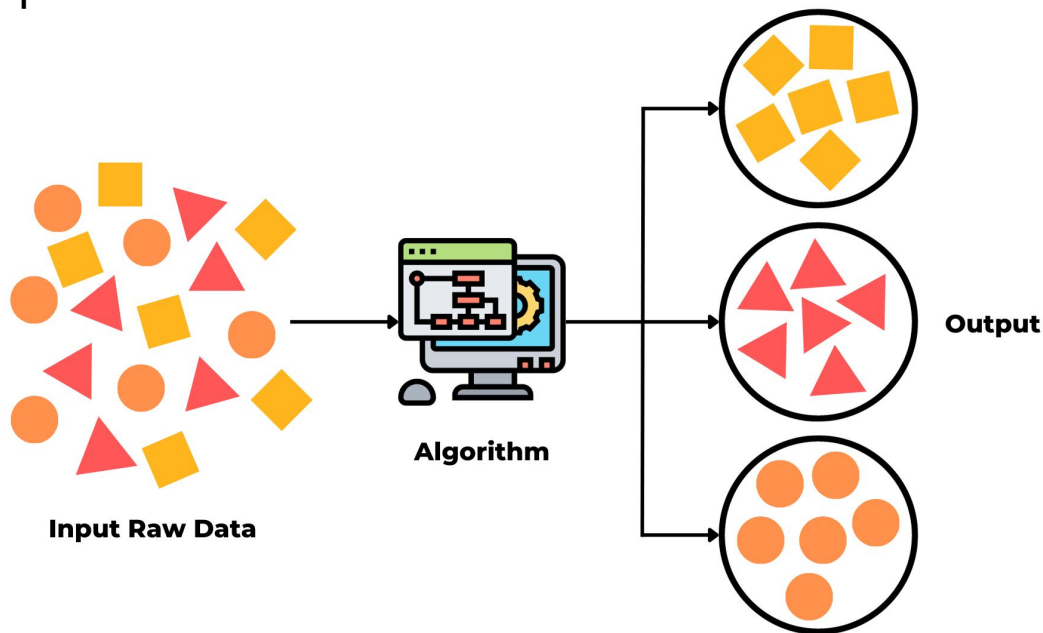
**Pre-requisite skills:**

Basic analytical background

**Course difficulty:** Advanced



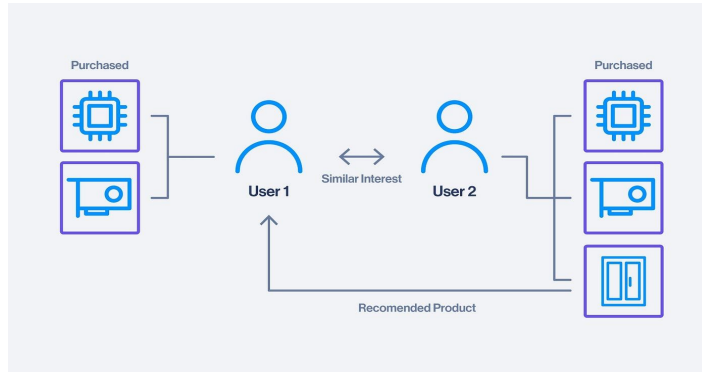
**Unsupervised Learning** is a machine learning technique where algorithms discover patterns and relationships in data without being given labeled examples or explicit instructions.





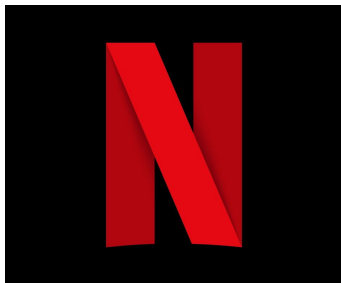
01. Announcements
02. Unsupervised Learning
03. Recommender Systems
04. The Final Boss – Project Kick-off 🤖

- Recommender systems utilizes unsupervised machine learning algorithms to identify patterns and relationships within user-item interaction data.
- Methodologies include:
  - Collaborative filtering → Recommends items based on similarities in preferences between users.
  - Content-based filtering → Uses the attributes of items, like genre or keywords, to recommend items similar to what a user has already liked.
  - Hybrid Systems → combine both collaborative and content-based methods for even more refined recommendations.



## Companies incorporating recommender systems into their Successes

Netflix recommends movies based on a history of movies you have watched or based on popularity of items watched.



Spotify recommends songs that you might like based on a history of streams.



Takealot recommends products based on the history of purchases customers have bought.



# Quiz Time!



01. Announcements
02. Unsupervised Learning
03. Recommender Systems
04. The Final Boss – Project Kick-off 🧐

- Build a collaborative **and** content-based recommender system for a collection of anime titles, capable of accurately **predicting how a user will rate an anime title they have not yet viewed**, based on their historical preferences.
- This **end-to-end project covers the entire workflow**, including data loading, preprocessing, model training, evaluation, and final deployment.
- Facilitators will be technical mentors for the project. What does this mean?
  - You book appointments with us with prepared questions. You may not get the same facilitator.



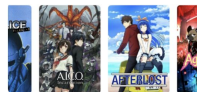


## Kaggle Competition



### Anime Recommender System Project 2024

Create a recommender system using your unsupervised learning skills



[Overview](#) [Data](#) [Code](#) [Models](#) [Discussion](#) [Leaderboard](#) [Rules](#)

#### Overview



**Competition Host**  
ExploreAI Academy



#### Prizes & Awards

Kudos  
Does not award Points or Medals

#### Participation

1 Entrants  
0 Participants  
0 Teams  
0 Submissions

### Kaggle Competition Participation (Not Required for Part-Time Students)

- Use the invite link to access the Kaggle competition page.
- On the Kaggle page, click **Late Submission** to join.
- Complete two pop-up windows:
  - **Competition Rules Agreement:** Click "Agree."
  - **Email Permission:** You may decline this.
- After completing these steps, you will have access to the dataset.



### GitHub|Git and Notebook



kaggle colab

- **Create a private repository** and ensure all teammates and **all facilitators** are added as collaborators. The facilitators' GitHub usernames can be [found here](#).
- The GitHub repository will require a [README file](#).
- Include all the **packages used in a requirements.txt file in your GitHub repository, and add instructions in the README** on how to recreate the environment using Anaconda. Helpful links can be found [here](#) and [here](#).
- Exporting your conda environment:

```
conda activate <env>
conda install pip
#get list of packages and pipe to txt file
pip list --format=freeze > requirements.txt
```
- For more resources consider using Kaggle and/or Google Colab to run your .ipynb notebooks.



### MLOps







- **MLOps** is a set of practices that helps manage and improve the process of **building, deploying, and maintaining machine learning models in real-world applications.**
- **MLflow, an MLOps tool, helps track hyperparameter tuning** by logging and comparing different model configurations.
- By using MLflow in your MLOps workflow, you can easily **identify and select the best-performing model based on logged metrics.**
- For a refresher, [here is the recording.](#)

## Streamlit App

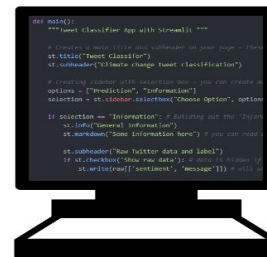


- Build **your own recommender app** using Streamlit's open source framework.
- The app could be outlined with pages/sections such as your team page, project overview, EDA, and more. Aim to **create a user-friendly interface**.
- Include **content-based filtering** and one pickled model for the **model-based collaborative filtering** approach.

Utility Matrix

	Items			
				
Bob	✓			✓
Xolisa	✓	✓		
Jacque				✓
Jon	✓		?	

MODEL



STREAMLIT



DEPLOY



## Streamlit App



- App should be able to **recommend using Content-Based and Collaborative-Based Filtering**. See the example below:



Select an algorithm

- ☒ Content Based Filtering  
☐ Collaborative Based Filtering

Enter Your Three Favorite Movies

First Option

So Proudly We Hail! (1943)

Second Option

Star Wars: Episode VII - The Force Awakens (2015)

Third Option

Ten Little Indians (Ein Unbekannter rechnet ab) (And Then There Were None) (1974)

Recommend

## Presentation



- Your final task will be to create a presentation **slide deck using Google Slides or Canva.**
- The slide deck could include an introduction, insights, recommenders systems, models, a demo of your Streamlit application, and a conclusion, along with other elements that help tell your story.
- Additionally, ensure you follow the rubric once it is provided to cover all required elements.
- Create a well-rounded presentation that presents these findings in a way that appeals to both a technical and non-technical audience.
- You will not need to present your slides.

### Communication and Project Management



- Use **email for formal communications**, such as reporting issues or team member statuses, and send facilitators the names of the Team Lead, Project Manager, and GitHub Manager.
- One team member should be designated as the **Project Manager to create and manage the Trello board**.
- For more information on using Trello, [watch this video](#).

## Week 1

- [Fill out this project form by Wednesday, 23 Oct '24 11:59 PM](#)
- Get to know your teammates and set out your team's ways of work.
- Elect a team lead and project manager (manages trello) (*Share this info with the facilitator your team is named after on Discord*).
- Set up all other collaborative and development tools required for the project ( Trello, 2 x GitHub repos and slide deck) (*Share this info with the facilitator your team is named after on Discord*).

## Week 2

- Data cleaning and EDA.
- Start creating your skeleton slide deck and Streamlit app (test if you can deploy the app).
- Start applying preprocessing tasks

### Week 3

- Train with a minimum of three models (notebook).
- Apply model evaluation metrics (*RMSE*) to compare model performance.

### Week 4

- Fine-tune model parameters and hyperparameters to achieve the best performance. Use MLflow to keep track of fine-tuning and optimizing your models.
- Add all relevant information to the notebook and slide deck.

### Week 5

- Make sure your Streamlit app includes all its features, including the recommender systems, and that it is ready for deployment.

### Week 6

- Finalise your notebook, README files for both GitHub repos, Streamlit app is deployed and slide deck.



Please find below, important links:

- Facilitator Github Usernames: Click [here](#)
- Managing Environments: Click [here](#)
- Creating environments from requirements.txt using "*conda create* ": Click [here](#)
- Jupyter notebook markdown cheatsheet: Click [here](#)
- Video on how to set-up your Trello board: click [here](#)
- MLFlow Guide: Click [here](#)
- Streamlit Guide: Click [here](#)
- Tutorial on Recommender systems: Click [here](#)
- Configuring Git Large File Storage: Click [here](#)

