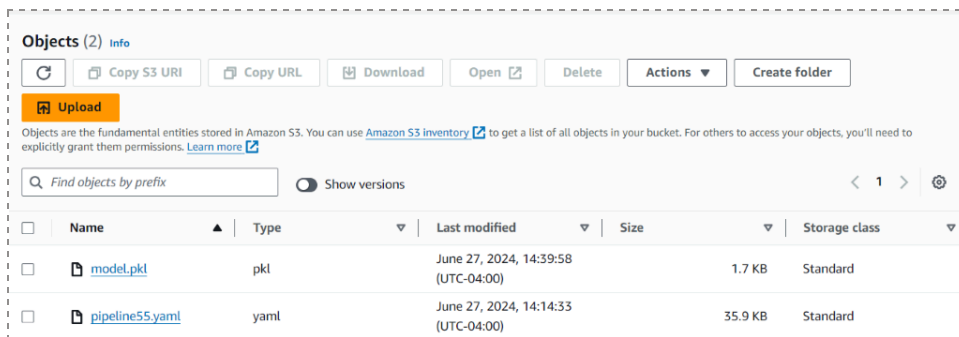


Kubernetes and Kubeflow - Create Inference Service

Go to AWS and select s3 buckets

1. Place model in s3 bucket/create s3 bucket if not created



2. Create or add IAM policies (aws-s3-access-programatic)
 - Go to Policies in IAM roles and go to Policy editor



3. Create IAM role in aws and connect to policies made (aws-s3-access)
 - Update the trust relationship to include the name of the policy
4. Create a Kubernetes service account on the cluster
 - Create a .yaml file named: service-account.yaml
 - The configuration should look like this:

yaml

Copy code

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: example-service-account
  namespace: default
```

Apply using:

bash

Copy code

```
kubectl apply -f service-account.yaml
```

5. Connect service account to IAM
 - `kubectl annotate serviceaccount ew-intern-2024 eks.amazonaws.com/role-arn=arn:aws:iam::YOUR_ACCOUNT_ID:role/YOUR_IAM_ROLE_NAME -n your-namespace`
6. Open Kubeflow and create service

```
from kubernetes import client
from kserve import KServeClient
from kserve import constants
from kserve import utils
from kserve import V1beta1InferenceService
from kserve import V1beta1InferenceServiceSpec
from kserve import V1beta1PredictorSpec
from kserve import V1beta1SKLearnSpec

name = "logreg"
namespace = utils.get_default_target_namespace()
kservice_version = 'v1beta1'
api_version = constants.KSERVE_GROUP + '/' + kserve_version
isvc = V1beta1InferenceService(
    api_version=api_version,
    kind=constants.KSERVE_KIND,
    metadata=client.V1ObjectMeta(
        name=name,
        namespace=namespace,
        annotations={'sidecar.istio.io/inject': 'false'}
    ),
    spec=V1beta1InferenceServiceSpec(
        predictor=V1beta1PredictorSpec(
            service_account_name="s3-service-account",
            sklearn=V1beta1SKLearnSpec(
                storage_uri="s3://mlpipelineews/model.pkl",
            )
        )
    )
)

KServe = KServeClient()
KServe.create(isvc)
```

Delete the inference service:

```
kServe.delete(name, namespace=namespace)
```