

Introduction to Wandb

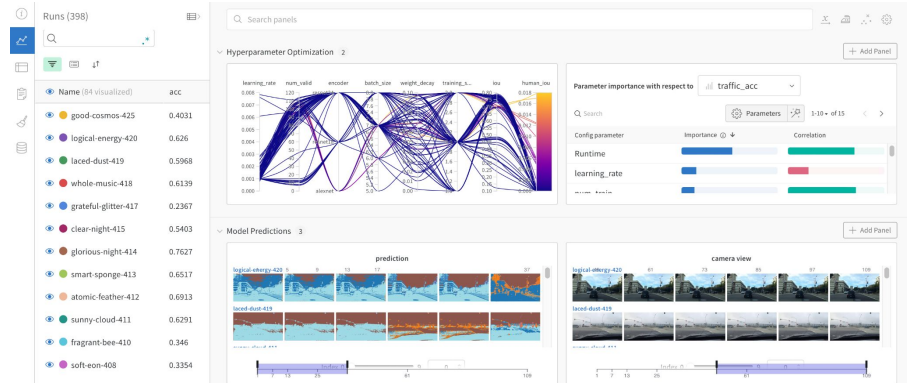


By Michael Kireeff and Aisha Opaluwa

What is WandB?

WandB, or Weights and Biases, is a powerful tool designed to help data scientist and machine learning engineers to track and manage their experiments easily through performance visualization and hyperparameter tuning.

It integrates seamlessly with popular machine learning frameworks like PyTorch and Tensorflow through its client library.



What are the Key Features of WandB

Performance Logging and Visualization

- Log various metrics and visualize them in real time
- Keep track of different model versions and training runs

Overview

Workspace

Logs

Files

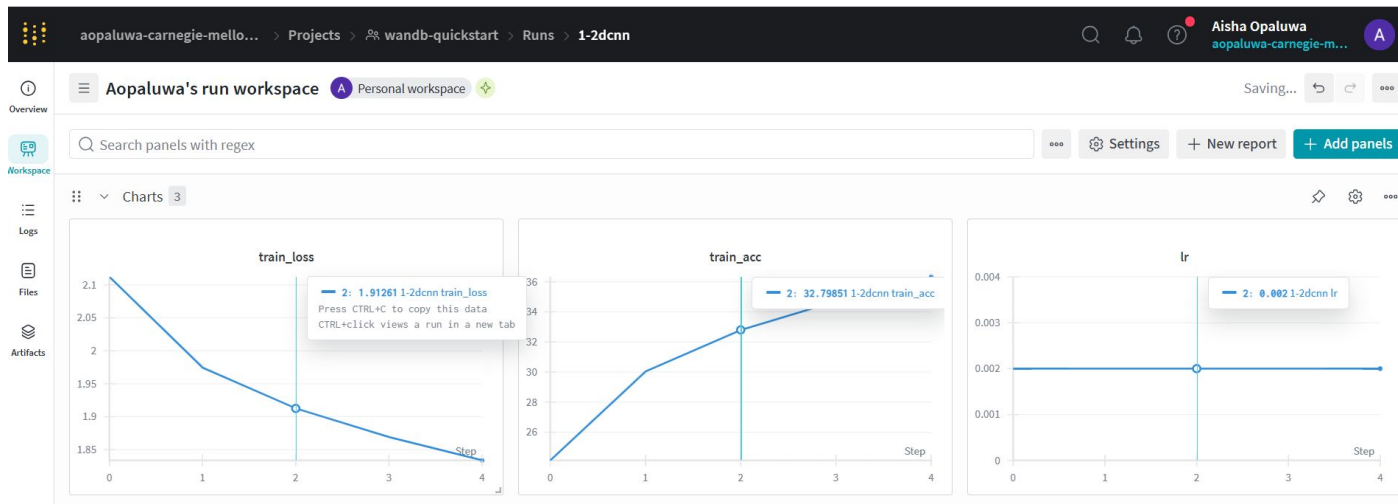
Artifacts

aopaluwa-carnegie-mello... > Projects > wandb-quickstart > Runs > 1-2dcnn

Timestamps visible

Search

1	2025-08-19 04:45:24	Train: 0% 0/782 [00:00<?, ?it/s]/tmp/ipython-input-2212453447.py:19: FutureWarning: 'utc' is deprecated and will be removed in a future version. Use 'utcnow' instead.
2	2025-08-19 04:45:24	with torch.cuda.amp.autocast():
3	2025-08-19 04:45:24	
4	2025-08-19 04:45:35	Epoch 1/5: Train Acc 24.1208%, Train Loss 2.1122, Learning Rate 0.0020
5	2025-08-19 04:45:47	Epoch 2/5: Train Acc 30.0472%, Train Loss 1.9743, Learning Rate 0.0020
6	2025-08-19 04:45:59	Epoch 3/5: Train Acc 32.7985%, Train Loss 1.9126, Learning Rate 0.0020
7	2025-08-19 04:46:11	Epoch 4/5: Train Acc 34.7087%, Train Loss 1.8690, Learning Rate 0.0020
8	2025-08-19 04:46:23	Epoch 5/5: Train Acc 36.3291%, Train Loss 1.8338, Learning Rate 0.0020



What are the Key Features of WandB

Experiment Resumption

- Easily restore model's state and resume experiments in case of runtime crashes

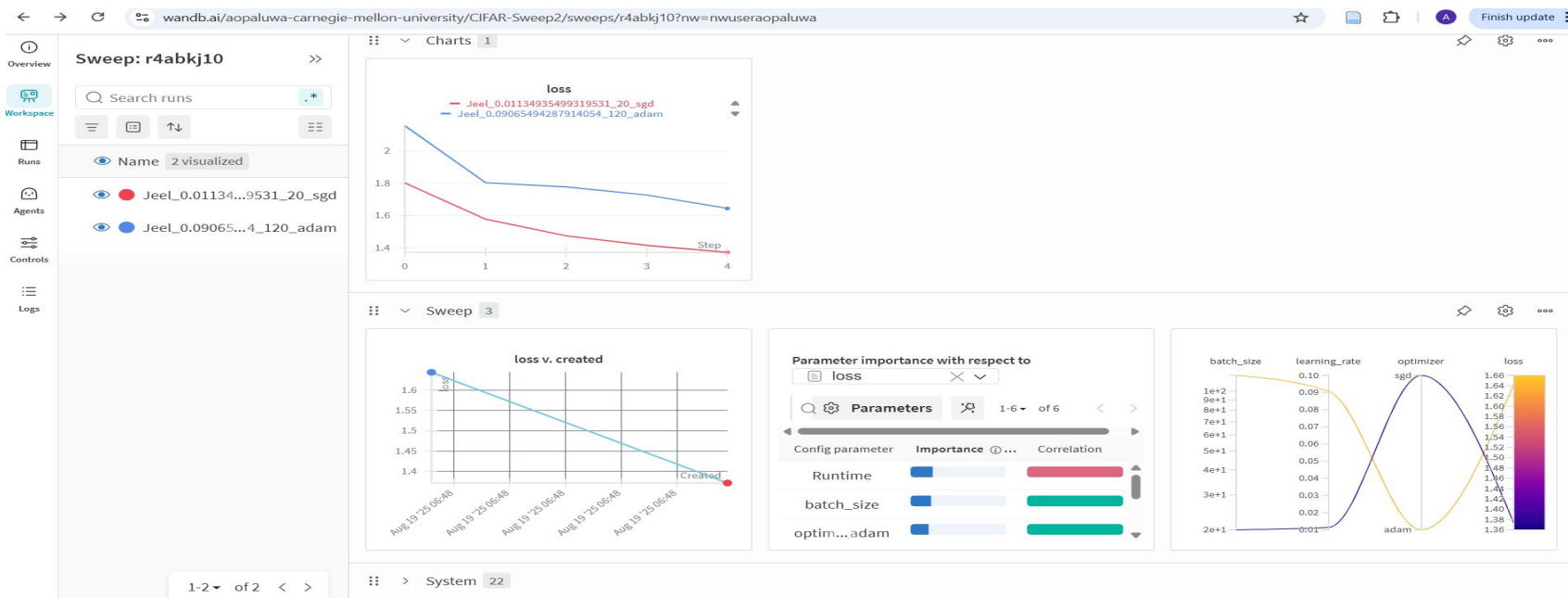
The screenshot displays the WandB interface for a workspace named "Aopaluwa's workspace". The left sidebar contains navigation icons for Overview, Workspace, Runs (selected), Automations, Sweeps, Reports, and Artifacts. The top right shows a "Saved just now" status and navigation controls. The main area is titled "Runs 1" and includes a search bar and filters. A table lists the runs, with one run named "test-lm" in a "Crashed" state. The table columns include Name, State, Notes, User, Tag, Created, Runtime, Sweep, Name, data.N, data.b, data.rc, data.sl, data.te, and data.tr.

<input type="checkbox"/>	<input checked="" type="checkbox"/> Name	1 visualized	State	Notes	Use	Tag	Created	Runtime	Sweep	Name	data.N	data.b	data.rc	data.sl	data.te	data.tr
<input checked="" type="checkbox"/>	test-lm		Crashed	Add notes	aopalu		4mo ago	5h 36m 30	-	aisha_opa	2	256	/kaggle/in	1	test	train

What are the Key Features of WandB

Hyperparameter Tuning

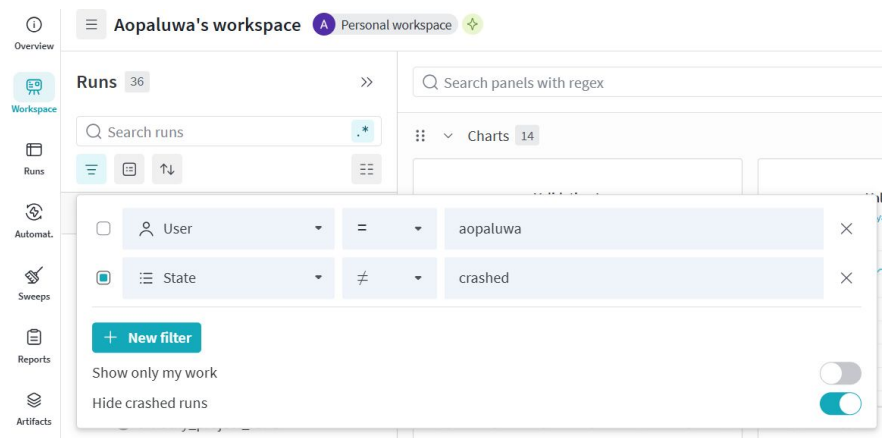
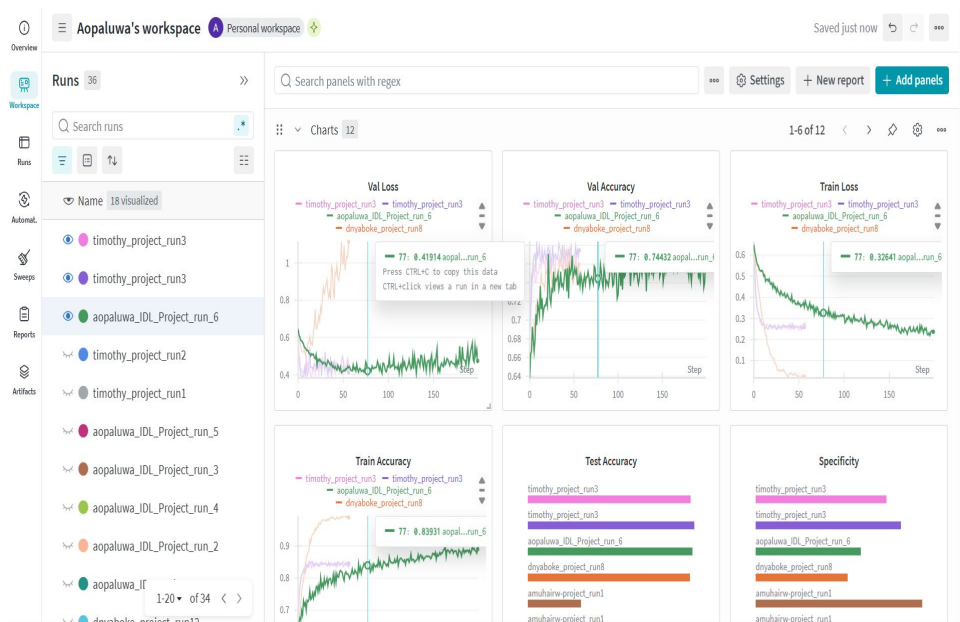
- Automate hyperparameter tuning to find the best configuration for your models through sweeps



What are the Key Features of WandB

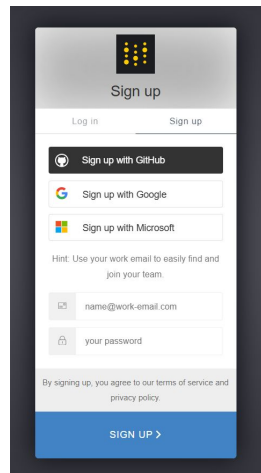
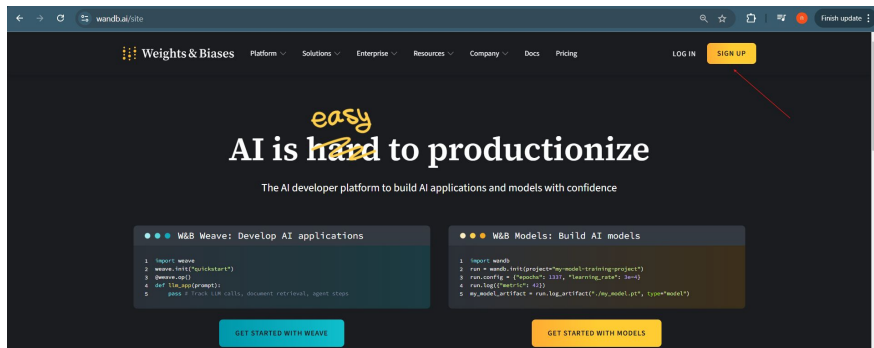
Collaboration

- Share and collaborate your projects with team members and experiment on different ablations while keeping track of others progress



Setting Up WandB Account and an API Key

Creating an account: <https://wandb.ai/site>



Generating an API key: <https://wandb.ai/settings>

API keys

Manage the API keys associated with your account.

Reveal

WandB Installation and Notebook Setup

Wandb Installation for Python

✓ Installation and Libraries

```
[ ] ## Installing WandB
    !pip install wandb -qqq
```

Notebook Setup


- Dataset: CIFAR10
- Neural Network: Convolutional Neural Network
- Code Format: all functionalities are built in functional blocks for automated access

Credits to Fall 2024 TAs for the Notebook

WandB Login and Initializing Project

Wandb Login in Notebook:

```
[ ] import wandb, os
os.environ['WANDB_API_KEY'] = "7328d336610dec777bdecff906dc57e2a464d0b4"#your key here
wandb.login()
```

 wandb: Using wandb-core as the SDK backend. Please refer to <https://wandb.me/wandb-core> for more information.
wandb: Logging into wandb.ai. (Learn how to deploy a W&B server locally: <https://wandb.me/wandb-server>)
wandb: You can find your API key in your browser here: <https://wandb.ai/authorize>
wandb: Paste an API key from your profile and hit enter, or press ctrl+c to quit:
wandb: Appending key for api.wandb.ai to your netrc file: /root/.netrc
True

Initializing Project:

```
[ ] run_config = {
    'model': '1-2dcnn',
    'optimizer': 'sgd',
    'lr': 2e-3,
    'batch_size': 64,
    'epochs': 5
}
```

```
[ ] run = wandb.init(
    project="wandb-quickstart",
    job_type="model-training",
    name=run_config['model'],
    config=run_config
)
```

 wandb: Currently logged in as: **nzafloris**. Use `wandb login --relogin` to force relogin
Tracking run with wandb version 0.18.7
Run data is saved locally in /content/wandb/run-20241218_123851-96ynisfu
Syncing run **1-2dcnn** to [Weights & Biases \(docs\)](#)
View project at <https://wandb.ai/nzafloris/wandb-quickstart>
View run at <https://wandb.ai/nzafloris/wandb-quickstart/runs/96ynisfu>

Logging Metrics and Saving Models

Logging Metrics

```
# What to log

metrics = {
    "train_loss": train_loss,
    "train_acc": train_acc,
    'lr': lr
}

# Log to run
wandb.log(metrics)
```

Saving Model to WandB

```
# Saving the model and optimizer states

torch.save({
    'model_state_dict': model.state_dict(),
    'optimizer_state_dict': optimizer.state_dict()
}, "Model.pth")

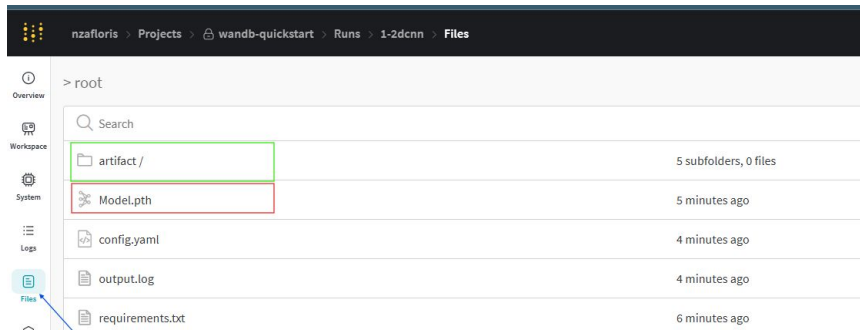
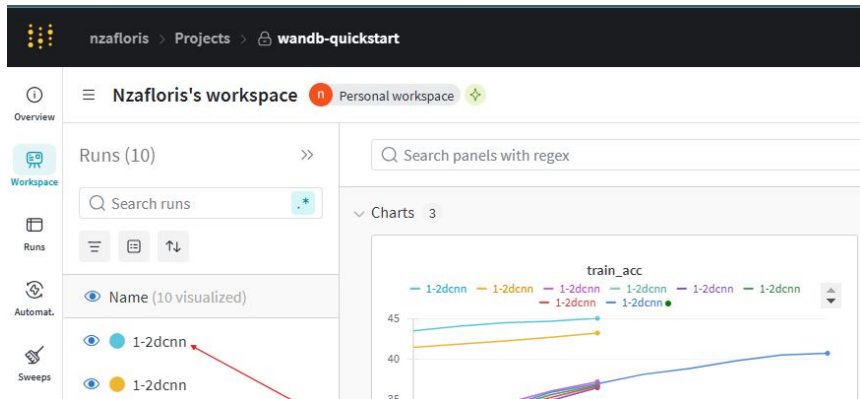
# ALTERNATIVE 1: Saving Files as Artifacts
# Creating Artifact
model_artifact = wandb.Artifact(run_config['model'], type='model')

# Adding model file to Artifact
model_artifact.add_file("Model.pth")

# Saving Artifact to WandB
run.log_artifact(model_artifact)

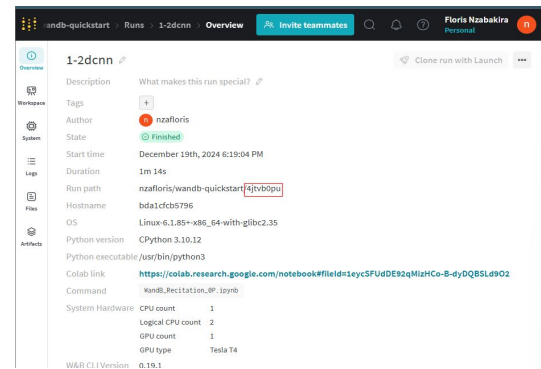
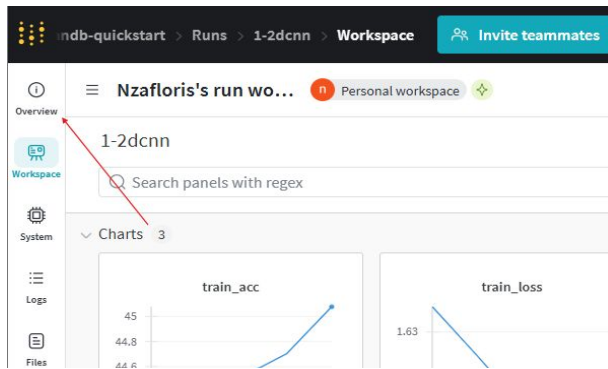
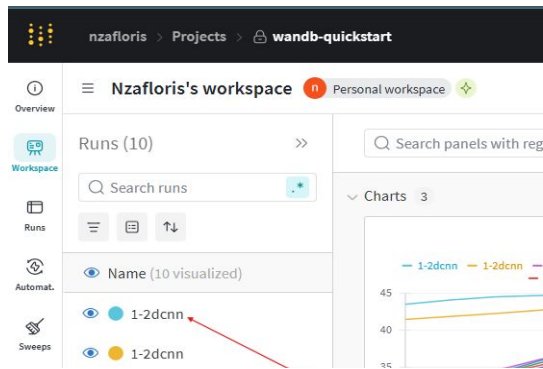
# ALTERNATIVE 2: Saving Files as Files
wandb.save("Model.pth")

if finish:
    wandb.finish()
```



Resuming a Run

Getting Run ID



Resuming Run in Notebook

```
run_id = "4jtvb0pu" ### Replace with run id string
run = wandb.init(
    id = run_id, ### Insert specific run id here if you want to resume a previous run
    resume = "must", ### You need this to resume previous runs, but comment out reinit = True when using this
    project = "wandb-quickstart", ### Project should be created in your wandb account
)
```

Automated Hyperparameter Tuning with Sweeps

```
[ ] # Hyperparameters to work with

parameters_dict = {
    'optimizer':{
        'values': ['sgd', 'adam']
    },
    'learning_rate':{
        'distribution':'uniform',
        'min':2e-4,
        'max':1e-1
    },
    'batch_size': {
        'distribution': 'q_log_uniform_values',
        'q':4,
        'min': 16,
        'max': 128
    },
    'epochs':{
        'value': 5
    }
}
sweep_config['parameters'] = parameters_dict
```

```
[ ] # Initializing the sweep

sweep_id = wandb.sweep(sweep_config, project="CIFAR-Sweep2")
```

```
def train_sweep(config = None):
    with wandb.init(config=config) as run:
        run.name=f"Jeel_{wandb.config.learning_rate}_{wandb.config.batch_size}_{wandb.config.optimizer}"
        config = wandb.config

        train_loader, test_loader = build_data(config.batch_size, data_train, data_test)

        model = Network().to(device)

        optimizer = get_optim(config.optimizer, config.learning_rate, model)

        criterion = nn.CrossEntropyLoss()

        scaler = torch.cuda.amp.GradScaler()

        for epoch in range(config.epochs):

            model, loss = train_epoch(model, train_loader, optimizer, criterion, scaler)

            wandb.log({'loss': loss})
```

```
[ ] # Running the sweep

wandb.agent(sweep_id, train_sweep, count=2)
```

Collaboration on WandB

Invite teammates

Create a new account

Account Name

wandb.ai/account-name

Choose your team name carefully! It cannot be changed after creation.

Account Type

Work

For users at companies, collaborating on ML projects at work.

Academic (limit 1 per user)

For academic research and projects outside of corporate use.

Invite team members (optional)

Separate emails or usernames with commas

```
[16] run = wandb.init(  
    entity="wandb-starter",  
    project="wandb-quickstart",  
    job_type="model-training",  
    name=run_config['model'],  
    config=run_config  
)
```

 wandb: Currently logged in as: **nzafloris** (**wandb-starter**). Use `wandb login --relogin` to force relogin

Tracking run with wandb version 0.19.1

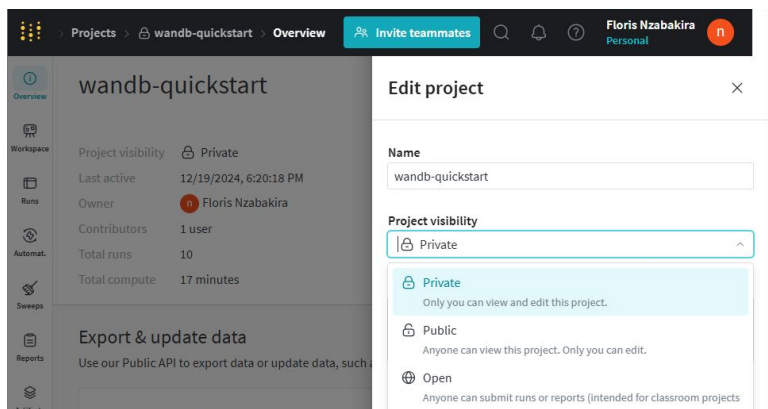
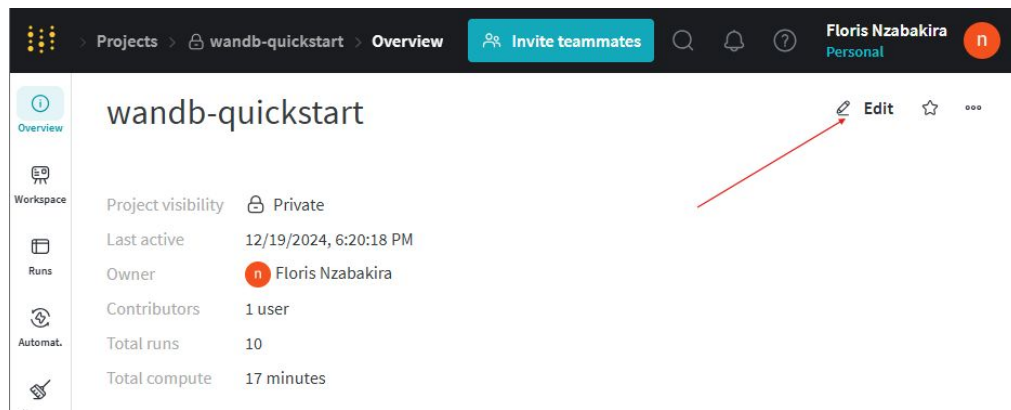
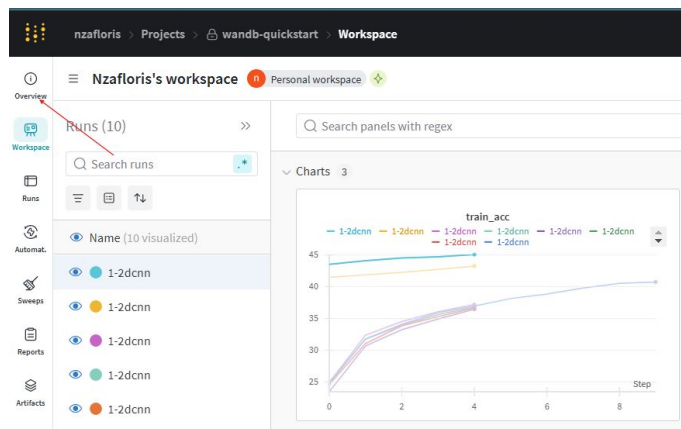
Run data is saved locally in `/content/wandb/run-20241219_225136-k7cu7hle`

Syncing run **1-2dcnn** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/wandb-starter/wandb-quickstart>

View run at <https://wandb.ai/wandb-starter/wandb-quickstart/runs/k7cu7hle>

Making your WandB Project Public



WandB Storage for Individual and Group Accounts

NB: Once your WandB free storage is full, you won't be able to access your runs.

You can view your storage status in the following way

Individual Account:

- Provides 100GB free
- Can be accessed through <https://wandb.ai/subscriptions>

Group Account:

- Provides 5GB free per team
- Can be accessed through <https://wandb.ai/account-settings/><Team-Name>/billing