

Developing and Deploying R model

How to use custom R image on AWS Sagemaker Studio

1. Power up local machine

```
#bash
pip install powerup --extra-index-url https://artifactory.gold.mgmt.carezen.net:443/artifactory/api/pypi/care-pypi-local/simple
eval $(powerup --account-alias czen-dataservices-dev yavapai-poc)

#Potential Errors
#1. Don't have full access to certain file during installation. Use 'chmod u+w <<File Name>>' to grant access
#2. Multiple powerups installed and the system is confused about which one to use. Run 'which powerup' to make sure the system is using the right one
```

2. Install AWS CLI tool if not already

```
#bash
brew install awscli
#would need to install brew first if using this installation method
```

3. Create remote repository on AWS ECR (Elastic Container Registry)

```
#bash
#aws ecr create-repository --repository-name <<repo name>> --region us-east-1
aws ecr create-repository --repository-name testrdeployment --region us-east-1
```

4. Create the docker image, tag the image and upload the image

Sample R Dockerfile with Python kernel so this can be directly used in Sagemaker Studio

<https://bitbucket.use.dom.carezen.net/projects/AN/repos/r-model-development-deployment/browse/Dockerfile%20w.%20Python%20Kernel>

```
#bash
#docker build -t <<image name>> .
#Create the docker image
docker build -t testrdeployment .

#tag the docker image
#docker tag <<image name>> <<AWS account number>>.dkr.ecr.us-east-1.amazonaws.com/<<image name>>:<<image version>>
docker tag testrdeployment 087946768277.dkr.ecr.us-east-1.amazonaws.com/testrdeployment:1

#Log onto ECR
#aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <<AWS account number>>.dkr.ecr.us-east-1.amazonaws.com/<<repo name>>
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 087946768277.dkr.ecr.us-east-1.amazonaws.com/testrdeployment

#Upload the image to ECR
#docker push <<AWS account number>>.dkr.ecr.us-east-1.amazonaws.com/<<image name>>:<<image version>>
docker push 087946768277.dkr.ecr.us-east-1.amazonaws.com/testrdeployment:1
```

5. Create the image and corresponding image version on ECR

```
#bash
#Create Sagemaker Image:
aws sagemaker create-image \
  --image-name testdeployment \
  --role-arn arn:aws:iam::087946768277:role/yavapai-sagemaker-fullaccess \
  --region us-east-1

#Create Image Version:
aws sagemaker create-image-version \
  --image-name testdeployment \
  --base-image 087946768277.dkr.ecr.us-east-1.amazonaws.com/testdeployment:1 \
  --region us-east-1
```

This step can be accomplished via AWS UI.

<https://console.aws.amazon.com/sagemaker/home?region=us-east-1#/images>

Amazon SageMaker X

Amazon SageMaker Studio

Dashboard

Search

Images

▼ Ground Truth

Labeling jobs

Labeling datasets

Labeling workforces

► Notebook

Amazon SageMaker > Images

Custom images [How to create SageMaker images](#)

	Name	Display name	Created	Status	Actions
<input checked="" type="radio"/>	feature-engineering-ml-dev	feature-engineering-ml-dev	Feb 04, 2021 13:09 UTC	✔ CREATED	Create version
<input type="radio"/>	customrsagemaker	-	Jan 28, 2021 03:39 UTC	✔ CREATED	Create version
<input type="radio"/>	customimagetest	-	Jan 28, 2021 01:55 UTC	✔ CREATED	Create version
<input type="radio"/>	seekerfraud	-	Jan 27, 2021 23:22 UTC	✔ CREATED	Create version

6. Attach the image to AWS SageMaker Studio Instance

<https://console.aws.amazon.com/sagemaker/home?region=us-east-1#/studio/d-40qpe4qkl76s>

Follow the instructions and enter the kernel name and kernel display name (in our case, it's Python 3)

Custom images attached to domain [How to attach SageMaker images](#)

You can attach up to 30 images. Only the latest version for each image is displayed below. If you try to attach more than 30 images, the request will fail. To attach additional images, you need to detach an image and all its versions and then try again.

<input type="checkbox"/>	Name	DisplayName	Latest attached version	Created	Actions
<input type="checkbox"/>	customimagetest	-	1	Jan 28, 2021 01:55 UTC	Attach version
<input type="checkbox"/>	customrsagemaker	-	1	Jan 28, 2021 03:39 UTC	Attach version
<input type="checkbox"/>	feature-engineering-ml-dev	feature-engineering-ml-dev	1	Feb 04, 2021 13:09 UTC	Attach version
<input type="checkbox"/>	seekerfraud	-	1	Jan 27, 2021 23:24 UTC	Attach version

image store.

☐ New image
Attaches 1 version.

☒ Existing image
Attaches 1 version.

Select an existing image from the SageMaker Image store

An attempt to add more versions than are remaining under the version limit will fail. In that case, detach some versions and try again.

	Name	DisplayName	Created
<input type="radio"/>	feature-engineering-ml-dev	feature-engineering-ml-dev	Feb 04, 2021 13:09 UTC
<input checked="" type="radio"/>	customrsagemaker	-	Jan 28, 2021 03:39 UTC
<input type="radio"/>	customimagetest	-	Jan 28, 2021 01:55 UTC
<input type="radio"/>	seekerfraud	-	Jan 27, 2021 23:22 UTC

Available image versions

The selected image version will be attached to the domain.

	Name	Version	Created
<input checked="" type="radio"/>	customrsagemaker	3	Jan 31, 2021 19:42 UTC

Cancel

Next

Attach image to domain

Studio configuration

EFS mount path

The path in the image where the Studio user's EFS home directory will be mounted. [Learn more about EFS mount paths](#)

/home/sagemaker-user

Kernel

The kernel included in the image. We recommend one kernel per image. [Learn more about kernels](#)

Kernel name

Python 3

1024 character max

Kernel display name - optional

Python 3

1024 character max

Delete

Add kernel

Configuration tags - optional

Add new tag

You can add up to 50 tags.

► Advanced configuration - optional

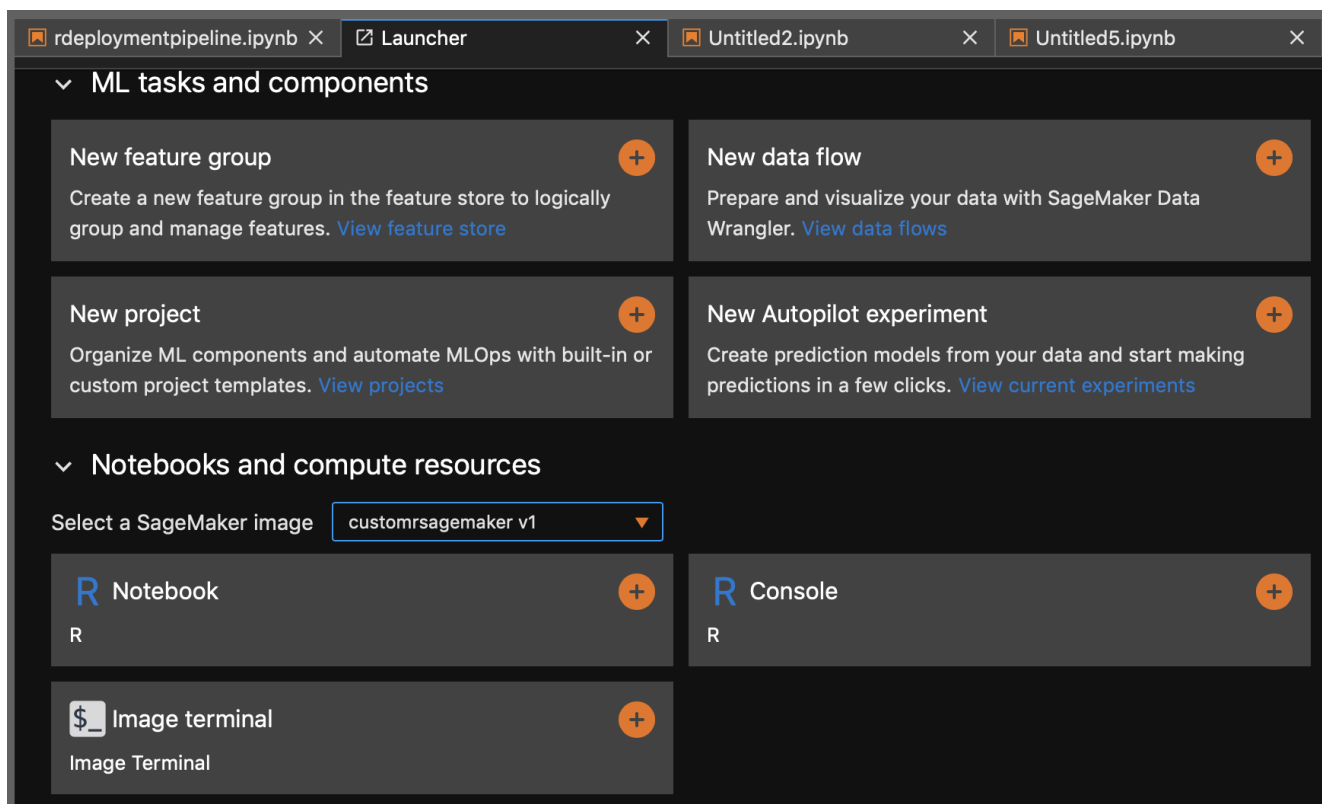
Recommended defaults have been selected for you.

Cancel

Back

Submit

7. Start the AWS SageMaker Studio instance and launch new notebook/terminal using the customized image (it might take some time for the new image to show up)



Model Training and Inferencing within AWS SageMaker

1. Create the training and inferencing image and

Sample repo:

<https://bitbucket.use.dom.carezen.net/projects/AN/repos/r-model-development-deployment/browse>

Sample Dockerfile:

<https://bitbucket.use.dom.carezen.net/projects/AN/repos/r-model-development-deployment/browse/Dockerfile>

Ubuntu 20.04 is used as the base image in the Dockerfiles. **R 4.0.3** is installed in the image.

2. Upload the image to ECR using the method above

3. Construct the data pipeline within SageMaker Studio and use the image above to train the model and perform the inference/prediction

Sample Jupyter Notebook: <https://bitbucket.use.dom.carezen.net/projects/AN/repos/r-model-development-deployment/browse/rdeploymentpipeline.ipynb>

