Represent

Serverless ML Deployment on AWS

Represent

Šimon Soták

- Studied Computer Graphics at Charles University in Prague
- Started out in Game Development (Keen SWH, Warhorse)
- Joined Represent in 2014

Represent

Andrej Hoos

- Studied Computer Science and Physics at University of Glasgow
- Joined Represent in 2014

Represent

Represent

Arnold Schwarzenegger

$1M

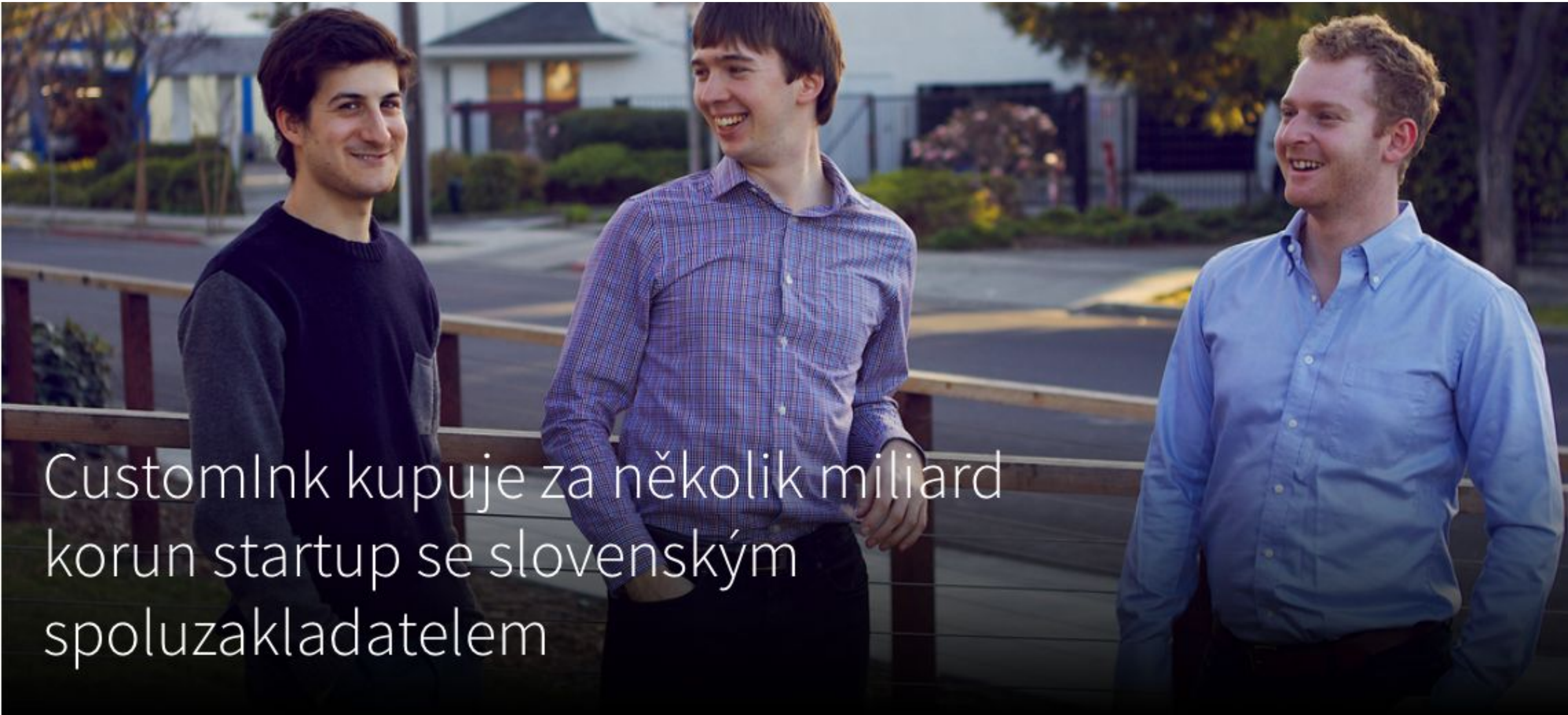
revenue

69K

products sold

Represent

CZECHCRUNCH



CustomInk kupuje za několik miliard korun startup se slovenským spoluzakladatelem

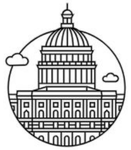
Represent

PRG

Prague, Czech Republic
Šaldova 12, Karlín

LA

Los Angeles, California
1680 Vine St, Hollywood



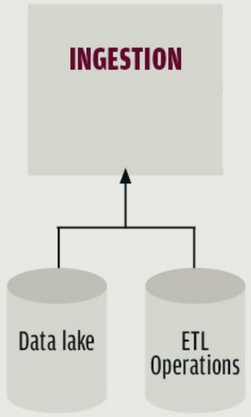
DC

Washington metropolitan area
2910 District Ave, Fairfax

Represent

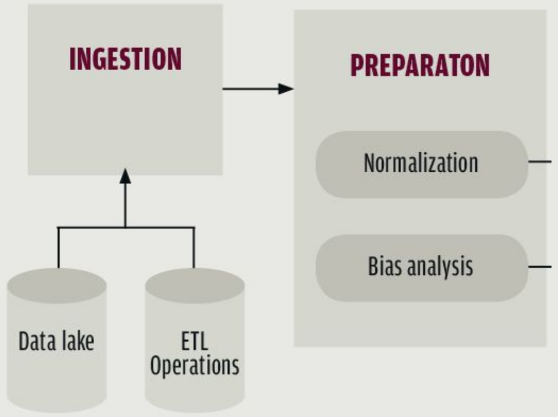
Machine Learning

Machine Learning Process



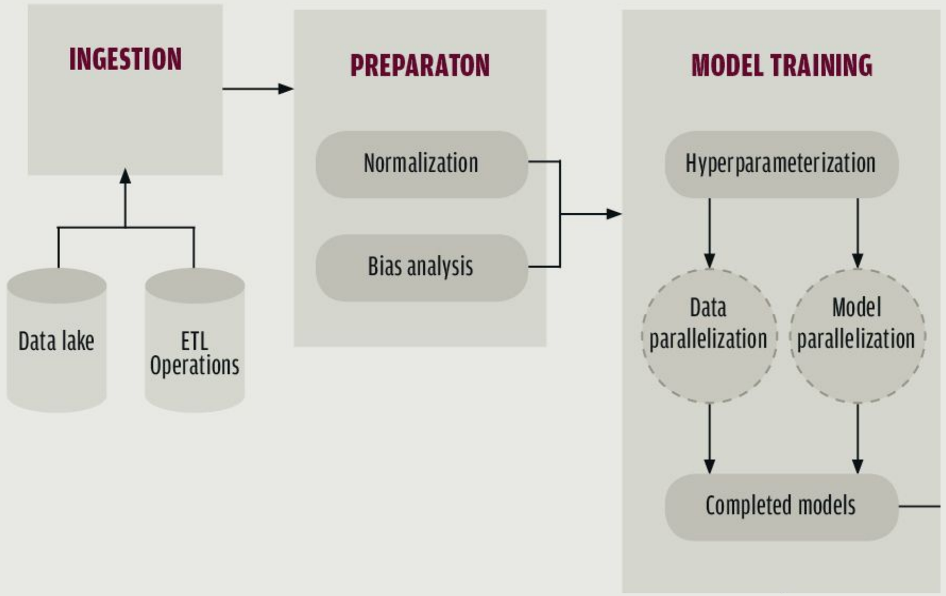
Represent

Machine Learning Process



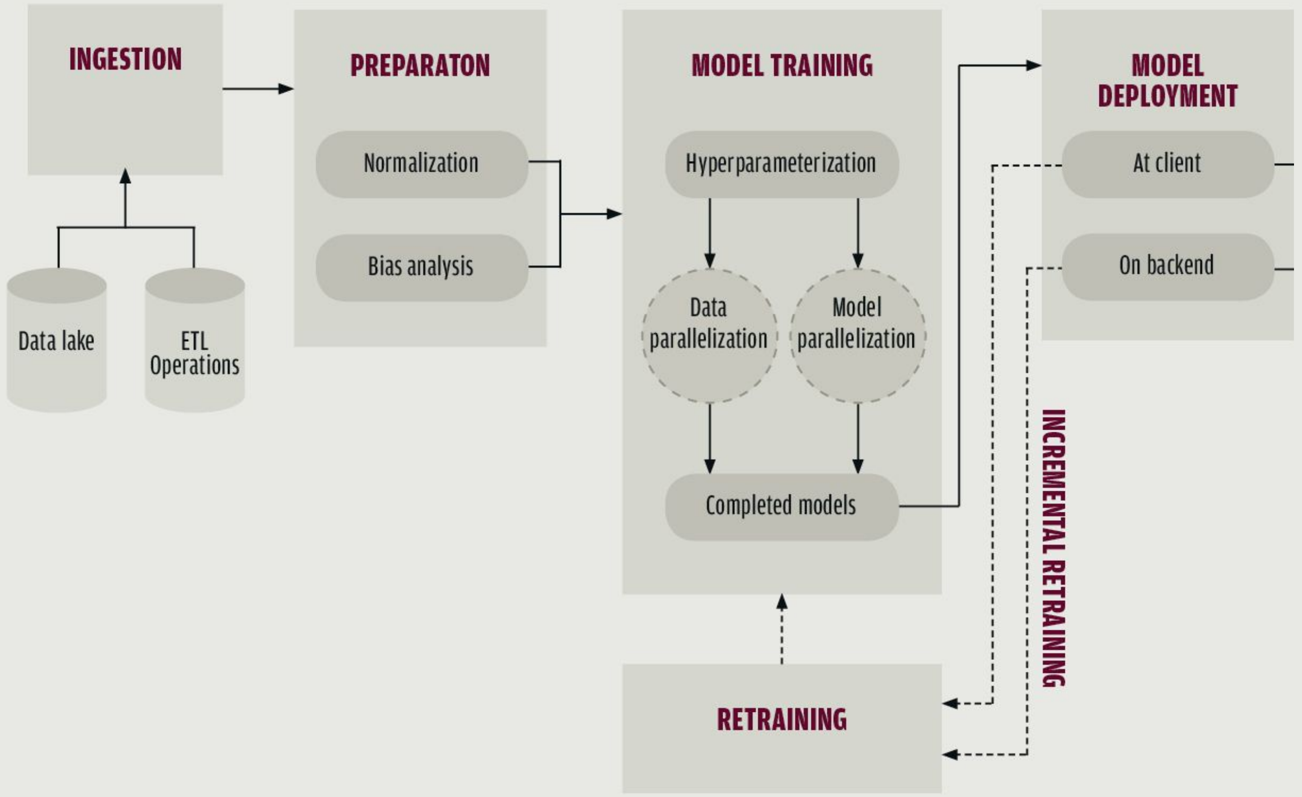
Represent

Machine Learning Process



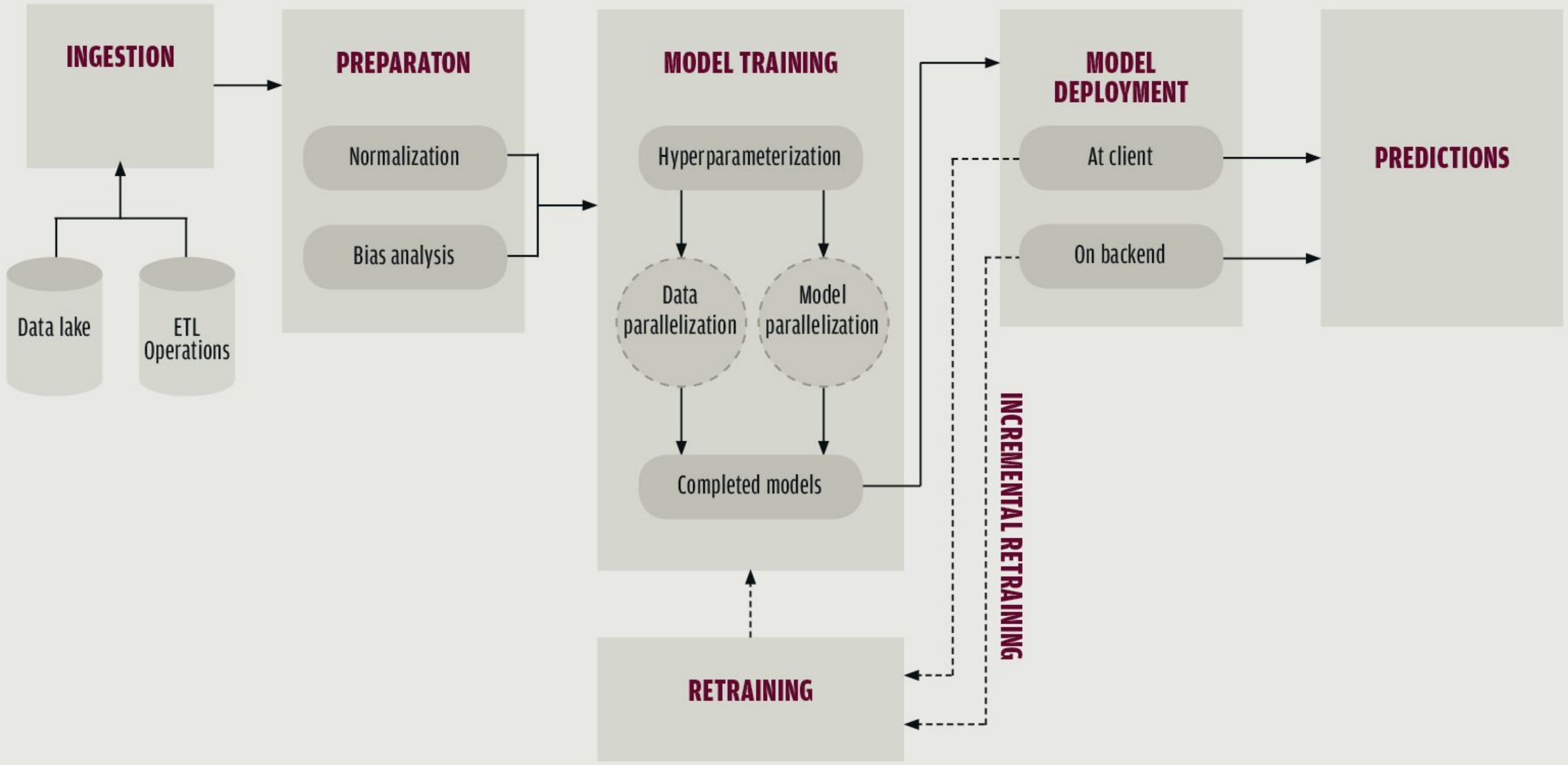
Represent

Machine Learning Process



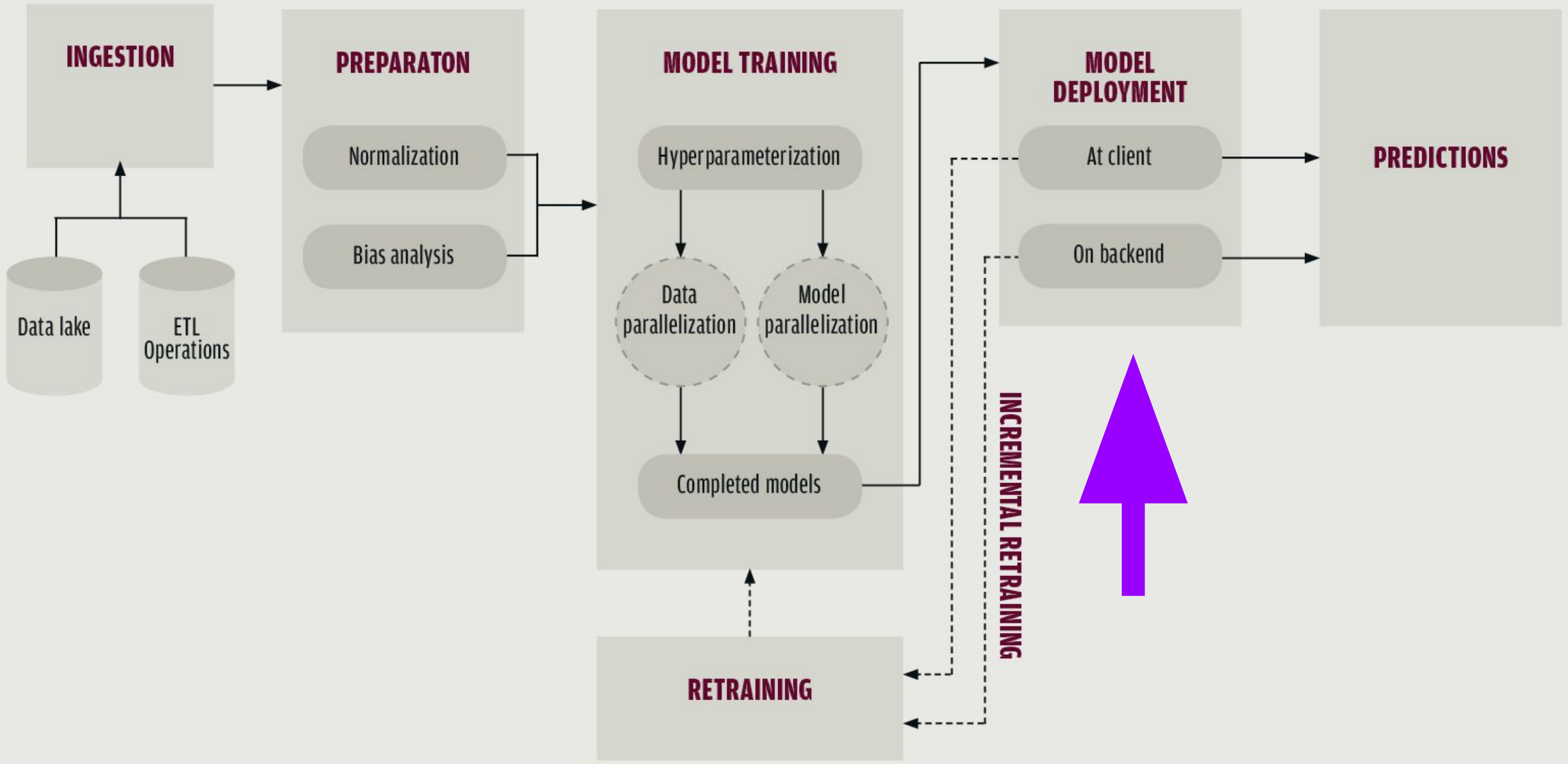
Represent

Machine Learning Process



Represent

This Talk



Represent

Serverless ML Deployment on AWS

Represent

How to deploy ML models

Approach	**AWS Service**
Hosted servers	EC2
Docker / Kubernetes containers	ECS / EKS
Serverless	**Lambda** / Fargate
ML-tailored solutions	Sagemaker

Represent

What is serverless?

- Function as a Service
- Fully managed
- You provide only the code to be run
- Advantages
  - No maintenance required
  - Scaling “for free”
  - Easier development
  - Lower infrastructure cost

Represent

Python ML Deployment on Serverless

- AWS Lambda
- Why Amazon?
- Why Python?

Represent

So, can I host my Python ML model on AWS Lambda?

Represent

Yes*…

Represent

Cold Start

- First request – “bootup” & init
- Instance “shuts down” after a while if no requests arrive
- Parallel cold starts
- Pre-warming
- ML Model loading

Represent

Issue to overcome: Package size

- AWS Lambda – 250 MB max package size
- Scikit-learn – 200+ MB
- Tensorflow – over 250 MB

Represent

Did you say “Easier development” earlier?

Represent

Introducing: `lambdipy`

- Tool for building python packages for AWS Lambda
- github.com/customink/lambdipy
- Builds “minified” packages for AWS Lambda
- Contains pre-built popular packages
- Active development – contributions welcome

Represent

What does `lambdipy` do

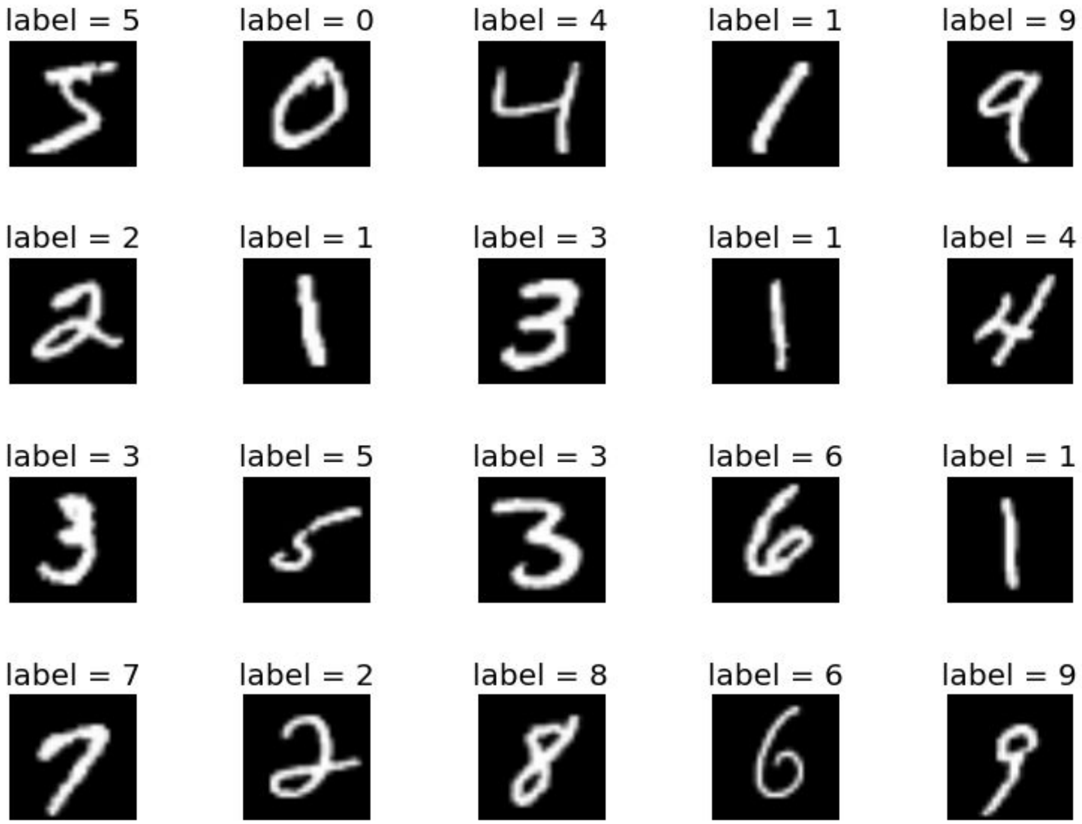
- Builds any python ML package
  - Uses Docker with Lambda-like environment
  - Downloads pre-built popular packages
- **`strip`** binaries
  - Binaries take the most space in ML packages
  - This reduces their file size significantly
- Remove some unneeded files
  - Tests, caches, package metadata
- scipy: 140 MB ⟶ 33 MB

Represent

Example

Represent

MNIST dataset



Represent

Simple image classifier

- Handwritten digit classification
- Inspired by blog post by Bikramjot Singh Hanzra[1]
- **`sam`** for deployment
- **`pipenv`** for dependency management

1. http://hanzratech.in/2015/02/24/handwritten-digit-recognition-using-opencv-sklearn-and-python.html

Example: `train.py`

```
dataset = datasets.fetch_mldata("MNIST Original")
```

Example: `train.py`

```
dataset = datasets.fetch_mldata("MNIST Original")

# Train a ML model
model = LinearClassifer()
model.fit(features, labels)
```

Represent

Example: `train.py`

```
dataset = datasets.fetch_mldata("MNIST Original")

# Train a ML model
model = LinearClassifer()
model.fit(features, labels)

joblib.dump(model, "model.pkl")
```

Example: `predict.py`

```
model = joblib.load("model.pkl")
```

Represent

Example: `predict.py`

```python
model = joblib.load("model.pkl")

def predict(context, _):
```

Represent

Example: `predict.py`

```python
model = joblib.load("model.pkl")

def predict(context, _):
    request_body = json.loads(context["body"])
```

Represent

Example: `predict.py`

```
model = joblib.load("model.pkl")

def predict(context, _):
    request_body = json.loads(context["body"])

    # Load image from url and extract its features
    prediction = model.predict(image_features)
```

Represent

Example: `predict.py`

```python
model = joblib.load("model.pkl")

def predict(context, _):
    request_body = json.loads(context["body"])

    # Load image from url and extract its features
    prediction = model.predict(image_features)

    return {
        "statusCode": 200,
        "body": json.dumps({"digit": int(prediction[0])})
    }
```

Represent

Example: template.yml

```
…

Resources:
  PredictFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: ./build
      Handler: predict.predict
…
```

Represent

Example: deployment

- Install dependencies
  - `pipenv install scikit-learn scikit-image numpy`
  - `pipenv install` **`lambdipy`** `aws-sam-cli --dev`
- Train the classifier
  - `pipenv run python train.py`
- Build the bundle
  - `pipenv run lambdipy build --from-pipenv -i predict.py -i model.pkl`
- Deploy
  - `pipenv run sam package --s3-bucket ...`
  - `pipenv run sam deploy --stack-name ...`

Represent

🤞🤞🤞 Live Demo 🤞🤞🤞

Represent

Example: testing the predictions

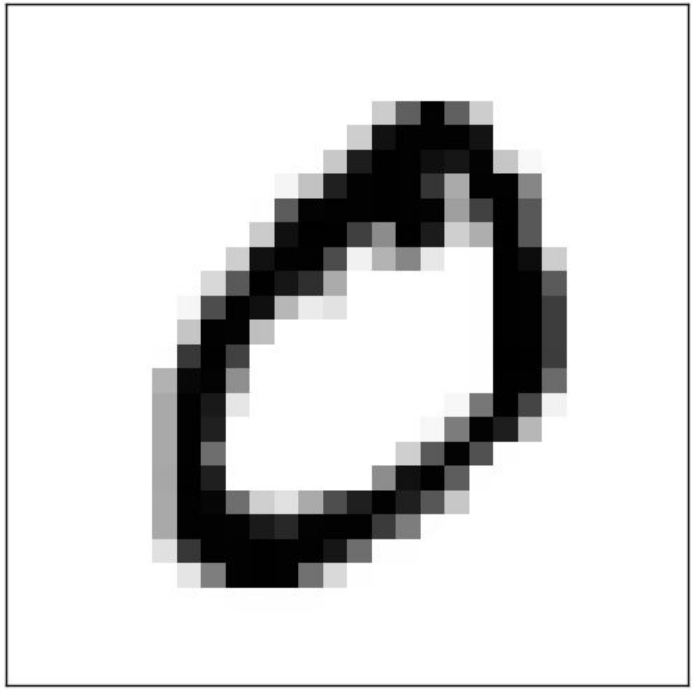
POST

https://r7my4vk3t0.execute-api.us-east-1.amazonaws.com/Prod/predict

```
{
  "url": "https://i.stack.imgur.com/FK0FB.png "
}
```

```
{ "digit": 0 }
```

```
curl -d '{"url": "https://i.stack.imgur.com/FK0FB.png"}' -X POST
https://r7my4vk3t0.execute-api.us-east-1.amazonaws.com/Prod/predict
```



Example: testing the predictions

POST

https://r7my4vk3t0.execute-api.us-east-1.amazonaws.com/Prod/predict

```
{
  "url": "https://goo.gl/K9ditV"
}
```

```
{ "digit": 2 }
```

```
curl -d '{"url": "https://goo.gl/K9ditV"}' -X POST
https://r7my4vk3t0.execute-api.us-east-1.amazonaws.com/Prod/predict
```

🎉 □ 📈

Production ML system complete!

Alternative approaches

- Zappa
    - Deploys Python web apps to serverless environments
- Google Cloud Functions
    - Python support currently in Beta

Thank you!

This presentation: tiny.cc/aws-lambdipy

We are hiring and always looking to have good conversations

represent.com/join

Represent