Microsoft

MLOPS with R: An end-to-end process for building machine learning applications

David Smith
Cloud Advocate, Microsoft
@revodavid

github.com/revodavid/mlops-r-gha

@revodavid at #nyrconf

RSTUDIO::CONF 2020 MODELING

MLOps for R with Azure Machine Learning

David Smith | January 31, 2020

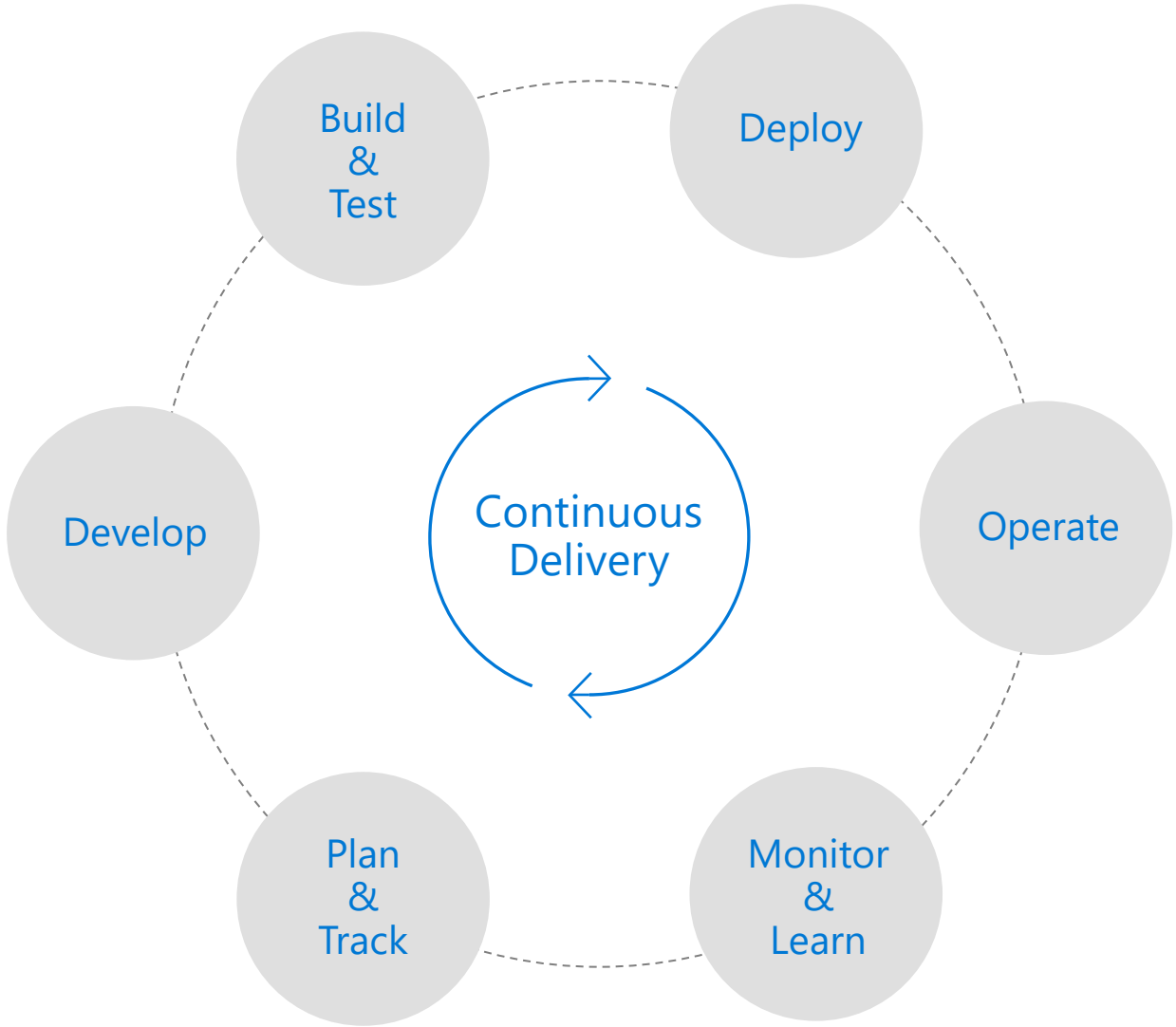
github.com/revodavid/mlops-r

What is DevOps?

People. Process. Products.

> DevOps is the union of **people**, **process**, and **products** to enable continuous delivery of value to your end users.

Donovan Brown, Microsoft
http://bit.ly/WhatIs-DevOps



DEVOPS	MLOPS
Manage code (source files)	Manage code (source files) Manage data files, notebooks, Rmd docs
Manage infrastructure (as code)	Manage infrastructure (as code) Manage environments (as code)
Source code control	Source code control Track experiment outcomes Manage data sets
Build executables Builds take hours (mostly) commodity compute	Train models Model training may take weeks or months GPU compute
Manage build versions	Manage model versions Manage reproducible environments
Tests (deterministic) Fix bugs with code	Tests (probabilistic) Fix bugs with code and/or data Model drift / model retraining

Accident Fatality Probability Estimator

Occupant Age:

16 69 95

16 24 32 40 48 56 64 72 80 88 95

Occupant gender:

f

Occupant role:

pass

Vehicle Year:

1,955 2,002

1,955 1,960 1,965 1,970 1,975 1,980 1,985 1,990 1,995 2,000 2,005

Seatbelt:

belted

Airbag:

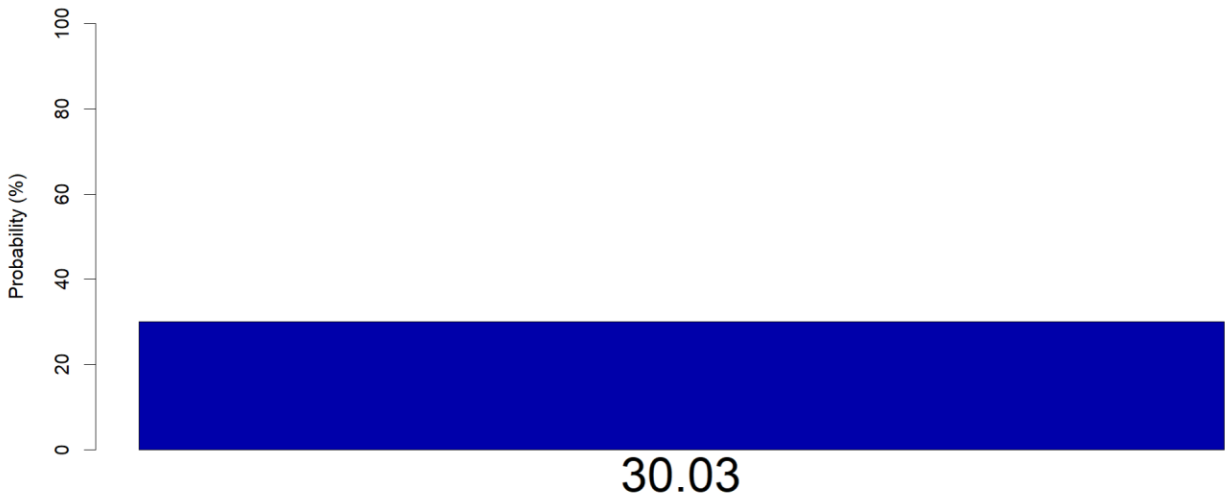
none

Impact speed:

40-54

Collision type:

notfrontal



Azure Machine Learning service

Set of Azure Cloud Services

Python & R SDKs

That enables you to:

- ✓ Prepare Data
- ✓ Build Models
- ✓ Train Models
- ✓ Manage Models
- ✓ Track Experiments
- ✓ Deploy Models

Azure Machine Learning service

Set of Azure Cloud Services + **Python & R SDKs** + **GitHub Actions**

That enables you to:

- ✓ Prepare Data
- ✓ Build Models
- ✓ Train Models

- ✓ Manage Models
- ✓ Track Experiments
- ✓ Deploy Models

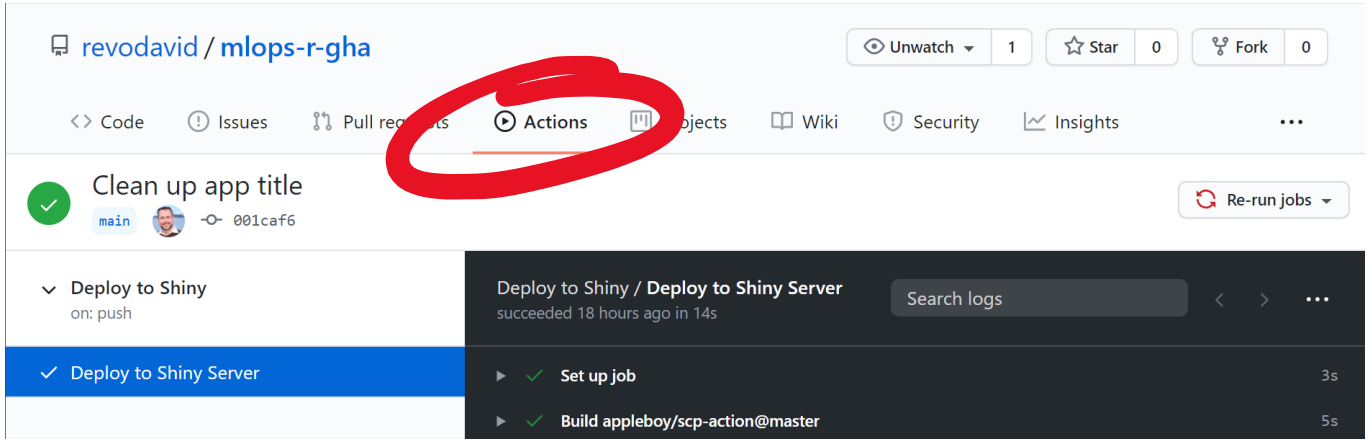
- ✓ Manage Code
- ✓ Collaborate
- ✓ Continuous Integration

GitHub Actions TL;DR

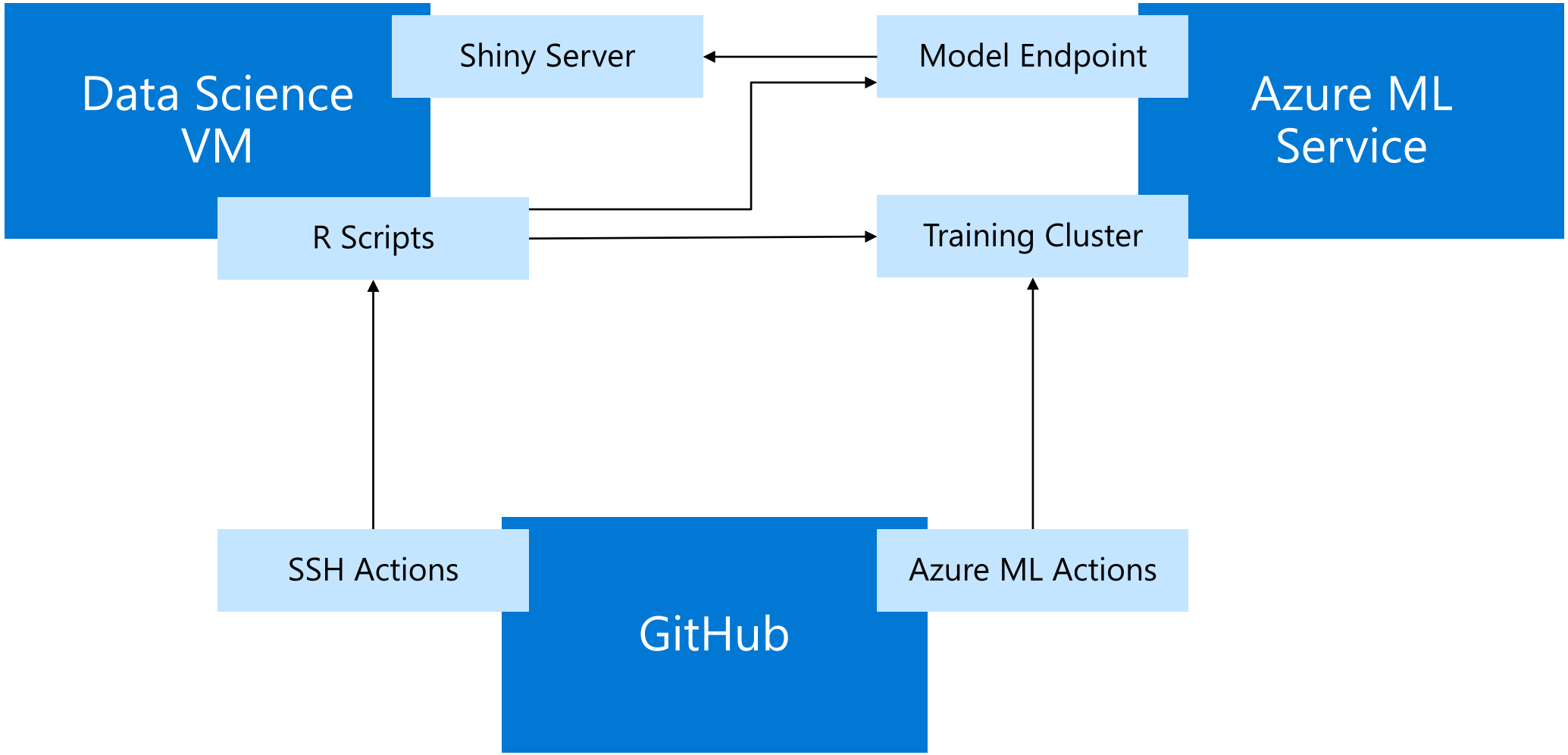
Create YAML files in `.github/workflows` to define jobs

Search GitHub Actions Marketplace for pre-defined templates

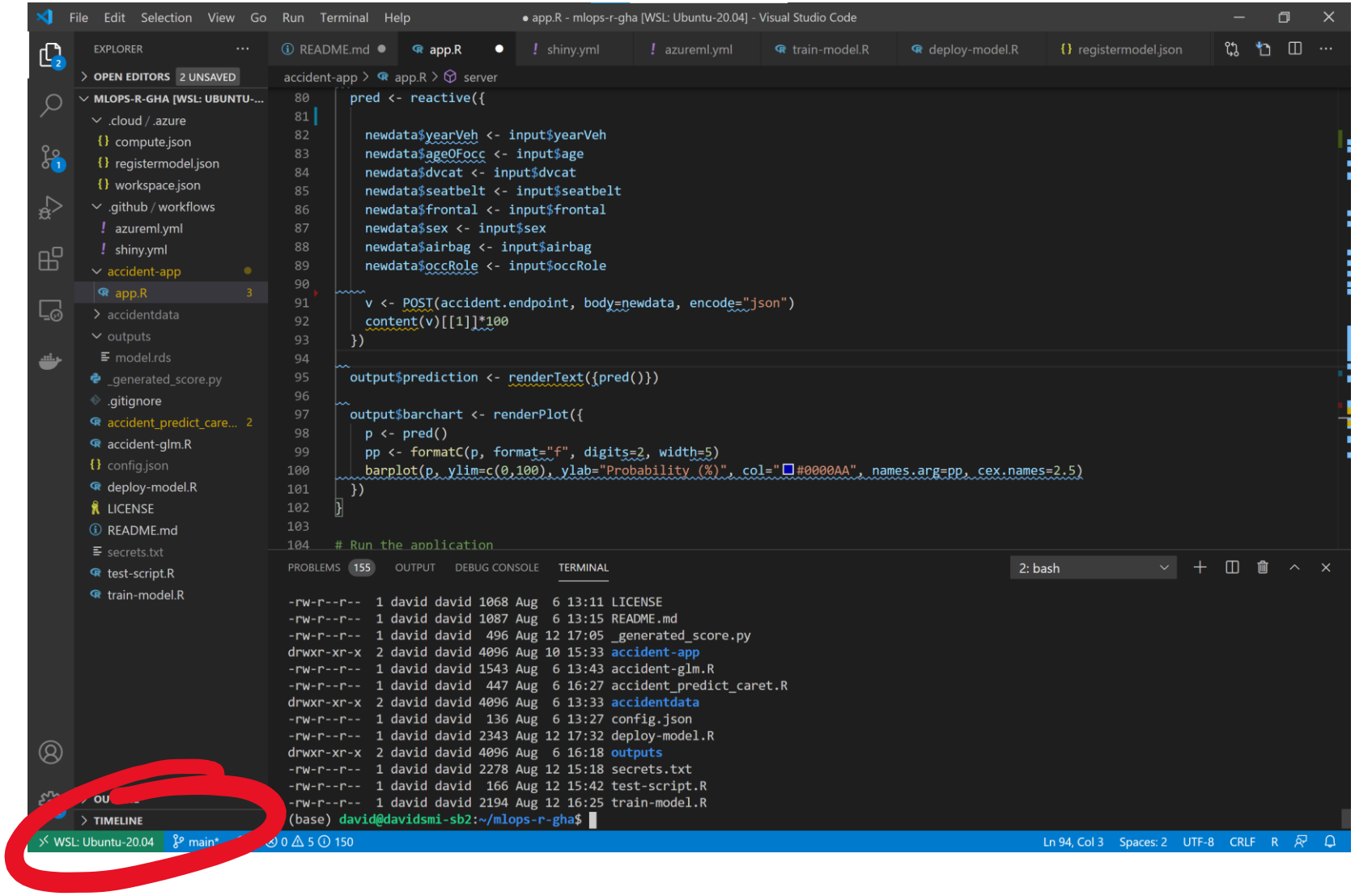
Push, then monitor workflows from the Actions tab



“Accident” Shiny Application Architecture



Dev environment: VS Code + WSL



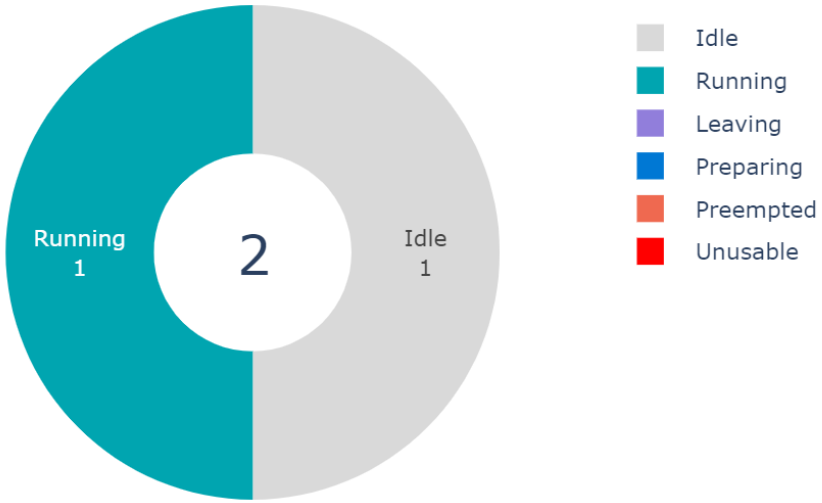
Create 4-node training cluster

azureml.yml

```yaml
# Connect or Create a Compute Target in Azure Machine Learning
- name: Connect/Create Azure Machine Learning Compute Target
  id: aml_compute_training
  uses: Azure/aml-compute@v1
  with:
      azure_credentials: ${{ secrets.AZURE_CREDENTIALS }}
```

compute.json

```json
{
    "name": "rcluster",
    "compute_type": "amlcluster",
    "min_nodes": 0,
    "max_nodes": 4,
    "idle_seconds_before_scaledown": 600
}
```



- Unused nodes de-allocate automatically
- Increase idle timeout when developing

Train model

azureml.yml

```yaml
# Connect to the Shiny VM to train the model
- name: Train model
  uses: JimCronqvist/action-ssh@master
  env:
    AZURE_CREDENTIALS: '${{ secrets.AZURE_CREDENTIALS}}'
  with:
    hosts: ${{ secrets.SHINYUSERNAME }}@${{ secrets.SHINYHOST }}
    privateKey: ${{ secrets.SHINYKEY }}
    command: |
      cd mlops-r-gha
      export AZURE_CREDENTIALS
      Rscript train-model.R
```

train-model.R

```r
est <- estimator(source_directory=".",
                 entry_script = "accident-glm.R",
                 script_params = list("--data_folder" = ds$path(target_pat
                 compute_target = compute_target)
run <- submit_experiment(exp, est)
```

Experiments tracked with source scripts and recorded metrics

Control execution with command line parameters

Most packages pre-loaded, custom packages supported

accident

Edit table | Refresh | Reset view | Add chart

Add filter | Include child runs

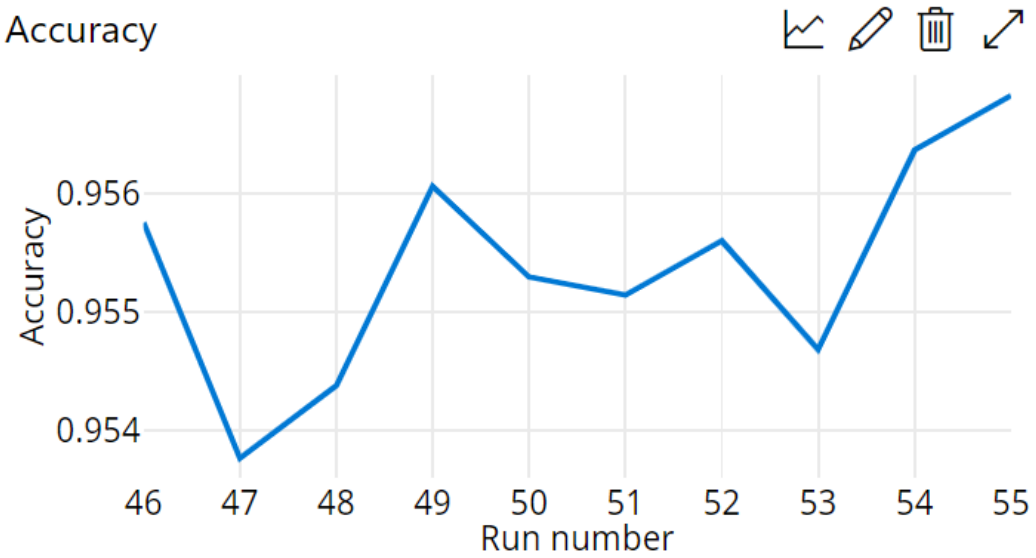
Run status

0 Running

52 Completed

3 Failed

0 Other



Show only selected rows (10 selected)

Run	Run ID	Status	Submitted time	Duration	Submitted by	Compute target
Run 55	accide...	Completed	Aug 13, 2020 3:52 PM	4m 40s	Service principal	rcluster

Deploy Model as REST endpoint

deploy-model.yml

```
inference_config <- inference_config(
  entry_script = "accident_predict_caret.R",
  source_directory = ".",
  environment = r_env)

aci_config <- aci_webservice_deployment_config(cpu_cores = 1, memory_gb = 0.5)

aci_service <- deploy_model(ws,
                        'accidents-gha',
                        list(model),
                        inference_config,
                        aci_config)
wait_for_deployment(aci_service, show_output = TRUE)
```

Name	Description	Created on	Created by	Updated on
accidents-gha	--	August 13, 2020 4:01 PM	d2fbafa0-336d-4ae2-83da-7c7df...	August 13, 2020 4:01 PM

Integrate model into Shiny app

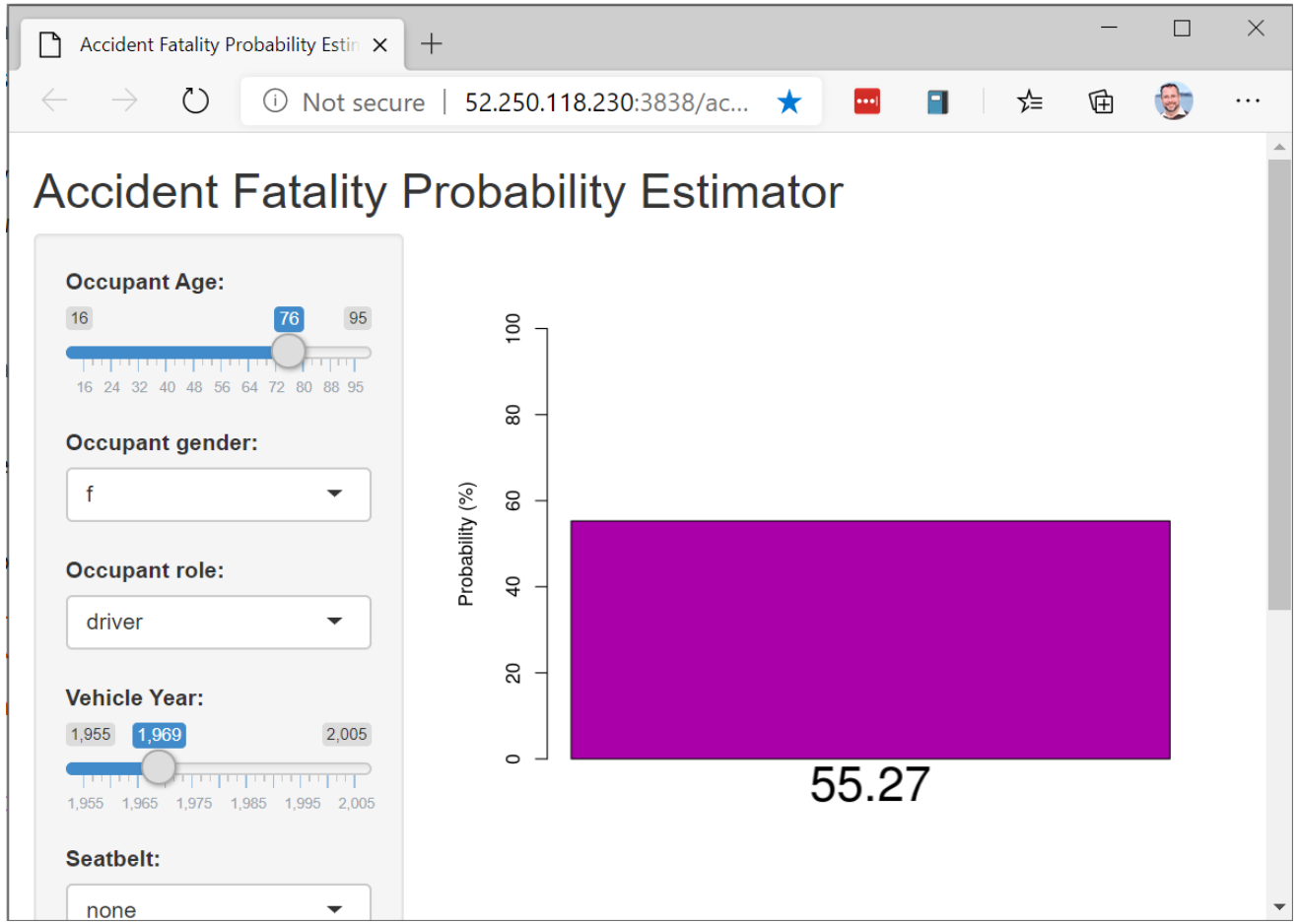
app.R

```
library(httr)
v <- POST(accident.endpoint,
          body=input,
          encode="json")

pred <- content(v)[[1]]
```

shiny.yml

```
- name: copy files via ssh key
  uses: appleboy/scp-action@master
  with:
    host: ${{ secrets.SHINYHOST }}
    username: ${{ secrets.SHINYUSERNAME }}
    key: ${{ secrets.SHINYKEY }}
    source: "accident-app/app.R"
    target: "~"
```



Microsoft

Demo:
Actions in action

Costs per day*

*$US. Cost vary by region and service. Example for illustration.

GitHub Actions	$0.00	
Azure ML service	$0.00	(Studio, logging, orchestration)
Training cluster	$0.10	(as needed)
Scoring endpoint	$1.67	(Azure Container Instances)
Shiny Server	$3.59	(Data Science VM, 4CPU 14Gb)
TOTAL	**$5.36**	

Azure subscription with $200 in free credits: **aka.ms/AML-NYR**

Thank you!

David Smith
Cloud Advocate, Microsoft
@revodavid

Slides and links:

github.com/revodavid/mlops-r-gha