Lecture 14: Cloud Computing I (Development and training)

Origin of Cloud Computing

Origin of Cloud Computing

- Andy Jassy (CEO of Amazon Web Services (AWS)) realised that their different teams were all solving the same problems over and over again.
- Amazon realised that they could solve these problems once with good API's and spead up all their teams.
- They also realised that this would be a huge business opportunity.
- What they did was they made each team consume other teams work as if they were customers.

Problems

Problems

- Training can be slow

Problems

- Training can be slow
- GPU machines are expensive (even many CPU machines)

Problems

- Training can be slow
- GPU machines are expensive (even many CPU machines)
- Running and managing hardware can be a full time job

Problems

- Training can be slow
- GPU machines are expensive (even many CPU machines)
- Running and managing hardware can be a full time job
- Allowing a project / website / company to **scale**

Structure of these lectures

Problem → Solution

Training

- I need a powerfull machine → Start a Virtual Machine
- I need to store my data and models → Use a bucket
- I need to run this training again → AI platform

Prediction

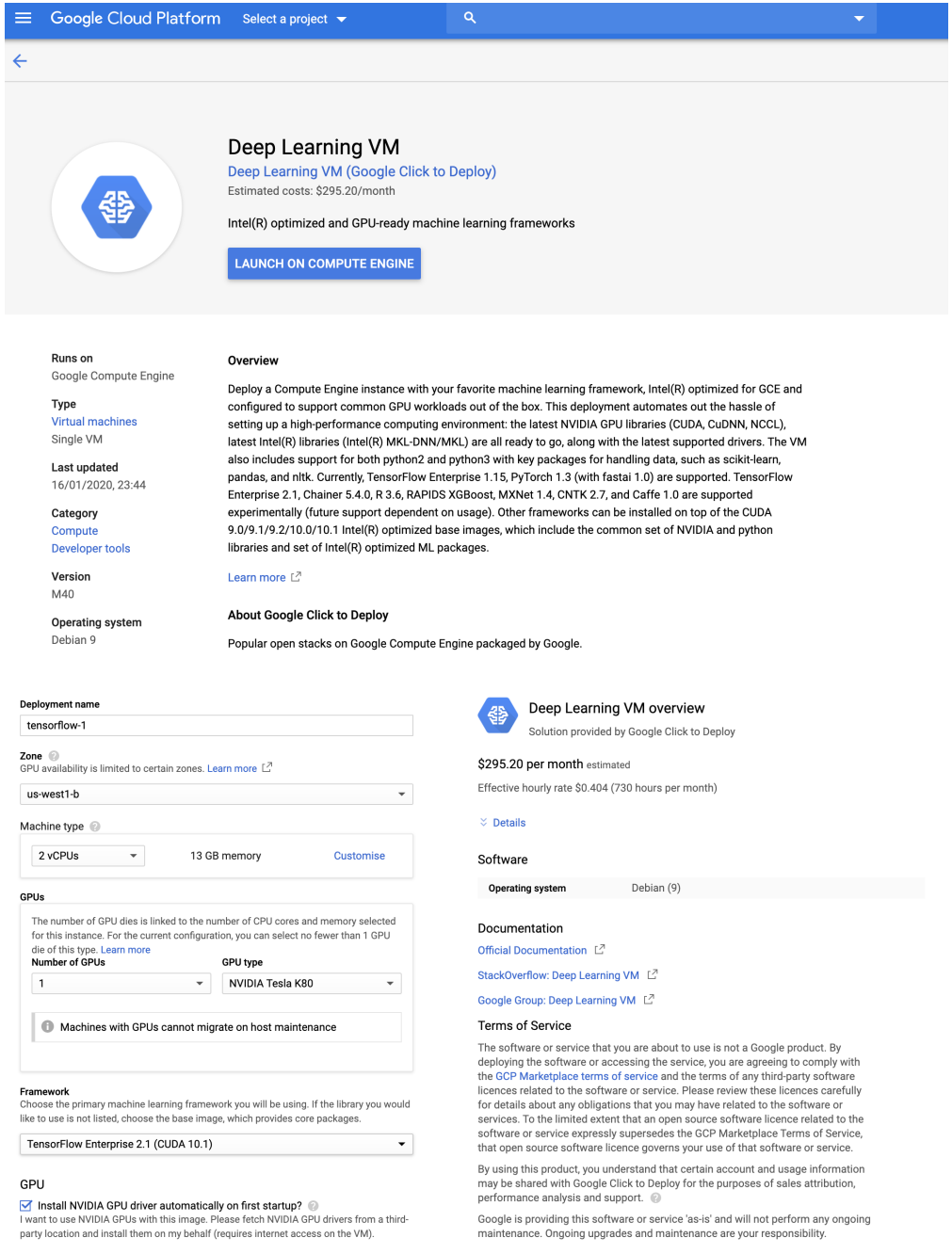
- I need to let other people use this model → AI platform
- I need to perform regular data operations → Airflow
- I need a simple front end for a website → Make a standard web app
- I need to have a database → SQL database
- I need to have a simple microservice → Make a cloud function
- I need to keep things seperate → Make good use of projects

For each problem I'll go through:

- a solution using the Google Cloud PLatform (GCP)
- things to consider
- the names or websites of similar products in AWS

I need a powerfull machine → Start a Virtual Machine.

- Choose what you want
- Pay only when you use it (almost)
- Make it even more powerfull if needed (or less)
- See https://cloud.google.com/deep-learning-vm/ (https://cloud.google.com/deep-learning-vm/)


Google Cloud Platform
Select a project
Deep Learning VM
Deep Learning VM (Google Click to Deploy)
Estimated costs: $295.20/month
Intel(R) optimized and GPU-ready machine learning frameworks
LAUNCH ON COMPUTE ENGINE
Runs on
Google Compute Engine
Type
Virtual machines
Single VM
Last updated
16/01/2020, 23:44
Category
Compute
Developer tools
Version
M40
Operating system
Debian 9
Overview
Deploy a Compute Engine instance with your favorite machine learning framework, Intel(R) optimized for GCE and configured to support common GPU workloads out of the box. This deployment automates out the hassle of setting up a high-performance computing environment: the latest NVIDIA GPU libraries (CUDA, CuDNN, NCCL), latest Intel(R) libraries (Intel(R) MKL-DNN/MKL) are all ready to go, along with the latest supported drivers. The VM also includes support for both python2 and python3 with key packages for handling data, such as scikit-learn, pandas, and nltk. Currently, TensorFlow Enterprise 1.15, PyTorch 1.3 (with fastai 1.0) are supported. TensorFlow Enterprise 2.1, Chainer 5.4.0, R 3.6, RAPIDS XGBoost, MXNet 1.4, CNTK 2.7, and Caffe 1.0 are supported experimentally (future support dependent on usage). Other frameworks can be installed on top of the CUDA 9.0/9.1/9.2/10.0/10.1 Intel(R) optimized base images, which include the common set of NVIDIA and python libraries and set of Intel(R) optimized ML packages.
Learn more
About Google Click to Deploy
Popular open stacks on Google Compute Engine packaged by Google.
Deployment name
tensorflow-1
Zone
GPU availability is limited to certain zones. Learn more
us-west1-b
Machine type
2 vCPUs
13 GB memory
Customise
GPUs
The number of GPU dies is linked to the number of CPU cores and memory selected for this instance. For the current configuration, you can select no fewer than 1 GPU die of this type. Learn more
Number of GPUs
1
GPU type
NVIDIA Tesla K80
Machines with GPUs cannot migrate on host maintenance
Framework
Choose the primary machine learning framework you will be using. If the library you would like to use is not listed, choose the base image, which provides core packages.
TensorFlow Enterprise 2.1 (CUDA 10.1)
GPU
Install NVIDIA GPU driver automatically on first startup?
I want to use NVIDIA GPUs with this image. Please fetch NVIDIA GPU drivers from a third-party location and install them on my behalf (requires internet access on the VM).
Deep Learning VM overview
Solution provided by Google Click to Deploy
$295.20 per month estimated
Effective hourly rate $0.404 (730 hours per month)
Details
Software
Operating system
Debian (9)
Documentation
Official Documentation
StackOverflow: Deep Learning VM
Google Group: Deep Learning VM
Terms of Service
The software or service that you are about to use is not a Google product. By deploying the software or accessing the service, you are agreeing to comply with the GCP Marketplace terms of service and the terms of any third-party software licences related to the software or service. Please review these licences carefully for details about any obligations that you may have related to the software or services. To the limited extent that an open source software licence related to the software or service expressly supersedes the GCP Marketplace Terms of Service, that open source software licence governs your use of that software or service.
By using this product, you understand that certain account and usage information may be shared with Google Click to Deploy for the purposes of sales attribution, performance analysis and support.
Google is providing this software or service 'as-is' and will not perform any ongoing maintenance. Ongoing upgrades and maintenance are your responsibility.

Things to consider

- If you're going to need it forever it *may be* worth buying one!
- After a year of complete usage it can cost as much as a laptop
- No:
  - set up time
  - risk of damage or theft
- Are you sure you're going to need the laptop constantly for a year?

Things to consider

- If you're going to need it forever it *may be* worth buying one!
- After a year of complete usage it can cost as much as a laptop
- No:
  - set up time
  - risk of damage or theft
- Are you sure you're going to need the laptop constantly for a year?

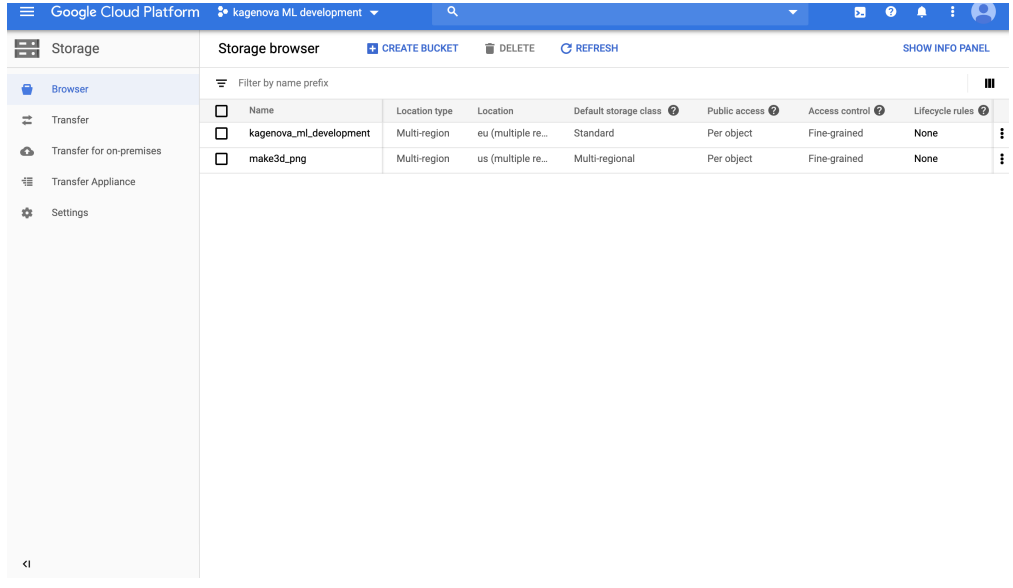
- **Remember to turn it off!**

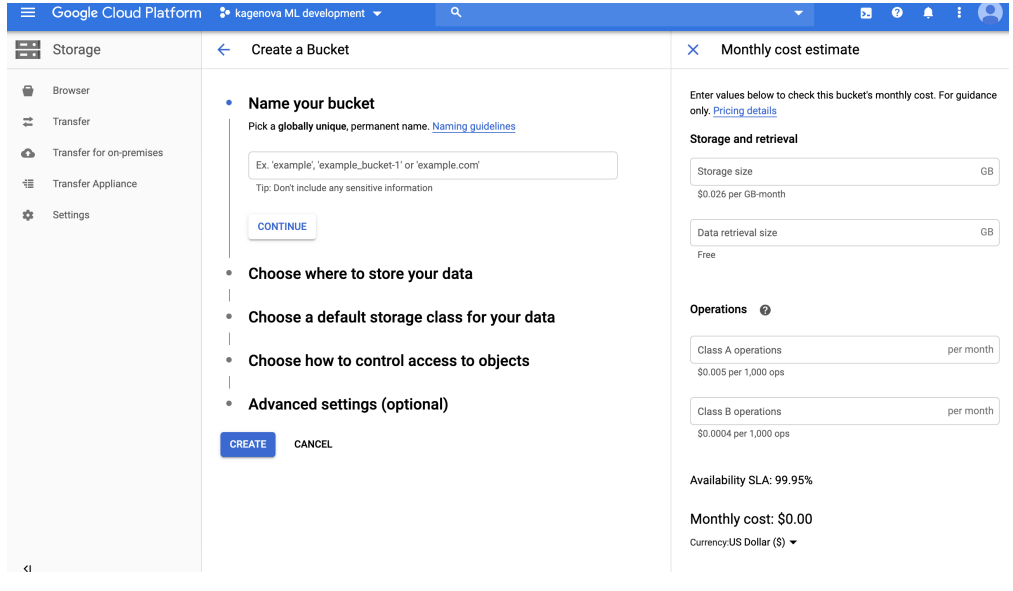
Other versions

AWS and Microsoft Azure have comparible set up (there is almost no difference).

I need to store my data and models → Use a bucket

- Infinite data storage (just pay for what you need)
- Can controll access to data (using permissons and making avalable)
- Can choose where the data is stored
  - Best to be close to location of machines and users
- See https://cloud.google.com/storage/docs/creating-buckets (https://cloud.google.com/storage/docs/creating-buckets)


Google Cloud Platform
kagenova ML development
Storage
Browser
Transfer
Transfer for on-premises
Transfer Appliance
Settings
Storage browser
CREATE BUCKET
DELETE
REFRESH
SHOW INFO PANEL
Filter by name prefix
Name
Location type
Location
Default storage class
Public access
Access control
Lifecycle rules
kagenova_ml_development
Multi-region
eu (multiple re...
Standard
Per object
Fine-grained
None
make3d_png
Multi-region
us (multiple re...
Multi-regional
Per object
Fine-grained
None


Google Cloud Platform
kagenova ML development
Storage
Browser
Transfer
Transfer for on-premises
Transfer Appliance
Settings
Create a Bucket
Name your bucket
Pick a globally unique, permanent name. Naming guidelines
Ex. 'example', 'example_bucket-1' or 'example.com'
Tip: Don't include any sensitive information
CONTINUE
Choose where to store your data
Choose a default storage class for your data
Choose how to control access to objects
Advanced settings (optional)
CREATE
CANCEL
Monthly cost estimate
Enter values below to check this bucket's monthly cost. For guidance only. Pricing details
Storage and retrieval
Storage size
GB
$0.026 per GB-month
Data retrieval size
GB
Free
Operations
Class A operations
per month
$0.005 per 1,000 ops
Class B operations
per month
$0.0004 per 1,000 ops
Availability SLA: 99.95%
Monthly cost: $0.00
Currency:US Dollar ($)

Thing to consider

- Cost per Gb is different with different settings
- Is one of the fastest ways to store data for general use (i.e. making a website)

Other versions

They are all the same!

I need to run this training again → AI platform

I need to run this training again → AI platform

- Run training on a GPU from a comand line script anywhere
- Only uses resources that are need (spins up, runs and ends)
- All data must be downloaded and saved elsewhere (e.g. on Google Cloud Service (GCS))
- Can even run hyper-parameter tuning in with ymal file
- See https://cloud.google.com/ai-platform/ (https://cloud.google.com/ai-platform/)

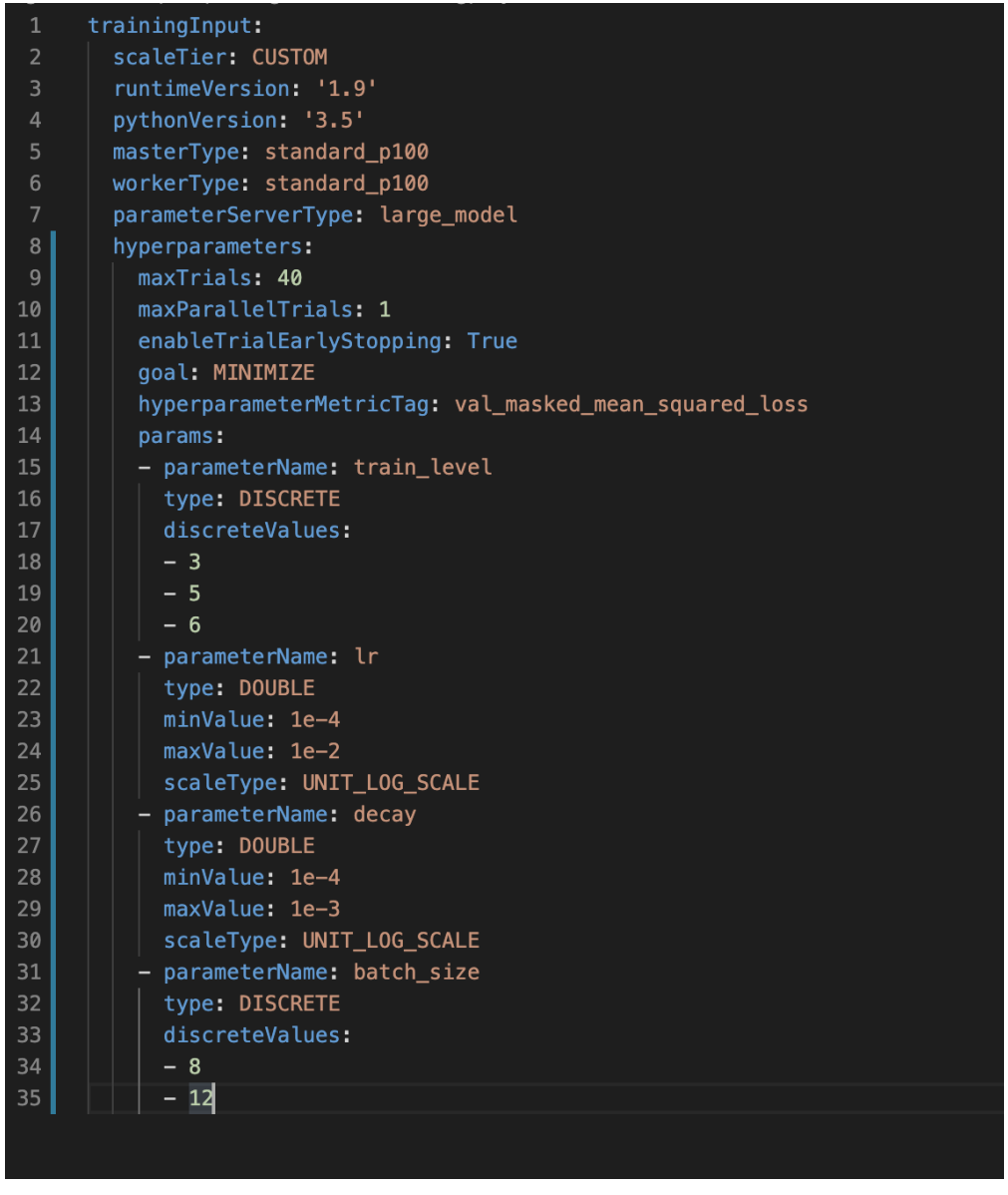
Comand line script to submit jobs.

```
JOBNAME="depth_nyu_make3d_"
JOBNAME+=`date '+%Y_%m_%d_%H_%M_%S'`
REPO_PATH="gs://kagelearn/outputs/depth-estimation-planar/"+$JOBNAME
REPO_PATH+="/training_code/"

gsutil -m cp -r . $REPO_PATH

gcloud ml-engine jobs submit training $JOBNAME \
    --module-name=trainer.train \
    --package-path=trainer \
    --job-dir=gs://kagelearn/outputs/depth-estimation-planar/ \
    --region=europe-west1 \
    --config=cloudml_gpu.yaml \
    -- \
    --testing=MAKE3D_IN_THE_CLOUD \
    --job-name=$JOBNAME \
```

Yaml file to define the training and hyperparameter tuning



```yaml
trainingInput:
  scaleTier: CUSTOM
  runtimeVersion: '1.9'
  pythonVersion: '3.5'
  masterType: standard_p100
  workerType: standard_p100
  parameterServerType: large_model
  hyperparameters:
    maxTrials: 40
    maxParallelTrials: 1
    enableTrialEarlyStopping: True
    goal: MINIMIZE
    hyperparameterMetricTag: val_masked_mean_squared_loss
    params:
    - parameterName: train_level
      type: DISCRETE
      discreteValues:
      - 3
      - 5
      - 6
    - parameterName: lr
      type: DOUBLE
      minValue: 1e-4
      maxValue: 1e-2
      scaleType: UNIT_LOG_SCALE
    - parameterName: decay
      type: DOUBLE
      minValue: 1e-4
      maxValue: 1e-3
      scaleType: UNIT_LOG_SCALE
    - parameterName: batch_size
      type: DISCRETE
      discreteValues:
      - 8
      - 12
```

Thing to consider

- Slow for quick iteration
- Code should be sure to work before using AI platform

Other versions

- AWS have Amazon SageMaker https://aws.amazon.com/machine-learning/ (https://aws.amazon.com/machine-learning/)