

開源軟體介紹 - 以MLOps為例

葉信和 / Hsin-Ho Yeh
Software Engineer / CEO @ 信誠金融科技
hsinho.yeh@footprint-ai.com

Download Slides

<http://bit.ly/3AisQmJ>



About me

- 2020 - Present at 信誠金融科技
 - **Shrimping**: A data-sharing platform
 - <https://get-shrimping.footprint-ai.com>
 - **Tintin**: a machine learning platform for everyone
 - <https://get-tintin.footprint-ai.com>
 - **KaFeiDo**: machine learning platform for green economy
- 2016 - 2020 at IglooInsure (16M+ in series A+ 2020)
 - Provide digital insurance for e-economic world
 - Funded in KUL, Headquartered in Singapore
 - First employee/ Engineering Lead / Regional Head/ Chief Engineer
- 2013 - 2016 at Studio Engineering @ hTC
 - Principal Engineer on Cloud Infrastructure Team
- 2009 - 2012 at IIS @ Academia Sinica
 - Computer vision, pattern recognition, and data mining
- CS@CCU, CS@NCKU alumni



Agenda

- What is Open Source?
- How to contribute to Open Source?
- Kubeflow Case Study
- FAQ

What is Open Source Software?

“Open-source software is a type of computer software ... the copyright holder grants users the rights to **use**, **study**, **change**, and **distribute** the software to anyone and for any purpose.” [1]

[1] https://en.wikipedia.org/wiki/Open-source_software

What is Open Source Software? (1/4)

	Open Source Software	Proprietary Software
Ownership	Everyone	A group of limited people
Usability	Difficult use for unskilled users Ex: Desktop Linux	Feature neat to end-user Ex: Microsoft Windows
Security	Better (?) Ex: <u>PHP source code is injected backdoors by hackers</u>	Less
Stability	Less	Better
Opacity	Transparent	Closed

The misunderstanding on open source software

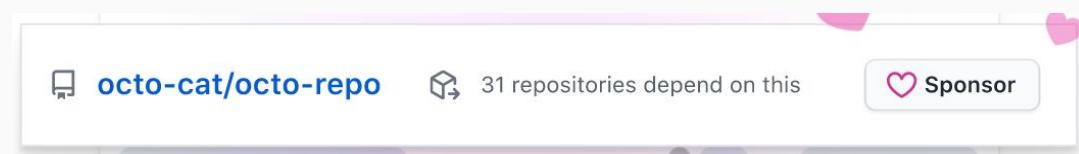
- Open source software is **Free** in terms of Freedom, not Monetary.
- Open source developer is **NOT** a free labor [1]
 - Sponsorship by companies
 - Add values(a.k.a layers) on the existing project
 - Professional consult
- Open source means **Pool** quality
 - Good quality when the project gained enough eyeballs
 - On average the density of code defect is smaller than proprietary software [2]

[1] <https://softwareengineering.stackexchange.com/questions/100685/making-money-with-open-source-as-a-developer>

[2] <https://blogs.worldbank.org/opendata/quality-open-source-software-how-many-eyes-are-enough>

Where open source communities lives?

- Github
 - Free for any public repositories
 - Github Sponsor



Key files to understand a github project

- README.md
 - How to build and use the project
- LICENSE
 - Terms and Conditions
- CONTRIBUTING.md
 - Rules on contributing
- ROADMAP.md
 - Project roadmap
- CODE OF CONDUCT.md
 - Guidance on rule, responsibilities

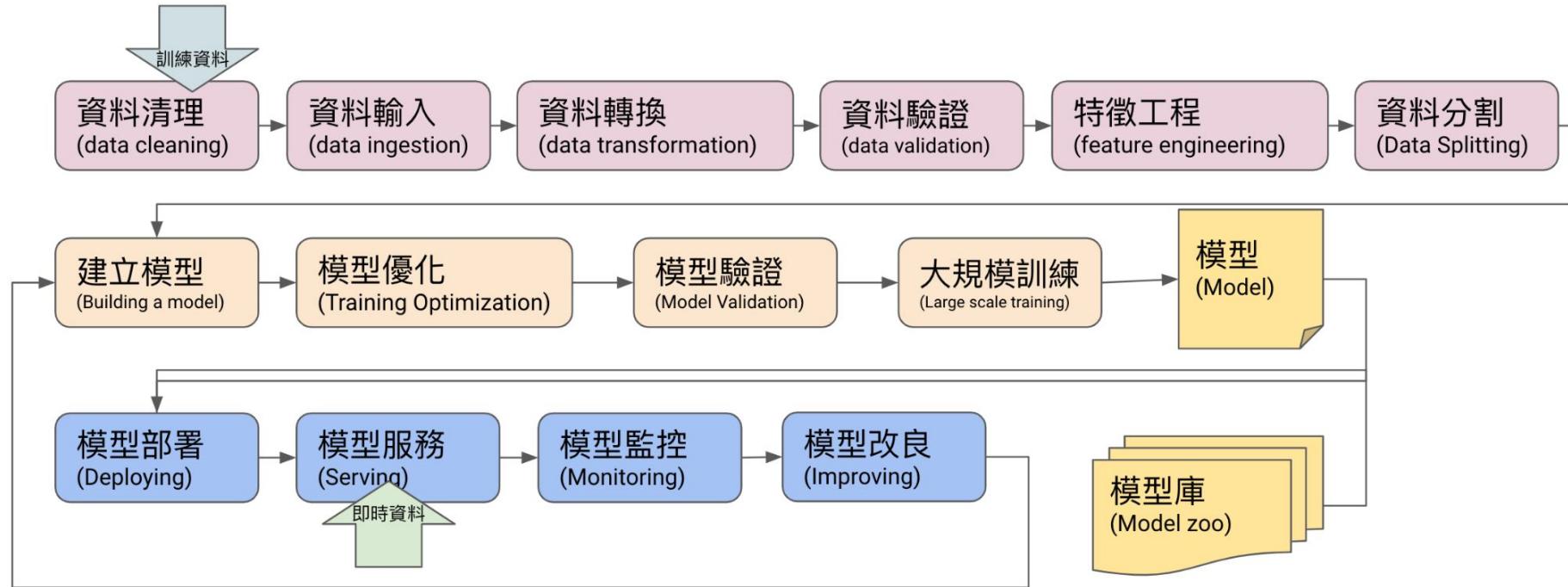
What is kubeflow?



Kubernetes + ML
= Kubeflow

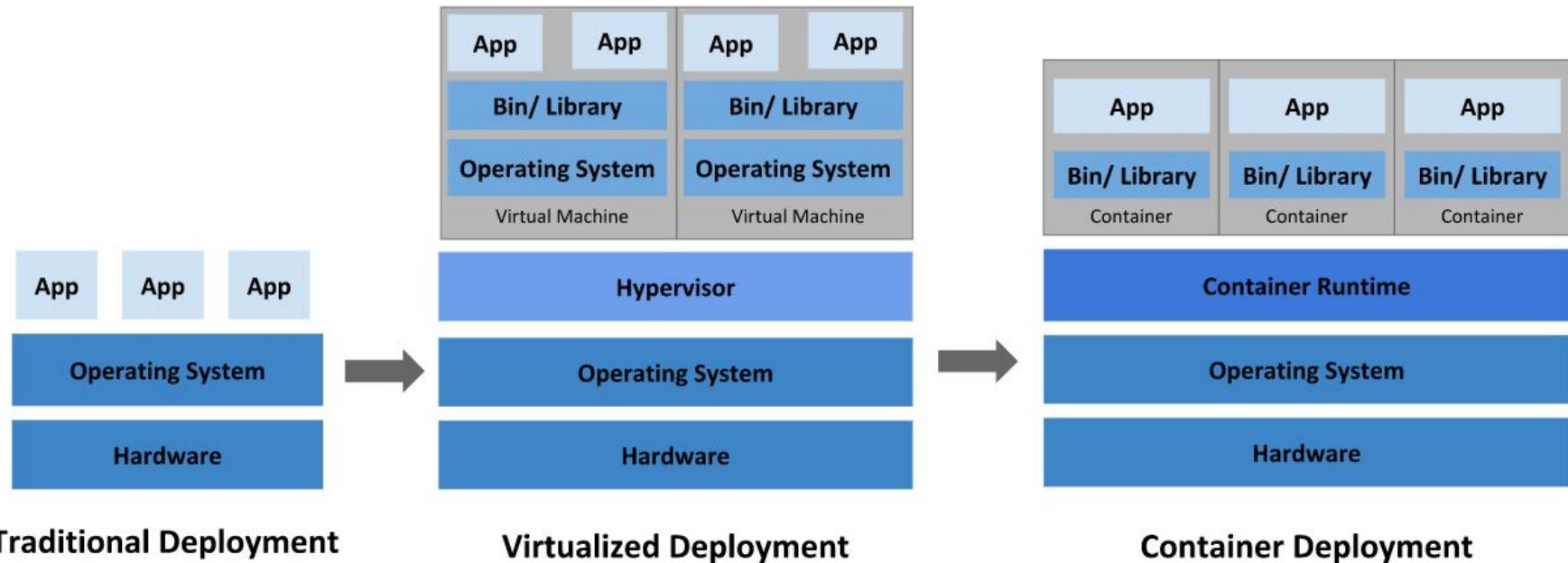
The Kubeflow project is dedicated to making deployments of machine learning (ML) workflows on Kubernetes simple, portable and scalable.

Real-world Machine Learning Application - End-to-End ML LifeCycle



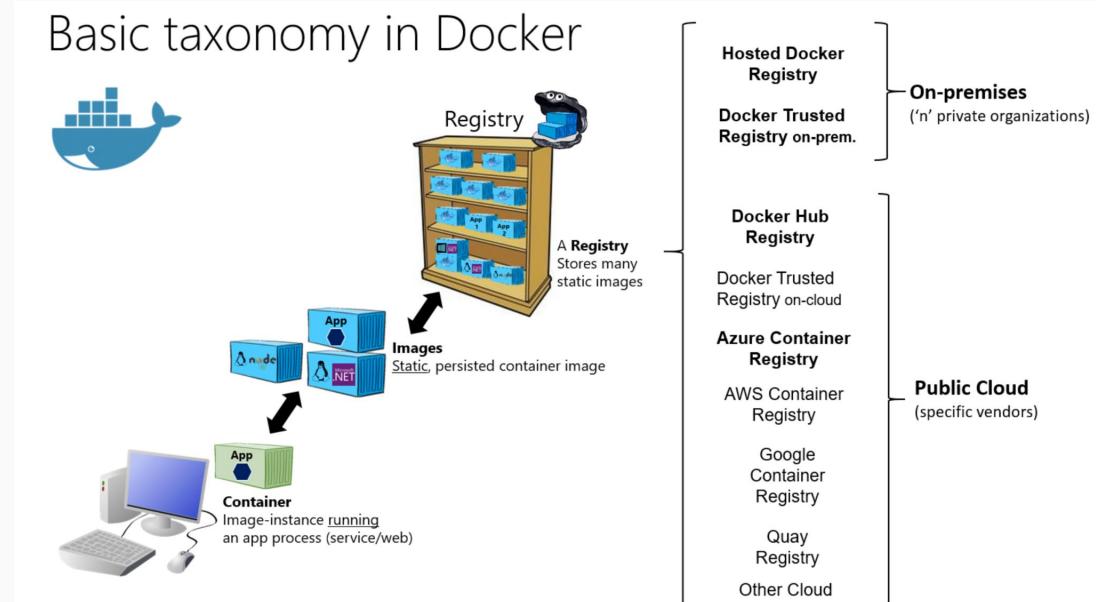
What is Container?

What is containerized deployment?



What is Container?

- Container
 - Container Image = Application code + dependencies
 - Runtime environment (cgroups, namespaces, env vars)
- Container Registry
 - [Dockerhub](#)
 - [Github container registry](#)



Ref:

<https://learn.microsoft.com/en-us/dotnet/architecture/microservices/container-docker-introduction/docker-containers-images-registriesg/practical-guide-on-writing-a-dockerfile-for-your-application-89376f88b3b5>

How to build a Docker Image

```
FROM php:7.0-apache
```

registry

```
COPY index.php /var/www/html/index.php
```

repository

```
EXPOSE 80
```

tag

```
docker build -t footprintai/k8sworkshop:php-demo -f Dockerfile .
```

```
=> [internal] load metadata for docker.io/library/php:7.0-apache 4.6s
```

```
...
```

```
=> [2/2] COPY index.php /var/www/html/index.php
```

```
0.8s
```

```
=> exporting to image
```

```
0.2s
```

```
=> => exporting layers
```

```
0.1s
```

```
=> => writing image
```

```
sha256:e74d16d21b10069d0beba2cc6daf7cc011723d7e51523c3830e50b1bc5338e88 0.0s
```

```
=> => naming to docker.io/footprintai/k8sworkshop:php-demo 0.0s
```

Install Docker runtime

- [Windows] install Docker Desktop
 - <https://docs.docker.com/desktop/install/windows-install/>
- Install Docker on your host machine
 - <https://docs.docker.com/engine/install/ubuntu/>

Hands on Docker

```
docker run -it ubuntu:20.04

root> apt-get update
root> apt-get install -y lsb-release
root> lsb_release -a

Distributor ID: Ubuntu
Description:    Ubuntu 20.04.2 LTS
Release:        20.04
Codename:       focal
```

k8s materials

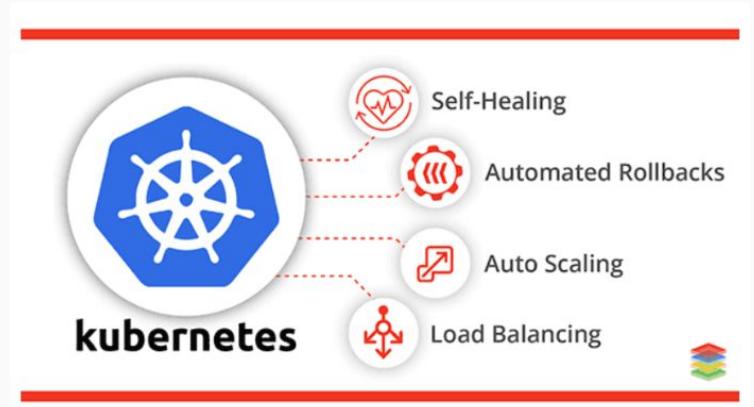
```
git clone https://github.com/FootprintAI/k8s-workshop
```

The screenshot shows the GitHub repository page for 'k8s-workshop'. The repository is public and has 1 branch and 0 tags. The 'Code' tab is selected. On the right, there is a 'Clone' section with three options: HTTPS, SSH, and GitHub CLI. The HTTPS URL is highlighted with a red box: `https://github.com/FootprintAI/k8s-workshop`. Below the URL, it says 'Use Git or checkout with SVN using the web URL'. The rest of the page displays the repository's contents, including files like 'deployment', 'install', 'probe', 'secrets', 'slide', 'LICENSE', 'README.md', 'ingress.yaml', and 'namespace.yaml', along with their commit history. At the bottom, there is a preview of the 'README.md' file which contains sections for 'k8s-workshop', 'Purpose', and 'Presentation'.

What is Kubernetes?

Why machine learning on Kubernetes?

- Portability
 - Dev/Staging/Prod
 - Laptop/Edge/Cloud environment
- Scalability
 - Hyperparameter tuning, production workloads
- Isolation
 - Workloads from different experiments



History Of Kubernetes

- Borg: the predecessor to Kubernetes
 - Google revealed the first time of its detail in an academic research paper, describing a “cluster manager that runs hundreds of thousands of jobs, from many thousands of different applications, across a number of clusters each with up to tens of thousands of machines.”[1]
 - A in-house cluster manager system inside Google for running every google services including Gmail, Google Maps, Google Docs...[2]
 - In a scale with ‘over 2 billion containers per week’ [3]
- The very first version of Kubernetes was released in 2015
- The latest version is v1.26, released at 2023.

[1] <https://research.google/pubs/pub43438/>

[2] <https://www.wired.com/2016/04/want-build-empire-like-googles-os/>

[3] <https://cloud.redhat.com/blog/building-kubernetes-bringing-google-scale-container-orchestration-to-the-enterprise>



Case Study: Open AI

OpenAI adopted Kubernetes since 2016 for portability, cost saving, and improved efficiency[1,2].

Years	Nodes	Estimated Cost [3]
2018	2,500	= 3 * 2500 * 24 = US\$ 180,000 / day
2021	7,500	= 3 * 7500 * 24 = US\$ 540,000 / day
2023	?	

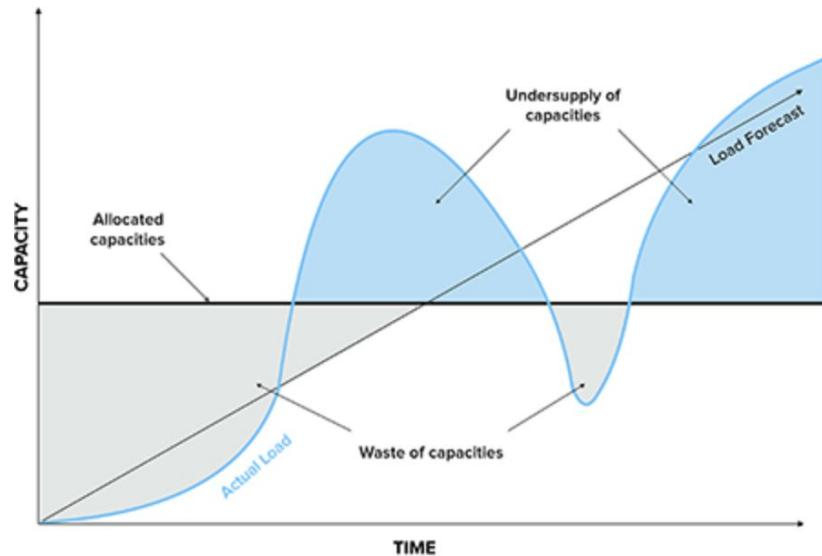
[1] <https://kubernetes.io/case-studies/openai/>
<https://blog.pichuang.com.tw/20230214-openai-scaling-kubernetes-to-7500-nodes.html>

[2] https://blog.pichuang.com.tw/20230214-openai-scaling-kubernetes-to-7500-nodes.html#_1

[3] AWS P3.2xlarge equips 8 vCores, 61 GB Memory, 1 Core of V100-16Gb GPU, charges US\$ 3 per hour.

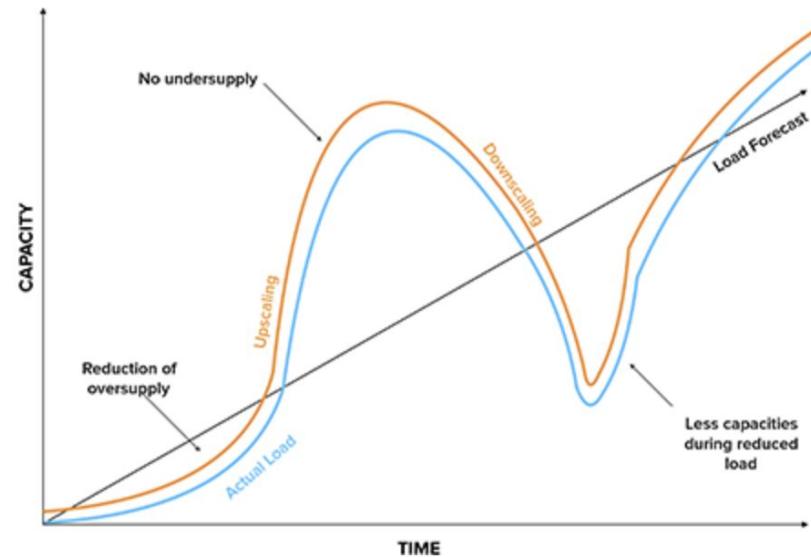
Cost, Cost, and Cost

Static Architecture



Static architectures are based on estimated load expectancy and are not flexible enough to adapt to unexpected load peaks or lulls.

Auto Scaling Architecture



Auto Scaling is the most cost-efficient solution for a fluctuating load. High performance and thereby user satisfaction are retained at all times.

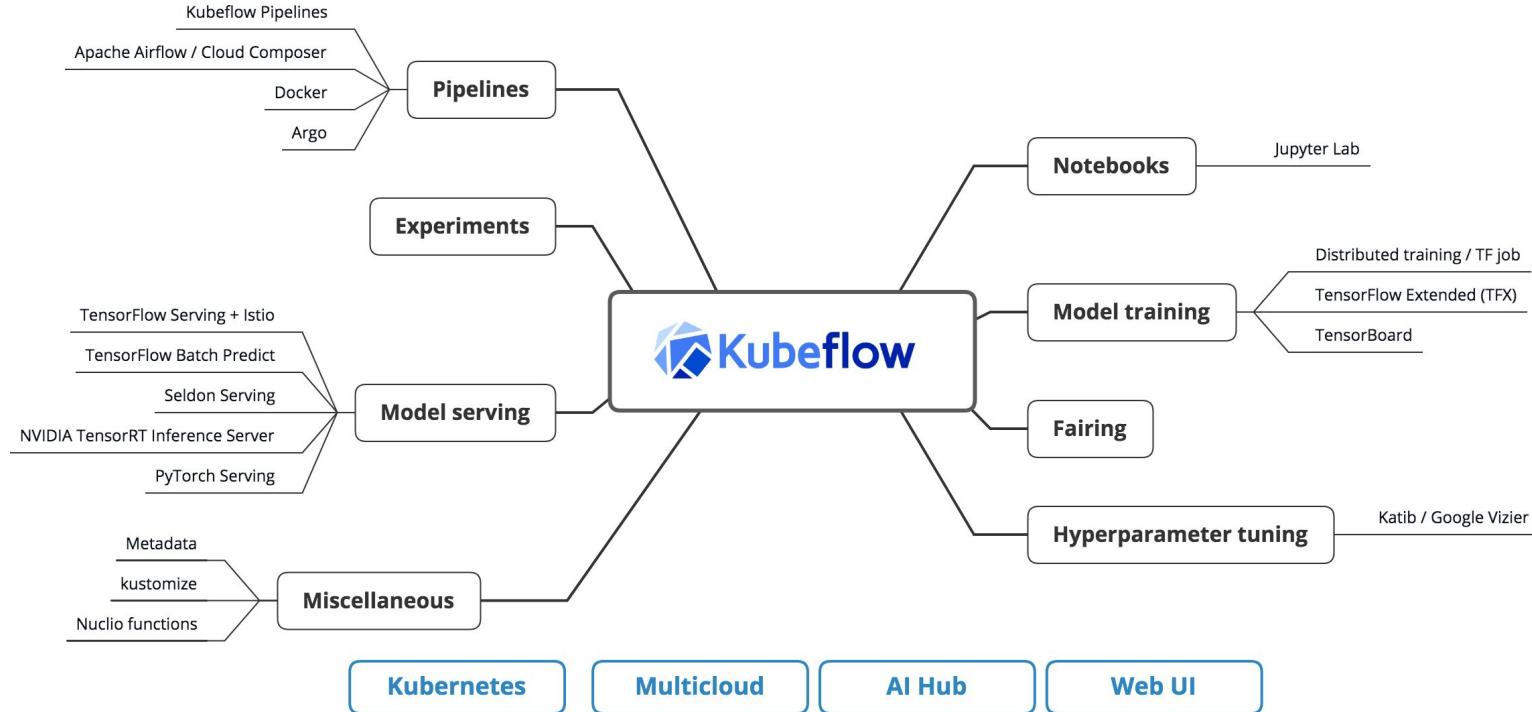
Oh, you want to use ML on K8s?

Before that, can you become an expert in:

- Containers
- Packaging
- Kubernetes service endpoints
- Persistent volumes
- Scaling
- Immutable deployments
- GPUs, Drivers & the GPL
- Cloud APIs
- DevOps
- ...



Architectures



How to contribute Open Source?

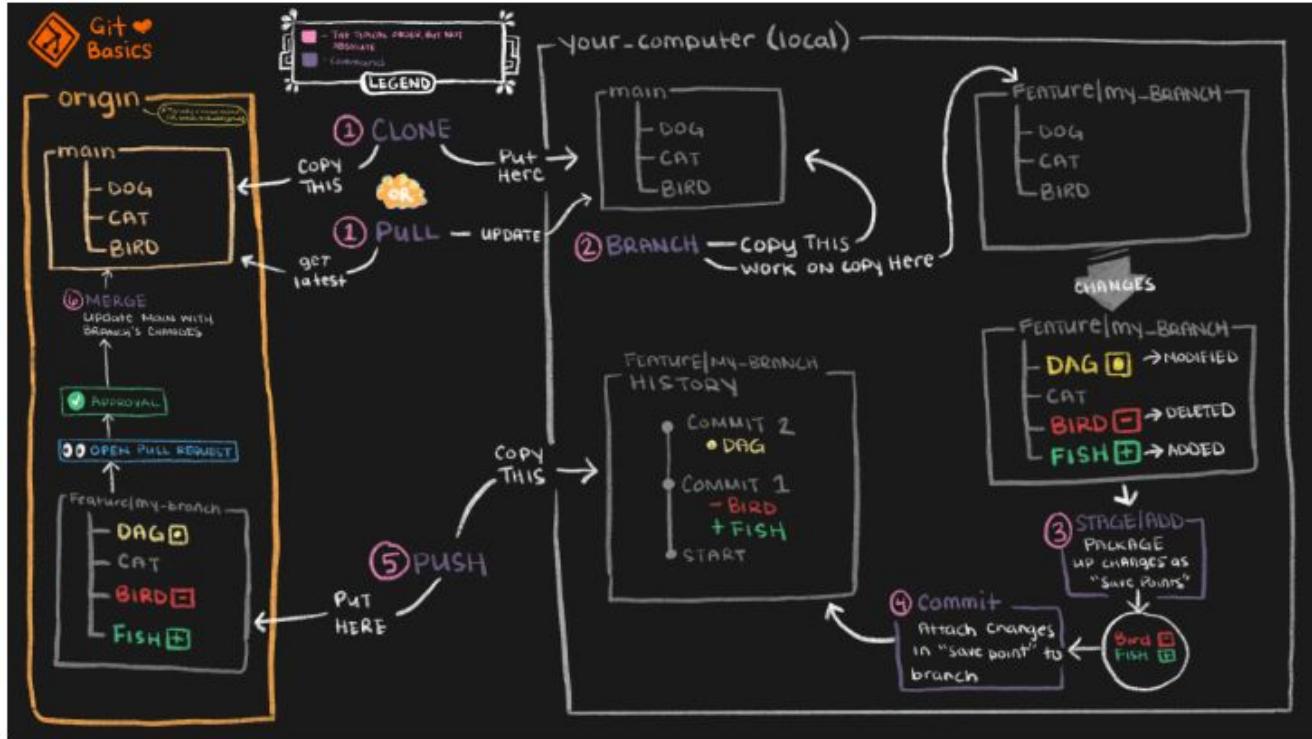
The two most popular way to contribute

1. Raising an issue where you can feel the project could be improved
 - a. Report a problem/bug (kind/bug)
 - b. Suggest new features (kind/feature)
 - c. Gap in documentation (doc)
2. Contribute to code
 - a. Pull-Request in Github

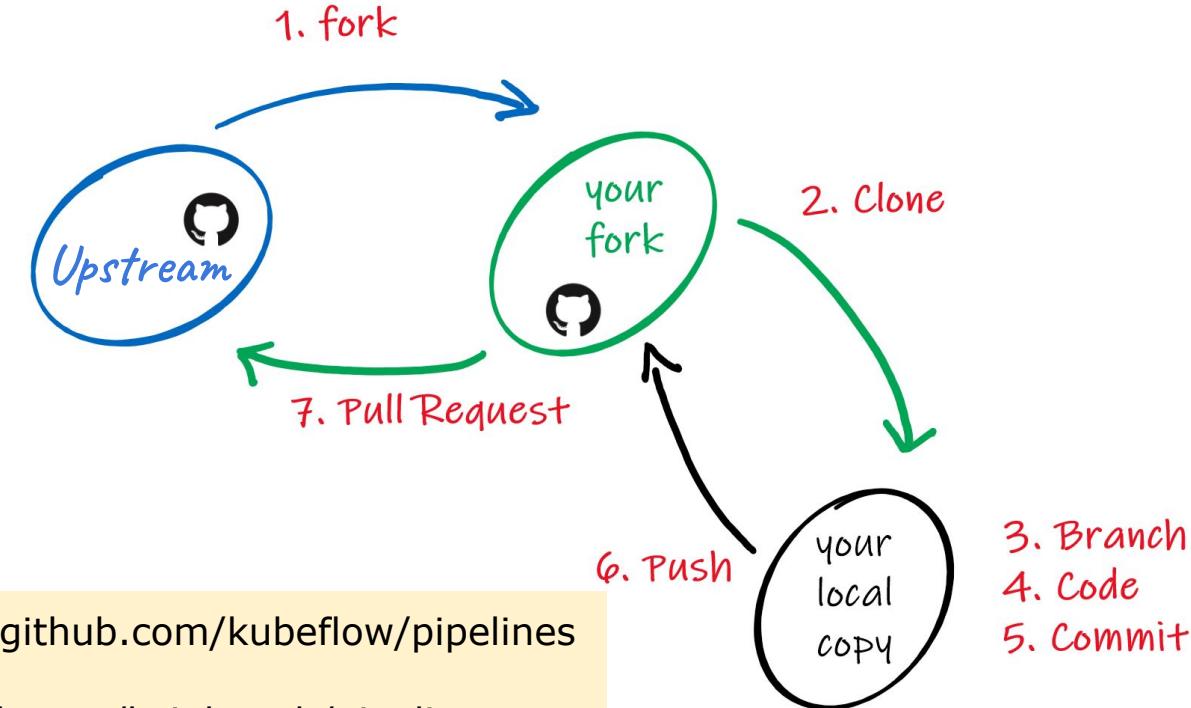
Issue list @kubeflow/pipeline

415 Open ✓ 1,720 Closed		Author ▾	Label ▾	Projects ▾	Milestones ▾	Assignee ▾	Sort ▾
⚠	[frontend] The sort, filter of artifact does not work	area/frontend	kind/bug				1
	#5565 opened 20 hours ago by capri-xiyue						
⚠	[Doc] Update "Visualize Results in the Pipelines UI" page						2
	#5564 opened yesterday by Bobgy						
⚠	[backend] OutputArtifact path is None	area/backend	kind/bug				
	#5560 opened 2 days ago by wilbry						
⚠	[feature] Configurable KFP Artifact types while storing to Minio	area/sdk	kind/feature				2
	#5558 opened 2 days ago by rado-narrak						
⚠	[Multi User] Kubeflow Pipelines Missing Headers	area/backend	kind/bug				
	#5556 opened 2 days ago by PoyerRamirez						
⚠	[feature] Allow jobs to be scheduled on AWS Fargate	area/backend	kind/feature				2
	#5555 opened 3 days ago by yuhuishi-convect						
⚠	[sdk] mlpipelines-metrics path is wrong when using kfp.components.load_component_from_text	area/sdk	kind/bug				
	#5548 opened 4 days ago by santiagoolivar2017						
⚠	pipeline_conf.set_ttl_seconds_after_finished(seconds=10) NOT WORKING	area/backend	kind/bug				
	#5545 opened 4 days ago by prasadkyp						
⚠	[sdk] input parameter created without default value	area/sdk	kind/bug				1
	#5544 opened 5 days ago by amitza						

Git Basic



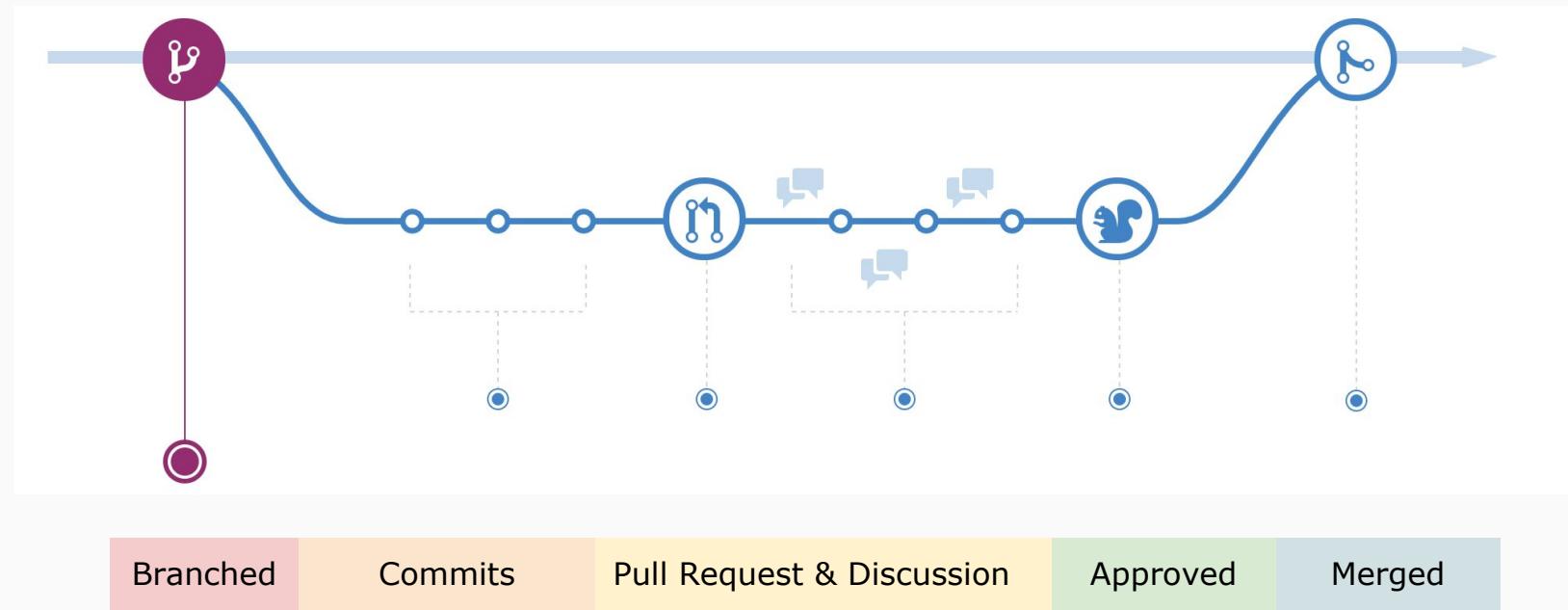
Pull-Request flow in Github



Upstream: github.com/kubeflow/pipelines

Fork: github.com/hsinhyeh/pipelines

Github Flow: branch-based workflow



Pull-Request list @kubeflow/pipeline

81 Open ✓ 3,342 Closed

Author ▾ Label ▾ Projects ▾ Milestones ▾ Reviews ▾ Assignee ▾ Sort ▾

chore: [Research] Add toxicity detection pipeline ● cla: yes needs-ok-to-test size/S
#5573 opened 1 hour ago by TheDutchDevil 1 of 1

feat(sdk.v2): reimplement custom job support ● cla: yes size/XL
#5572 opened 4 hours ago by chensun 1 of 1

Add OWNERS for v2 folder ● cla: yes lgmt size/XS
#5571 opened 10 hours ago by Bobgy 0 of 1

Add zijianjyo to frontend/OWNERS ● approved cla: yes size/XS
#5570 opened 10 hours ago by Bobgy 0 of 1

chore(deps): bu
#5569 opened 11 hours ago by TheDutchDevil

chore(google_c
size/M
#5554 opened 3 days ago by TheDutchDevil

feat(sdk): Add a
#5552 opened 3 days ago by TheDutchDevil

chore(v2): fix in
size/M
#5543 opened 5 days ago by TheDutchDevil

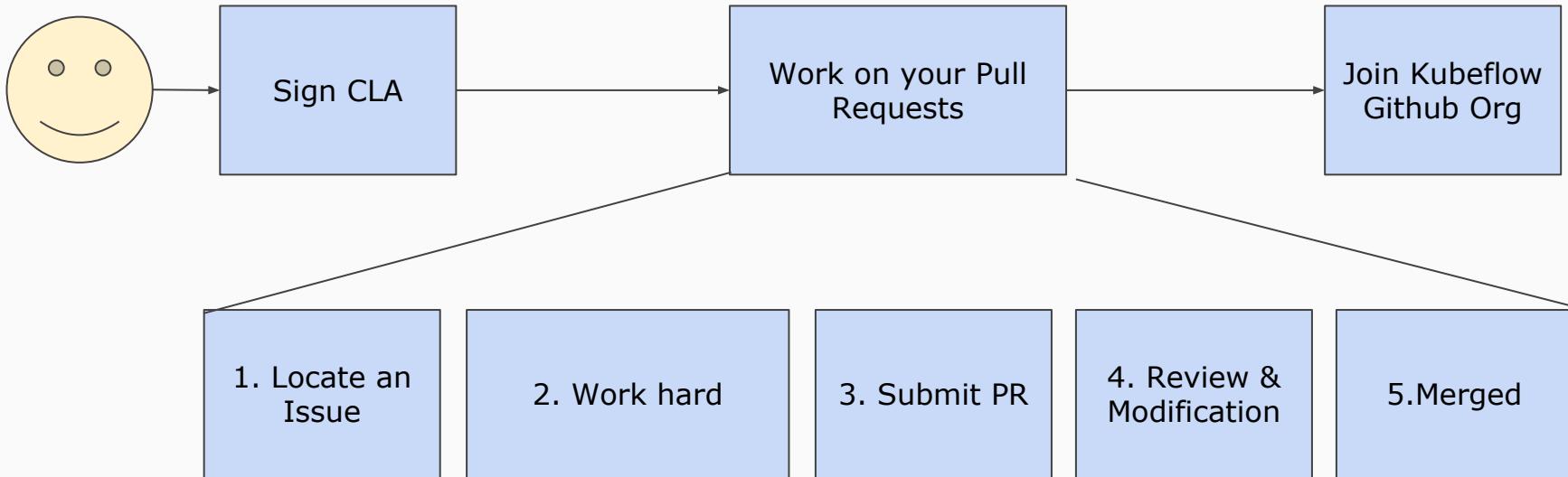
Ref: <https://www.conventionalcommits.org/en/v1.0.0/>

<type>(<scope>): <short summary>

- ↳ Summary in present tense. Not capitalized. No period at the end.
- ↳ Scope: common|compiler|authentication|core|
- ↳ Type: chore, docs, feat, fix, refactor, style, or test.

4 32

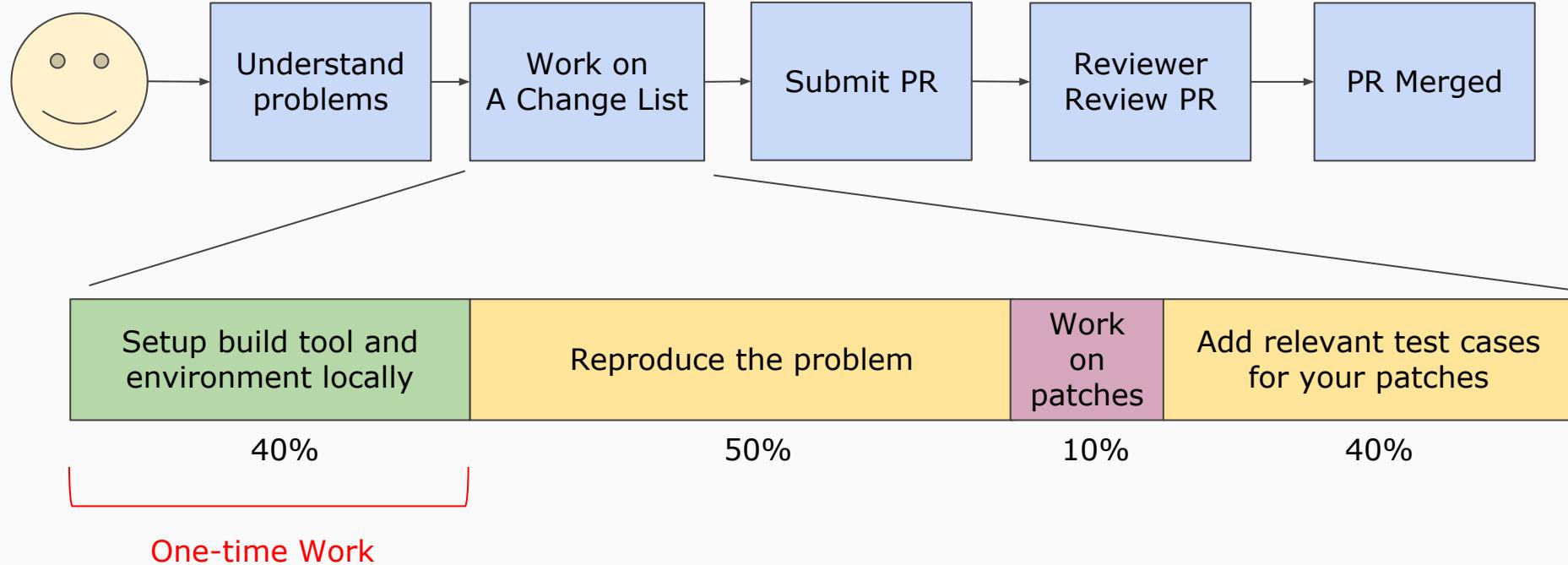
How to be a Kubeflow Contributor?



Ref: <https://www.kubeflow.org/docs/about/contributing/>

1. Existing issues
 - a. Issues labelled with `good first issue`:
kubeflow/pipeline
 - b. Issues labelled with `help wanted`:
kubeflow/pipeline
2. Organic issues
 - a. Create an issue to describe issues you encountered and also provide a solution (a.k.a Pull-request) for it

Why does the first contribution be important?



Kubeflow Case Study

Kubeflow/pipeline#4699 Case Study (1/2) - Existing issue

Feature suggestion: expose the duration of each pipeline step #4699

New issue

Closed

gcampax opened this issue on Oct 31, 2020 · 1 comment



gcampax commented on Oct 31, 2020

...

It would be very useful to know how long each pipeline step took, in addition to the total duration of the pipeline. This is especially useful when trying different hyperparameters that affect performance.

This information is already available on the Argo UI, but (1) one needs to know the UI exists (it is not exposed from the Kubeflow dashboard) and (2) one needs to cross reference the pod name from the kubeflow-pipelines UI and the Argo UI. It would be nice to have the duration in one place.

5

Bobgy added this to Needs triage in KFP Runtime Triage via automation on Nov 5, 2020

Bobgy added good first issue area/frontend kind/feature help wanted labels on Nov 5, 2020



Bobgy commented on Nov 5, 2020

Collaborator

...

Contribution instructions: <https://github.com/kubeflow/pipelines/tree/master/frontend>

Run details page react component:

```
pipelines/frontend/src/pages/RunDetails.tsx  
Line 342 in e3992fa  
342 {sidepanelSelectedTab === SidePaneTab.VISUALIZATIONS &&
```

Assignees

No one assigned

Labels

area/frontend good first issue
help wanted kind/feature

Projects

KFP Runtime Triage
Closed

Milestone

No milestone

Linked pull requests

Successfully merging a pull request may close this issue.

feat(frontend): Add Task Details tab whe...

Notifications

Customize

Subscribe

You're not receiving notifications from this thread.

2 participants



Bobgy moved this from Needs triage to Awaits Contributor in KFP Runtime Triage on Nov 15, 2020

Kubeflow/pipeline#4699 Case Study (2/2) - Existing issue

feat(frontend): Add Task Details tab when clicking a node: Info includes ID, Name, Time, Status. Fixes #4699 #5023

Merged google-oss-robot merged 10 commits into `kubeflow:master` from `zijianjoy:duration` on Jan 26

Conversation 41 Commits 10 Checks 0 Files changed 5 +137 -12

zijianjoy commented on Jan 22 · edited

Fixes #4699

Description of your changes:
As part of fulfilling feature request #4699, creating a Task Detail tab which allows people to view the task ID, display name, status, start time, end time, runtime duration of a node when they click on it.

Checklist:

- The title for your pull request (PR) should follow our title convention. [Learn more about the pull request title convention used in this repository](#).
- Do you want this pull request (PR) cherry-picked into the current release branch?

Screenshot:

Success

file-passing-pipelines-v8k56-1117908985

Task Detail

Task ID	file-passing-pipelines-v8k56-1117908985
Task name	print-text-3
Status	Succeeded
Started at	1/21/2021, 12:17:11 AM
Finished at	1/21/2021, 12:17:12 AM
Duration	0:00:01

Contributor

Reviewers

- Bobby**
- capri-xiyue
- Ark-kun
- elikatsis

Assignees

- Bobby**

Labels

- approved
- cla: yes
- lgtm** (highlighted with a red box)
- size/L

Projects

None yet

Milestone

No milestone

Linked issues

Successfully merging this pull request may close these issues.

Feature suggestion: expose the duration...

Notifications Customize

LGTM?

Kubeflow/kubeflow#7525 Case Study - Organic issue (1/2)

[backend] health service response should be consumed by application/json not text/plain #7525

Edit

New issue

Open

hsinhyeh opened this issue 16 days ago · 2 comments



hsinhyeh commented 16 days ago

...

Environment

- How did you deploy Kubeflow Pipelines (KFP)? install via kubeflow manifest with version v1.4
- KFP version: v1.7.0
- KFP SDK version: v1.7.0 (golang version under backend/api/go_http_client)

Steps to reproduce

1. start pipeline backend api server by listening port 8888
2. sent http request with swagger generated code stub with the following code example

```
package main

import (
    "github.com/go-openapi/strfmt"
    "github.com/go-openapi/runtime"
    healthz_client "github.com/kubeflow/pipelines/backend/api/go_http_client/healthz_client"
    healthz_service "github.com/kubeflow/pipelines/backend/api/go_http_client/healthz_client/healthz_service"
)

func main() {
    healthServiceClient := healthz_client.New(
        client.New(
            "localhost:8888",
            healthz_client.DefaultbasePath,
            []string{"http"}),
        strfmt.NewFormats(),
    )

    clientAuthInfoWriter := runtime.ClientAuthInfoWriterFunc(func(clientRequest runtime.ClientRequest, reg))
    return nil
}
```

Assignees

No one assigned

Labels

area/backend kind/bug

Projects

KFP Runtime Triage
Awaiting Contributor

Milestone

No milestone

Development

No branches or pull requests

Notifications

Customize

Unsubscribe

You're receiving notifications because you were mentioned.

2 participants



Kubeflow/kubeflow#7525 Case Study - Organic issue (2/2)

fix(backend): fixes healthz response by adding json content type. Fixes #7525

[Edit](#) [Code](#)

[Open](#) hsinhyeh wants to merge 1 commit into `kubeflow:master` from `hsinhyeh:master`

Conversation 4 Commits 1 Checks 1 Files changed 3 +117 -0

hsinhyeh commented 14 days ago

Description of your changes:

1. explicit content type with application/json on healthz response.
2. add healthz client api under common/client for integration test
3. add integration test for healthz endpoint

Checklist:

The title for your pull request (PR) should follow our title convention. [Learn more about the pull request title convention used in this repository.](#)

fix(backend): fixes healthz response by adding json content type. Fixes 977d08e

google-oss-prov bot added the size/L label 14 days ago

google-oss-prov bot commented 14 days ago

Hi @hsinhyeh. Thanks for your PR.

I'm waiting for a `kubeflow` member to verify that this patch is reasonable to test. If it is, they should reply with `/ok-to-test` on its own line. Until that is done, I will not automatically test new commits in this PR, but the usual testing commands by org members will still work. Regular contributors should [join the org](#) to skip this step.

Once the patch is verified, the new status will be reflected by the `ok-to-test` label.

I understand the commands that are listed [here](#).

Details

Reviewers

- zijianjoy
- chensun
- Linchin

Assignees

No one assigned

Labels

- needs-ok-to-test
- size/L

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you were mentioned.

Kserve/Kserve#2355 Case Study: kind/feature

Add an example of Lime text explainer from AIX360 toolkit #2355

Merged

kserve-oss-bot merged 9 commits into kserve:master from C1berwiz:master on Oct 16, 2022

Conversation 26

Commits 9

Checks 43

Files changed 7

+236 -6



C1berwiz commented on Jul 26, 2022 • edited by yuzisun

Contributor

What this PR does / why we need it:

This PR added an example of LIME text explainer of AIX360 toolkit

Since there are 3 modes of LIME explainer (image, text and tabular), but currently there is only a sample of image (MNIST)

We suppose that adding a text example would help others better understanding LIME explainer

Type of changes

Please delete options that are not relevant.

New feature (non-breaking change which adds functionality)

Feature/Issue validation/testing:

Please describe the tests that you ran to verify your changes and relevant result summary. Provide instructions so it can be reproduced.

Please also list any relevant details for your test configuration.

Test A

- like the mnist sample of aix and mentioned in README.md , this sample should execute just like mnist sample
- The first step is to [determine the ingress IP and ports](#) and set INGRESS_HOST and INGRESS_PORT

```
MODEL_NAME=aix-explainer
SERVICE_HOSTNAME=$(kubectl get inferenceservice ${MODEL_NAME} -o jsonpath='{.status.url}' | cut -d "/" -f 3)
python query_explain.py http://${INGRESS_HOST}:${INGRESS_PORT}/v1/models/${MODEL_NAME}:explain ${SERVICE_HOSTNAME}
```

Reviewers

yuzisun

theofpa

yuzliu

drewbutlerbb4

Tomcli

Still in progress? Learn about draft PRs

Assignees

yuzisun

Labels

approved lgtm

Projects

None yet

Milestone

No milestone

Hands on Kubeflow

Pre-requirement

- Be comfortable with UNIX command line
 - Navigating directories with `cd` or `tree`
 - Editing files, like `vim`, `nano`
 - Bash scripting, like env or looping
- Be an export with `Google`
 - <https://letmegooglet.com/?q=you+can+google+it>
- It is totally OK if you don't know what is Container and Kubernetes

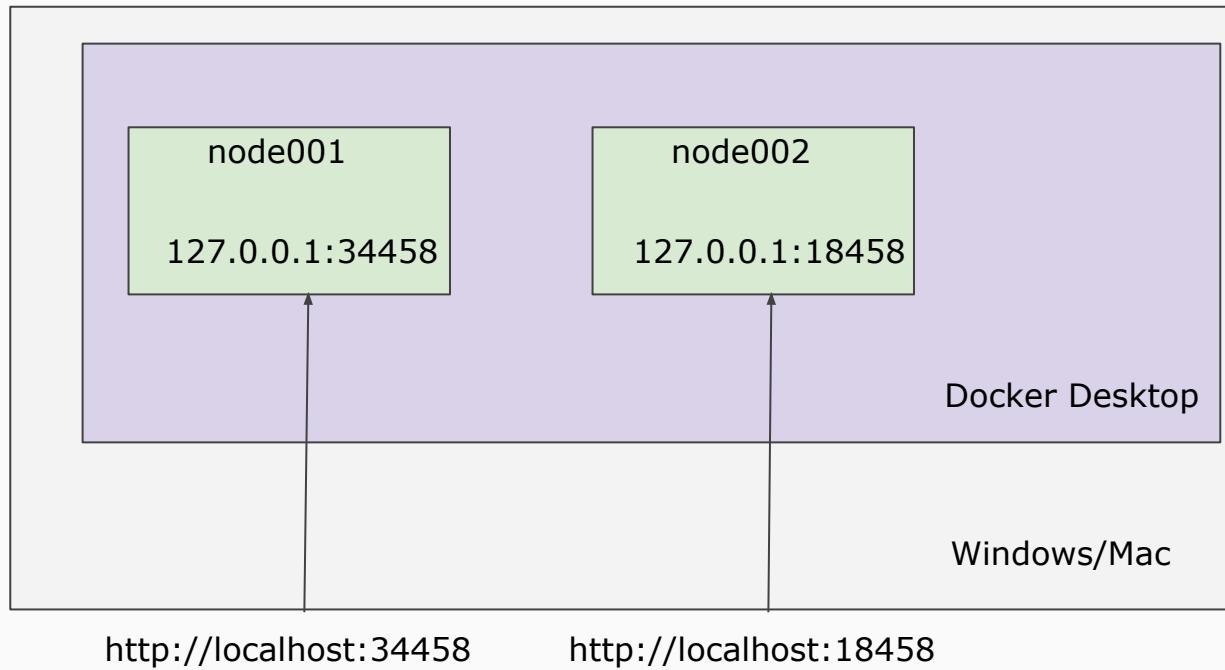
孩子，您多久沒唸中文了？

荀子《儒效篇》

「不聞不若聞之，聞之不若見之，見之不若知之，知之不若行之；學至于行之而止矣。」

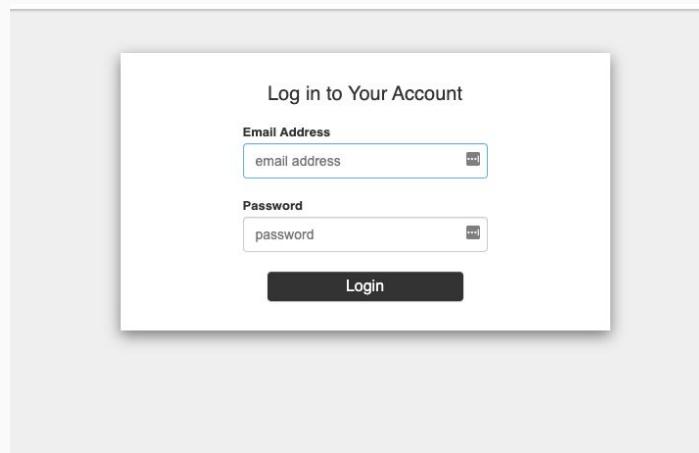
multikf: One-click Installation

- Multikf: <https://github.com/footprintai/multikf>



Wait! 所以我說那個帳號密碼呢?

Account: user@example.com
Password: 12341234



Step1: Use Notebook as Online IDE (1/3)

The screenshot shows the Kubeflow web interface. On the left, a sidebar menu is visible with the following items:

- Home
- Notebooks** (highlighted with a red box)
- Tensorboards
- Volumes
- Models
- Experiments (AutoML)
- Experiments (KFP)
- Pipelines
- Runs
- Recurring Runs
- Artifacts
- Executions

At the bottom of the sidebar, there is a link to "Privacy • Usage Reporting" and a note about the build version: "build version dev_local".

The main content area is titled "Notebooks" and displays a table with columns: Status, Name, Type, Age, Image, GPUs, CPUs, Memory, and Volumes. There is no data in the table.

A large red arrow points upwards from the "+ NEW NOTEBOOK" button, which is located in the top right corner of the main content area. The "+ NEW NOTEBOOK" button is also highlighted with a red box.

Step1: Use Notebook as Online IDE (2/3)

The screenshot shows the Kubeflow interface for creating a new Notebook Server. On the left is a sidebar with various options: Home, Notebooks (selected), Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, Artifacts, and Executions. Below these are Manage Contributors and Privacy & Help links.

The main area is titled "Specify the name of the Notebook Server and the Namespace it will belong to." It contains two input fields: "Name" (containing "demo") and "Namespace" (containing "kubeflow-user-example-com").

Below this is a section for the "Image". It includes a "Custom Image" checkbox (unchecked), a dropdown menu showing "jupyterlab" selected (with options 1 and 2), and a detailed view of the "jupyterlab" entry. This view shows the image URL "j1r0q0g6/notebooks/notebook-servers/jupyter-tensorflow-full:v1.4".

Under "Advanced Options", there is a "CPU / RAM" section. It asks to "Specify the total amount of CPU and RAM reserved by your Notebook Server. For CPU-intensive workloads, you can choose more than 1 CPU (e.g. 1.5)". Two input fields are shown: "Requested CPU" (containing "0.5") and "Requested Memory (in GiB)" (containing "1").

A large red box highlights the "Name" field. Two red arrows point from the text "Specify a name and resources like CPU and Memory" to the "Requested CPU" and "Requested Memory" fields respectively.

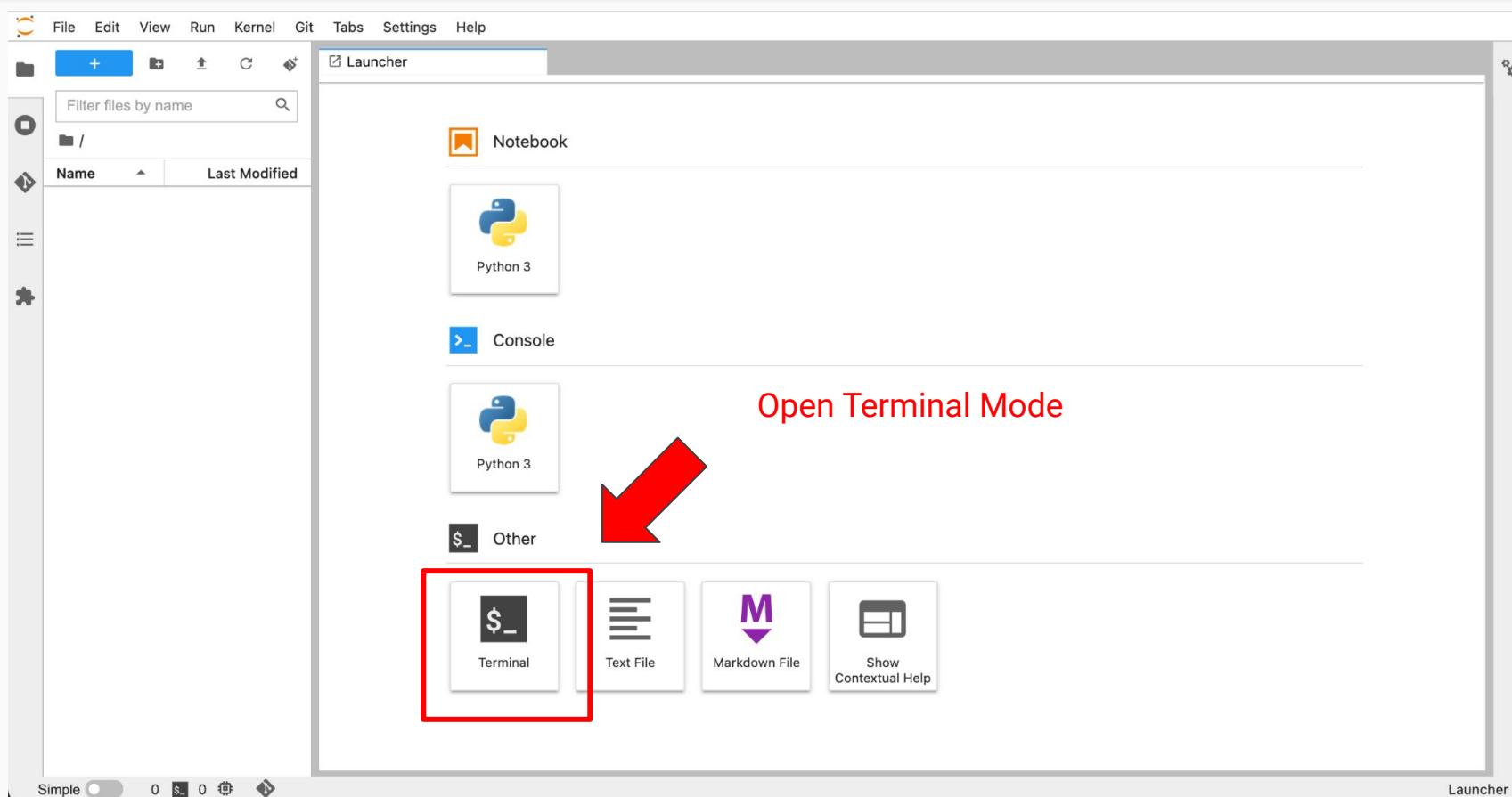
Specify a name and resources like CPU and Memory

Step1: Use Notebook as Online IDE (3/3)

The screenshot shows the Kubeflow interface with the 'Notebooks' section selected in the sidebar. The main area displays a table of notebooks with columns for Status, Name, Type, Age, Image, GPUs, CPUs, Memory, and Volumes. Two notebooks are listed: 'demo1' and 'demo2'. Both are in a 'Running' status. The 'demo1' row includes a 'CONNECT' button with a small icon. The 'demo2' row also includes a 'CONNECT' button with a small icon. A red box highlights the 'CONNECT' button for 'demo2'. A large red arrow points upwards from the bottom of the screen towards the 'CONNECT' button for 'demo2'.

Status	Name	Type	Age	Image	GPUs	CPUs	Memory	Volumes
✓	demo1	jupyter	20 hours ago	jupyter-scipy:v1.4	0	0.5	1Gi	
✓	demo2	jupyter	2 hours ago	jupyter-tensorflow-full:v1.4	0	0.5	1Gi	

Step2: Use terminal to download the materials (1/3)



Step2: Use terminal to download the materials (2/3)

The screenshot shows a Jupyter Notebook interface. On the left is a file browser sidebar. In the center is a terminal window titled "Terminal 1" containing a command-line session. A red box highlights the command "git clone https://github.com/footprintai/kubeflow-workshop". A large red arrow points upwards from a green box containing the same command to the highlighted terminal text. The green box is positioned below the terminal window.

```
groups: cannot find name for group ID 1222
(base) jovyan@dem01-0:~$ git clone https://github.com/footprintai/kubeflow-workshop
Cloning into 'kubeflow-workshop'...
remote: Enumerating objects: 164, done.
remote: Counting objects: 100% (164/164), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 164 (delta 89), reused 133 (delta 62), pack-reused 0
Receiving objects: 100% (164/164), 1.71 MiB | 8.29 MiB/s, done.
Resolving deltas: 100% (89/89), done.
(base) jovyan@dem01-0:~$
```

git clone https://github.com/footprintai/kubeflow-workshop

Simple 1 0 Terminal 1 51

Step2: Use terminal to download the materials (3/3)

The screenshot shows the Jupyter Notebook interface. On the left is a file browser pane with a sidebar containing icons for file operations like new file, copy, move, and delete. A search bar at the top of the browser allows filtering by file name. Below the search bar is a list of files under the path `/kubeflow-workshop / pipelines /`. The list includes:

- img (directory)
- 0.helloworld.ipynb
- 1.conditional-flow.ipynb
- 2.persistvolume.ipynb
- 3.calc_metrics.ipynb
- 4.mnist.ipynb
- 5.auto-mnist-with-katib.ipynb
- 6.kfserving.ipynb
- 7.kfserving-canary-rollout.ipynb
- kfp.ipynb
- testdata.jpg
- testdata2.jpg
- testdata3.jpg

A red box highlights the first seven items in the list: 0.helloworld.ipynb through 7.kfserving-canary-rollout.ipynb.

On the right is a terminal window titled "Terminal 1". It displays the output of a `git clone` command:

```
groups: cannot find name for group ID 1337
(base) jovyan@demo1-0:~$ git clone https://github.com/footprintai/kubeflow-workshop
Cloning into 'kubeflow-workshop'...
remote: Enumerating objects: 164, done.
remote: Counting objects: 100% (164/164), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 164 (delta 89), reused 133 (delta 62), pack-reused 0
Receiving objects: 100% (164/164), 1.71 MiB | 8.29 MiB/s, done.
Resolving deltas: 100% (89/89), done.
(base) jovyan@demo1-0:~$
```

The bottom status bar indicates the notebook is in "Simple" mode, has one tab open, and shows page numbers 1, 2, and 0. The terminal window is labeled "Terminal 1".

Kubeflow Terms

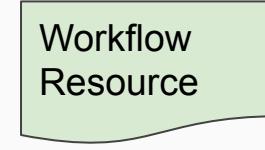
詞彙說明



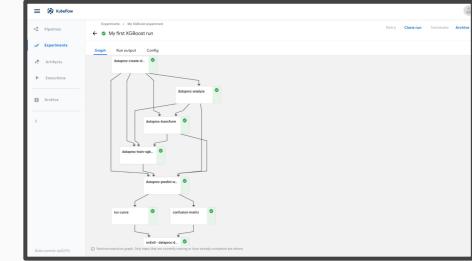
```
1.1 with open('requirements.txt', 'w') as f:  
    f.write("pyyaml<4.4.0,  
    pip install --upgrade --use-feature  
    --no-deps  
    --no-cache-dir  
    --no-binary=*
```

```
1.2 import sys  
1.3 import os  
1.4  
1.5 def echo(text):  
1.6     return os.GlobbingPattern(  
1.7         escape='\\',  
1.8         ignore='!library!(dash|4,4,2)',  
1.9         arguments=[text],  
2.0         text=text)  
2.1  
2.2 def echo_task1():  
2.3     echo('task 1')  
2.4     echo('task 1')  
2.5     echo('task 1')  
2.6     echo('task 1')  
2.7  
2.8 def echo_task2():  
2.9     echo('task 2')  
3.0  
3.1 def pipeline():  
3.2     echo('Execution order pipeline')  
3.3     echo_task1()  
3.4     echo_task2()  
3.5     echo('Execution order pipeline')  
3.6  
3.7 if __name__ == '__main__':  
3.8     pipeline()  
3.9  
3.10 # generate workflow artifacts in .zip format  
3.11 MyWorkflow().compile('MyWorkflow', 'yellowworld_pipeline', 'yellowworld.zip')
```

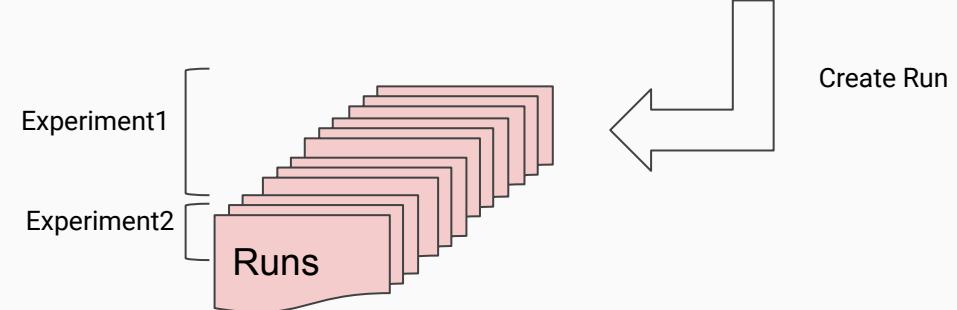
Compiled



Create a Pipeline



Pipeline Code



Hello World Example

Step4: Compile helloworld.ipynb (1/2)

The screenshot shows a Jupyter Notebook interface with several panes:

- File Explorer:** Shows a directory structure under "/kubeflow-workshop / pipelines /". The file "0.helloworld.ipynb" is selected.
- Terminal 1:** Displays Python code for pipeline compilation. A red box highlights the last line of code:

```
# generate workflow artifacts in .zip format
kfp.compiler.Compiler().compile(execution_order_pipeline, 'helloworld.zip')
```
- Code Editor:** Shows the Python code for defining the pipeline, including imports, function definitions for echo1_op and echo2_op, and the @dsl.pipeline decorator.
- Status Bar:** Shows "Simple" mode, Python 3 | Idle, and the file "0.helloworld.ipynb".

Step4: Compile helloworld.ipynb (2/2)



File Edit View Run Kernel Git Tabs Settings Help

Terminal 1 × 0.helloworld.ipynb Python 3

Filter files by name

/ kubeflow-workshop / pipelines /

Name	Last Modified
img	7 minutes ago
0.helloworld.ipynb	seconds ago
1.conditional-flow.ipynb	7 minutes ago
2.persistvolume.ipynb	7 minutes ago
3.calc_metrics.ipynb	7 minutes ago
4.mnist.ipynb	7 minutes ago
5.auto-mnist-with-katib.ipynb	7 minutes ago
6.kfserving.ipynb	7 minutes ago
7.kfsserving_end2end_order.ipynb	7 minutes ago
helloworld.zip	seconds ago
kfp.ipynb	7 minutes ago
requirements.txt	seconds ago
testdata.jpg	7 minutes ago
testdata2.jpg	7 minutes ago
testdata3.jpg	7 minutes ago

```
name='echo1',
image='library/bash:4.4.23',
command=['sh', '-c'],
arguments=['echo "$0"', text1]

def echo2_op(text2):
    return dsl.ContainerOp(
        name='echo2',
        image='library/bash:4.4.23',
        command=['sh', '-c'],
        arguments=['echo "$0"', text2])

@dsl.pipeline(
    name='Execution order pipeline',
    description='A pipeline to demonstrate execution order management.')
def execution_order_pipeline(text1='message 1', text2='message 2'):
    """A two step pipeline with an explicitly defined execution order."""
    step1_task = echo1_op(text1)
    step2_task = echo2_op(text2)
    step2_task.after(step1_task)

[3]: # generate workflow artifacts in .zip format
kfp.compiler.Compiler().compile(execution_order_pipeline, 'helloworld.zip')

/opt/conda/lib/python3.8/site-packages/kfp/dsl/_container_op.py:1150: FutureWarning: Please create reusable components instead of constructing ContainerOp instances directly. Reusable components are shareable, portable and have compatibility and support guarantees. Please see the documentation on: https://www.kubeflow.org/docs/pipelines/sdk/component-development/#writing-your-component-definition-file The components can be created manually (or, in case of python, using kfp.components.create_component_from_func or func_to_container_op) and then loaded using kfp.components.load_component_from_file, load_component_from_uri or load_component_from_text: https://kubeflow-pipelines.readthedocs.io/en/stable/source/kfp.components.html#kfp.components.load\_component\_from\_file
warnings.warn(
```

generated helloworld.zip

Step4: Create a Pipeline (1/7)

The screenshot shows the Kubeflow Pipelines interface. On the left, a sidebar menu is visible with the following items:

- Home
- Notebooks
- Tensorboards
- Volumes
- Models
- Experiments (AutoML)
- Experiments (KFP) (highlighted with a red box)
- Pipelines (highlighted with a red box)
- Runs
- Recurring Runs
- Artifacts
- Executions

The main content area is titled "Pipelines" and displays a list of existing pipelines. A red box highlights the "Upload pipeline" button in the top right corner. A large red arrow points upwards towards this button. The table lists the following pipelines:

<input type="checkbox"/>	Pipeline name	Description	Upload
<input type="checkbox"/>	[Tutorial] V2 lightweight Python com...	source code Shows different component input and output options for KFP v2 components.	11/30/2021, 1:02:25 PM
<input type="checkbox"/>	[Tutorial] DSL - Control structures	source code Shows how to use conditional execution and exit handlers. This pipeline will randomly fail to demonstr...	11/30/2021, 1:02:24 PM
<input type="checkbox"/>	[Tutorial] Data passing in python co...	source code Shows how to pass data between python components.	11/30/2021, 1:02:23 PM
<input type="checkbox"/>	[Demo] TFX - Taxi tip prediction mod...	source code GCP Permission requirements. Example pipeline that does classification with model analysis based on...	11/30/2021, 1:02:22 PM
<input type="checkbox"/>	[Demo] XGBoost - Iterative model tra...	source code This sample demonstrates iterative training using a train-eval-check recursive loop. The main pipeline ...	11/30/2021, 1:02:21 PM

At the bottom right of the main area, there is a "Rows per page: 10" dropdown and navigation arrows (< >).

Step4: Create a Pipeline (2/7)

The screenshot shows the Kubeflow interface for creating a new pipeline. The left sidebar lists various options like Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, Artifacts, and Executions. The main area is titled 'Upload Pipeline or Pipeline Version' and has two radio button options: 'Create a new pipeline' (selected) and 'Create a new pipeline version under an existing pipeline'. A red box highlights the 'Pipeline Name' field containing '0.helloworld', with a large red arrow pointing to it labeled '1.Pipeline Name'. Below it is a 'Pipeline Description' field with '0.helloworld'. The next section asks to choose a pipeline package file, with a red box highlighting the 'File*' input field containing 'helloworld (19).zip' and a 'Choose file' button, with a large red arrow pointing to it labeled '2.specify zip file location'. There's also an 'Import by url' option with a 'Package Url' input field. At the bottom, a red box highlights the 'Create' button, with a large red arrow pointing to it labeled '3.Create'.

Kubeflow

kubeflow-user-example-c... ▾

Pipeline Versions

← Upload Pipeline or Pipeline Version

Create a new pipeline Create a new pipeline version under an existing pipeline

Upload pipeline with the specified package.

Pipeline Name*
0.helloworld

Pipeline Description*
0.helloworld

Choose a pipeline package file from your computer, and give the pipeline a unique name.
You can also drag and drop the file here.
For expected file format, refer to [Compile Pipeline Documentation](#).

Upload a file
helloworld (19).zip Import by url
Package Url

Code Source (optional)

Create Cancel

1.Pipeline Name

2.specify zip file location

3.Create

Step4: Create a Pipeline (3/7)

The screenshot shows the Kubeflow Pipelines interface. On the left is a dark sidebar with navigation links: Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, Artifacts, and Executions. Below this is a "Manage Contributors" section.

The main area is titled "Pipelines" and shows a pipeline named "0.helloworld (0.helloworld)". It displays a "Graph" view of the pipeline, which consists of two nodes: "echo1" at the top and "echo2" below it, connected by a downward arrow. There is also a "YAML" tab. A "Simplify Graph" button is present.

At the top right of the main area are three buttons: "+ Create run", "+ Upload version", and "+ Create experiment". The "+ Create experiment" button is highlighted with a red box and a large red arrow pointing towards it from the bottom right. To the right of the arrow, the text "Create an experiment" is written in red.

A summary panel at the bottom left provides details about the pipeline:

Summary	
ID	6f25028f-01e3-4acd-9389-7ec2031fb04b
Version	0.helloworld
Version source	(dropdown menu)

Step4: Create a Pipeline (4/7)

The screenshot shows the Kubeflow interface for creating a new experiment. On the left is a dark sidebar with navigation links: Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP) (which is expanded to show Pipelines, Runs, Recurring Runs, Artifacts, and Executions), Manage Contributors, and GitHub integration.

The main area is titled "Experiments" and "← New experiment". It has a sub-section titled "Experiment details" with a descriptive text: "Think of an Experiment as a space that contains the history of all pipelines and their associated runs".

A form is displayed for entering experiment details:

- A text input field containing "0.helloworld.exp" is highlighted with a red border.
- An optional "Description (optional)" input field is empty.
- At the bottom are two buttons: "Next" (highlighted with a red border) and "Cancel".

A large red arrow points from the text instruction to the "Next" button.

Enter experiment name and press Next button.

Step4: Create a Pipeline (5/7)

The screenshot shows the Kubeflow UI for creating a pipeline run. The left sidebar contains navigation links: Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP) (selected), Pipelines, Runs, Recurring Runs, Artifacts, Executions, Manage Contributors, GitHub, Documentation, Privacy + Usage Reporting, and build version dev_local.

The main area is titled "Run details". It includes fields for "Pipeline", "Pipeline Version", "Run name", "Description (optional)", "Experiment", "Service Account (Optional)", "Run Type" (set to "One-off"), and "Run parameters". The "Run parameters" section contains fields for "text1" and "text2". At the bottom are "Start" and "skip this step" buttons.

Two red arrows point from the right towards the "Pipeline" and "Experiment" fields, which are highlighted with red boxes. Red text annotations provide instructions:

1. Run a pipeline and specify its version
2. Add its experiment name

Step4: Create a Pipeline (6/7)

Kubeflow

kubeflow-user-example-c... ▾

Experiments

← 0.helloworld.exp

Recurring run configs 0 active

Experiment description

Manage

Refresh Archive

Runs

+ Create run + Create recurring run Compare runs Clone run Archive

Active Archived

Filter runs

Run name	Status	Duration	Pipeline Version	Recurring Run	Start time
Run of 0.helloworld (1e261)	?	-	0.helloworld	-	12/2/2021, 4:33:10 PM

Rows per page: 10 < >

Run List

Step4: Create a Pipeline (7/7)

The screenshot shows the Kubeflow interface for creating a pipeline. On the left, the sidebar includes options like Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, Artifacts, and Executions. The main area displays a pipeline graph titled "Run of 0.helloworld (1e261)". The graph consists of two nodes: "echo1" and "echo2", connected by a downward arrow. A red box highlights this graph. To the right, a modal window titled "execution-order-pipeline-fxxfn-4223123588" shows the "Input/Output" tab. It lists "Input parameters" (text1: message 1) and "Input artifacts". Under "Output parameters" and "Output artifacts", it shows "main-logs" with the value "minio://mipeline/artifacts/execution-order-pipeline-qvlt5/2021/11/30/execution-order-pipeline-qvlt5-137025724/main.log" and "message 1". A large red arrow points upwards from the "Output artifacts" section towards the text "運行結果輸出" (Execution Result Output) at the bottom right.

Graph

Run output

Config

Simplify Graph

echo1

echo2

execution-order-pipeline-fxxfn-4223123588

Input/Output

Input parameters

text1 message 1

Input artifacts

Output parameters

Output artifacts

main-logs

minio://mipeline/artifacts/execution-order-pipeline-qvlt5/2021/11/30/execution-order-pipeline-qvlt5-137025724/main.log

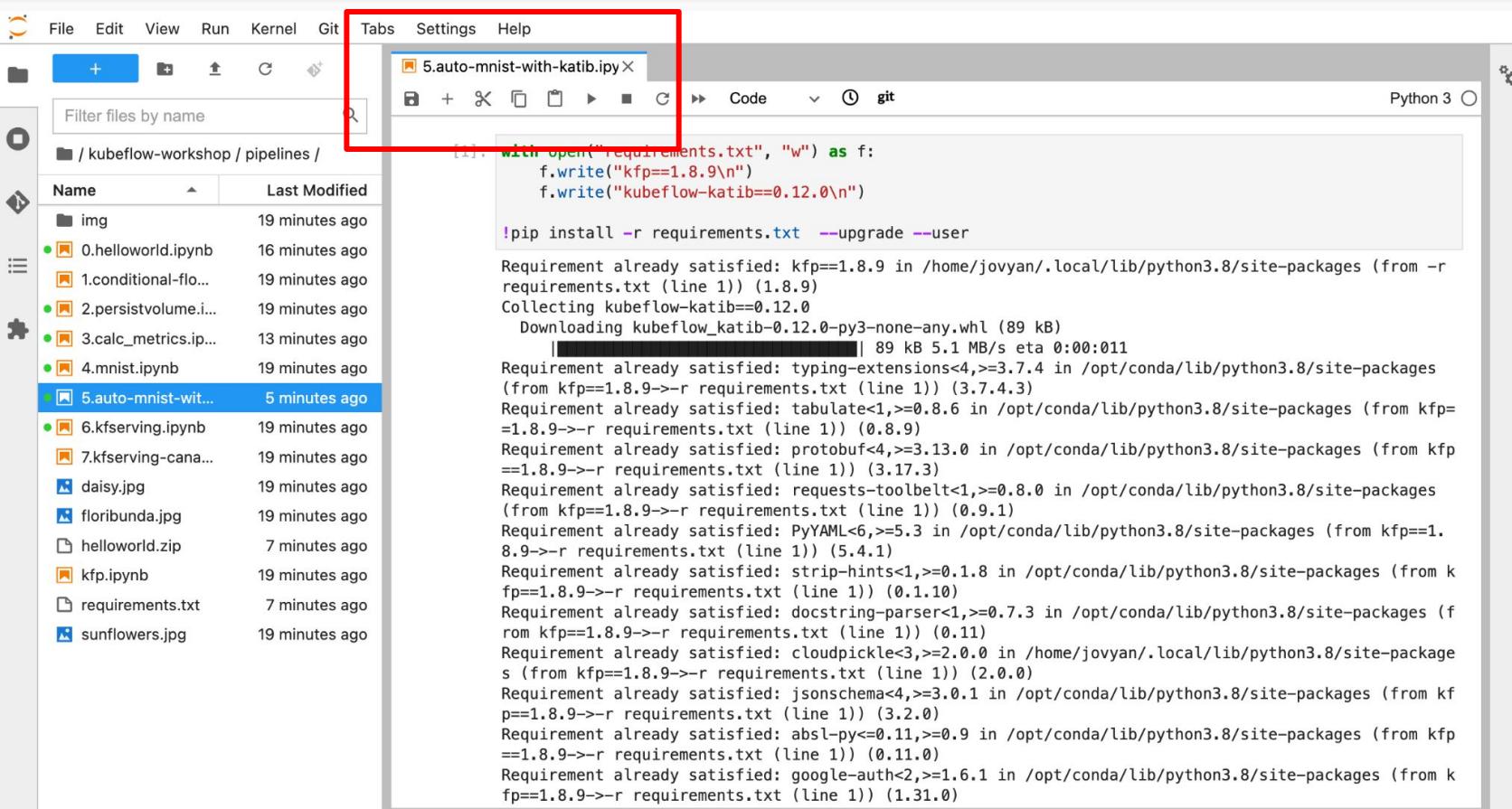
message 1

View All

運行結果輸出

Hyperparameter Example

Step5: Hyperparameter tuning with katib (1/4)



File Edit View Run Kernel Git Tabs Settings Help

5.auto-mnist-with-katib.ipynb

```
[1]: with open("requirements.txt", "w") as f:  
    f.write("kfp==1.8.9\n")  
    f.write("kubeflow-katib==0.12.0\n")  
  
!pip install -r requirements.txt --upgrade --user  
  
Requirement already satisfied: kfp==1.8.9 in /home/jovyan/.local/lib/python3.8/site-packages (from -r requirements.txt (line 1)) (1.8.9)  
Collecting kubeflow-katib==0.12.0  
    Downloading kubeflow_katib-0.12.0-py3-none-any.whl (89 kB)  
     ██████████| 89 kB 5.1 MB/s eta 0:00:011  
Requirement already satisfied: typing-extensions<4,>=3.7.4 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.7.4.3)  
Requirement already satisfied: tabulate<1,>=0.8.6 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.8.9)  
Requirement already satisfied: protobuf<4,>=3.13.0 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.17.3)  
Requirement already satisfied: requests-toolbelt<1,>=0.8.0 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.9.1)  
Requirement already satisfied: PyYAML<6,>=5.3 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (5.4.1)  
Requirement already satisfied: strip-hints<1,>=0.1.8 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.1.10)  
Requirement already satisfied: docstring-parser<1,>=0.7.3 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.11)  
Requirement already satisfied: cloudpickle<3,>=2.0.0 in /home/jovyan/.local/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (2.0.0)  
Requirement already satisfied: jsonschema<4,>=3.0.1 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.2.0)  
Requirement already satisfied: absl-py<=0.11,>=0.9 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.11.0)  
Requirement already satisfied: google-auth<2,>=1.6.1 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (1.31.0)
```

Step5: Hyperparameter tuning with katib (2/4)

The screenshot shows the Kubeflow interface. On the left, a sidebar navigation bar includes Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, and Artifacts. The 'Runs' item is selected. The main area displays the 'Experiments' page for the experiment 'hello1'. The 'Experiments' section shows 'Recurring run configs 0 active' and an 'Experiment description' input field. The 'Runs' section has tabs for 'Active' (selected) and 'Archived', with a 'Create run' and 'Create recurring run' button. A red box highlights the first row of the 'Runs' table, which lists three runs: 'Run of hello-world_version...' (Status: ?, Duration: -, Pipeline Version: hello-world_version..., Start time: 5/17/2022, 9:42:...), 'Run of hello-world_version...' (Status: ✓, Duration: 0:06:19, Pipeline Version: hello-world_version..., Start time: 5/17/2022, 9:30:...), and 'Run of hello-world_version...' (Status: ✓, Duration: 0:01:24, Pipeline Version: hello-world_version..., Start time: 5/17/2022, 9:23:...). The last two columns of the table are 'quotient' and 'remainder'.

Run name	Status	Duration	Pipeline Version	Recurri...	Start time	quotient	remainder
Run of hello-world_version...	?	-	hello-world_version...	-	5/17/2022, 9:42:...		
Run of hello-world_version...	✓	0:06:19	hello-world_version...	-	5/17/2022, 9:30:...		
Run of hello-world_version...	✓	0:01:24	hello-world_version...	-	5/17/2022, 9:23:...	0.000	6.000

Step5: Hyperparameter tuning with katib (3/4)

The screenshot shows the Kubeflow interface. On the left, a sidebar menu is visible with the following items:

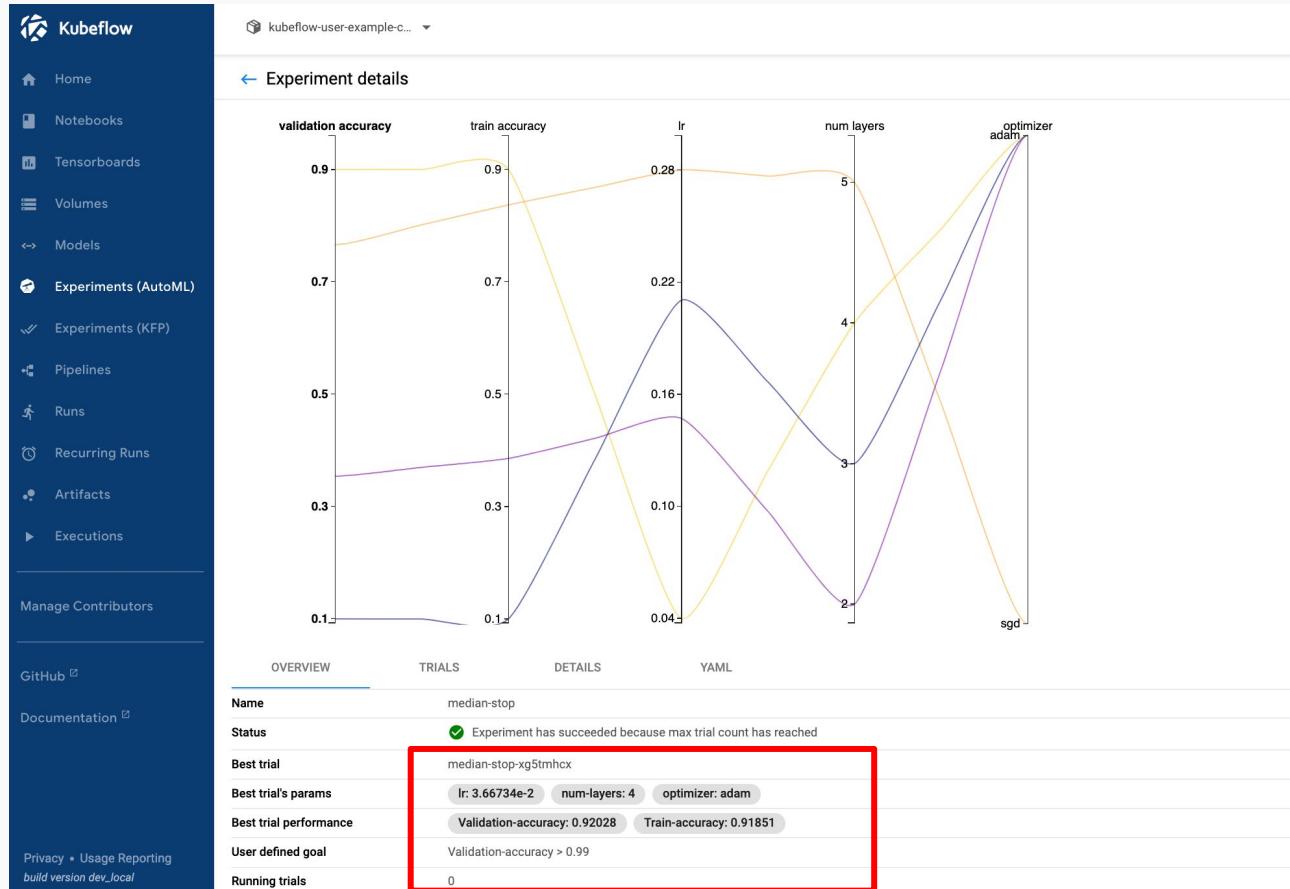
- Home
- Notebooks
- Tensorboards
- Volumes
- Models
 - Experiments (AutoML) **(highlighted by a red box)**
 - Experiments (KFP)
- Pipelines
- Runs
- Recurring Runs
- Artifacts

The main content area is titled "Experiments". It displays a table with the following columns: Status, Name, Age, Successful trials, Running trials, Failed trials, and Optimal trial. A single experiment entry is shown:

Status	Name	Age	Successful trials	Running trials	Failed trials	Optimal trial
✓	median-stop	8 minutes ago	4	0	0	Validation accuracy: 0.92028

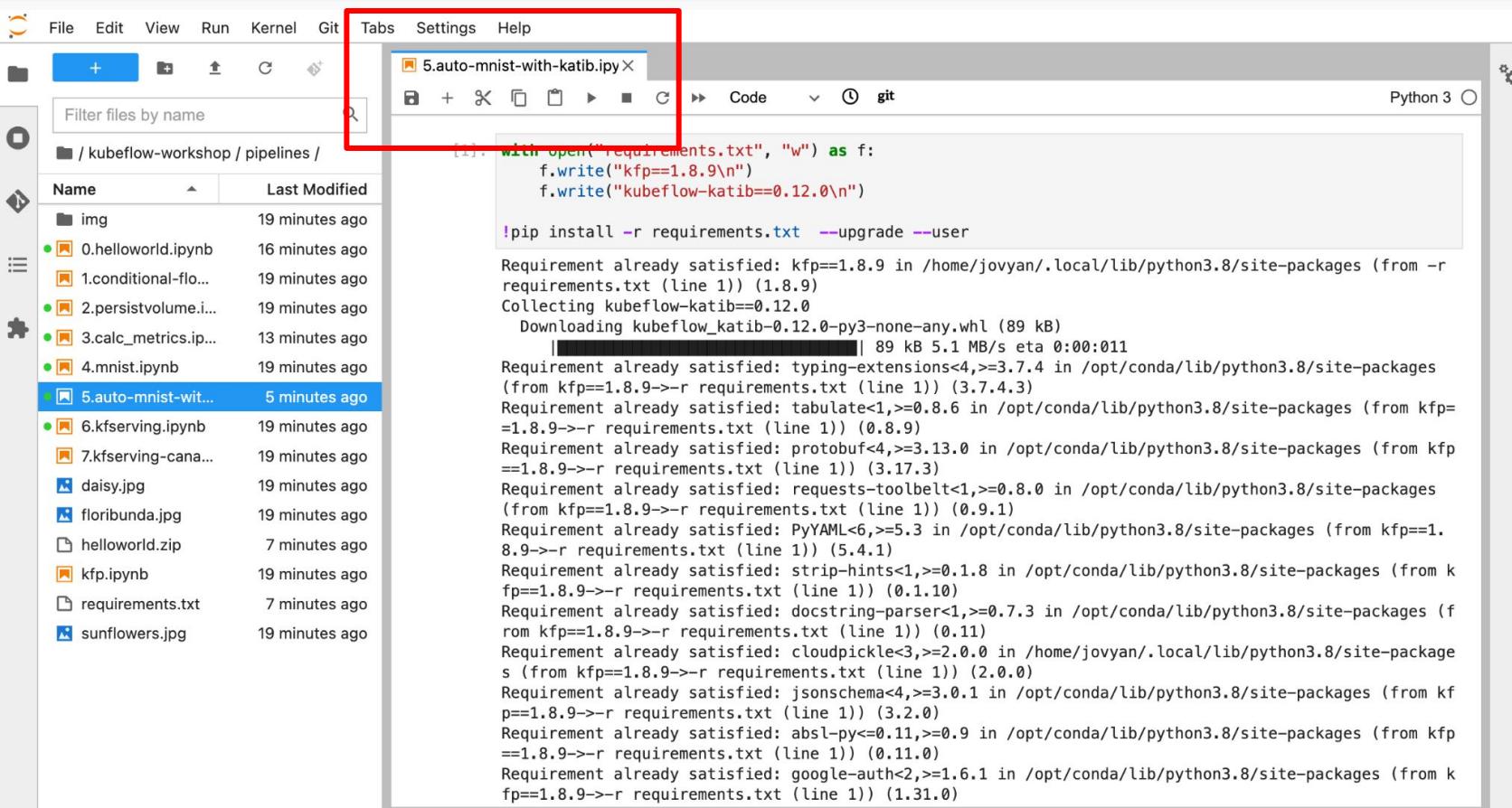
A blue "+ NEW EXPERIMENT" button is located in the top right corner of the experiments table.

Step5: Hyperparameter tuning with kubeflow (4/4)



Kserve Example

Step6: Serving model with Kserve (1/4)



```
File Edit View Run Kernel Git Tabs Settings Help  
5.auto-mnist-with-katib.ipynb  
+ Filter files by name  
/ kubeflow-workshop / pipelines /  
Name Last Modified  
img 19 minutes ago  
0.helloworld.ipynb 16 minutes ago  
1.conditional-flo... 19 minutes ago  
2.persistvolume.i... 19 minutes ago  
3.calc_metrics.ip... 13 minutes ago  
4.mnist.ipynb 19 minutes ago  
5.auto-mnist-wit... 5 minutes ago  
6.kfserving.ipynb 19 minutes ago  
7.kfserving-cana... 19 minutes ago  
daisy.jpg 19 minutes ago  
floribunda.jpg 19 minutes ago  
helloworld.zip 7 minutes ago  
kfp.ipynb 19 minutes ago  
requirements.txt 7 minutes ago  
sunflowers.jpg 19 minutes ago  
  
with open("requirements.txt", "w") as f:  
    f.write("kfp==1.8.9\n")  
    f.write("kubeflow-katib==0.12.0\n")  
  
!pip install -r requirements.txt --upgrade --user  
  
Requirement already satisfied: kfp==1.8.9 in /home/jovyan/.local/lib/python3.8/site-packages (from -r requirements.txt (line 1)) (1.8.9)  
Collecting kubeflow-katib==0.12.0  
  Downloading kubeflow_katib-0.12.0-py3-none-any.whl (89 kB)  
 |██████████| 89 kB 5.1 MB/s eta 0:00:011  
Requirement already satisfied: typing-extensions<4,>=3.7.4 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.7.4.3)  
Requirement already satisfied: tabulate<1,>=0.8.6 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.8.9)  
Requirement already satisfied: protobuf<4,>=3.13.0 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.17.3)  
Requirement already satisfied: requests-toolbelt<1,>=0.8.0 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.9.1)  
Requirement already satisfied: PyYAML<6,>=5.3 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (5.4.1)  
Requirement already satisfied: strip-hints<1,>=0.1.8 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.1.10)  
Requirement already satisfied: docstring-parser<1,>=0.7.3 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.11)  
Requirement already satisfied: cloudpickle<3,>=2.0.0 in /home/jovyan/.local/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (2.0.0)  
Requirement already satisfied: jsonschema<4,>=3.0.1 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (3.2.0)  
Requirement already satisfied: absl-py<=0.11,>=0.9 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (0.11.0)  
Requirement already satisfied: google-auth<2,>=1.6.1 in /opt/conda/lib/python3.8/site-packages (from kfp==1.8.9->-r requirements.txt (line 1)) (1.31.0)
```

Step6: Serving model with Kserve (2/4)

The screenshot shows the Kubeflow interface for managing machine learning experiments. On the left, a sidebar lists various options: Home, Notebooks, Tensorboards, Volumes, Models, Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, and Artifacts. The main area is titled 'Experiments' and shows a sub-section for 'hello1'. It includes a 'Recurring run configs' section with a 'Manage' button, an 'Experiment description' field, and a 'Create run' button. Below this is a 'Runs' section with tabs for 'Active' (selected) and 'Archived'. A red box highlights the first row of the 'Active' runs table, which contains the following data:

Run name	Status	Duration	Pipeline Version	Recurri...	Start time	quotient	remainder
Run of hello-world_version_1	?	-	hello-world_version_1	-	5/17/2022, 9:42:...		
Run of hello-world_version_1	✓	0:06:19	hello-world_version_1	-	5/17/2022, 9:30:...		
Run of hello-world_version_1	✓	0:01:24	hello-world_version_1	-	5/17/2022, 9:23:...	0.000	6.000

Step6: Serving model with Kserve (3/4)

The screenshot shows the Kubeflow interface. On the left, a dark sidebar lists various components: Home, Notebooks, Tensorboards, Volumes, Models (which is highlighted with a red box), Experiments (AutoML), Experiments (KFP), Pipelines, Runs, Recurring Runs, and Artifacts. The main area is titled "Model Servers" and contains a table with the following data:

Status	Name	Age	Predictor	Runtime	Protocol	Storage URI
✓	tensorflow-sample	2 minutes ago	Tensorflow	1.14.0		gs://kfserving-samples/models/tensorflow/flo...

A red box highlights both the "Models" item in the sidebar and the "tensorflow-sample" row in the table. To the right of the table, there are "NEW MODEL SERVER" and "Edit" buttons.

Step6: Serving model with Kserve (4/4)

The screenshot shows a Jupyter Notebook interface with the following details:

- File Bar:** File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help.
- File Explorer:** Shows a directory structure under "/kubeflow-workshop / pipelines /".
- Code Cell:** The active cell (cell 4) contains Python code for sending a REST API request to an Istio-ingressgateway. The code uses the `requests` library to encode an image file as base64 and send it to the endpoint `http://istio-ingressgateway.istio-system/v1/models/tensorflow-sample:predict`. The response is printed to the console.
- Output Cell:** The output of cell 4 is visible below the code cell, showing the JSON response from the server.
- Network Tab:** A network traffic capture tool is open, showing a table of cookies. It includes columns for Name, Value, Domain, Path, Expires/M..., Size, HttpOnly, Secure, SameSite, SameParty, and Priority. One cookie, `authservice_session`, is highlighted.
- Page Number:** The bottom right corner indicates page 74.

```
[4]: ## the following example use python's request to send restapi requests
import base64
import json
import requests

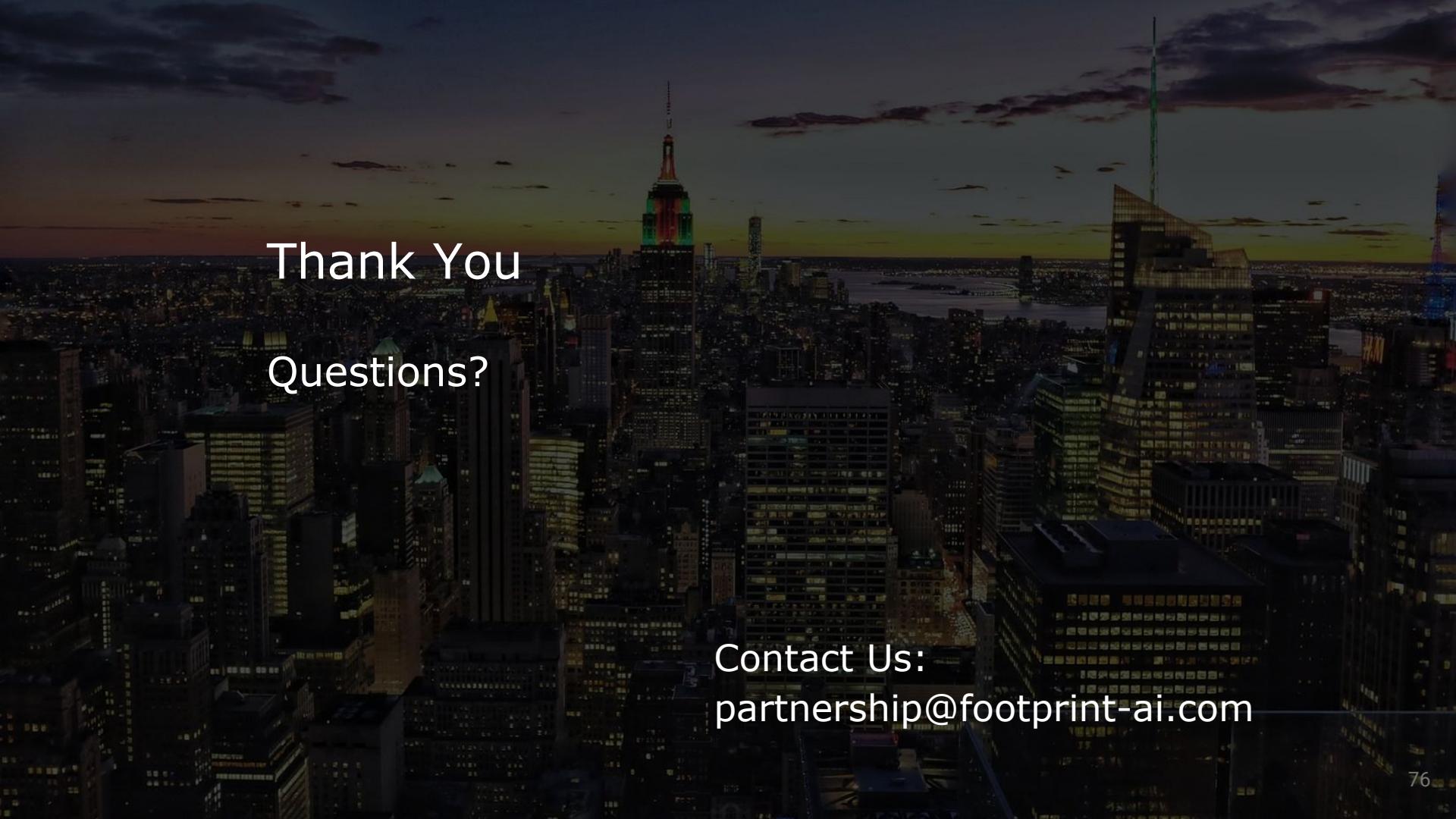
with open('floribunda.jpg', 'rb') as f:
    image_content = f.read()
    image_64_encode = base64.encodebytes(image_content).decode('utf-8')
headers = {"Cookie": "authservice_session=MTY1Mjc5MzE5M3x0d3dBTkZkUk4xZER0MGRZVmpZMldsTkpWVFUzU2s5V1dF
    "Host": "tensorflow-sample.kubeflow-user-example-com.example.com"}
payload = {"instances": [{"image_bytes": {"b64": image_64_encode}, "key": "1"}]}
resp = requests.post('http://istio-ingressgateway.istio-system/v1/models/tensorflow-sample:predict', he
print(resp.text)

{
    "predictions": [
        {
            "scores": [1.30671893e-07, 3.01086693e-08, 0.814807534, 9.6436537e-08, 0.185192183, 3.4390
2293e-08],
            "prediction": 2,
            "key": "1"
        }
    ]
}
```

```
[1]: ## the following example use curl to send restapi requests
```

Conclusion

- Open source is a public artifacts that help you grow a reputation and a career.
- The work to contribute to Open Source could take days or weeks, but the rewards could keep years.
- Let's make changes to the software world now, even steps are small :)



Thank You

Questions?

Contact Us:
partnership@footprint-ai.com