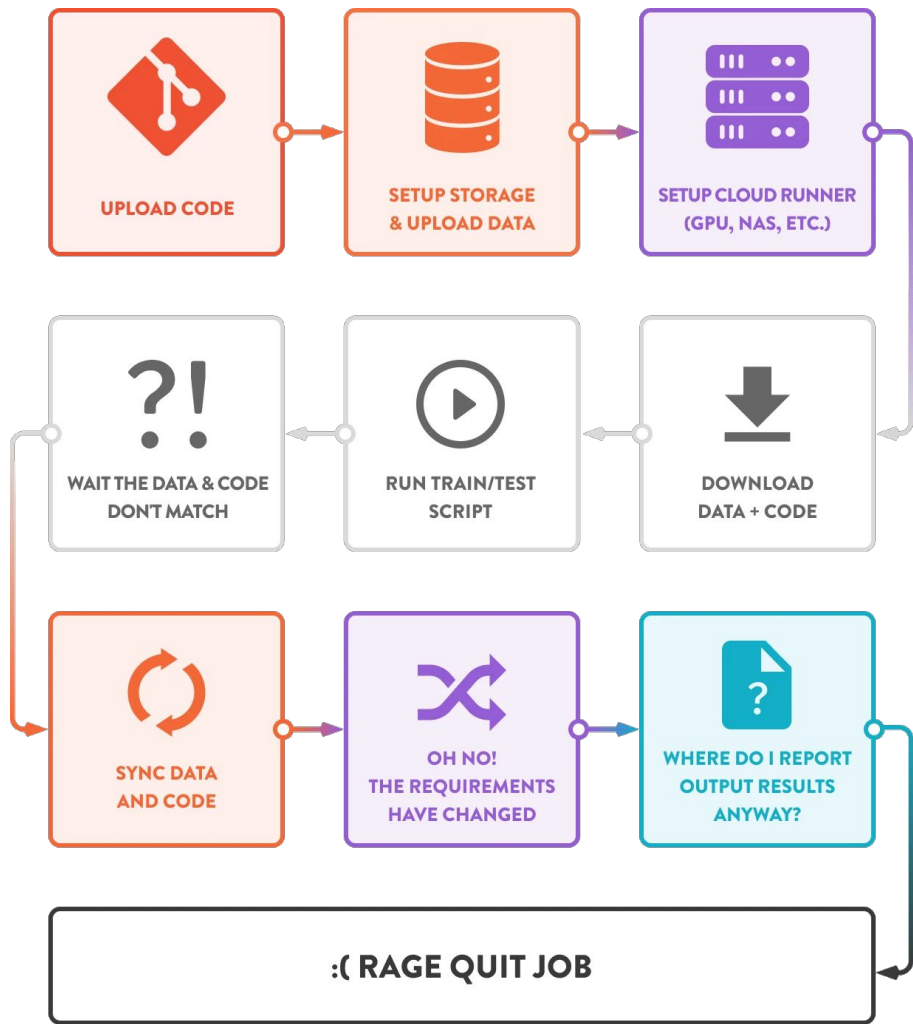# CI/CD for ML

Alex Kim @alex000kim

Introduction

- Common problems with ML workflows
- Key Assumption about ML workflows
- What's CML (Continuous Machine Learning)?
    - Automated Reporting
    - Infrastructure provisioning
- Live Demo
- Summary

Common problems with ML workflows

Key Assumption:

ML workflows
≈
SWE workflows

Best practices in SWE workflows

- Git repo is source of truth
- Experiment/feature branches
- Automated Reporting
- MRs/PRs and Code Reviews

What's CML (Continuous Machine Learning)?

- Open-source command line tool
- Intended to work within CI tools
    - GitHub Actions
    - GitLab CI/CD
    - Bitbucket Cloud
- Enables GitFlow for ML
- Auto-reporting for PRs/MRs
- Integration with Cloud Services

⭐ 3.5k

Create CI workflow

```
name: train-model
on: push
jobs:
  train-model:
    ...
    steps:
      ...
      - name: TrainModel
        ...
        run: |
          pip install -r requirements.txt
          dvc pull
          dvc repro
          dvc push
          # Create CML report
          echo "## Metrics" >> report.md
          dvc metrics show --md >> report.md
          echo "## Feature Importances" >> report.md
          csv2md reports/feat_imp.csv >> report.md
          echo "## Confusion Matrix" >> report.md
          cml publish reports/figures/cm.png --md >> report.md
          cml send-comment report.md
```

1. Install dependencies and run model training

2. Create custom report

3. Submit report as a comment:
cml comment

View reports in GitLab | GitHub | BitBucket

Need more compute power? CML Runner!


```yaml
name: Run Experiment
on: push
jobs:
  deploy-runner:
    runs-on: ubuntu-latest
    ...
      - name: cml runner
        ...
        run: |
          cml runner \
            --cloud=aws \
            --cloud-region=us-east-1 \
            --cloud-type=g4dn.xlarge \
            --labels=cml-runner
  train-model:
    needs: deploy-runner
    runs-on: [self-hosted, cml-runner]
    container:
      image: iterativeai/cml:0-dvc2-base1
      options: --gpus all
    environment: cloud
    ...
      - name: dvc-repro-cml
        run: |
          # Install dependencies
          pip install -r requirements.txt
          dvc pull
          dvc repro
          dvc push
          ### Create CML report ###
          echo "## Metrics" >> report.md
          dvc metrics show --md >> report.md
          ...
          ##########################
          cml send-comment --pr --update report.md
```

1. Provision VM in Cloud:
cml runner

2. Install dependencies and run model training

3. Create custom report

4. Submit report as a comment

Deploy Runner -> Train Model -> Create Report

CML PR: automatically create a MR (PR)

- create & checkout a new branch
- add files
- commit
- git push
- create a PR
- checkout the original branch

```
$ SHA="$(git log -n1 --format=%h)"
$ BASE="$(git branch)"
$ git checkout -b "${BASE}-cml-pr-${SHA}"
$ git add *.py *.json
$ git commit -m "CML PR for ${SHA} [skip ci]"
$ git push
$ gh pr create -B ${BASE} -t "Suggestions"
$ git checkout -
```

```
- run: |
    echo 'lr = 1.0' >> params.py
    python train.py --output metrics.json
    cml pr *.py *.json
```

Team collaboration:

CML reports become part of your code review process!

CML Workflow

Data Scientist — push → Code Repo → GitHub Actions

CML Magic

GitHub Actions → cml.yaml → Cloud Runner → Train/Test → Generate Report → Published as Comment

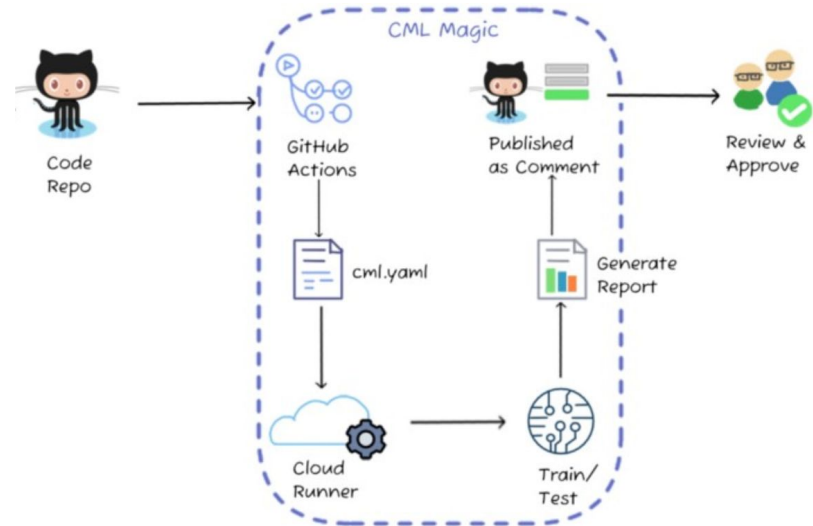Published as Comment → Review & Approve — merge → Deploy Model
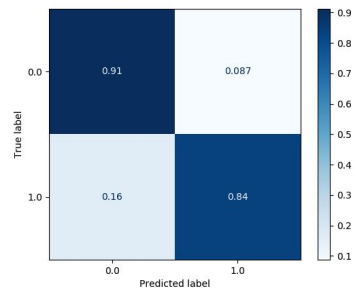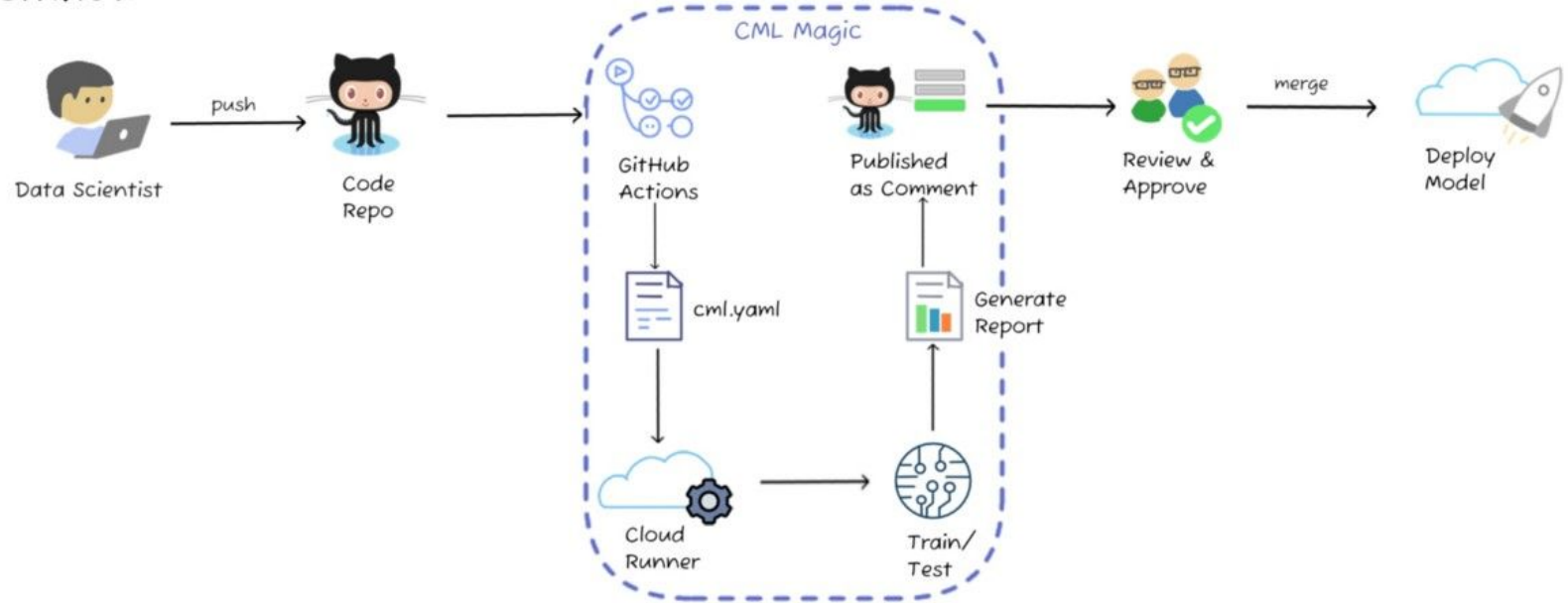
Summary

∞

CML

- ML workflows ≈ SWE workflows
- Model training as part of CI/CD
- Automated Reporting
- Easy resource (de)provisioning
- Improved team collaboration!

CML Magic

Code Repo → GitHub Actions → cml.yaml → Cloud Runner → Train/Test → Generate Report → Published as Comment → Review & Approve

Practice time!