

# NSF HDR ML Challenge Codabench Tutorial



PHY-2117997

Yuan-Tang Chou (UW)  
A3D3 Institute  
for HDR ML Challenge team

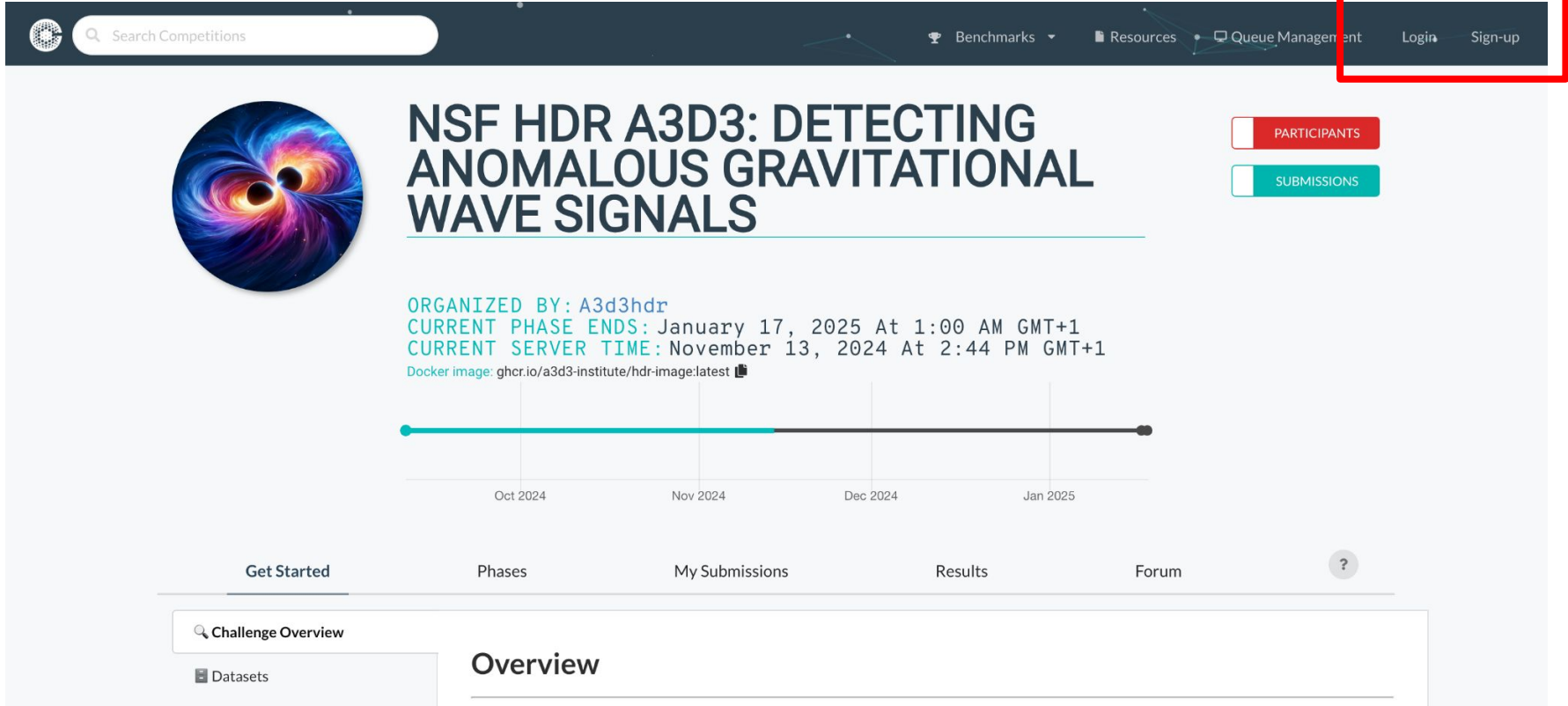


<https://a3d3.ai/>

# Reminder

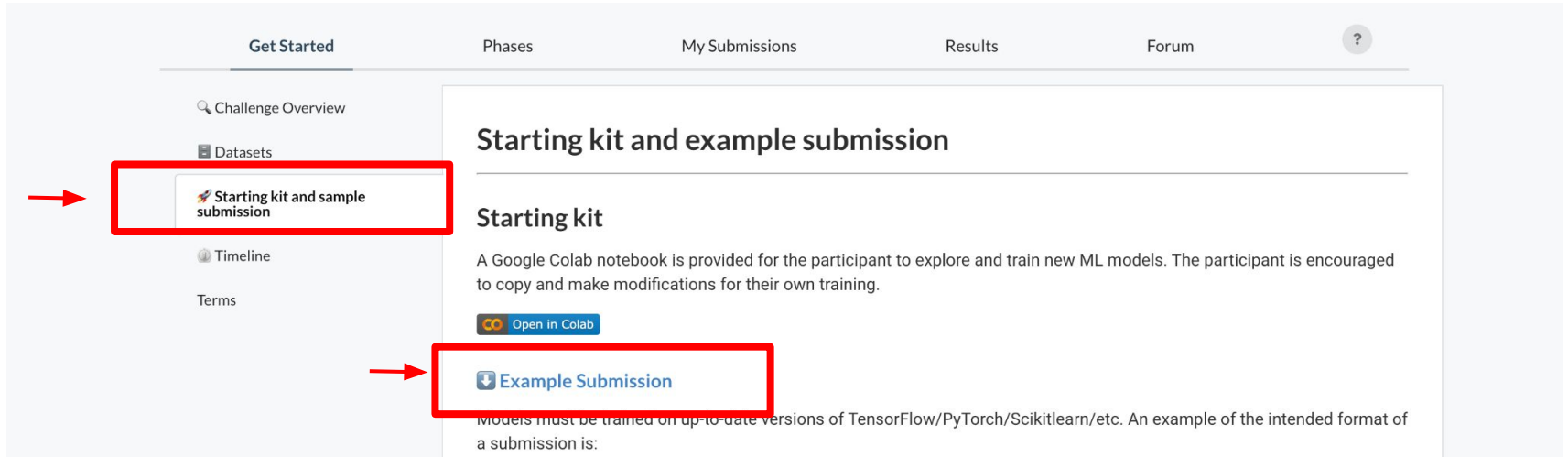
- The jobs submitted must be run manually and approved by **organizer team** in the systems. **Please inform us the time you want to run the hackathon.**
- We will create a Indico for all events <https://indico.cern.ch/category/19105/>
- Please let me know once the date is decided. (Yuan-tang Chou [ytchou@uw.edu](mailto:ytchou@uw.edu))
- Avoid running event during NERSC Perlmutter maintenance [[Link](#)] Next one is 12/18/24!
  
- Poster template: [[Link](#)]
- Google forum template [[Link](#)]
- Slack channel for local organizer support: <https://hdrecosystem.slack.com/archives/C07UJUWT1D5>

# 1. Login or Create Account on Codabench



The screenshot displays the Codabench website interface for a specific challenge. At the top, a dark navigation bar contains a search bar on the left and several menu items on the right: 'Benchmarks', 'Resources', 'Queue Management', 'Login', and 'Sign-up'. The 'Login' and 'Sign-up' buttons are highlighted with a red rectangular box, and a red arrow points downwards towards this box. Below the navigation bar, the main content area features a circular image of a gravitational well on the left. To its right, the challenge title 'NSF HDR A3D3: DETECTING ANOMALOUS GRAVITATIONAL WAVE SIGNALS' is prominently displayed. Further right, two buttons are visible: a red 'PARTICIPANTS' button and a teal 'SUBMISSIONS' button. Below the title, the text indicates the challenge is organized by 'A3d3hdr', with the current phase ending on January 17, 2025, and the server time as of November 13, 2024. A progress bar at the bottom of the challenge section shows the timeline from October 2024 to January 2025. At the very bottom, a navigation menu includes 'Get Started', 'Phases', 'My Submissions', 'Results', 'Forum', and a help icon. A search bar at the bottom left contains the text 'Challenge Overview' and 'Datasets'.

## 2. Download Dummy Submission



The screenshot shows a web interface with a navigation bar at the top containing 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A help icon is on the right. A left sidebar contains 'Challenge Overview', 'Datasets', 'Starting kit and sample submission', 'Timeline', and 'Terms'. The main content area has the heading 'Starting kit and example submission' and a sub-heading 'Starting kit'. Below this is a paragraph about a Google Colab notebook and an 'Open in Colab' button. At the bottom, there is an 'Example Submission' button with a download icon. Two red boxes highlight the 'Starting kit and sample submission' menu item and the 'Example Submission' button, with red arrows pointing to them from the left.

Get Started Phases My Submissions Results Forum ?

Challenge Overview

Datasets

Starting kit and sample submission

Timeline

Terms

### Starting kit and example submission

#### Starting kit

A Google Colab notebook is provided for the participant to explore and train new ML models. The participant is encouraged to copy and make modifications for their own training.

[Open in Colab](#)

[Example Submission](#)

Models must be trained on up-to-date versions of TensorFlow/PyTorch/Scikitlearn/etc. An example of the intended format of a submission is:

### 3. Register in the Competition

The screenshot shows a navigation bar with the following items: "Get Started", "Phases", "My Submissions" (highlighted with a red arrow), "Results", "Forum", and a help icon (a question mark in a circle). Below the navigation bar, the text reads: "You have not yet registered for this competition." followed by "To participate in this competition, you must accept its specific [terms and conditions](#). This competition **does not** require approval, once you register, you will immediately be able to participate." A red box highlights the registration form, which contains a checkbox with the text "I accept the terms and conditions of the competition." and a blue "Register" button. A red arrow points to the "Register" button.

## 4. Submit Dummy Submission

The screenshot shows a web interface for managing submissions. At the top, there are navigation tabs: "Get Started", "Phases", "My Submissions", "Results", and "Forum". A red arrow points to the "My Submissions" tab. Below the tabs, there are two buttons: "Development Phase" and "Final Phase".

Below the buttons, there are two summary cards:

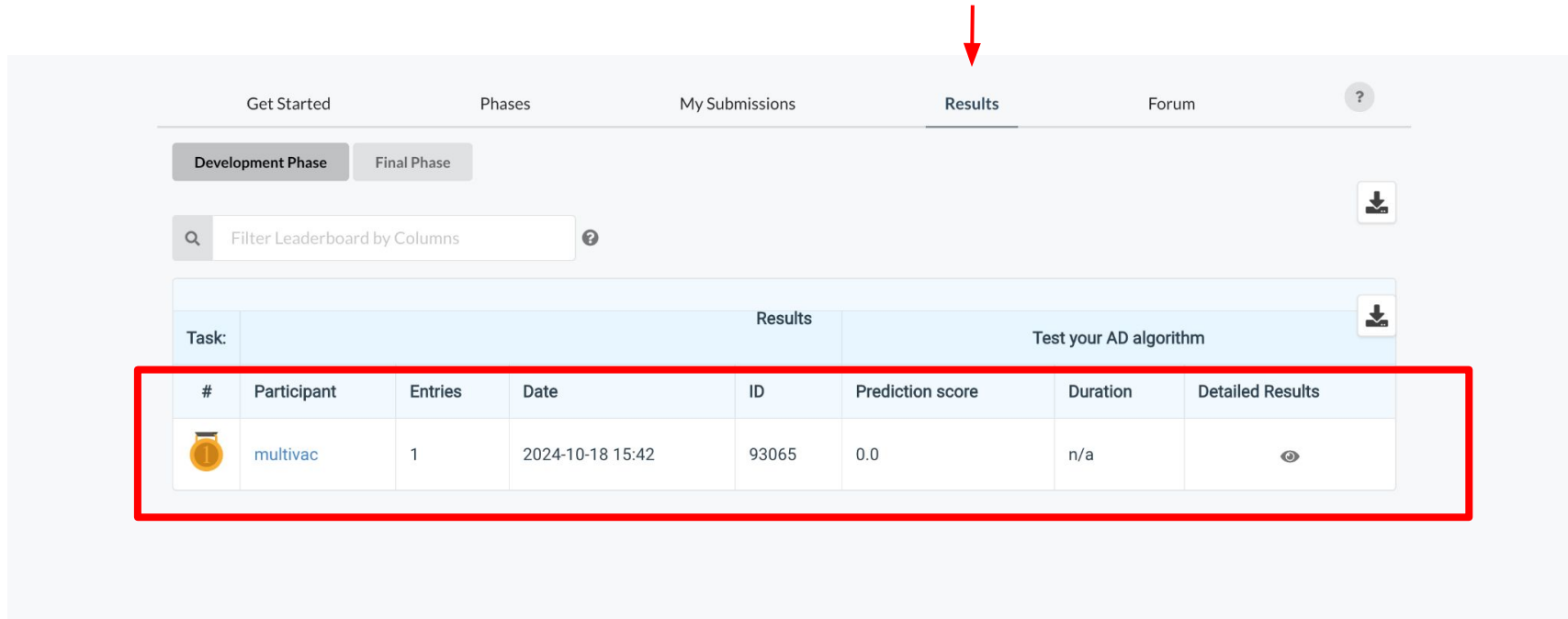
- Number of submissions used for the day:** 0 out of 500
- Number of total submissions used:** 0 out of 1000

Below the summary cards is the "Submission upload" section. It includes a "Submit as:" dropdown menu with "Yourself" selected. A red arrow points to this dropdown. Below the dropdown is a large text input field with a file upload icon (a paperclip) on the left. This input field is highlighted with a red rectangular border.



Below the input field, there is a search bar with "Search..." and a magnifying glass icon, and a "Status" dropdown menu.

At the bottom, there is a table with the following columns: "ID #", "File name", "Date", "Status", "Score", "Detailed Results", and "Actions". Below the table, there is a message: "No submissions found! Please make a submission".

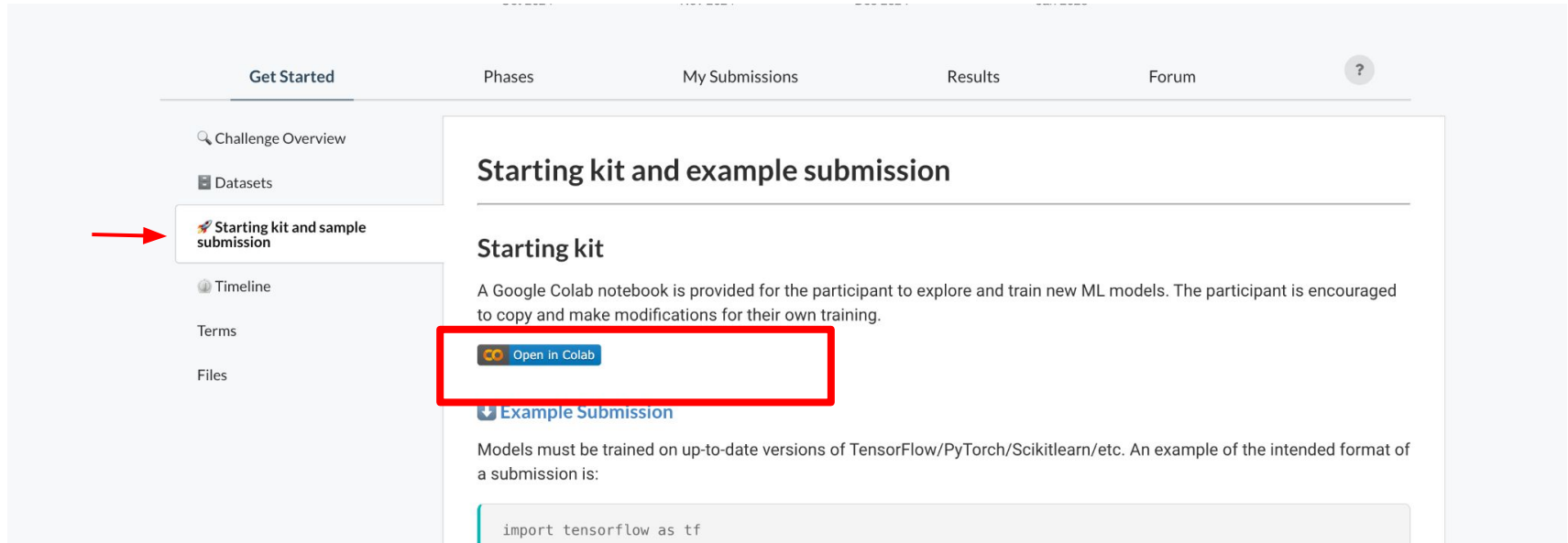
## 5. Check results in the leaderboard



The screenshot shows a web interface with a navigation bar at the top containing 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A red arrow points to the 'Results' tab. Below the navigation bar, there are two buttons: 'Development Phase' and 'Final Phase'. A search bar labeled 'Filter Leaderboard by Columns' is present. The main content area features a table with a light blue header. The table has a 'Task' column and a 'Results' column. The 'Results' column is further divided into 'ID', 'Prediction score', 'Duration', and 'Detailed Results'. The first row of the table is highlighted with a red border and contains the following data:

#	Participant	Entries	Date	ID	Prediction score	Duration	Detailed Results
	<a href="#">multivac</a>	1	2024-10-18 15:42	93065	0.0	n/a	

## 6. Check out the starting kit



The screenshot displays a web interface for a challenge. The top navigation bar includes 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A sidebar on the left lists navigation options: 'Challenge Overview', 'Datasets', 'Starting kit and sample submission' (highlighted with a red arrow), 'Timeline', 'Terms', and 'Files'. The main content area is titled 'Starting kit and example submission' and contains the following text:

### Starting kit

A Google Colab notebook is provided for the participant to explore and train new ML models. The participant is encouraged to copy and make modifications for their own training.

[Open in Colab](#)

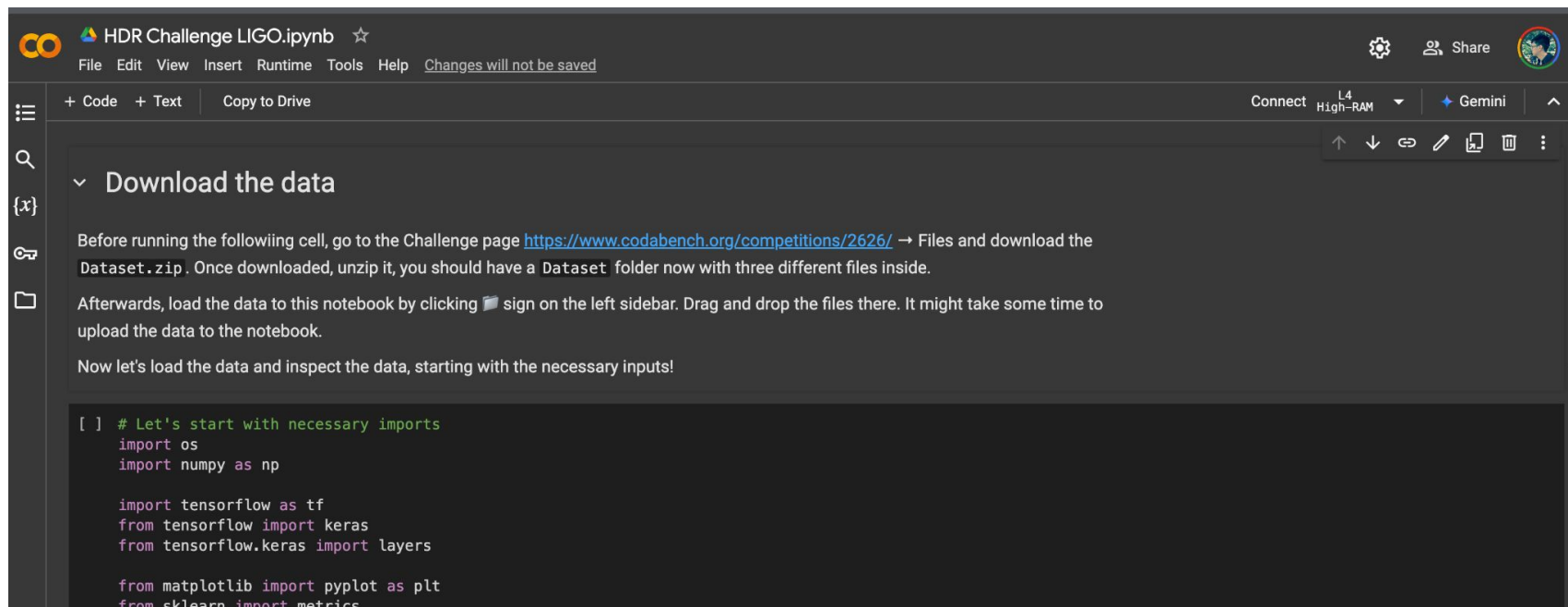
### Example Submission

Models must be trained on up-to-date versions of TensorFlow/PyTorch/Scikitlearn/etc. An example of the intended format of a submission is:

```
import tensorflow as tf
```




# 7. Starting kit as a Google Colab Notebook



The screenshot shows a Google Colab notebook interface. At the top, the title bar reads "HDR Challenge LIGO.ipynb" with a star icon. Below the title bar is a menu bar with options: File, Edit, View, Insert, Runtime, Tools, Help, and a status message "Changes will not be saved". On the right side of the title bar, there are icons for settings, a "Share" button, and a globe icon. Below the title bar, there are tabs for "+ Code", "+ Text", and "Copy to Drive". On the far right of this bar, there are "Connect" and "High-RAM" options, and a "Gemini" button. The main content area is titled "Download the data" and contains the following text:

Before running the following cell, go to the Challenge page <https://www.codabench.org/competitions/2626/> → Files and download the Dataset.zip. Once downloaded, unzip it, you should have a Dataset folder now with three different files inside.

Afterwards, load the data to this notebook by clicking  sign on the left sidebar. Drag and drop the files there. It might take some time to upload the data to the notebook.

Now let's load the data and inspect the data, starting with the necessary inputs!

```
[ ] # Let's start with necessary imports
import os
import numpy as np

import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers

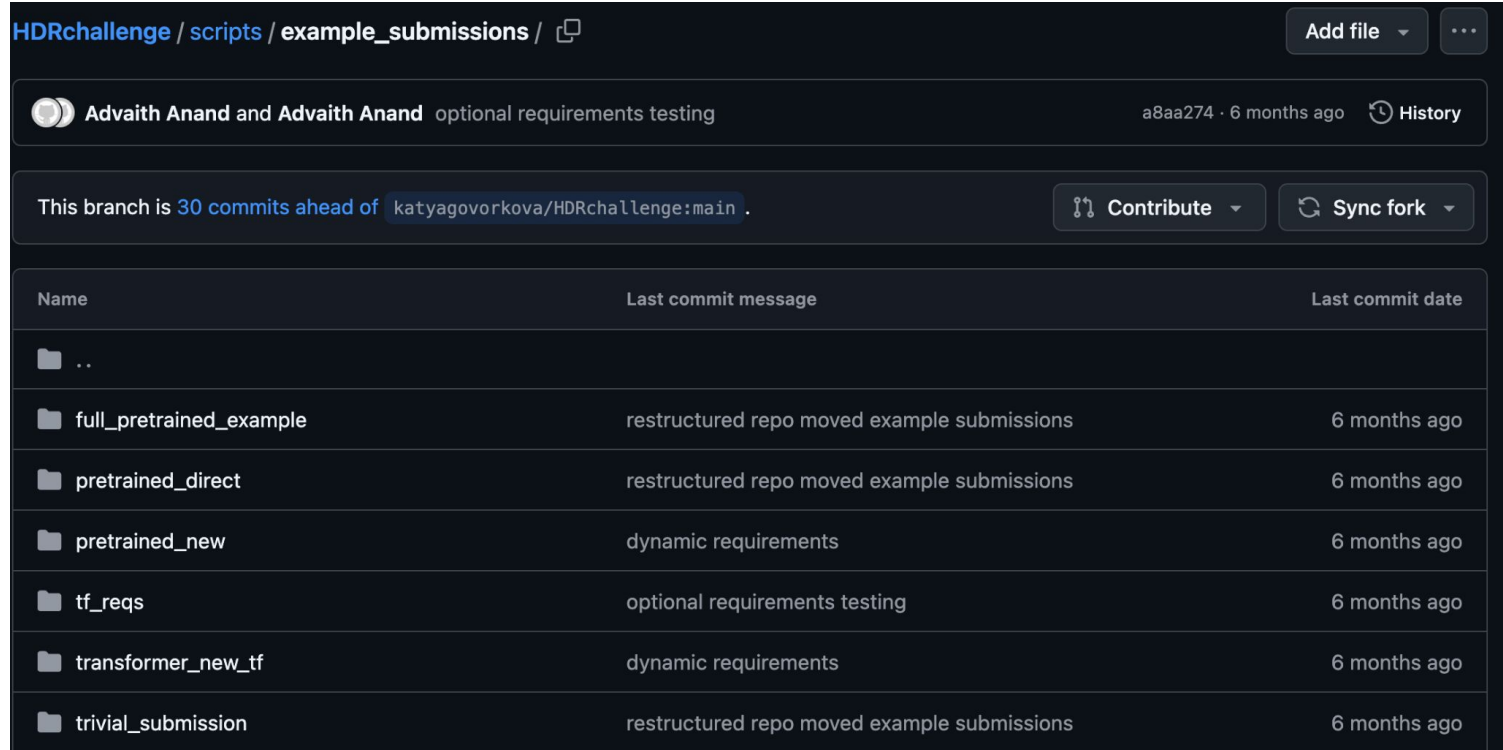
from matplotlib import pyplot as plt
from sklearn import metrics
```

# 8. Get Public Data

The screenshot shows a challenge interface with a timeline at the top and a navigation menu on the left. The timeline spans from October 2024 to January 2025. The navigation menu includes 'Challenge Overview', 'Datasets', 'Starting kit and sample submission', 'Timeline', 'Terms', and 'Files'. A red arrow points to the 'Files' menu item. The main content area displays a table with the following data:

Download	Phase	Task	Type	Size
<a href="#">solution @ 04-09-2024 19:28</a>	Development Phase	Test your AD algorithm	Solution	522 B
<a href="#">Dataset</a>	Development Phase	-	Public Data	473.26 MB

# 9. Checkout example submissions



The screenshot shows a GitHub repository page for 'HDRchallenge / scripts / example\_submissions'. The repository is owned by 'Advaith Anand and Advaith Anand' and is currently on a branch that is 30 commits ahead of the main branch of the upstream repository 'katyagovorkova/HDRchallenge:main'. The page displays a list of folders with their last commit messages and dates.

Name	Last commit message	Last commit date
..		
full_pretrained_example	restructured repo moved example submissions	6 months ago
pretrained_direct	restructured repo moved example submissions	6 months ago
pretrained_new	dynamic requirements	6 months ago
tf_reqs	optional requirements testing	6 months ago
transformer_new_tf	dynamic requirements	6 months ago
trivial_submission	restructured repo moved example submissions	6 months ago

# 11. Code submission structure [\[Example\]](#)

```
1 import tensorflow as tf
2 import json
3 import os
4
5 class Model:
6     def __init__(self):
7         # You could include a constructor to initialize your model here, but all calls will be made to the load meth
8         self.clf = None
9
10    def predict(self, X):
11        # This method should accept an input of any size (of the given input format) and return predictions appropri
12        preds = self.clf.predict(X)
13        print(preds)
14        return preds
15
16    def load(self):
17        # This method should load your pretrained model from wherever you have it saved
18
19        with open(os.path.join(os.path.dirname(__file__), 'config.json'), 'r') as file:
20            for line in file:
21                self.clf = tf.keras.models.model_from_json(line)
22            self.clf.load_weights(os.path.join(os.path.dirname(__file__), 'model.weights.h5'))
```

[\*] Do not zip the whole folder. Select the model.py and relevant files to make the tarball

[\*] Follow the example to load your model. Avoid hard-coded path to model weight



# NSF HDR ML Challenge

