Unified AI & ML runtime

How SageMaker Distribution aims to radically simplify ML development

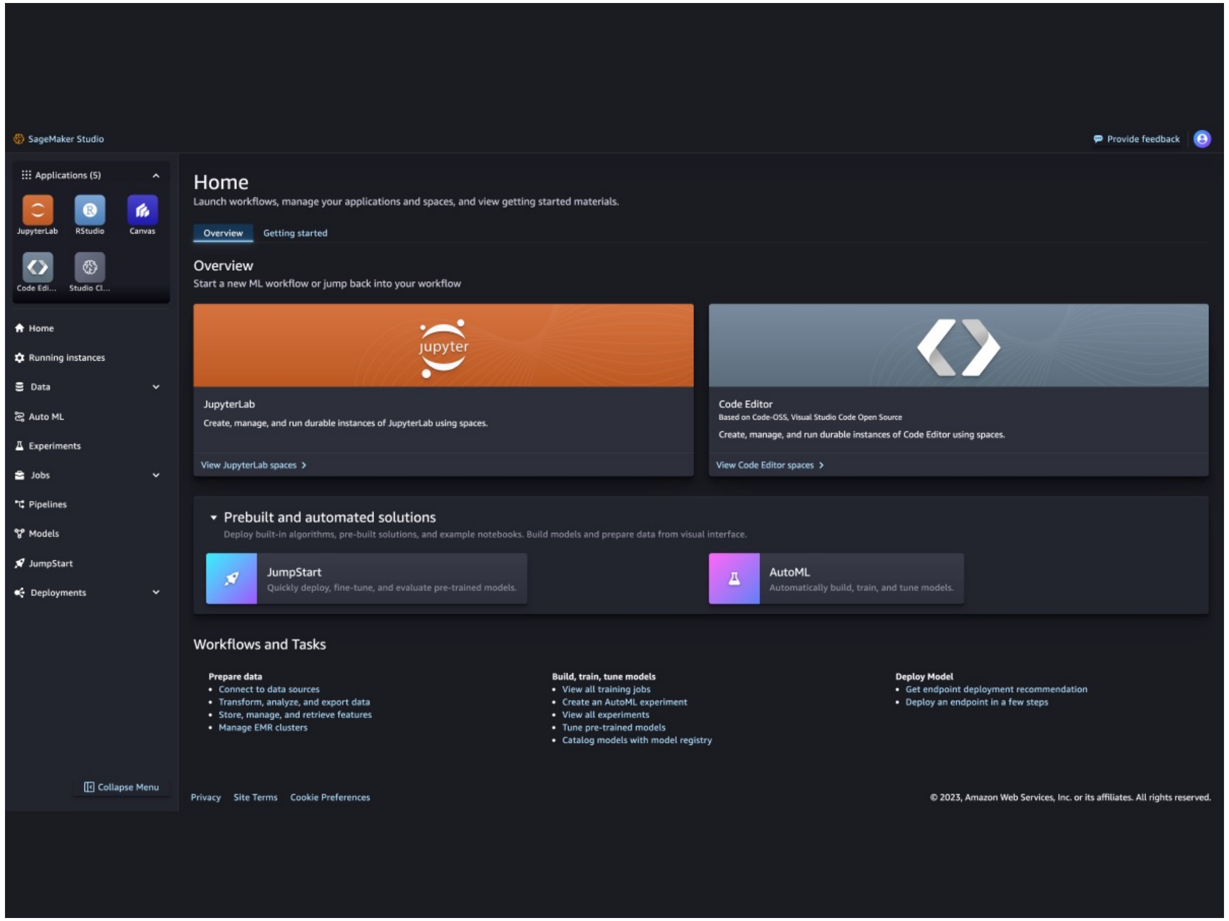
Presented by: Ketan Vijayvargiya

10/10/2024

Introduction

- Principal Engineer @ AWS AI/ML.
- First time speaker at Community over Code!

Background: SageMaker Studio



Home

Launch workflows, manage your applications and spaces, and view getting started materials.

Overview

Start a new ML workflow or jump back into your workflow

JupyterLab

Create, manage, and run durable instances of JupyterLab using spaces.

Code Editor

Based on Code-OSS, Visual Studio Code Open Source

Create, manage, and run durable instances of Code Editor using spaces.

Prebuilt and automated solutions

Deploy built-in algorithms, pre-built solutions, and example notebooks. Build models and prepare data from visual interface.

JumpStart

Quickly deploy, fine-tune, and evaluate pre-trained models.

AutoML

Automatically build, train, and tune models.

Workflows and Tasks

Prepare data
- Connect to data sources
- Transform, analyze, and export data
- Store, manage, and retrieve features
- Manage EMR clusters

Build, train, tune models
- View all training jobs
- Create an AutoML experiment
- View all experiments
- Tune pre-trained models
- Catalog models with model registry

Deploy Model
- Get endpoint deployment recommendation
- Deploy an endpoint in a few steps

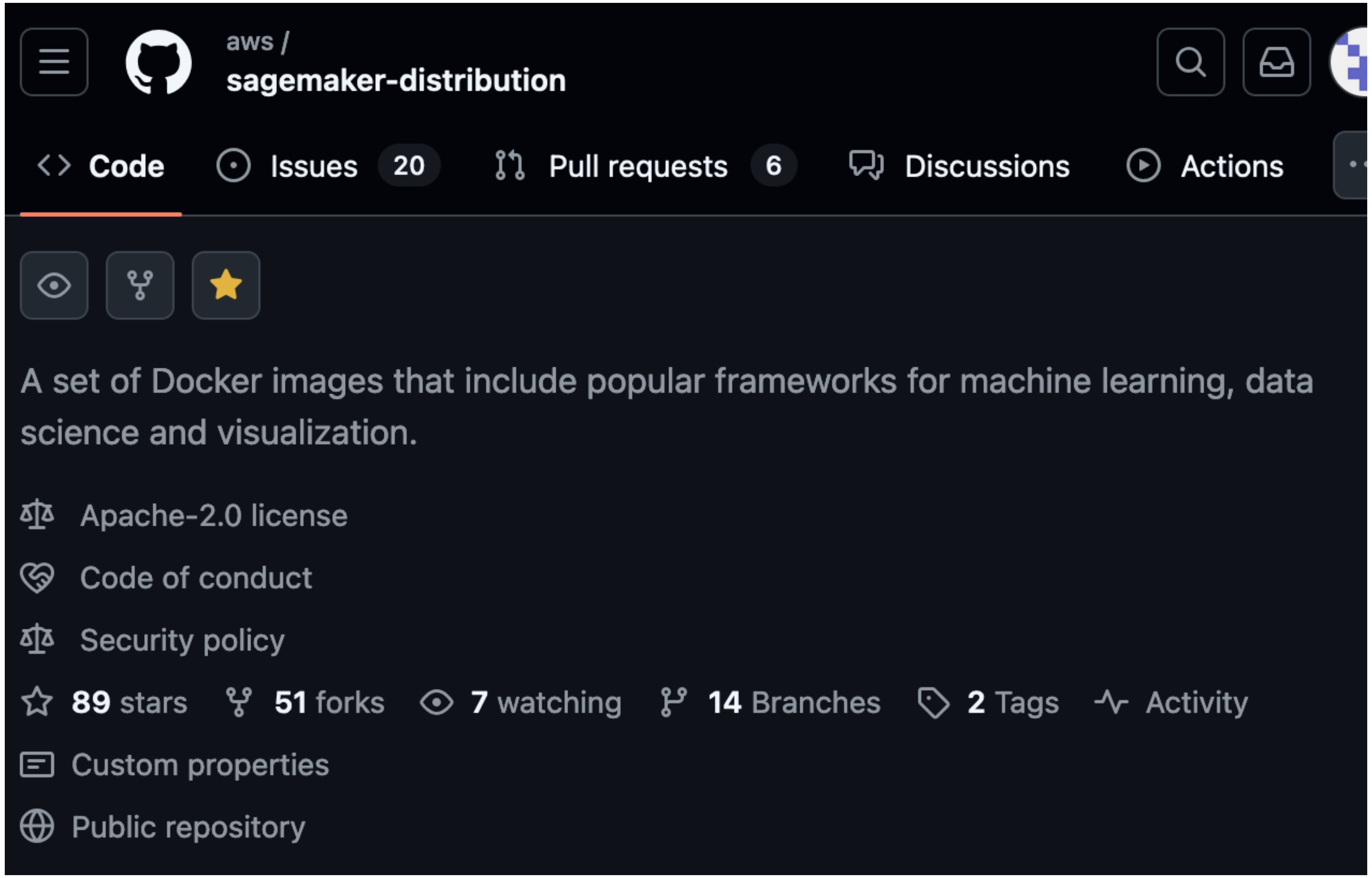
Privacy Site Terms Cookie Preferences

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Motivation

- Purpose-built or generalized runtime?
- Open-source.

SageMaker Distribution



aws / sagemaker-distribution

A set of Docker images that include popular frameworks for machine learning, data science and visualization.

Apache-2.0 license

Code of conduct

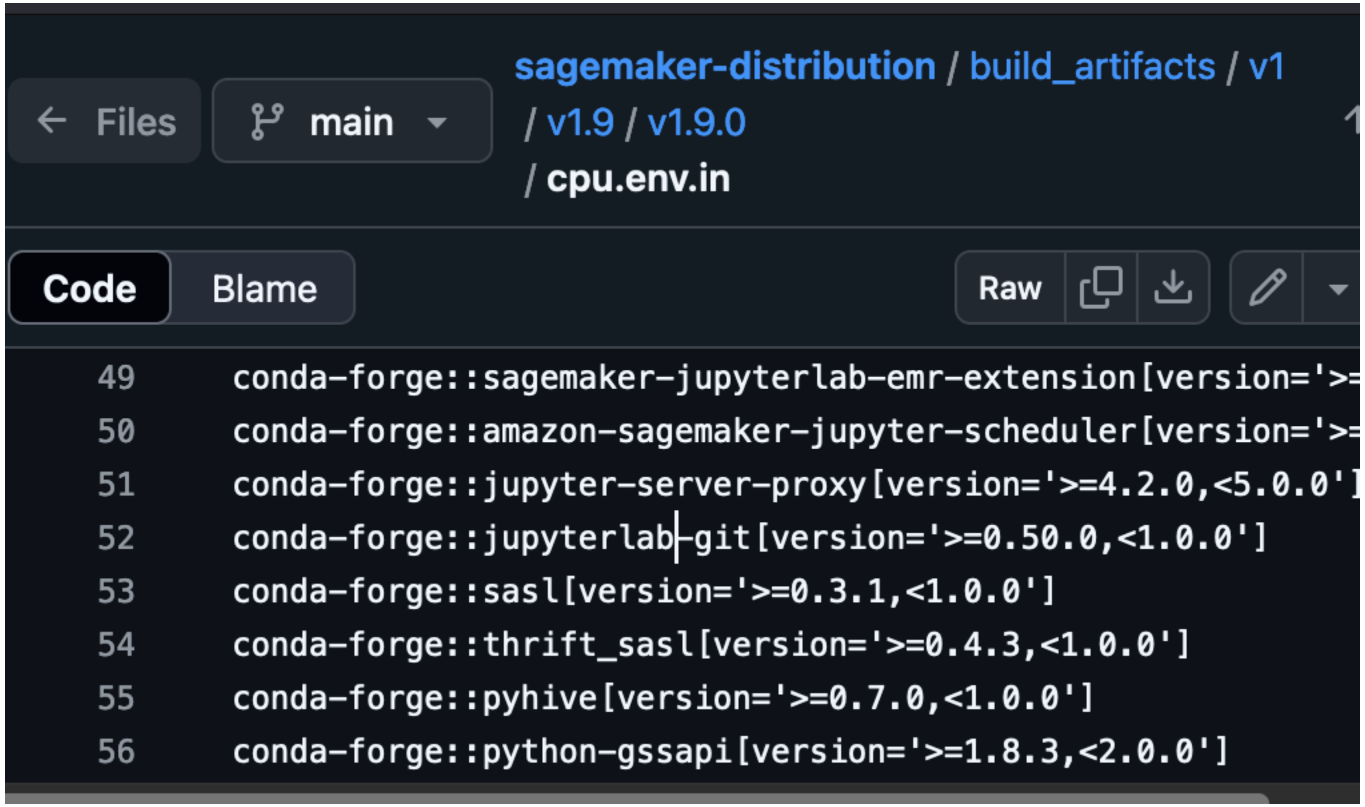
Security policy

89 stars   51 forks   7 watching   14 Branches   2 Tags   Activity

Custom properties

Public repository

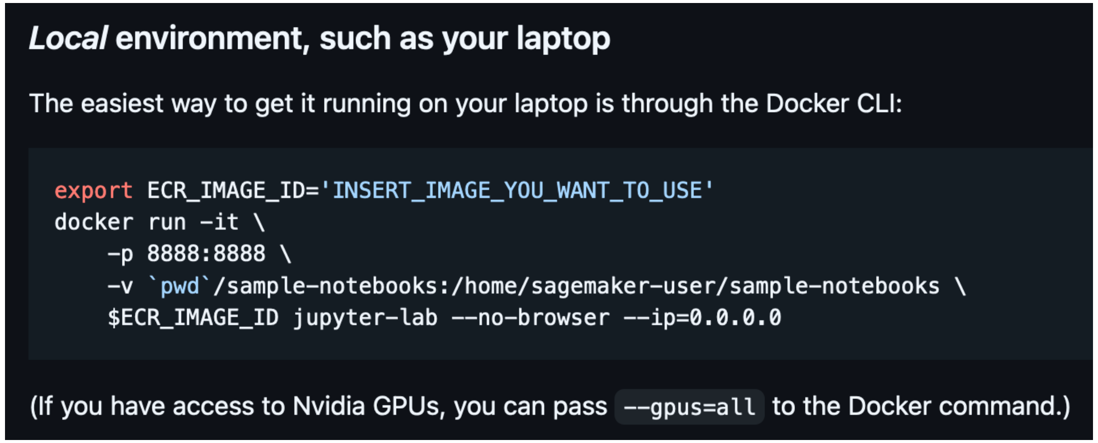
“Unified” runtime



sagemaker-distribution / build_artifacts / v1 / v1.9 / v1.9.0 / cpu.env.in

```
conda-forge::sagemaker-jupyterlab-emr-extension[version='>=
conda-forge::amazon-sagemaker-jupyter-scheduler[version='>=
conda-forge::jupyter-server-proxy[version='>=4.2.0,<5.0.0']
conda-forge::jupyterlab-git[version='>=0.50.0,<1.0.0']
conda-forge::sasl[version='>=0.3.1,<1.0.0']
conda-forge::thrift_sasl[version='>=0.4.3,<1.0.0']
conda-forge::pyhive[version='>=0.7.0,<1.0.0']
conda-forge::python-gssapi[version='>=1.8.3,<2.0.0']
```

Pre-built Docker images, with a built-in IDE

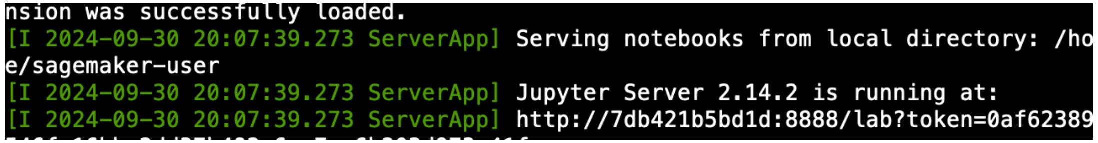


Local environment, such as your laptop

The easiest way to get it running on your laptop is through the Docker CLI:

```
export ECR_IMAGE_ID='INSERT_IMAGE_YOU_WANT_TO_USE'
docker run -it \
    -p 8888:8888 \
    -v `pwd`/sample-notebooks:/home/sagemaker-user/sample-notebooks \
    $ECR_IMAGE_ID jupyter-lab --no-browser --ip=0.0.0.0
```

(If you have access to Nvidia GPUs, you can pass `--gpus=all` to the Docker command.)



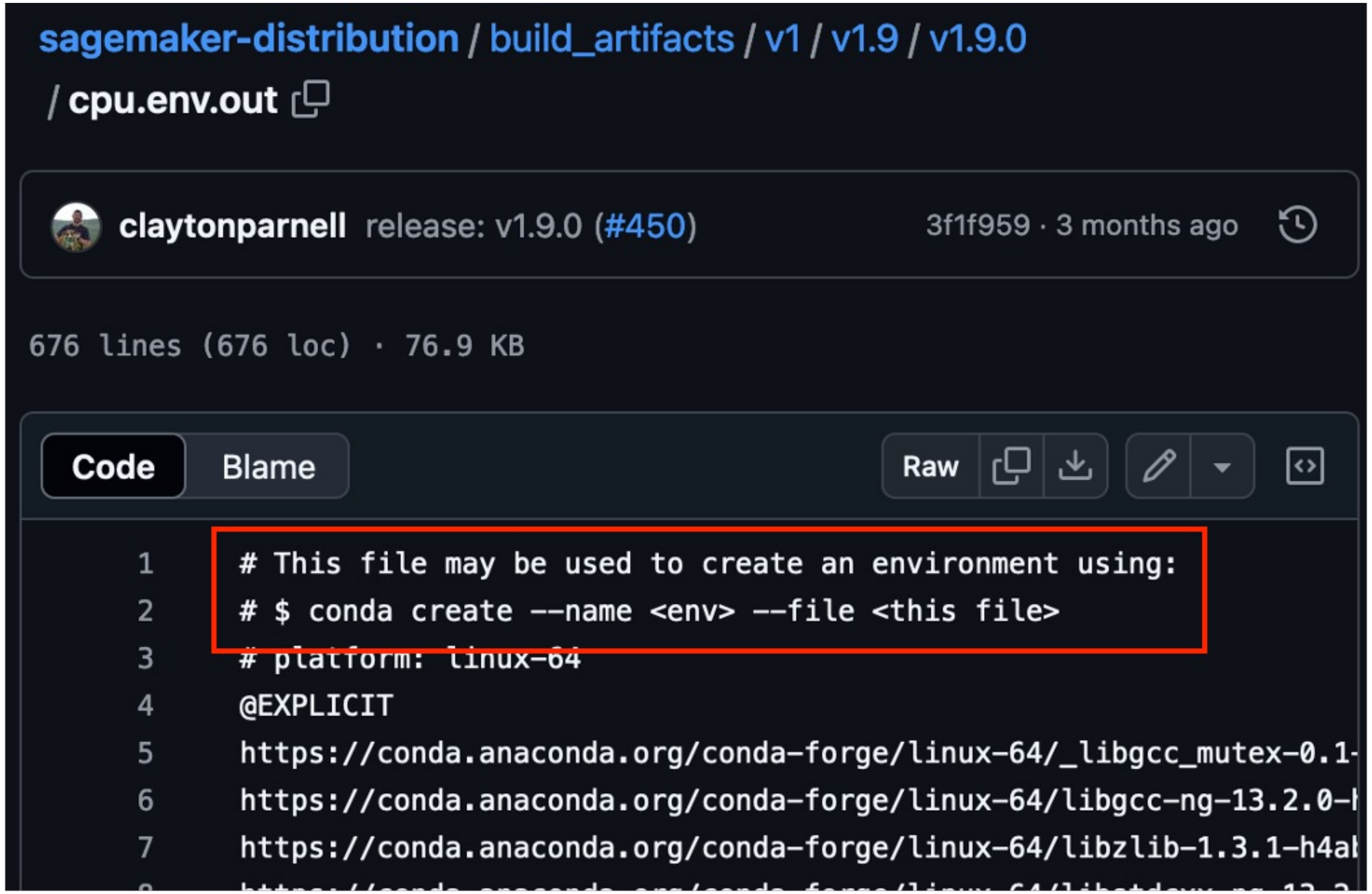
```
nsion was successfully loaded.
[I 2024-09-30 20:07:39.273 ServerApp] Serving notebooks from local directory: /ho
e/sagemaker-user
[I 2024-09-30 20:07:39.273 ServerApp] Jupyter Server 2.14.2 is running at:
[I 2024-09-30 20:07:39.273 ServerApp] http://7db421b5bd1d:8888/lab?token=0af62389
```

Image tags

Example: *v3.2.1* is the latest release. Images tagged with:

1. *latest-gpu*
2. *v3-gpu*
3. *v3.2-gpu*
4. *v3.2.1-gpu* (immutable).

Reproducible environments



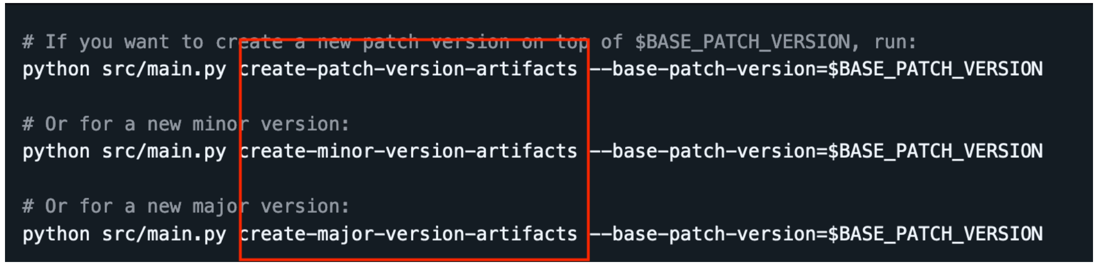
sagemaker-distribution / build_artifacts / v1 / v1.9 / v1.9.0 / cpu.env.out

claytonparnell release: v1.9.0 (#450) 3f1f959 · 3 months ago

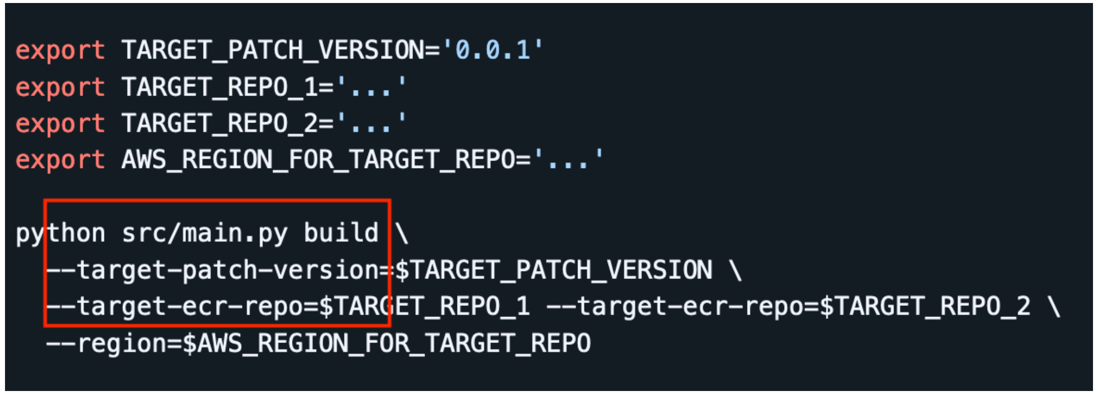
676 lines (676 loc) · 76.9 KB

```
# This file may be used to create an environment using:
# $ conda create --name <env> --file <this file>
# platform: linux-64
@EXPLICIT
https://conda.anaconda.org/conda-forge/linux-64/_libgcc_mutex-0.1-
https://conda.anaconda.org/conda-forge/linux-64/libgcc-ng-13.2.0-h
https://conda.anaconda.org/conda-forge/linux-64/libzlib-1.3.1-h4ab
```

Automated version upgrades

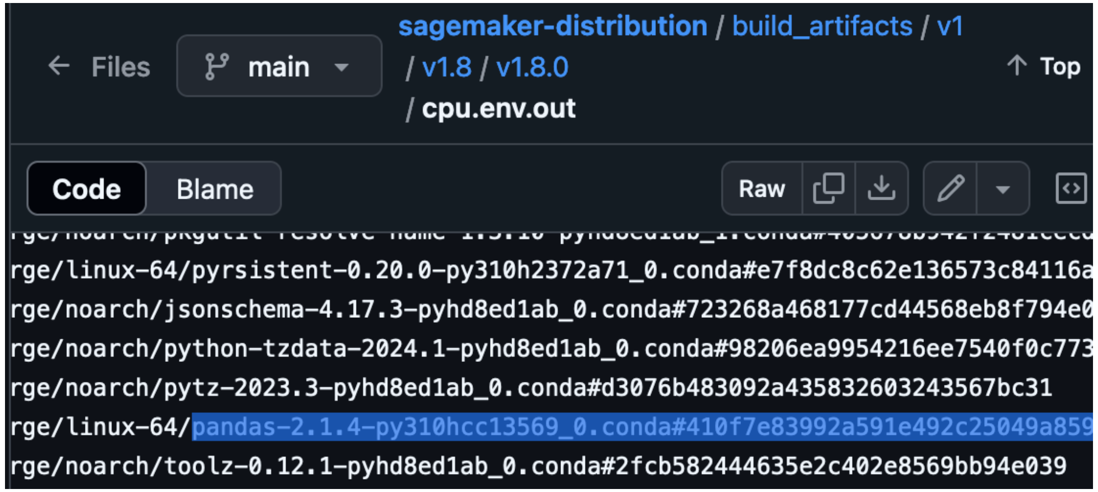


```
# If you want to create a new patch version on top of $BASE_PATCH_VERSION, run:
python src/main.py create-patch-version-artifacts --base-patch-version=$BASE_PATCH_VERSION

# Or for a new minor version:
python src/main.py create-minor-version-artifacts --base-patch-version=$BASE_PATCH_VERSION

# Or for a new major version:
python src/main.py create-major-version-artifacts --base-patch-version=$BASE_PATCH_VERSION
```



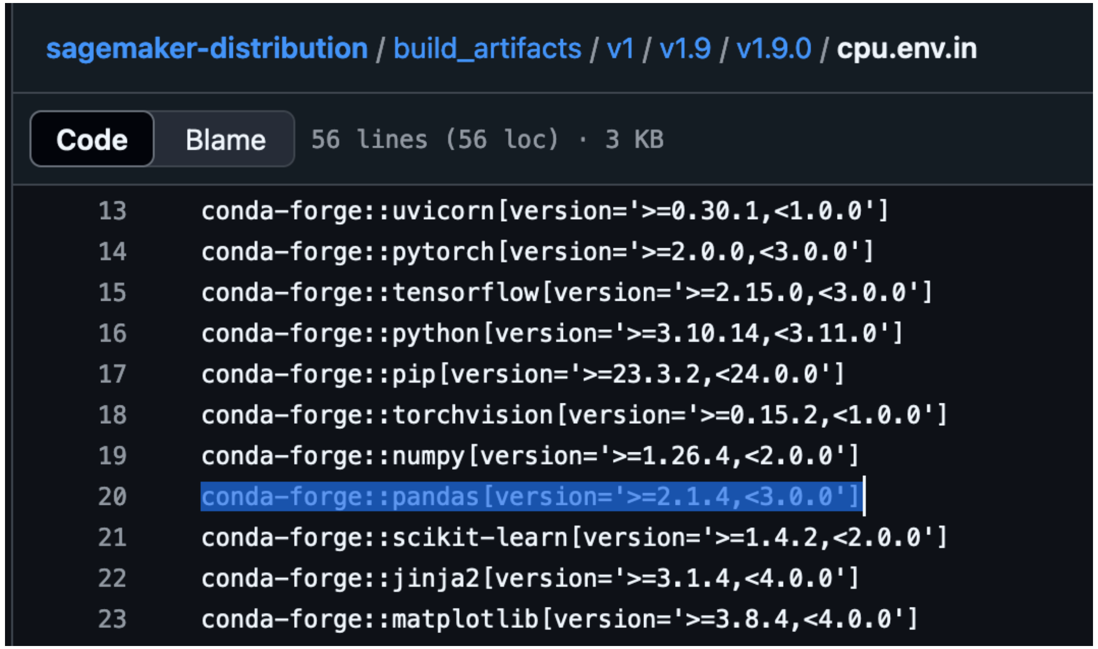
```
export TARGET_PATCH_VERSION='0.0.1'
export TARGET_REPO_1='...'
export TARGET_REPO_2='...'
export AWS_REGION_FOR_TARGET_REPO='...'

python src/main.py build \
  --target-patch-version=$TARGET_PATCH_VERSION \
  --target-ecr-repo=$TARGET_REPO_1 --target-ecr-repo=$TARGET_REPO_2 \
  --region=$AWS_REGION_FOR_TARGET_REPO
```

Example: “minor” upgrade from v1.8.x to v1.9



sagemaker-distribution / build_artifacts / v1 / v1.8 / v1.8.0 / cpu.env.out

```
rge/linux-64/pyrsistent-0.20.0-py310h2372a71_0.conda#e7f8dc8c62e136573c84116a
rge/noarch/jsonschema-4.17.3-pyhd8ed1ab_0.conda#723268a468177cd44568eb8f794e0
rge/noarch/python-tzdata-2024.1-pyhd8ed1ab_0.conda#98206ea9954216ee7540f0c773
rge/noarch/pytz-2023.3-pyhd8ed1ab_0.conda#d3076b483092a435832603243567bc31
rge/linux-64/pandas-2.1.4-py310hcc13569_0.conda#410f7e83992a591e492c25049a859
rge/noarch/toolz-0.12.1-pyhd8ed1ab_0.conda#2fcb582444635e2c402e8569bb94e039
```

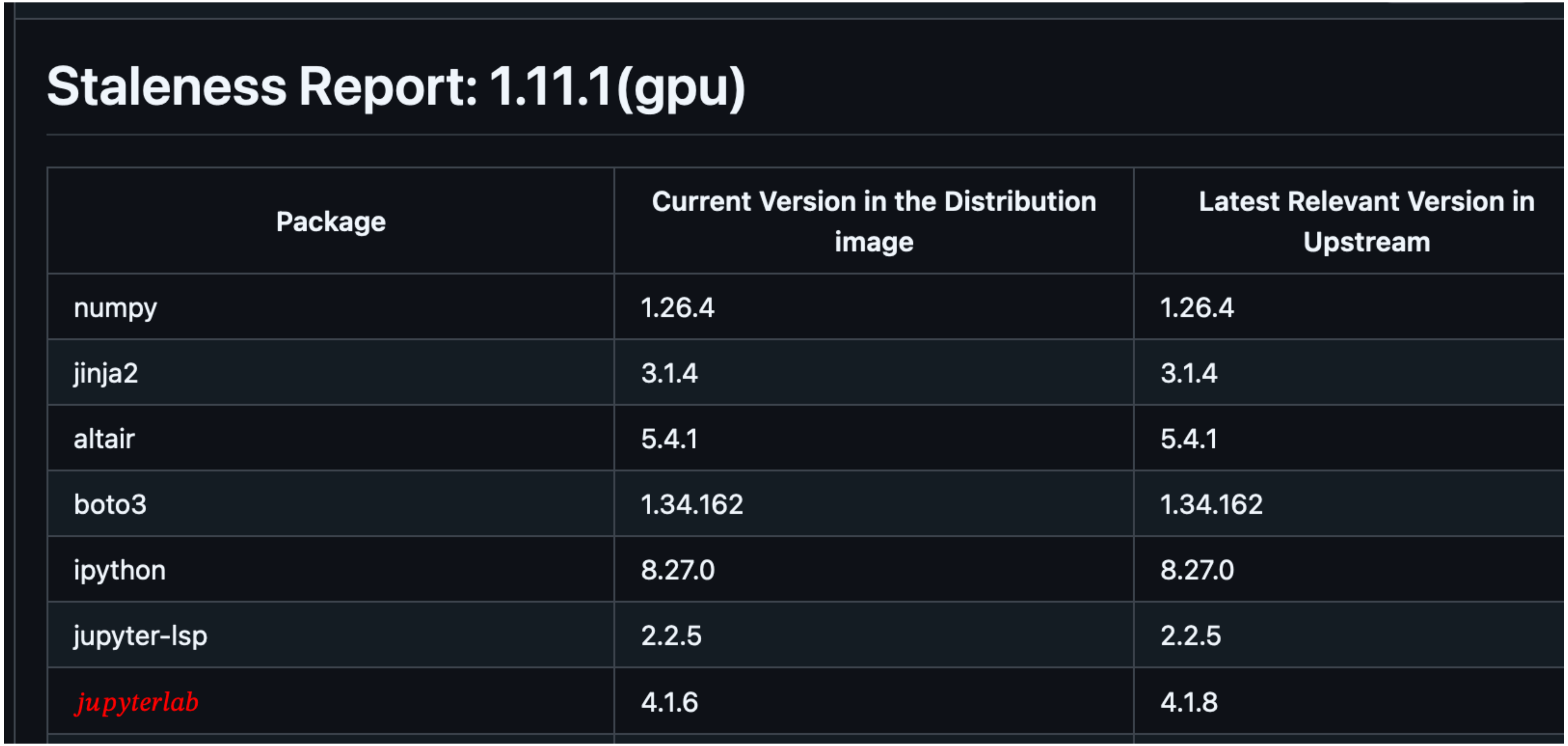


sagemaker-distribution / build_artifacts / v1 / v1.9 / v1.9.0 / cpu.env.in

56 lines (56 loc) · 3 KB

```
conda-forge::uvicorn[version='>=0.30.1,<1.0.0']
conda-forge::pytorch[version='>=2.0.0,<3.0.0']
conda-forge::tensorflow[version='>=2.15.0,<3.0.0']
conda-forge::python[version='>=3.10.14,<3.11.0']
conda-forge::pip[version='>=23.3.2,<24.0.0']
conda-forge::torchvision[version='>=0.15.2,<1.0.0']
conda-forge::numpy[version='>=1.26.4,<2.0.0']
conda-forge::pandas[version='>=2.1.4,<3.0.0']
conda-forge::scikit-learn[version='>=1.4.2,<2.0.0']
conda-forge::jinja2[version='>=3.1.4,<4.0.0']
conda-forge::matplotlib[version='>=3.8.4,<4.0.0']
```

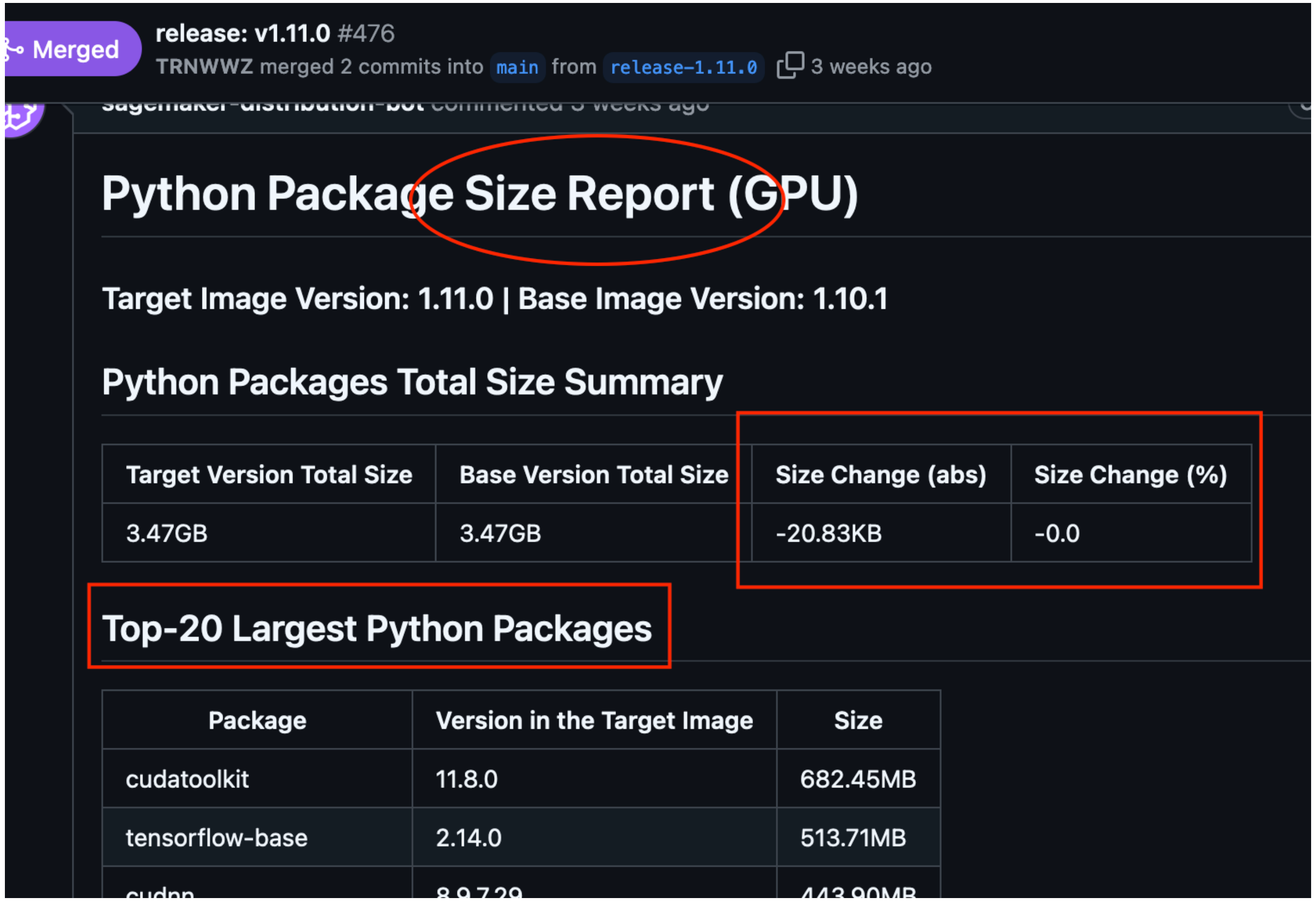
Automated staleness reporting



Staleness Report: 1.11.1(gpu)

Package	Current Version in the Distribution image	Latest Relevant Version in Upstream
numpy	1.26.4	1.26.4
jinja2	3.1.4	3.1.4
altair	5.4.1	5.4.1
boto3	1.34.162	1.34.162
ipython	8.27.0	8.27.0
jupyter-lsp	2.2.5	2.2.5
jupyterlab	4.1.6	4.1.8

Automated image-size reporting



release: v1.11.0 #476

TRNWWZ merged 2 commits into `main` from `release-1.11.0` 3 weeks ago

sagemaker-distribution-bot commented 3 weeks ago

Python Package Size Report (GPU)

Target Image Version: 1.11.0 | Base Image Version: 1.10.1

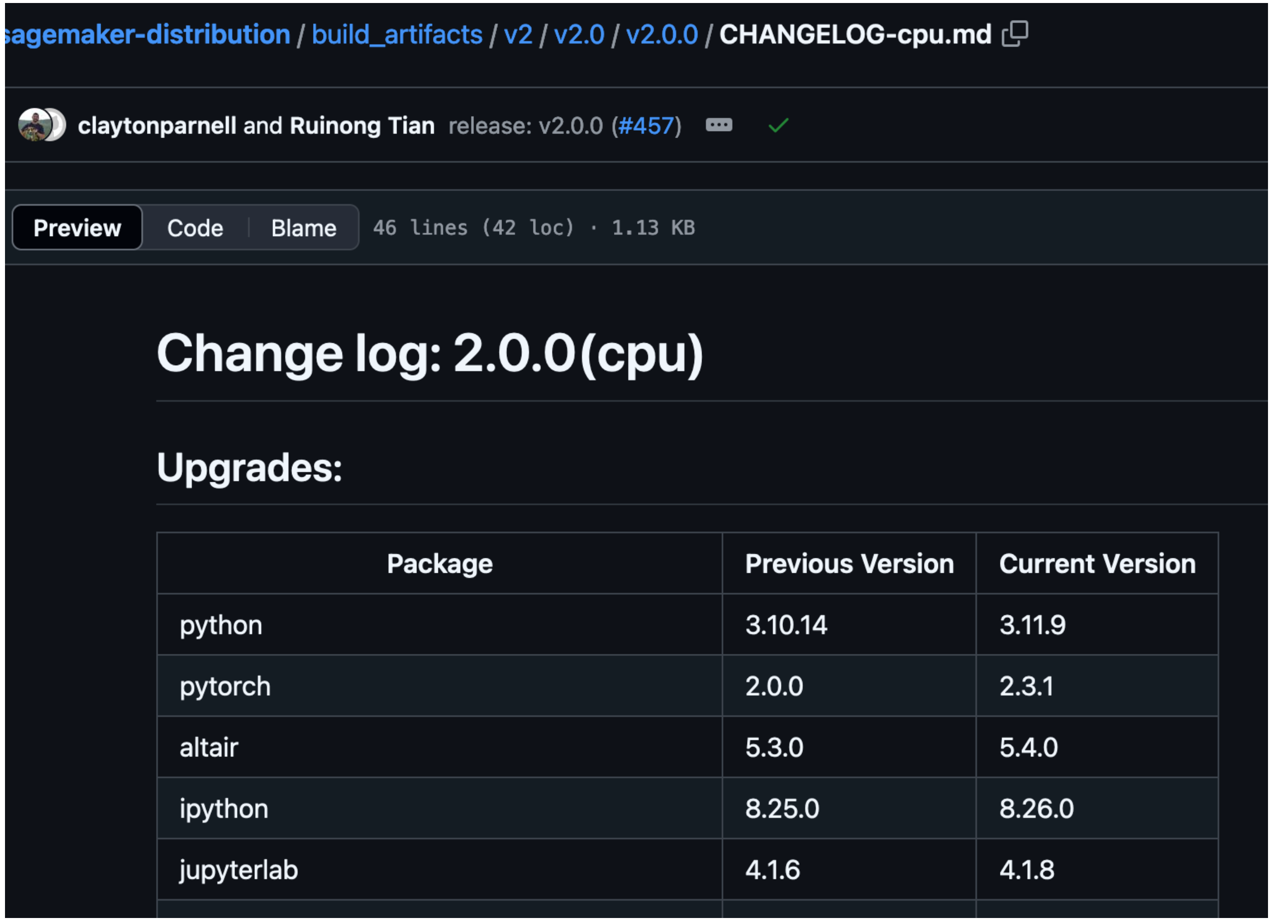
Python Packages Total Size Summary

Target Version Total Size	Base Version Total Size	Size Change (abs)	Size Change (%)
3.47GB	3.47GB	-20.83KB	-0.0

Top-20 Largest Python Packages

Package	Version in the Target Image	Size
cudatoolkit	11.8.0	682.45MB
tensorflow-base	2.14.0	513.71MB
cudnn	8.9.7.29	443.90MB

Automated changelog



sagemaker-distribution / build_artifacts / v2 / v2.0 / v2.0.0 / CHANGELOG-cpu.md

claytonparnell and Ruinong Tian release: v2.0.0 (#457) ✓

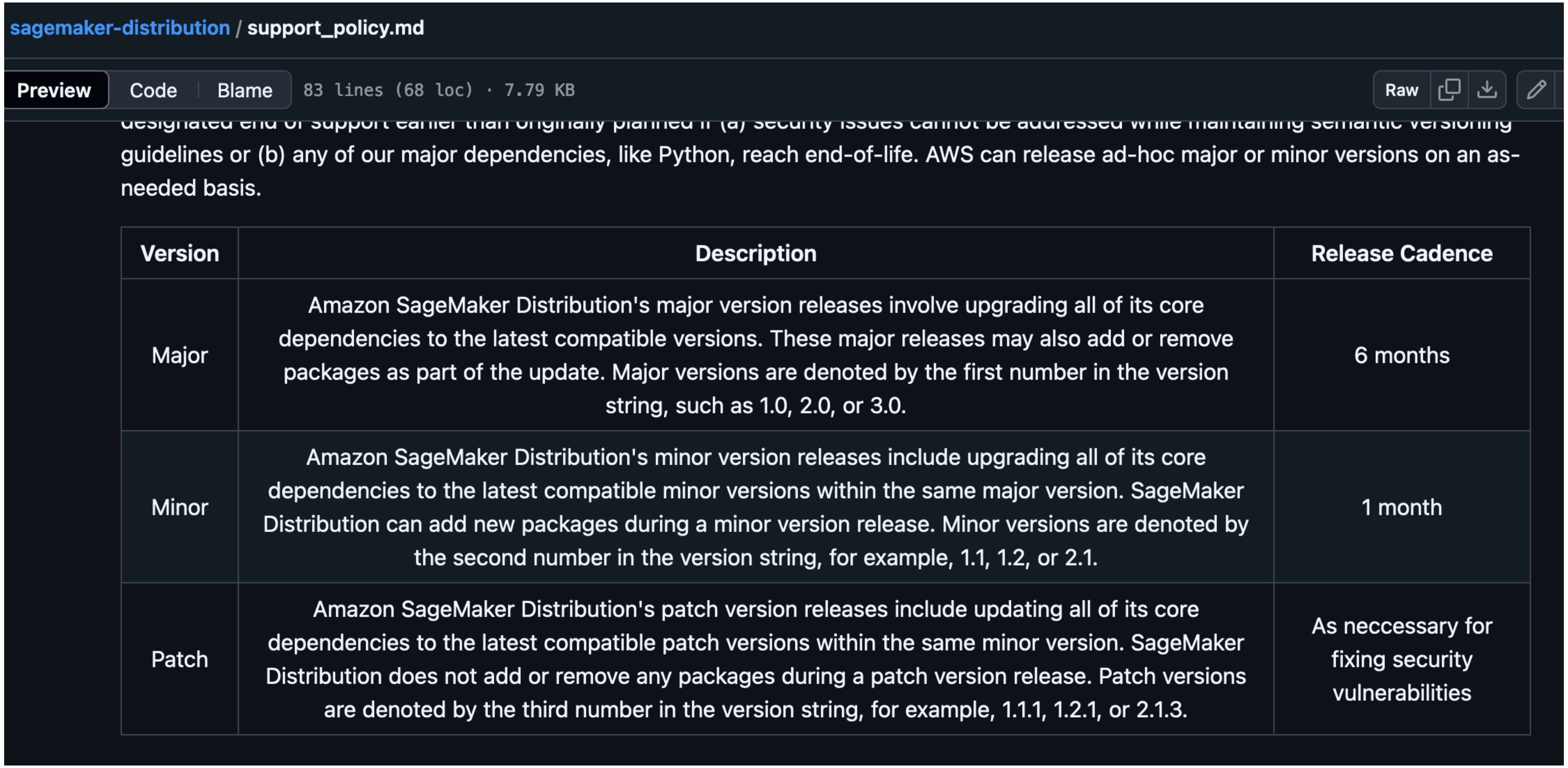
46 lines (42 loc) · 1.13 KB

Change log: 2.0.0(cpu)

Upgrades:

Package	Previous Version	Current Version
python	3.10.14	3.11.9
pytorch	2.0.0	2.3.1
altair	5.3.0	5.4.0
ipython	8.25.0	8.26.0
jupyterlab	4.1.6	4.1.8

Public release cadence and support policy



sagemaker-distribution / support_policy.md

83 lines (68 loc) · 7.79 KB

designated end of support earlier than originally planned if (a) security issues cannot be addressed while maintaining semantic versioning guidelines or (b) any of our major dependencies, like Python, reach end-of-life. AWS can release ad-hoc major or minor versions on an as-needed basis.

Version	Description	Release Cadence
Major	Amazon SageMaker Distribution's major version releases involve upgrading all of its core dependencies to the latest compatible versions. These major releases may also add or remove packages as part of the update. Major versions are denoted by the first number in the version string, such as 1.0, 2.0, or 3.0.	6 months
Minor	Amazon SageMaker Distribution's minor version releases include upgrading all of its core dependencies to the latest compatible minor versions within the same major version. SageMaker Distribution can add new packages during a minor version release. Minor versions are denoted by the second number in the version string, for example, 1.1, 1.2, or 2.1.	1 month
Patch	Amazon SageMaker Distribution's patch version releases include updating all of its core dependencies to the latest compatible patch versions within the same minor version. SageMaker Distribution does not add or remove any packages during a patch version release. Patch versions are denoted by the third number in the version string, for example, 1.1.1, 1.2.1, or 2.1.3.	As neccessary for fixing security vulnerabilities

Security

- Automated scanning.
- Patch releases to fix security vulnerabilities.

Future work

1. Decouple tooling from build artifacts.
2. Semver for OS level packages.
3. Support for AWS accelerators (Inferentia and Trainium) with Neuron SDK.

Resources

- GitHub: https://github.com/aws/sagemaker-distribution
- Container registry: https://gallery.ecr.aws/sagemaker/sagemaker-distribution
- AWS SageMaker: https://aws.amazon.com/sagemaker/

Thanks!

Contact:

- vijayvar@amazon.com
- https://ketanvijayvargiya.com/

Questions?