

HDR ML Challenge

*Detect Anomalies in Science
with Machine Learning!*



Carnegie
Mellon
University

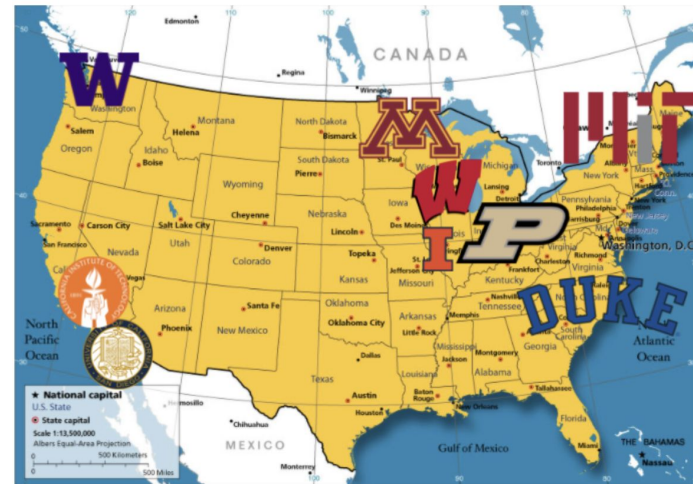
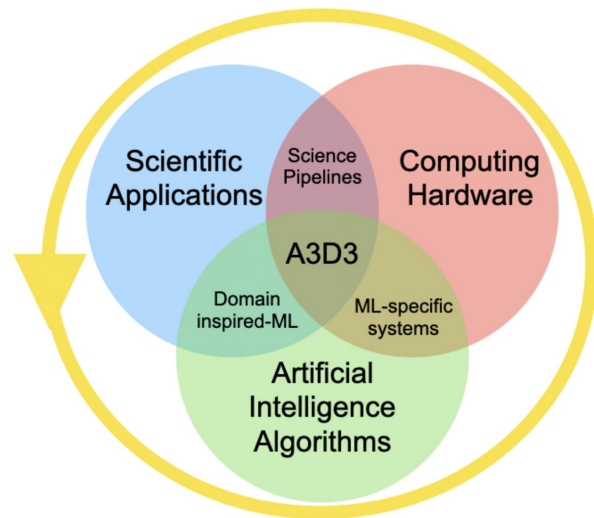


December 2nd, 2024

Harnessing the Data Revolution

- The HDR program aims to develop **AI solutions** to accelerate science & discovery
 - Funded by National Science Foundation (**NSF**)
 - Active, multi-disciplinary collaboration via the:

*Accelerated AI Algorithms for Data-Driven Discovery (**A3D3**) Institute*



CMU only recently joined A3D3!

Harnessing the Data Revolution

- The HDR program aims to develop **AI solutions** to accelerate science & discovery
 - Funded by National Science Foundation ([NSF](#))
 - Active, multi-disciplinary collaboration via the:

Accelerated AI Algorithms for Data-Driven Discovery ([A3D3](#)) Institute

- **A3D3 challenges YOU** to develop Machine Learning models for **anomaly detection!**
 - Detect hybrid butterfly species
 - Find unmodeled gravitational waves
 - Detect unusual fluctuations in water levels



- A solid prize pool, including funded invitations to the [AAAI 2025 conference!](#)
(Challenge last until 2025/01/17)



Prize Pool for HDR ML Challenges	
💰	Total cash prizes \$2500
👤	\$3000 in AWS cloud computing credits
🏆	Extra award sponsored by AMD (details pending)
🌟	Special jury prizes include funded invitations to AAAI 2025
Potential for additional award.	

Harnessing the Data Revolution



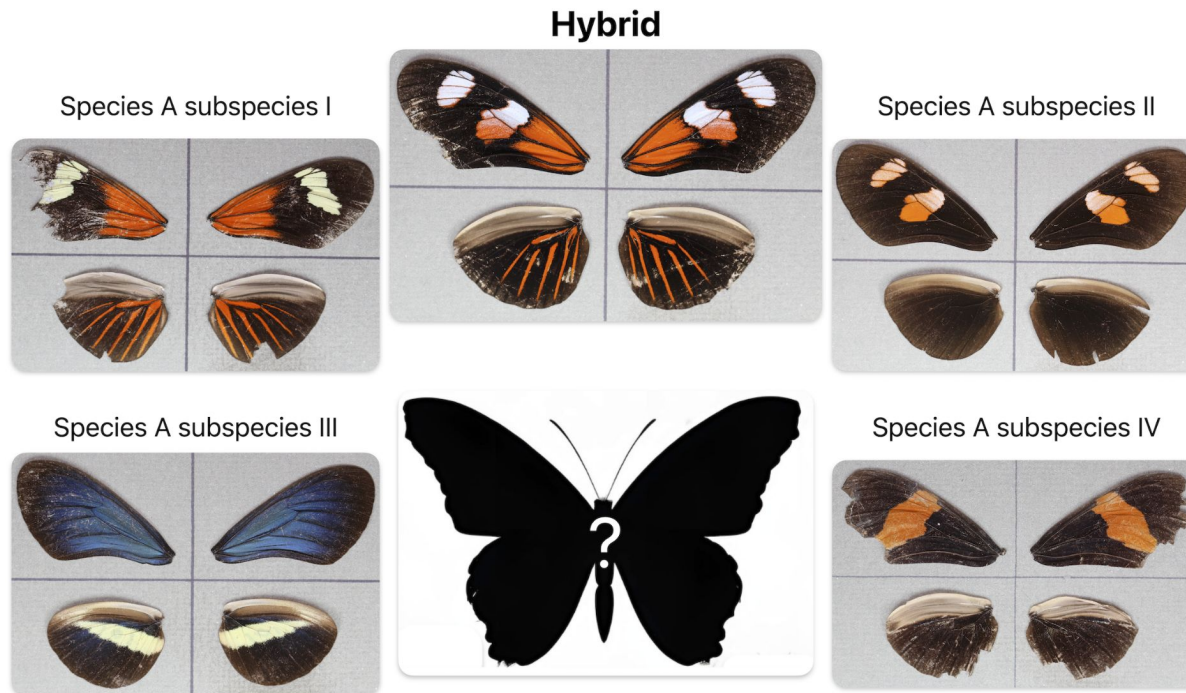
Agenda

- Harnessing the Data Revolution
- Introduction to the challenges
 - Butterfly Hybrid Detection
 - Sea Level Anomaly detection
 - Detecting Anomalous Gravitational Wave
- Model Submission Platform
- References & Practicalities

The Challenges

Butterfly Hybrid Detection

- Hybridization may lead to a variety of resulting butterfly wing patterns
 - Identifying hybrids requires knowledge of their parent species/subspecies
- Can ML models automatically identify (unseen) hybrid cases?



Butterfly Hybrid Detection

- All information + a starter kit given in [codabench](#)
- **Training data** = ~2200 images of Species A
 - Includes:
 - Multiple *sub*species.
 - Selected signal hybrids of two *sub*species
- **Test/Dev data** = ~1100 images
 - Includes:
 - All Species A subspecies.
 - Signal hybrids from training data.
 - Further introduces:
 - Other Species A hybrids (non-signal).
 - Species B: Mimics of Species A signal hybrid parents (& their hybrids).

Butterfly Hybrid Detection

- Among Species A & B, can your algorithm find...
 - Species A signal hybrids?
 - Species A non signal hybrids?
 - Species B hybrids (mimics of Species A signal hybrids)?

Species A subspecies I



Species A subspecies II



Species A subspecies III



Species A subspecies IV



Species B subspecies II



Species B subspecies I



Sea-Level Anomaly Detection



AS OUR OCEAN WARMS, SEA LEVEL RISES

We know seas are rising and we know why. The urgent questions are by how much and how quickly.



Sea-Level Anomaly Detection

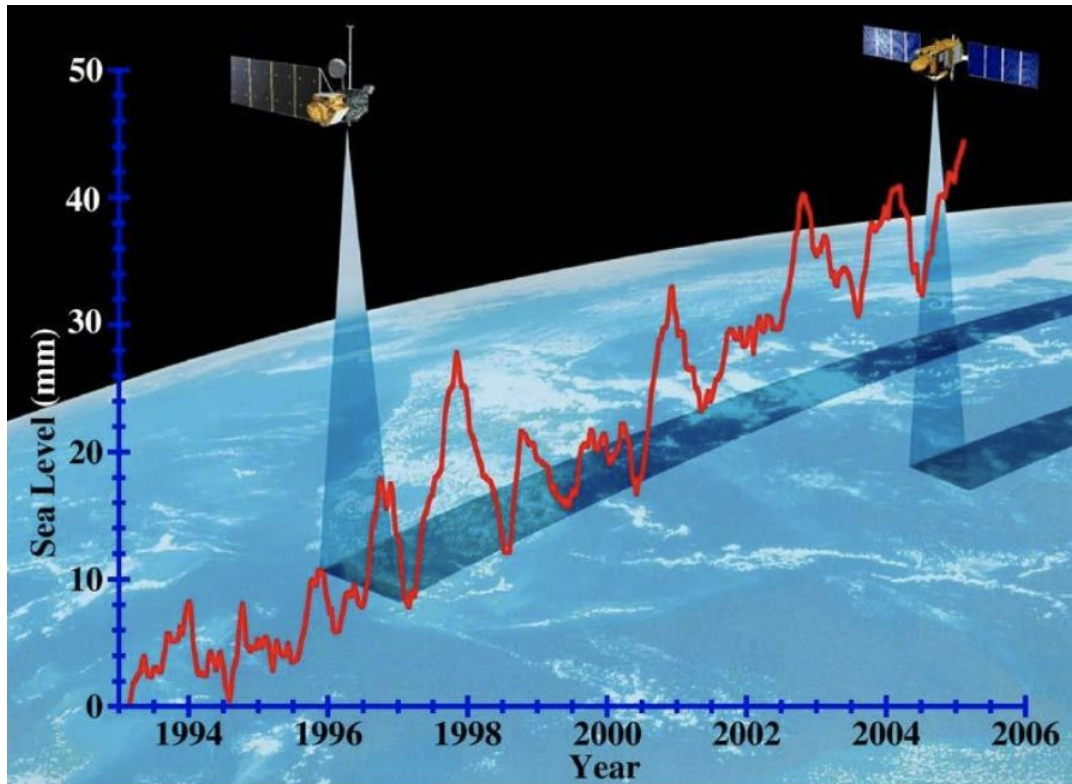
SEA LEVEL RISE AFFECTS US ALL

More than **160 million people** live along coasts in the U.S., about half the nation's population. **Eleven of the world's 15 largest cities** lie along shores, including New York City. Sea level rise means the ocean will gradually inundate low-lying areas, and storms like hurricanes, bolstered by even higher seas, will extend their reach inland. All of society bears the burden for storm damage and those costs are expected to rise: Annual losses from flooding in the world's biggest coastal cities could rise from about **\$6 billion a year** today to **\$1 trillion a year** by 2050.



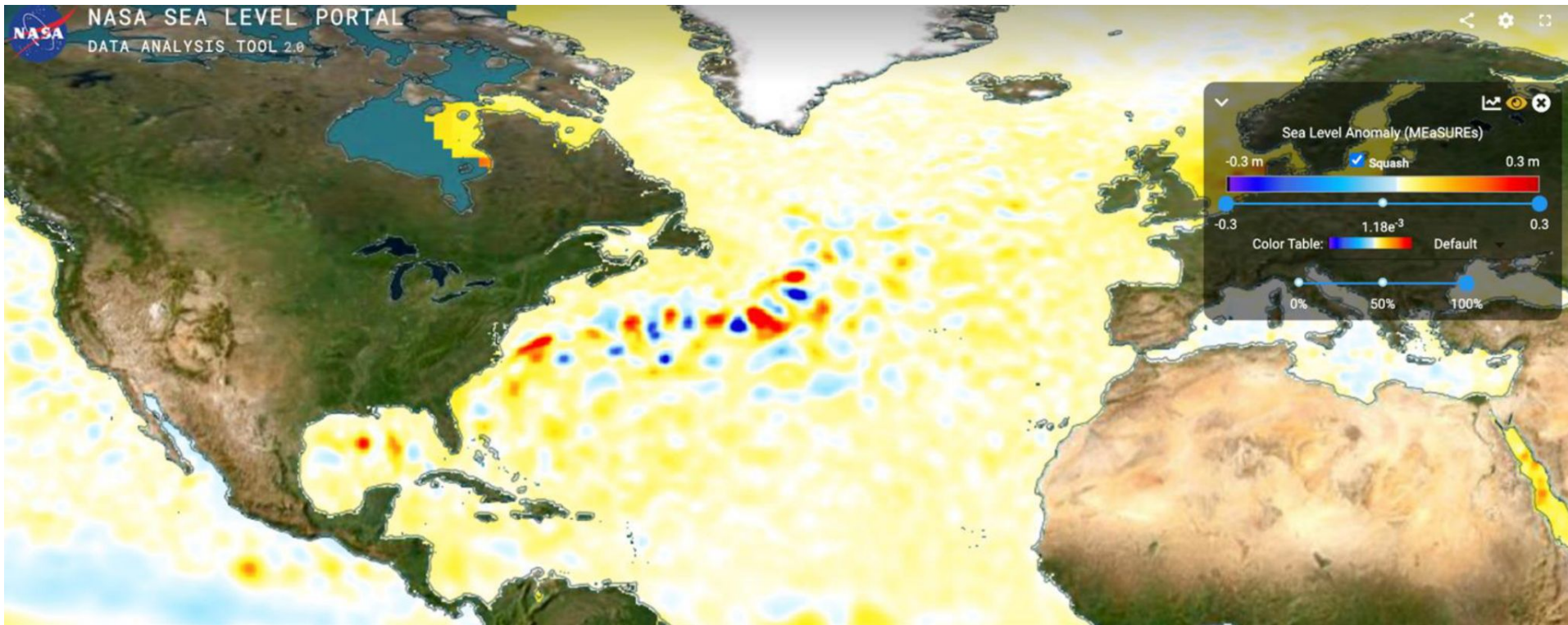
Sea-Level Anomaly Detection

- As the ocean rises, the ability to provide even more precise information about coastal sea level rise is crucial



Sea-Level Anomaly Detection

- **Goal:** Detect anomalous flooding events along the US East Coast with the maps of sea-level over the North Atlantic

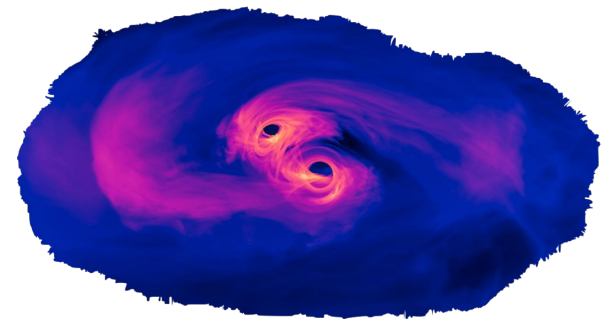
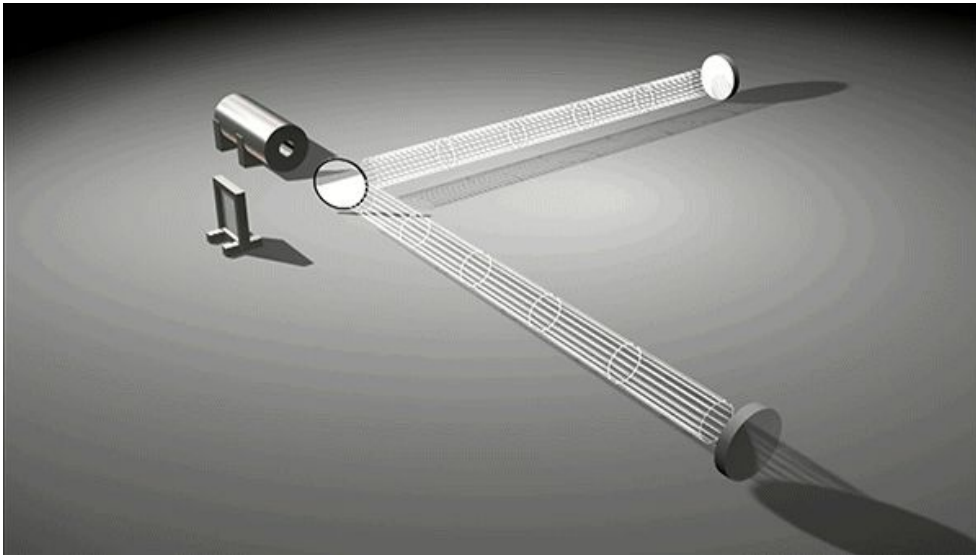


Sea-Level Anomaly Detection

- **Goal:** Detect anomalous flooding events along the US East Coast with the maps of sea-level over the North Atlantic
- All information + a starter kit given in [codabench](#)
- Data provided:
 - Satellite maps of sea level data over the North Atlantic for the past 30 years
 - Labeled Anomalous Flood: Dates of anomalous flooding along the US East Coast

Anomalous Gravitational Wave Detection

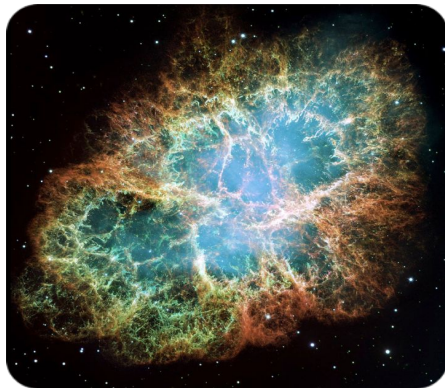
- Accelerating masses produce deformations in space-time that we can detect via interferometers
 - Multiple large-scale interferometers throughout the world (LIGO, VIRGO, ...)
 - A signal will appear in at least two interferometers, with small time delay



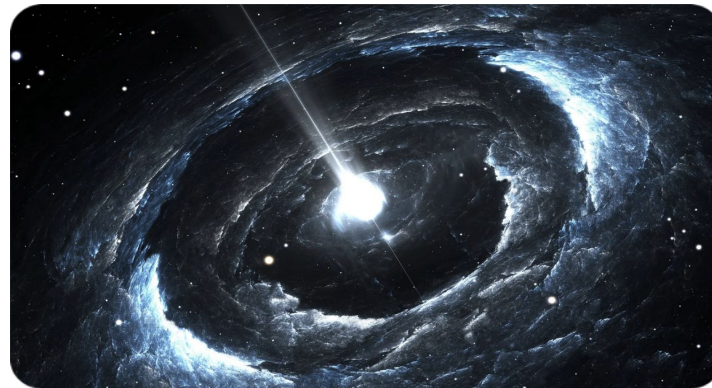
Anomalous Gravitational Wave Detection

- Accelerating masses produce deformations in space-time that we can detect via interferometers
 - Multiple large-scale interferometers throughout the world (LIGO, VIRGO, ...)
 - A signal will appear in at least two interferometers, with small time delay

CORE-COLLAPSE SUPERNOVA



NEUTRON STAR GLITCHES

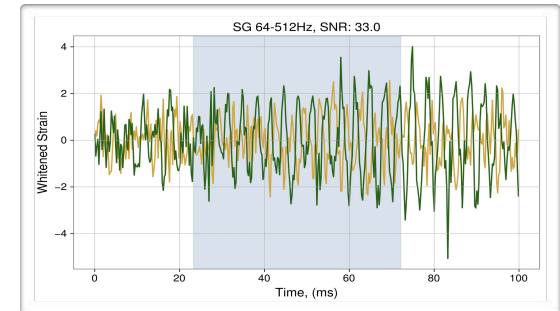


Known unknowns: We know they exist, but they are poorly modeled, so hard to detect!

Unknown unknowns: New anomalous GW sources we haven't thought off...

Anomalous Gravitational Wave Detection

- **Goal:** Design a ML model that identifies anomalous gravitational waves from (un)known unknowns

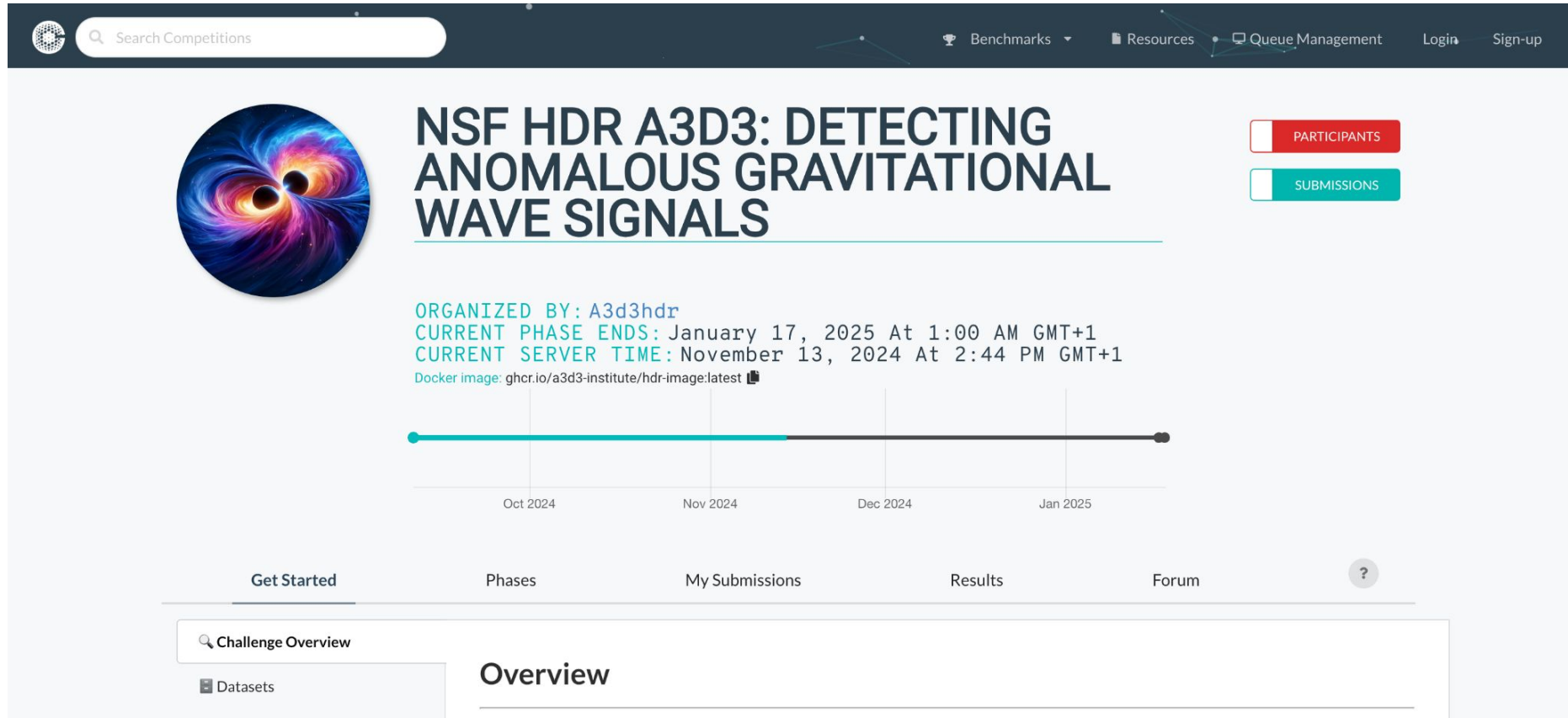


- All information + a starter kit given in [codabench](#)
- Data provided = Continuous time series of interferometer data at 4096 Hz
 - Already pre-processed: whitened & band-passed ($30 < f < 1500$ Hz)
 - Divided into 50 ms segments, each containing 200 data points
($50 \text{ ms} * 4096 \text{ samples/second} = 200 \text{ samples}$)
 - Dimension of input data is $(N, 200, 2)$, with N representing the number of data segments. The last dimension of 2 corresponds to the datastreams from the two LIGO interferometers in **Hanford (WA)** & **Livingston (LA)**

Submission Platform



1. Login or Create Account on Codabench



The screenshot displays the Codabench website interface for the NSF HDR A3D3 challenge. At the top, there is a dark navigation bar with a search bar on the left and links for Benchmarks, Resources, Queue Management, Login, and Sign-up on the right. The main content area features a large 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. Below the title, two progress bars are shown: a red one for "PARTICIPANTS" and a teal one for "SUBMISSIONS". Further down, the text indicates the challenge is organized by A3d3hdr, with the current phase ending on January 17, 2025, and the current server time being November 13, 2024. A timeline below this text shows the phase progress from October 2024 to January 2025. At the bottom, a navigation menu includes "Get Started", "Phases", "My Submissions", "Results", "Forum", and a help icon. On the left side of the page, there is a sidebar with a search bar and a "Datasets" link.

Search Competitions

Benchmarks Resources Queue Management Login Sign-up

NSF HDR A3D3: DETECTING ANOMALOUS GRAVITATIONAL WAVE SIGNALS

PARTICIPANTS

SUBMISSIONS

ORGANIZED BY: A3d3hdr
CURRENT PHASE ENDS: January 17, 2025 At 1:00 AM GMT+1
CURRENT SERVER TIME: November 13, 2024 At 2:44 PM GMT+1
Docker image: ghcr.io/a3d3-institute/hdr-image:latest

Oct 2024 Nov 2024 Dec 2024 Jan 2025

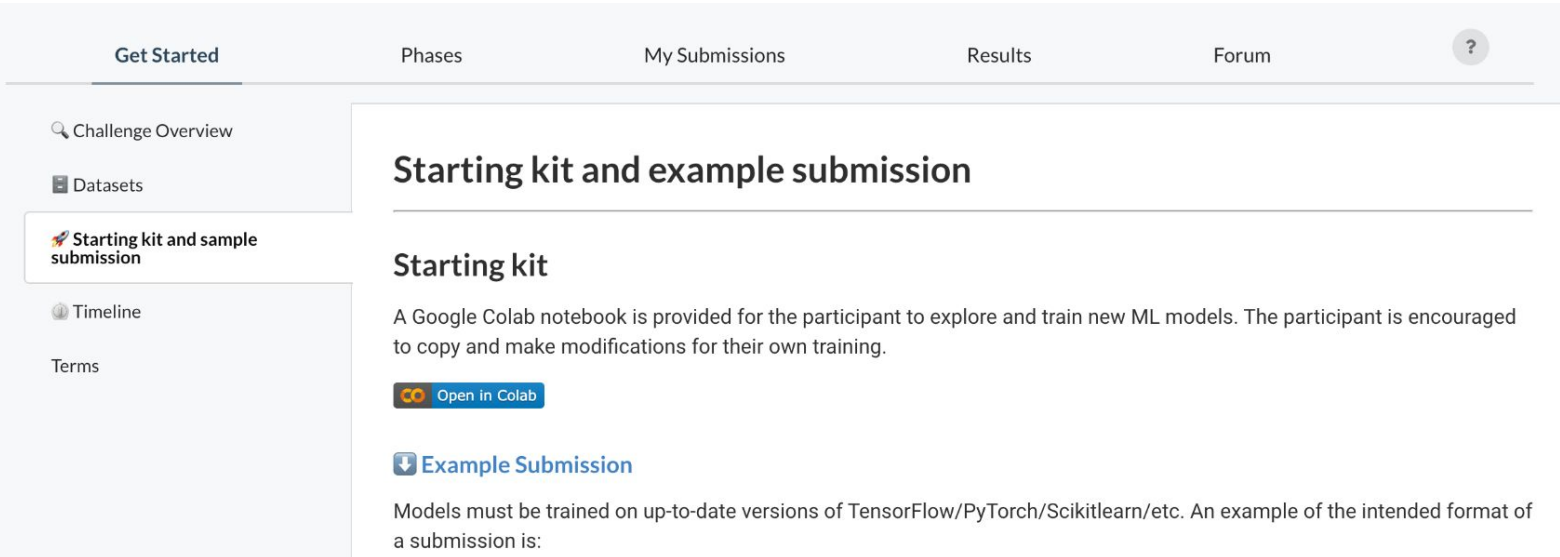
Get Started Phases My Submissions Results Forum ?

Challenge Overview

Datasets

Overview

2. Download Dummy Submission



The screenshot shows a web interface for a challenge. At the top, there is a navigation bar with tabs: 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A help icon (?) is also present. On the left side, there is a sidebar menu with the following items: 'Challenge Overview', 'Datasets', 'Starting kit and sample submission' (which is highlighted), 'Timeline', and 'Terms'. The main content area is titled 'Starting kit and example submission'. Under this title, there is a sub-section 'Starting kit' with a paragraph of text: '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.' Below this text is a blue button with the Google Colab logo and the text 'Open in Colab'. Further down, there is a blue link with a download icon and the text 'Example Submission'. Below this link is another paragraph: '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

Get Started

Phases

My Submissions

Results

Forum



You have not yet registered for this competition.

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.

I accept the terms and conditions of the competition.

Register

4. Submit Dummy Submission

Get Started Phases **My Submissions** Results Forum ?


Development Phase Final Phase


? Number of submissions used for the day 0 out of 500 Number of total submissions used 0 out of 1000

Submission upload

Submit as: ?

Yourself



Search...  Status



ID # ▾	File name	Date	Status	Score	Detailed Results	Actions
<i>No submissions found! Please make a submission</i>						

5. Check results in the leaderboard

Get Started Phases My Submissions **Results** Forum ?

Development Phase Final Phase

Filter Leaderboard by Columns ?

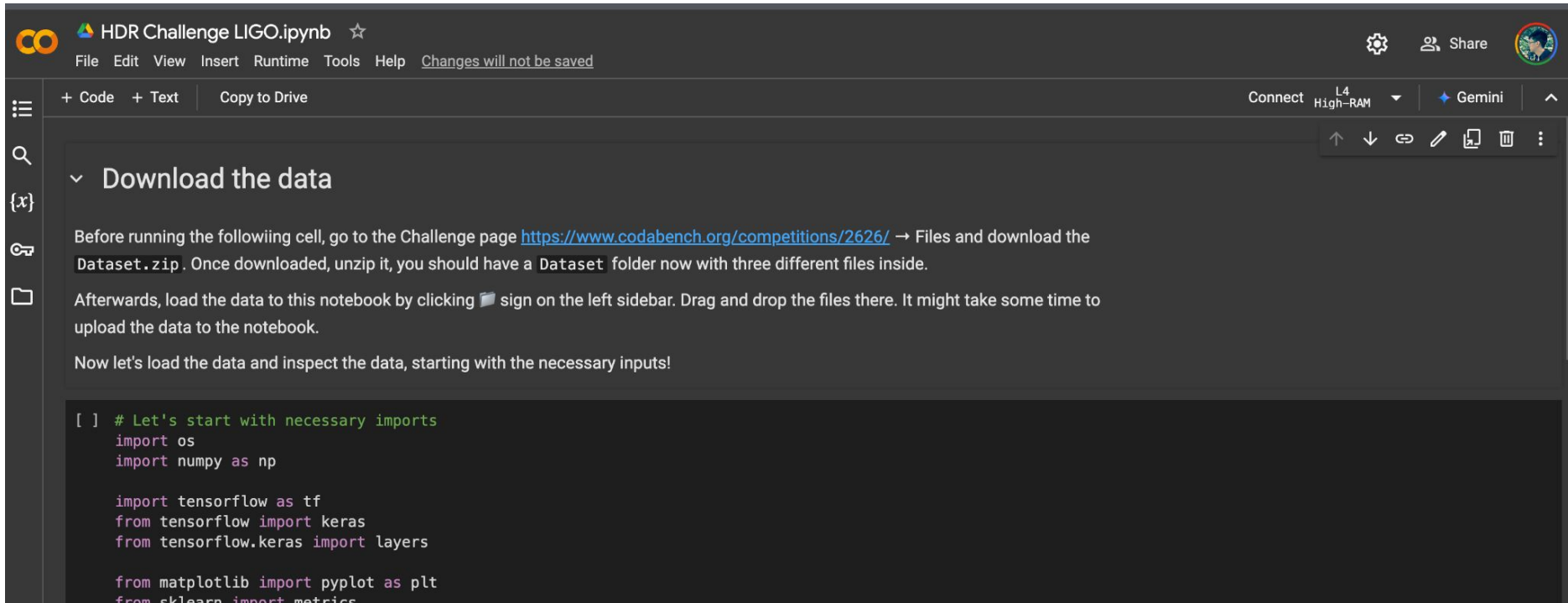
Task:		Results			Test your AD algorithm		
#	Participant	Entries	Date	ID	Prediction score	Duration	Detailed Results
	multivac	1	2024-10-18 15:42	93065	0.0	n/a	

6. Check out the starting kit

The screenshot shows a web interface for a competition. At the top, there are navigation tabs: 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A help icon (?) is also present. On the left side, there is a sidebar menu with the following items: 'Challenge Overview', 'Datasets', 'Starting kit and sample submission' (which is highlighted), 'Timeline', 'Terms', and 'Files'. The main content area is titled 'Starting kit and example submission'. Under the 'Starting kit' sub-heading, it states: '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.' Below this text is a blue button with the Colab logo and the text 'Open in Colab'. Under the 'Example Submission' sub-heading, it states: 'Models must be trained on up-to-date versions of TensorFlow/PyTorch/Scikitlearn/etc. An example of the intended format of a submission is:'. Below this text is a code block containing the line:

```
import tensorflow as tf
```

7. Starting kit as a Google Colab Notebook



HDR Challenge LIGO.ipynb ☆


File Edit View Insert Runtime Tools Help Changes will not be saved

+ Code + Text Copy to Drive

Connect L4 High-RAM Gemini

Download the data

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

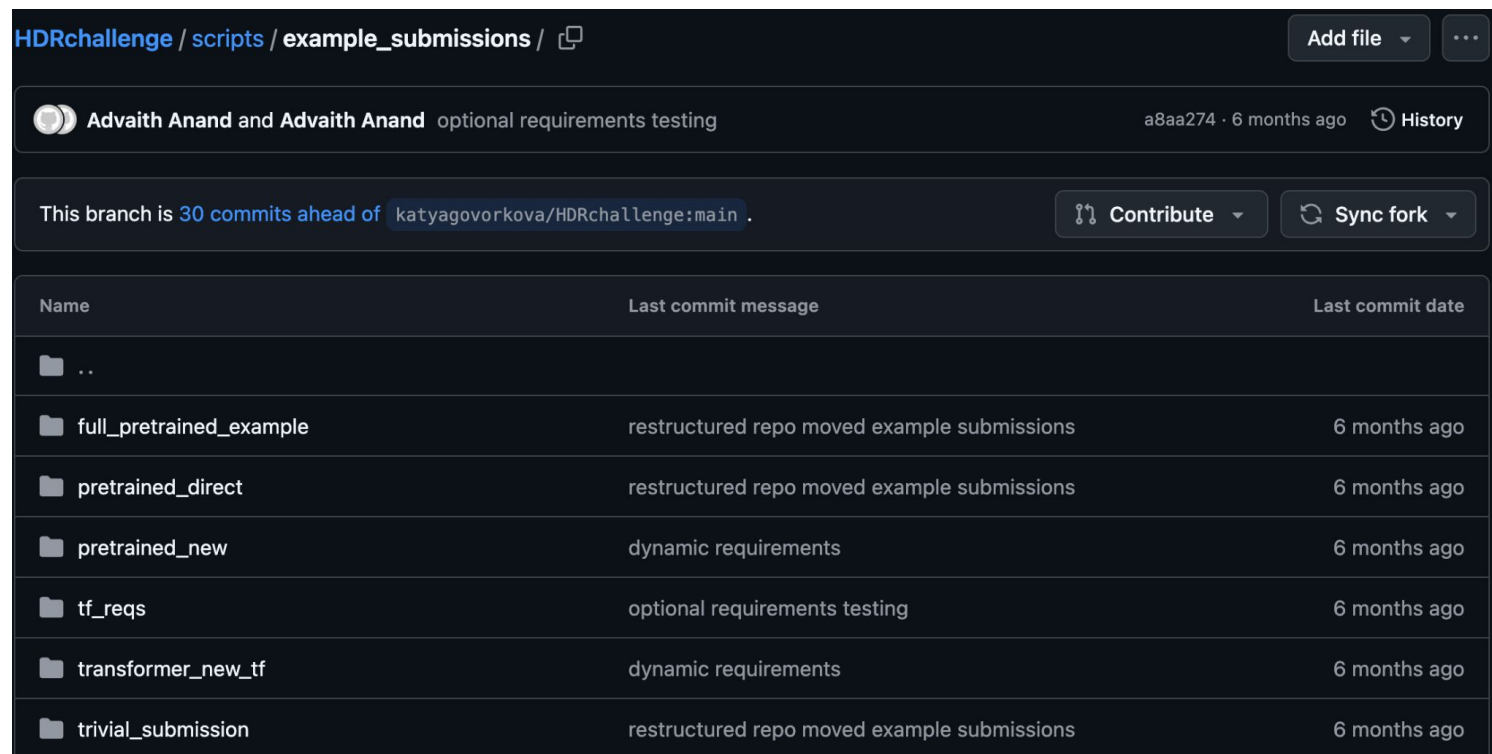
from matplotlib import pyplot as plt
from sklearn import metrics
```

8. Get Public Data

The screenshot displays a challenge interface. At the top, a horizontal timeline shows the period from October 2024 to January 2025. Below the timeline is a navigation bar with tabs for 'Get Started', 'Phases', 'My Submissions', 'Results', and 'Forum'. A sidebar on the left contains a search icon and links to 'Challenge Overview', 'Datasets', 'Starting kit and sample submission', 'Timeline', 'Terms', and 'Files'. The main content area features a table with the following data:

Download	Phase	Task	Type	Size
solution @ 04-09-2024 19:28	Development Phase	Test your AD algorithm	Solution	522 B
Dataset	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 titled 'optional requirements testing'. The current branch is 'a8aa274' and was updated '6 months ago'. The repository is 30 commits ahead of the upstream 'katyagovorkova/HDRchallenge:main' branch. There are buttons for 'Contribute' and 'Sync fork'. Below this, a table lists the files and folders in the repository, along 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

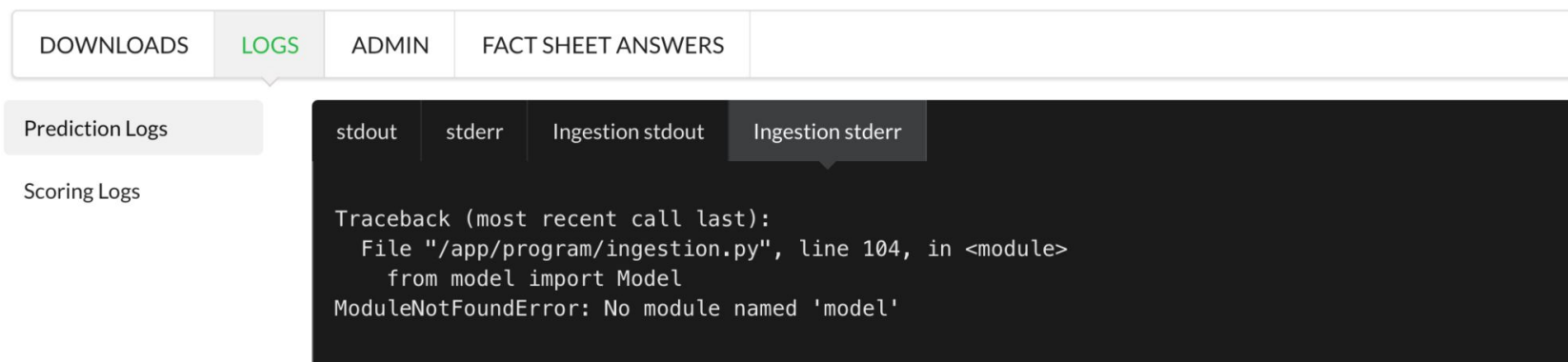
10. Code submission structure [[example](#)]

```
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'))
```

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

Common issue

[!!] Do not zip the whole folder. ONLY select the model.py and relevant weight files to make the tarball



DOWNLOADS LOGS ADMIN FACT SHEET ANSWERS

Prediction Logs

Scoring Logs

stdout stderr Ingestion stdout Ingestion stderr

```
Traceback (most recent call last):
  File "/app/program/ingestion.py", line 104, in <module>
    from model import Model
ModuleNotFoundError: No module named 'model'
```

If you see the above error, mostly likely you zip the whole folder when making the tarball

References & Practicalities

References & Practicalities

- HDR ML Hackathon page: <https://www.nsfhdr.org/mlchallenge>
 - Today's event page: <https://indico.cern.ch/event/1482320/>
 - **Model submissions are accepted until January 17th, 2025**
- Details and starter-kits for each challenge:
 - [Butterfly hybrid detection](#)
 - [Sea-level anomaly detection](#)
 - [Gravitational wave detection](#)
- GPU resources are being organised through the Pittsburgh Supercomputing Center

HUGE thank you to them!



PSC Resources Overview

Allocations



PSC Bridges-2 RM | 300 Core Hours

PSC Bridges-2 GPU | 1k GPU Hours

- Set up an ACCESS account: <https://operations.access-ci.org/identity/new-user>
 - Email Zach with your ACCESS ID once it is set up: zbaldwin@cmu.edu
- The best resource is the [PSC Bridges-2 User's Guide](#) which provides all the necessary information on things like
 - Password Management/Acceptable Use Policy
 - Support/Reporting a problem (help@psc.edu)
 - Usage (environments, software, etc.)
- If any issues arise with PSC resources, email Zach!

or cc him in messages with support

PSC Quick Look

All information provided here (and more) can be found in extensive detail in the [User's guide](#)

- Connect to Bridges-2 login node (web browser or command line)
 - [OnDemand](#)
 - Command line interface | `ssh -Y <username>@bridges2.psc.edu`

`$HOME` - User's home directory

`$LOCAL` - Local file system (only visible to the node the current system is attached to)

`$PROJECT` - File storage

Transferring files - rsync, scp, sftp or Globus

Running Jobs - interactive, batch, or OnDemand

OpenMP & OpenACC available!

Backup
