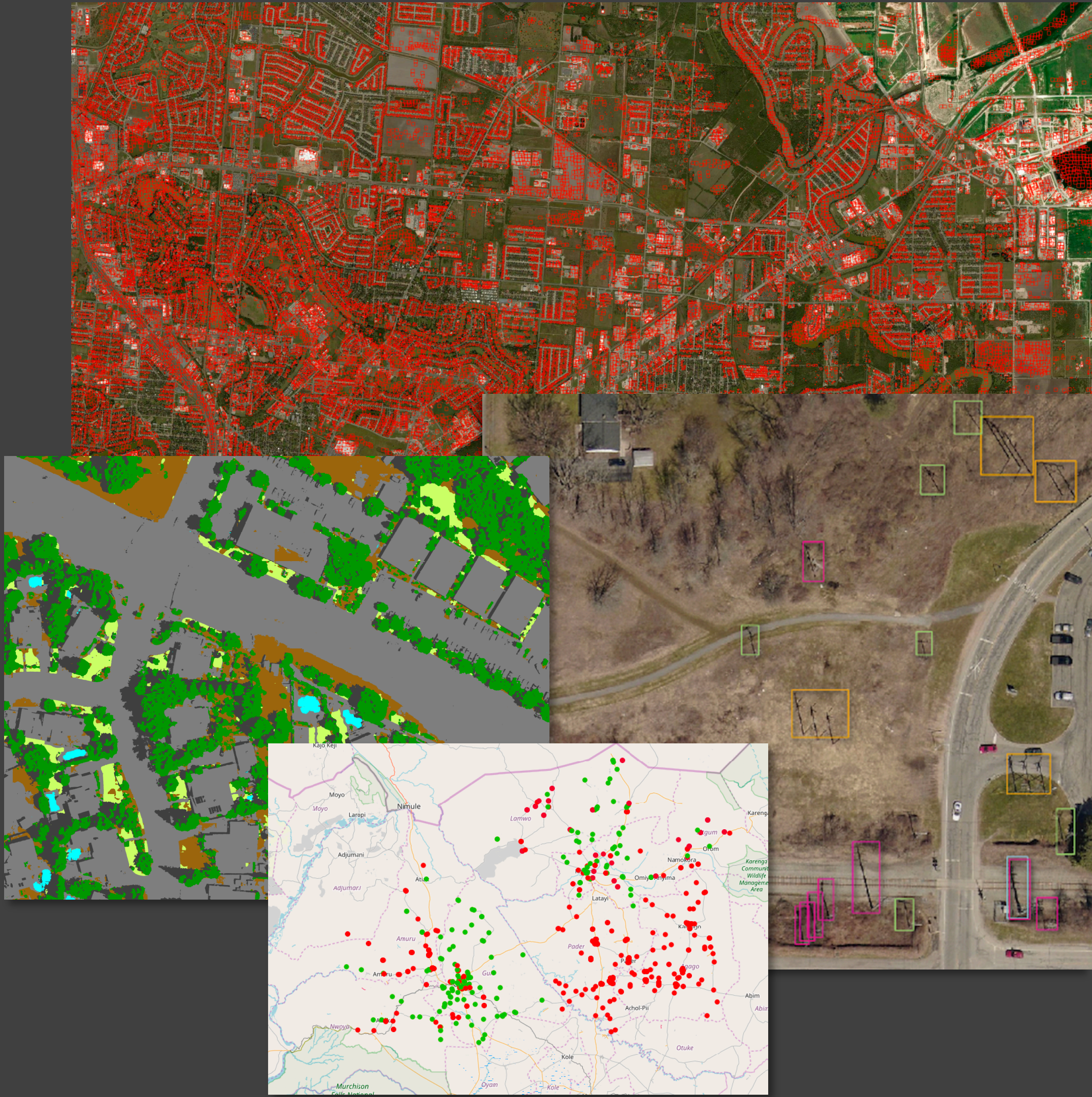
GEOML & MODEL DEMOCRATIZATION WITH OSM DATA

Shay Strong, PhD
Director of Data Science & ML (Eagleview)
Astrophysics, UT Austin
Affiliate, UW

@shaybstrong
shay.strong@eagleview.com

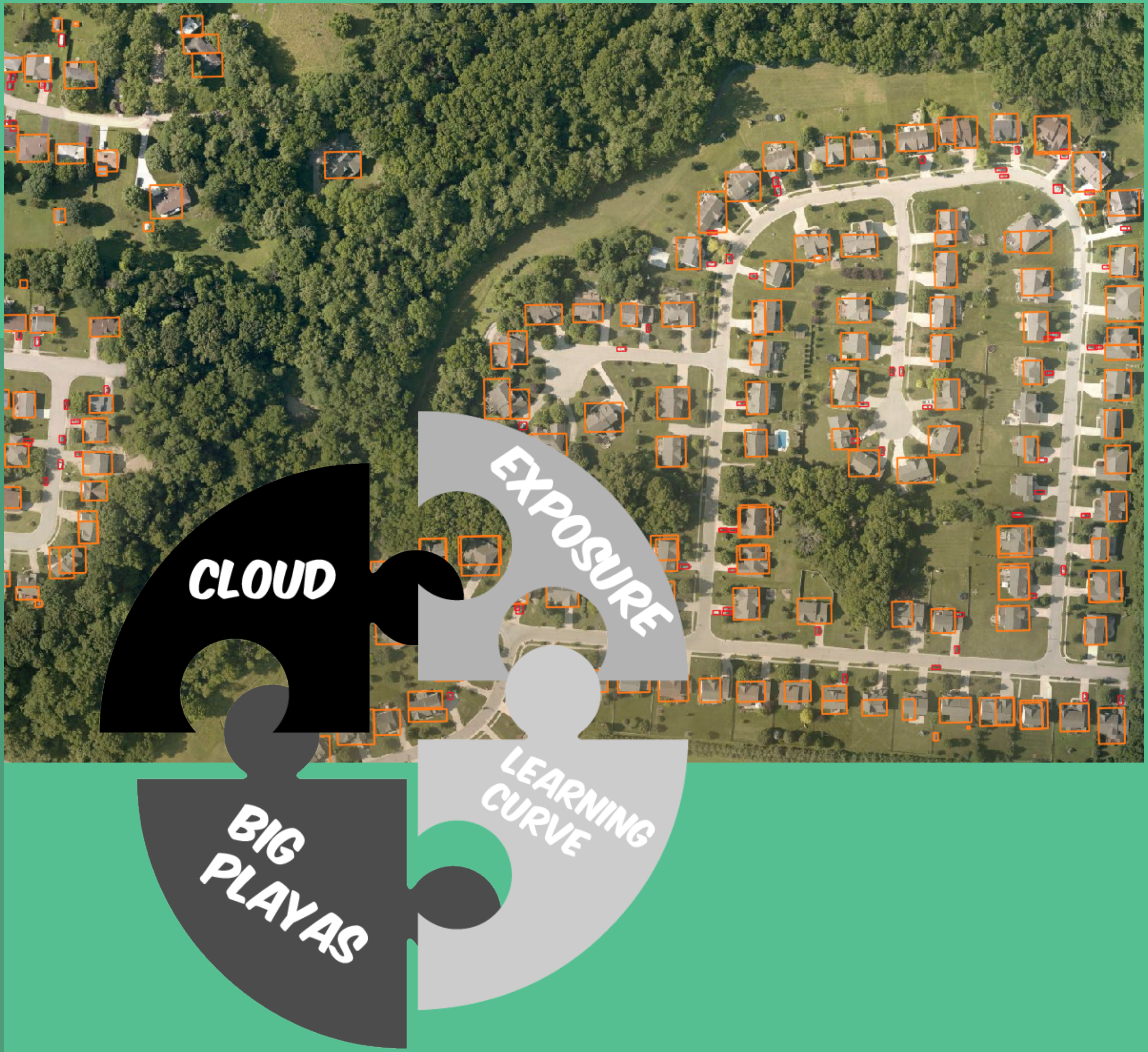
THE OSM + ML POTENTIAL

- ML (CNN) scales extraction of visual patterns w/ supervision
- A wealth of curated, crowd-sourced expertise that could be used for supervision
  - Still requires additional curation & extraction
- Avenue for adding contributions & education
  - Co-proposed creating a focus at UW (Geohackweek) to modify standard GIS content with end-to-end OSM to model generation
  - *Could we create a user 'market' of open-geoML models?* 🤔



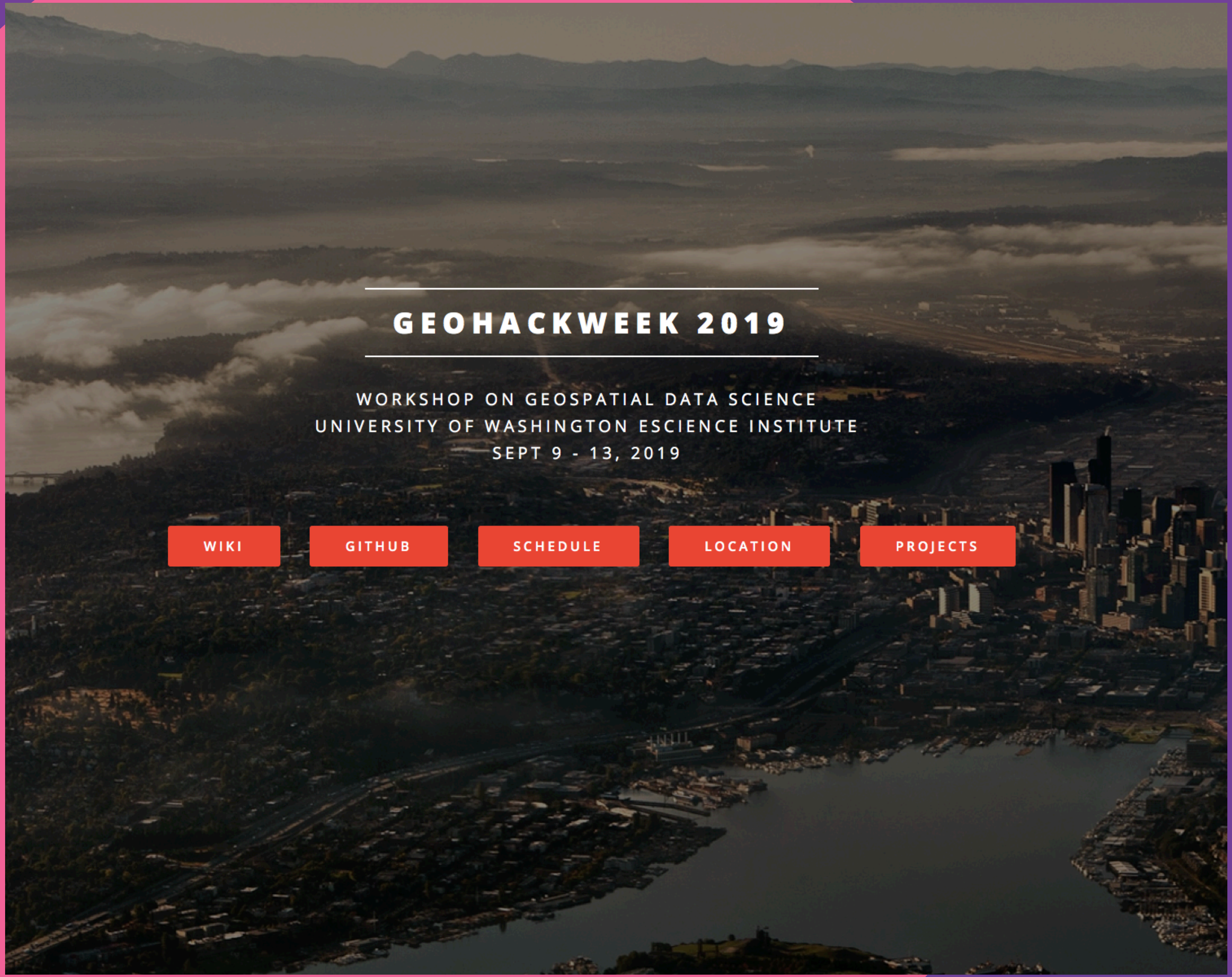
AN OVER-ABUNDANCE OF ML PLATFORMS

- Democratize dev of ‘geoML’, lower the barrier to entry for students & practitioners, & obliterate the ‘practice’ of platform commercialization
- geoML is not hard, but daunting
  - Difficulty as a DS with limited cloud-orchestration resources
  - Opensource/open-framework desires (common tools, community driven)
  - The POC is easy. The scaling is hard.
  - Big spenders ‘own’ the university programs (drive content)
  - Model marketplaces & commercialization that force you into platform use



UW GEOHACKWEEK

(https://geohackweek.github.io)



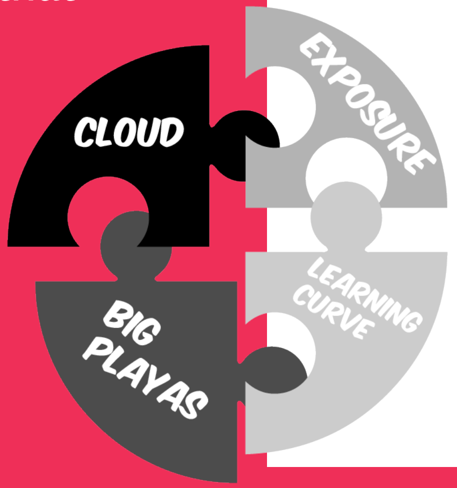
GEOHACKWEEK 2019

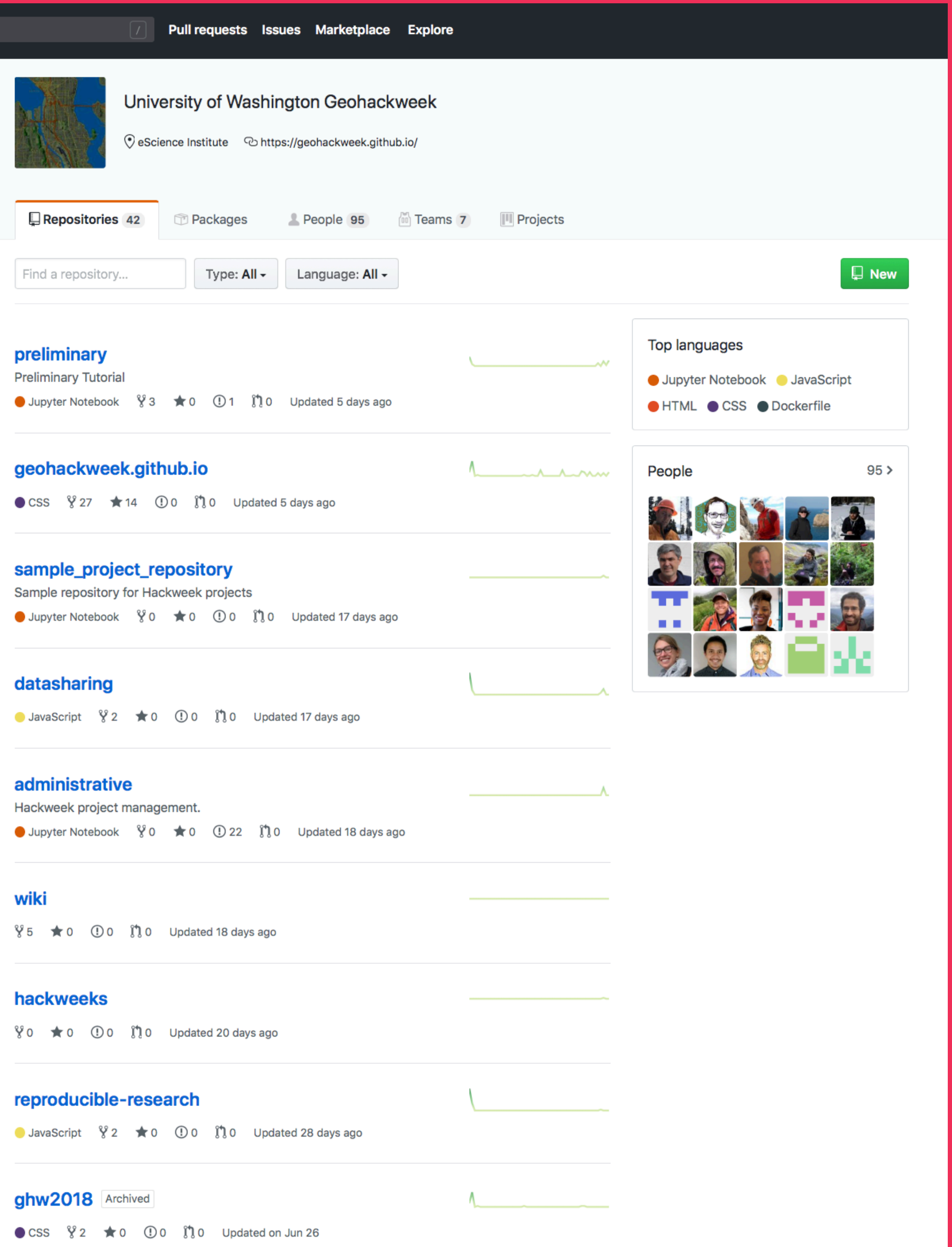
WORKSHOP ON GEOSPATIAL DATA SCIENCE
UNIVERSITY OF WASHINGTON ESCIENCE INSTITUTE
SEPT 9 - 13, 2019

- 2018 Software Carpentry Approach
  - Last minute volunteer to teach ML in 1 hr 1 month after moving to Seattle
  - 1 week to cover many aspects of geospatial + github + opensource
  - Team projects
- 2019 ML Revolution
  - We can do so much more!
  - Leverage HOT-OSM data sets as vector training data
  - **Raster -> Vector -> Machine Learning**
  - https://github.com/scottyhq/geohackweek2019-raster
  - https://github.com/geohackweek/tutorial_contents/tree/master/vector/notebooks
  - https://github.com/shaystrong/sagely/

EDUCATION

- **Leveraging OSM + ML + cloud compute**
  - University of Washington's GeoHackweek Multi-disciplinary, multi-background
  - HOT-OSM tasking (vector) → ML outcome building prediction in 1 week!
  - 50 new OSM mappers!!
  - AWS sponsorship (negotiated free compute!)
  - **Personal goal (unaffiliated)**: *Advance stagnant geospatial activities and create a community invested in producing optimal solutions that become foundational to advanced, open endeavors.*





University of Washington Geohackweek

eScience Institute https://geohackweek.github.io/

preliminary
Preliminary Tutorial
Jupyter Notebook 3 ★ 0 1 0 Updated 5 days ago

geohackweek.github.io
CSS 27 ★ 14 0 0 Updated 5 days ago

sample_project_repository
Sample repository for Hackweek projects
Jupyter Notebook 0 ★ 0 0 0 Updated 17 days ago

datasharing
JavaScript 2 ★ 0 0 0 Updated 17 days ago

administrative
Hackweek project management.
Jupyter Notebook 0 ★ 0 22 0 Updated 18 days ago

wiki
5 ★ 0 0 0 Updated 18 days ago

hackweeks
0 ★ 0 0 0 Updated 20 days ago

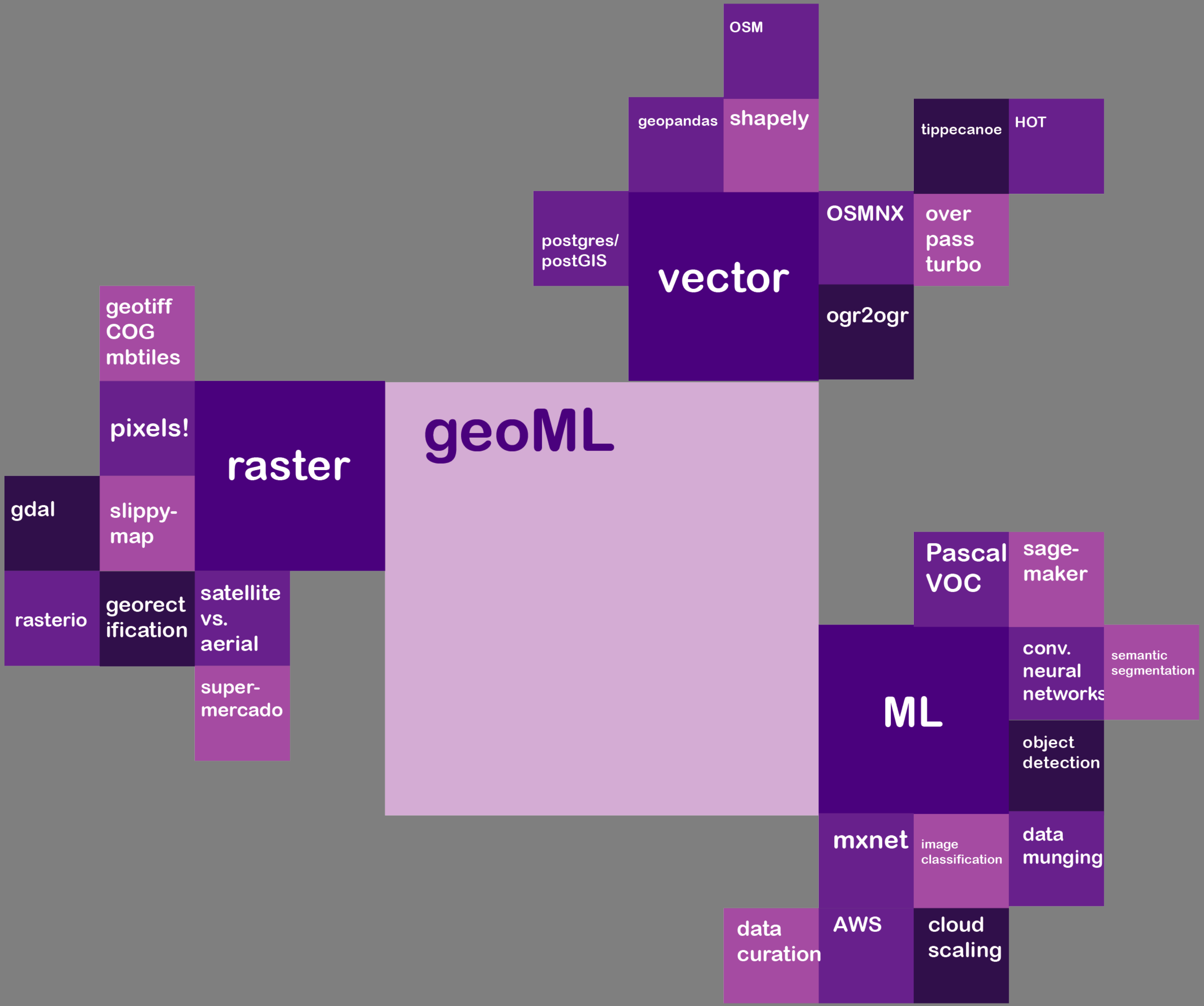
reproducible-research
JavaScript 2 ★ 0 0 0 Updated 28 days ago

ghw2018 Archived
CSS 2 ★ 0 0 0 Updated on Jun 26

Top languages

● Jupyter Notebook ● JavaScript ● HTML ● CSS ● Dockerfile

People 95 >



geoML

vector

OSM

geopandas

shapely

postgres/ postGIS

OSMNX

over pass turbo

tippecanoe

HOT

ogr2ogr

raster

geotiff COG mbtiles

pixels!

gdal

slippy-map

rasterio

georectification

satellite vs. aerial

super-mercado

ML

Pascal VOC

sage-maker

conv. neural networks

semantic segmentation

object detection

mxnet

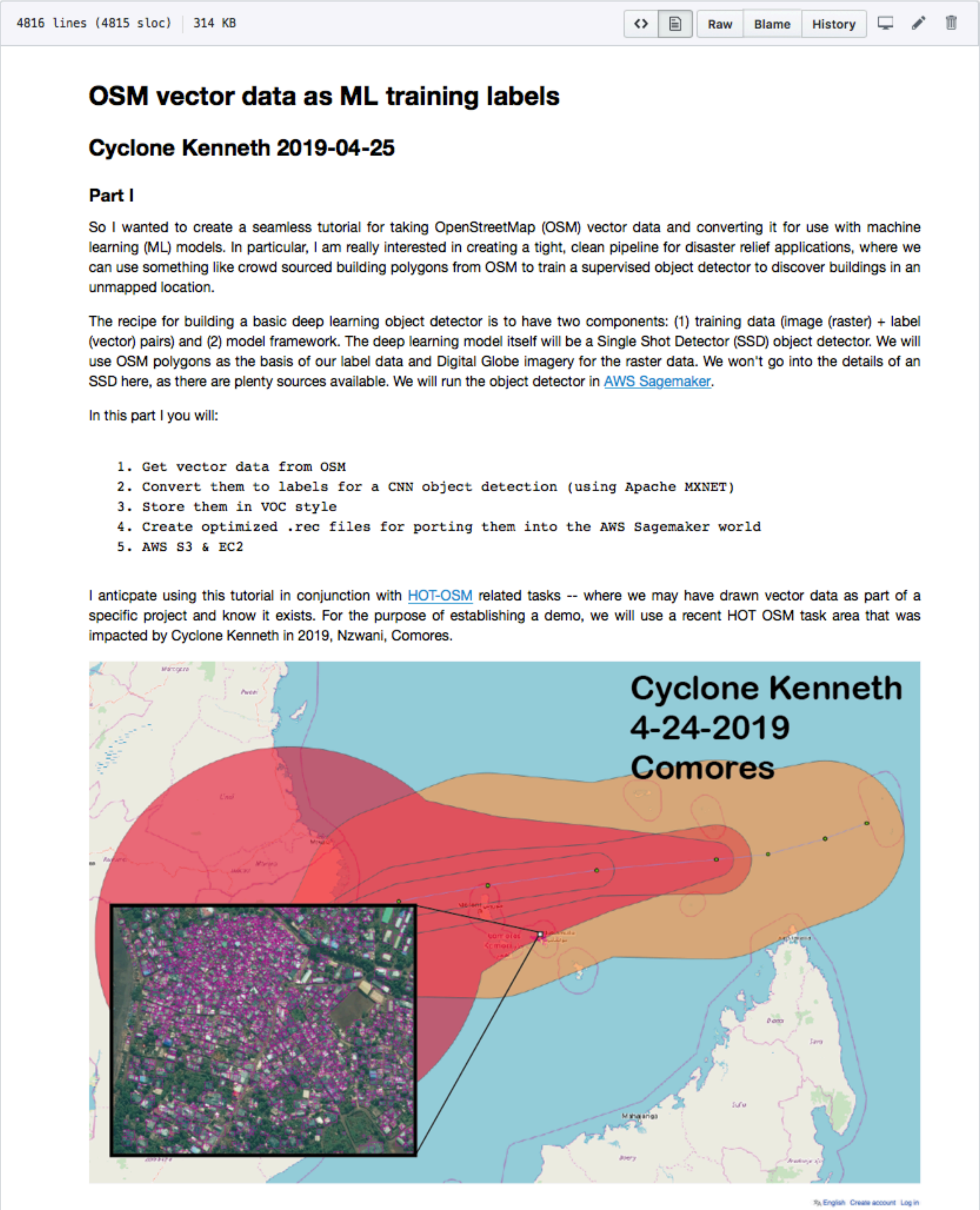
image classification

data munging

data curation

AWS

cloud scaling



OSM vector data as ML training labels

Cyclone Kenneth 2019-04-25

Part I

So I wanted to create a seamless tutorial for taking OpenStreetMap (OSM) vector data and converting it for use with machine learning (ML) models. In particular, I am really interested in creating a tight, clean pipeline for disaster relief applications, where we can use something like crowd sourced building polygons from OSM to train a supervised object detector to discover buildings in an unmapped location.

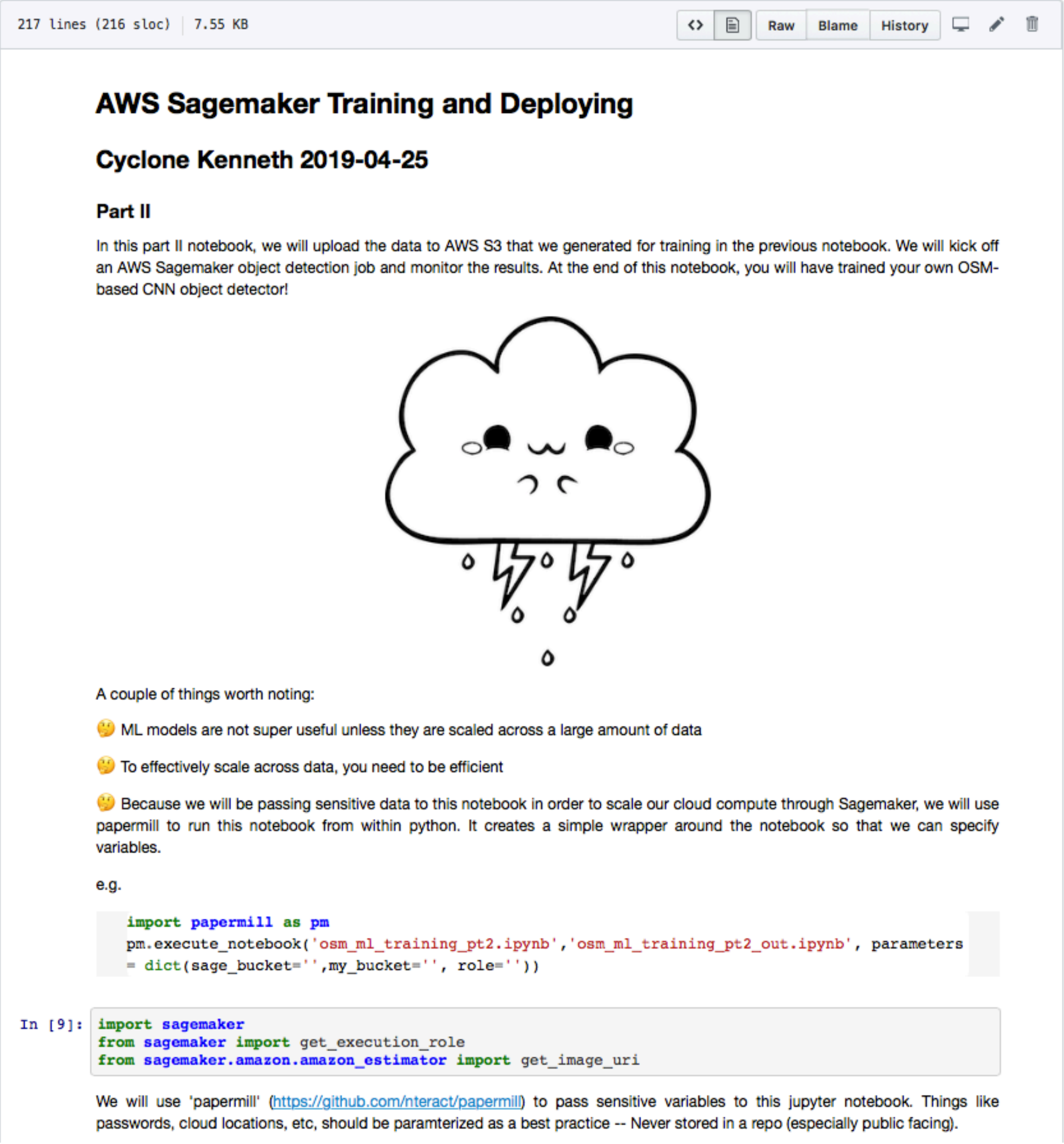
The recipe for building a basic deep learning object detector is to have two components: (1) training data (image (raster) + label (vector) pairs) and (2) model framework. The deep learning model itself will be a Single Shot Detector (SSD) object detector. We will use OSM polygons as the basis of our label data and Digital Globe imagery for the raster data. We won't go into the details of an SSD here, as there are plenty sources available. We will run the object detector in AWS Sagemaker.

In this part I you will:

```
1. Get vector data from OSM
2. Convert them to labels for a CNN object detection (using Apache MXNET)
3. Store them in VOC style
4. Create optimized .rec files for porting them into the AWS Sagemaker world
5. AWS S3 & EC2
```

I anticpate using this tutorial in conjunction with HOT-OSM related tasks -- where we may have drawn vector data as part of a specific project and know it exists. For the purpose of establishing a demo, we will use a recent HOT OSM task area that was impacted by Cyclone Kenneth in 2019, Nzwani, Comores.

https://github.com/shaystrong/sagely



AWS Sagemaker Training and Deploying

Cyclone Kenneth 2019-04-25

Part II

In this part II notebook, we will upload the data to AWS S3 that we generated for training in the previous notebook. We will kick off an AWS Sagemaker object detection job and monitor the results. At the end of this notebook, you will have trained your own OSM-based CNN object detector!

A couple of things worth noting:

🤔 ML models are not super useful unless they are scaled across a large amount of data

🤔 To effectively scale across data, you need to be efficient

🤔 Because we will be passing sensitive data to this notebook in order to scale our cloud compute through Sagemaker, we will use papermill to run this notebook from within python. It creates a simple wrapper around the notebook so that we can specify variables.

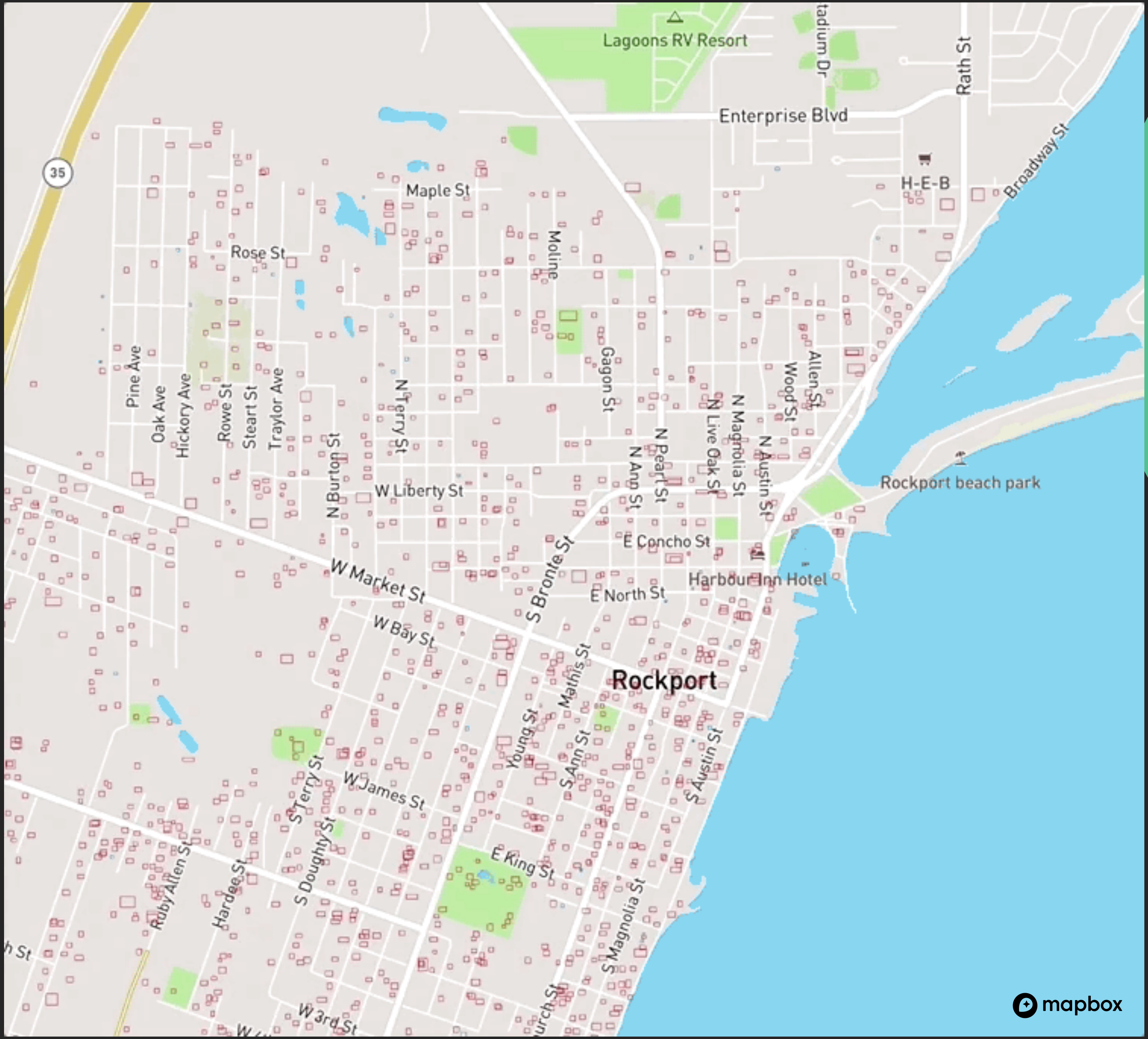
e.g.

```
import papermill as pm
pm.execute_notebook('osm_ml_training_pt2.ipynb','osm_ml_training_pt2_out.ipynb', parameters = dict(sage_bucket='',my_bucket='', role=''))
```

```
In [9]: import sagemaker
        from sagemaker import get_execution_role
        from sagemaker.amazon.amazon_estimator import get_image_uri
```

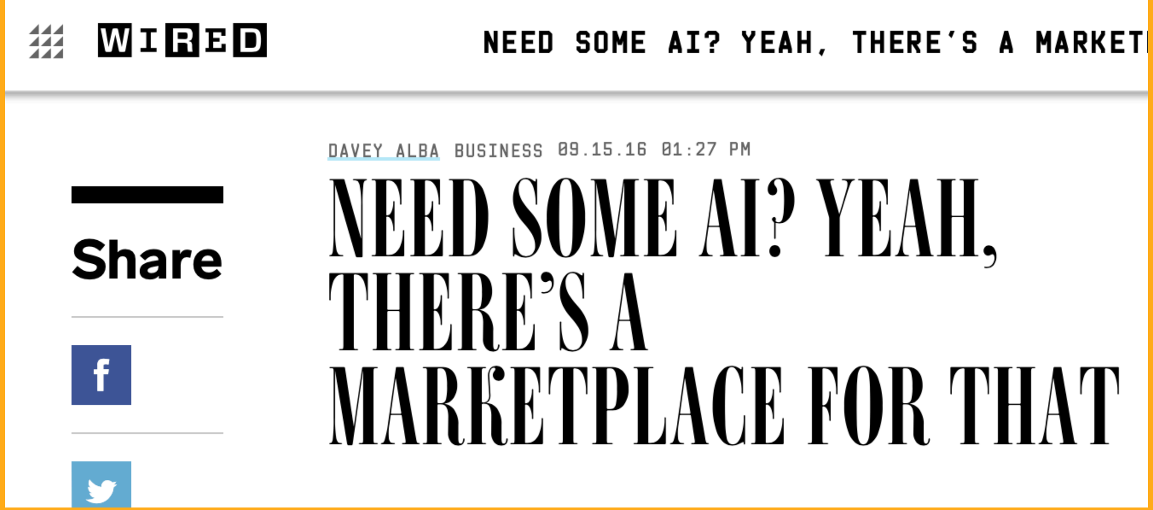
We will use 'papermill' (https://github.com/nteract/papermill) to pass sensitive variables to this jupyter notebook. Things like passwords, cloud locations, etc, should be paramterized as a best practice -- Never stored in a repo (especially public facing).

OSM TO STORM RESPONSE



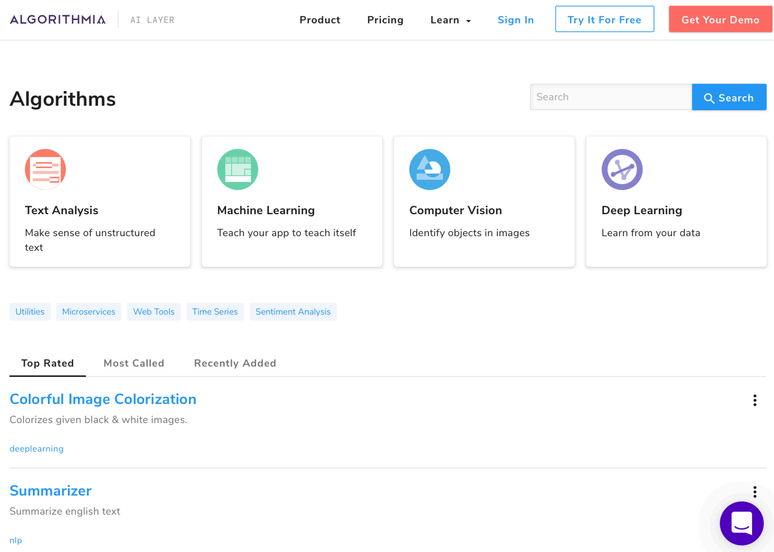
PLANS

- Flood commercialized marketplaces with models leverage-able by a broader community
  - Extricate 'free' models from their marriage to black-box platforms
- What could incentivization or a social currency approach look like in the long run?
- Challenges: Imagery, cloud compute, annotation cleansing



DAVEY ALBA BUSINESS 09.15.16 01:27 PM

NEED SOME AI? YEAH, THERE'S A MARKETPLACE FOR THAT



Algorithms

Text Analysis
Make sense of unstructured text

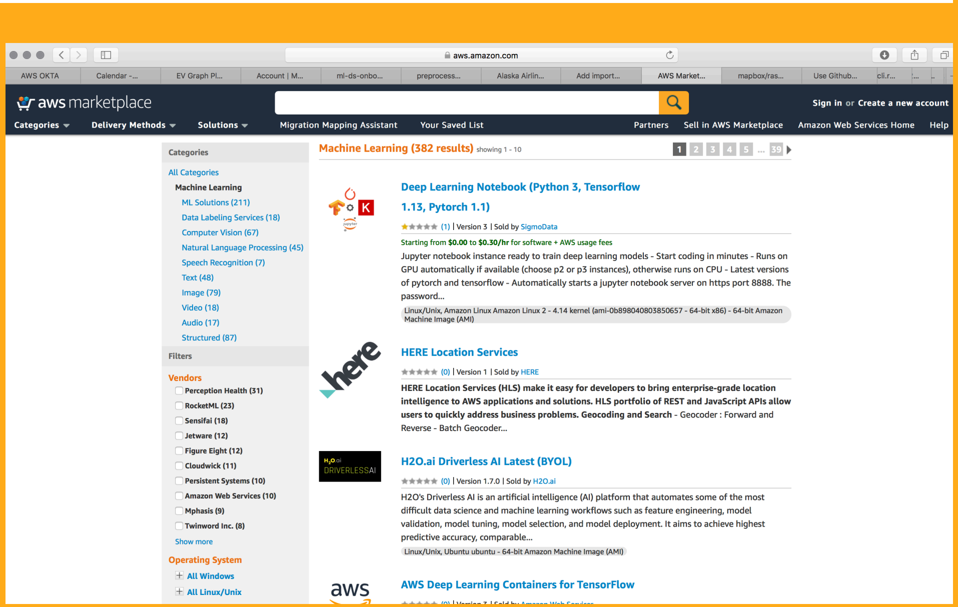
Machine Learning
Teach your app to teach itself

Computer Vision
Identify objects in images

Deep Learning
Learn from your data

Colorful Image Colorization
Colorizes given black & white images.
deeplearning

Summarizer
Summarize english text
nlp



Machine Learning (382 results) showing 1 - 10

Categories

All Categories
Machine Learning
ML Solutions (211)
Data Labeling Services (18)
Computer Vision (67)
Natural Language Processing (45)
Speech Recognition (7)
Text (48)
Image (79)
Video (18)
Audio (17)
Structured (87)

Filters

Vendors
□ Perception Health (31)
□ RocketML (23)
□ Sensifai (18)
□ Jetware (12)
□ Figure Eight (12)
□ Cloudwick (11)
□ Persistent Systems (10)
□ Amazon Web Services (10)
□ Mphasis (9)
□ Twinword Inc. (8)
Show more

Operating System
+ All Windows
+ All Linux/Unix

Deep Learning Notebook (Python 3, Tensorflow 1.13, Pytorch 1.1)
(1) | Version 3 | Sold by SigmoData
Starting from $0.00 to $0.30/hr for software + AWS usage fees
Jupyter notebook instance ready to train deep learning models - Start coding in minutes - Runs on GPU automatically if available (choose p2 or p3 instances), otherwise runs on CPU - Latest versions of pytorch and tensorflow - Automatically starts a jupyter notebook server on https port 8888. The password...
Linux/Unix, Amazon Linux Amazon Linux 2 - 4.14 kernel (ami-0b898040803850657 - 64-bit x86) - 64-bit Amazon Machine Image (AMI)

HERE Location Services
(0) | Version 1 | Sold by HERE
HERE Location Services (HLS) make it easy for developers to bring enterprise-grade location intelligence to AWS applications and solutions. HLS portfolio of REST and JavaScript APIs allow users to quickly address business problems. Geocoding and Search - Geocoder : Forward and Reverse - Batch Geocoder...

H2O.ai Driverless AI Latest (BYOL)
(0) | Version 1.7.0 | Sold by H2O.ai
H2O's Driverless AI is an artificial intelligence (AI) platform that automates some of the most difficult data science and machine learning workflows such as feature engineering, model validation, model tuning, model selection, and model deployment. It aims to achieve highest predictive accuracy, comparable...
Linux/Unix, Ubuntu ubuntu - 64-bit Amazon Machine Image (AMI)

AWS Deep Learning Containers for TensorFlow