Introduction to Pytorch

CSU 2021
Rasmus F. Ørsøe
18-08-2021

Overview

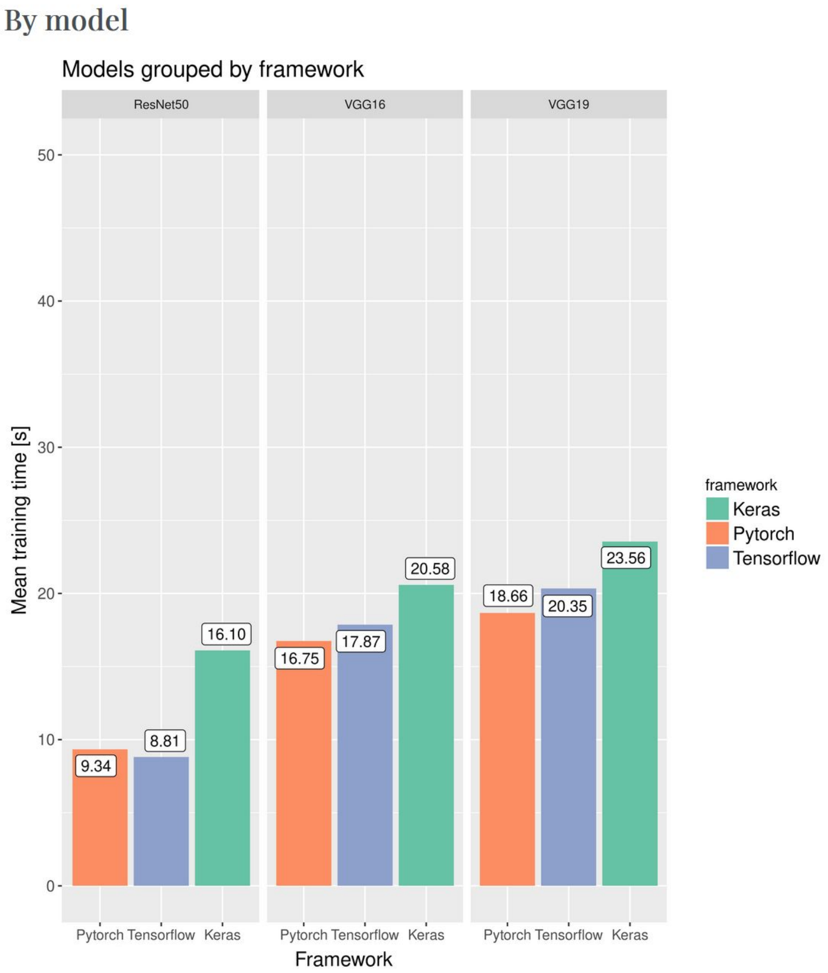
1) Pros and Cons of Pytorch

2) Tensors, devices and gradients

3) Keras and PyTorch Code Comparison

Pros

- Speed!


By model
Models grouped by framework
ResNet50
VGG16
VGG19
Mean training time [s]
0
10
20
30
40
50
9.34
8.81
16.10
16.75
17.87
20.58
18.66
20.35
23.56
Pytorch
Tensorflow
Keras
Framework
framework
Keras
Pytorch
Tensorflow

Pros

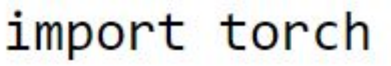
- Speed!

- (nearly) Complete Control
    - Dynamic computational graphs
    - Data parallelism

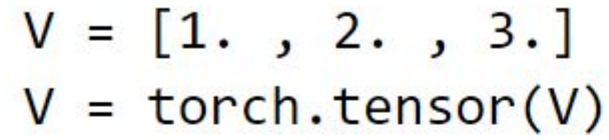
Cons

- Much to take in

Tensors

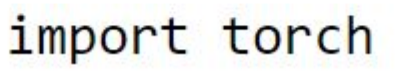


```
import torch
```

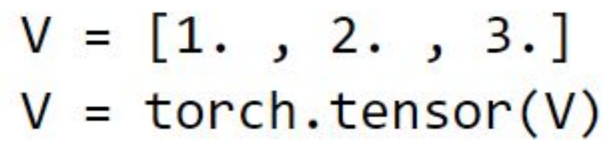


```
V = [1. , 2. , 3.]
V = torch.tensor(V)
```

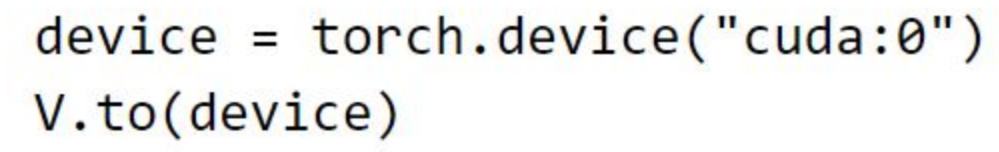
Tensors



```
import torch
```

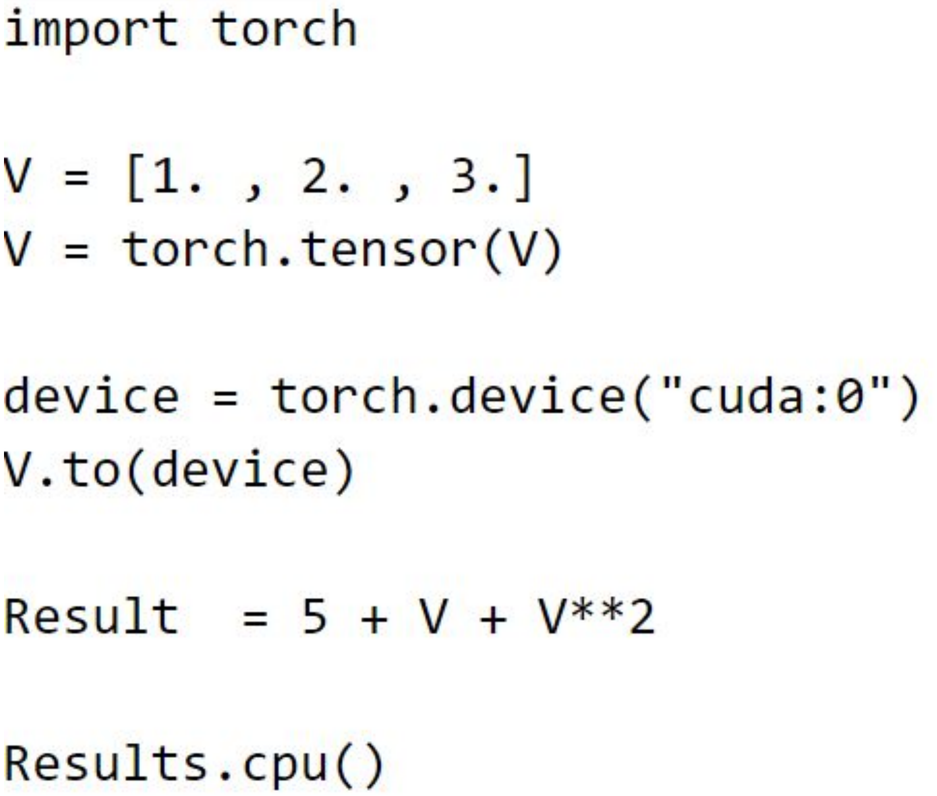


```
V = [1. , 2. , 3.]
V = torch.tensor(V)
```



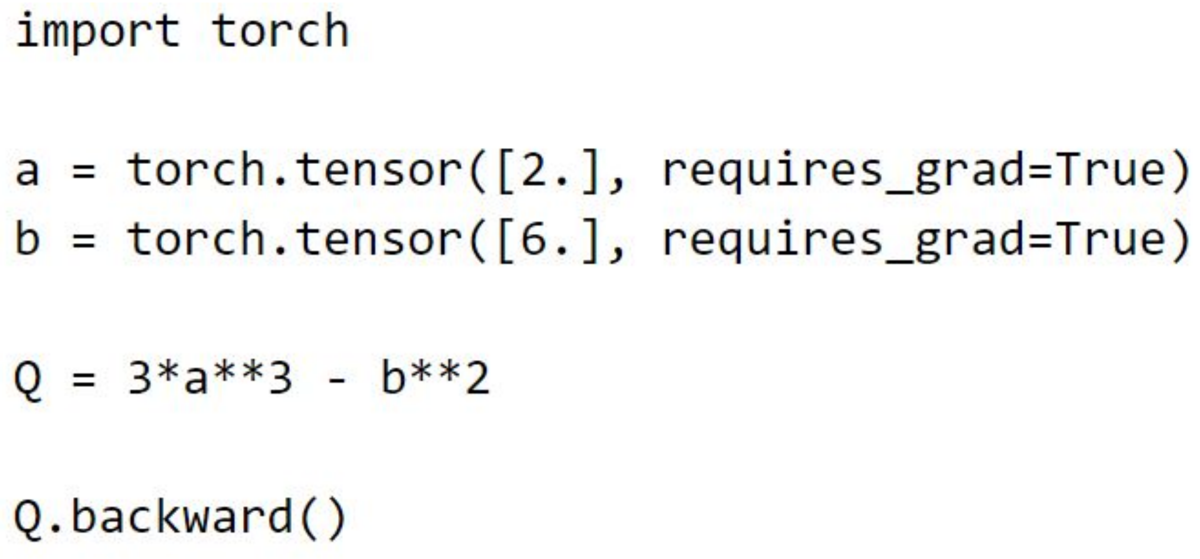
```
device = torch.device("cuda:0")
V.to(device)
```

Tensors



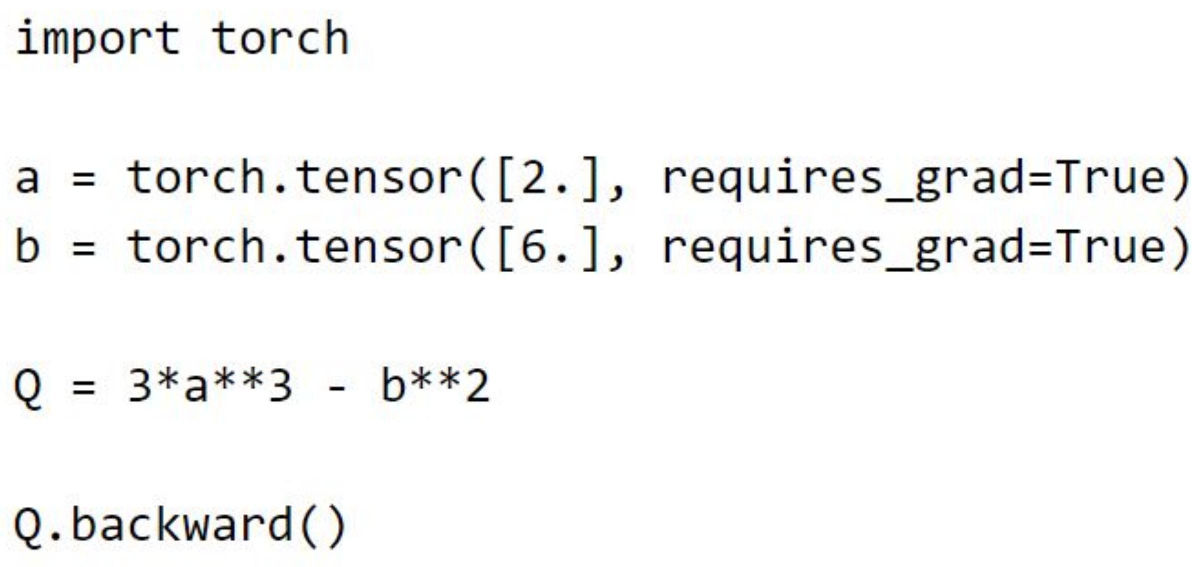
```
import torch

V = [1. , 2. , 3.]
V = torch.tensor(V)

device = torch.device("cuda:0")
V.to(device)

Result  = 5 + V + V**2

Results.cpu()
```

Tensors



```
import torch

a = torch.tensor([2.], requires_grad=True)
b = torch.tensor([6.], requires_grad=True)

Q = 3*a**3 - b**2

Q.backward()
```

Tensors

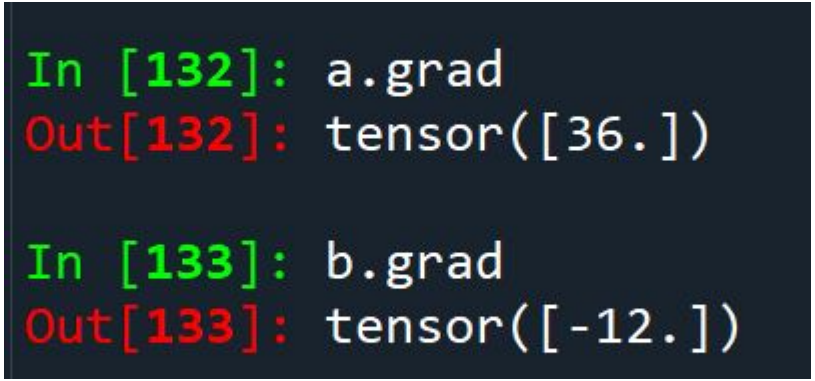


```
import torch

a = torch.tensor([2.], requires_grad=True)
b = torch.tensor([6.], requires_grad=True)

Q = 3*a**3 - b**2

Q.backward()
```

$$\frac{\partial Q}{\partial a} = 9a^2$$ = 36

$$\frac{\partial Q}{\partial b} = -2b$$ = -12

Tensors



```
In [132]: a.grad
Out[132]: tensor([36.])

In [133]: b.grad
Out[133]: tensor([-12.])
```

$$\frac{\partial Q}{\partial a} = 9a^2 = 36$$

$$\frac{\partial Q}{\partial b} = -2b = -12$$

Time for notebooks!