What frameworks to use?

Whow numpy is different from tensorflow?

Wwhat keras is?

Do we need to know python?

What is anaconda?

Wwhat is venv?

How to setup python packages?

Will it work on windows?

Do we need gpu?

How to deal with python notebooks?

First things first

To have anything possible, we need to setup python properly.

Examples will be shown for windows machines now, but it is more or less the same for linux also

# Installing python

You can easily install python just going to <https://www.python.org/> and get it from there.

But I would suggest different approach.

Go to anaconda site <https://www.continuum.io/downloads> and download python 3.6 64-bit installer

By doing that you'll get not only python, but awesome package manager, 'conda' which is kinda similar to nuget packages you're familiar with

When click-click-next installation is ready, lanuch terminal window (Win+R, cmd, Enter)

And type

```

python --version

```

You will see something like this

C: - -version 
Python 3.6.ø : : Anaconda 4.3.ø (64-bit) 

This means you have running python on your machine

Let's double check that it works properly

Python 3.6.ø 'Anaconda 4.3.ø (64-bit)l (default, 
Type "help", "copyright", "credits" or "license" 
1+1 
Dec 23 2ø16, 11:57:41) v. 19W 64 bit (x«h4)] on Win32 
for •re information . 

Seems like it can make some addition, which is definitely helpful for our machine learning tutorial

# working with virtual environments

Because different project might use different set of tools and libraries, we need to avoid dll hell somehow.

In python world it is done by so-called virtual environments.

You can think of virtual environment as a docker-ish sandbox with python interpreter and set of libraries inside.

One virtual environment lives inside some folder.

Important thing : virtual environment can be accessed from anywhere, you need to 'activate'

But before we do activation, let's create one and see how easy it is

```

conda create **--**name batman python**=**3 anaconda

*From <*[*https://conda.io/docs/using/envs.html*](https://conda.io/docs/using/envs.html)*>*

```

C: create - -name batman python—3 anacM'da 
Fetching package rtadata . . 
Solving package . 
Package plan for installatim in environ—nt C: 
The following NN packages 
will be INSTALLED: 
Ii cense : 
alabaster : 
anaconda : 
anacorG-cIient: 
1. I-py36_1 
e. 7.9-py36_ø 
4.3. I-np111py36_ø 
1.6.ø-py36_ø 

It will ask you to agree

Proceed ( 丨 刃 ) ~ y 
anaconda—proJe 
anaconda—navlg 
【 vc14 亅 
Time: 
Time: 
e: W: 797 · 54 kB/S 
e: : 03 1 · 22 

And will install A LOT of packages that are included in anaconda package.

What is anaconda? It is set of libraries for machine learning and stuff, packaged and being supported by conda.io guys.

After few minutes/seconds/years of installation, you'll end up with virtual environment

How can you check which environments do you have on the machine?

Just as easy as

```

conda env list

```

C: env 
# conda 
d eeplearn 
deepIearn2 
deepIearn3 
C: user s\oleg 
list 
C: batman 
C: \deeplearn 
C: \deepIearn2 
C: \deepIearn3 

With asterisk you can see that current environment is 'root'

Let's change our environment to batman, by 'activating' it

```

activate batman

```

C:\Users\oleg\AppData\Local\Temp\msohtmlclip1\02\clip_image006.png

Now you can see that you live in batman world

If you want to deactivate it, just use

```

deactivate

```

(batman) C : \Users 
C: user s\oleg 

And batman is gone for good

Let's see what we have inside of our batman world

C:users\oleg>actlvate batman 
(batman) C:Wsers\oleg>python 
Python 3.6. e IAnaconda 4.3 · ø (64-bit)l (default, 
Type "help", "copyrlght", "credItS" 
23 2a16 , 11 : 57 : 41 ) [l 
fo 「 Infor•tlon. 
> > > numpy as np 
> > > zeros(lø) 
array([ 
> > > zeros(lø) + 2 
array([ 2 · 
> > > np. zeros(lø) + np.mes(lø) 
array([ 1 · 
> > > np.zeros(lø) + • · s ( le ) /ø. 
array([ 3 · 33333333 , 3.33333333 , 
3 · 33333333 , 3.33333333 , 
” > exit() 
3.33333333 , 
3.33333333 , 
2. 亅 ) 
1. 亅 ) 
3.33333333 , 
3.33333333 , 
3 · 33333333 , 
3 · 33333333 亅 ) 

You can see nice little session here with numpy included.

Numpy is biiig and fancy library for doing matrix operation.

Numpy is used by tensorflow, tensorflow is used by keras.

For deep learning stuff we will try to stick with keras, but this is leaky abstraction. You will need to learn keras, some tensorflow, some numpy, and maybe some python, because everything is written in it.

# installing tensorflow

Tensorflow is deep learning library that will rule the world, because it is developed by google and is being pushed really hard.

If you want to install GPU support for tensorflow, please visit this page <https://www.tensorflow.org/install/install_windows>

If you just want to use good old school CPU version, you can just do following in our batman world

```

**pip install --ignore-installed --upgrade** [**https://storage.googleapis.com/tensorflow/windows/cpu/tensorflow-1.0.1-cp35-cp35m-win\_amd64.whl**](https://storage.googleapis.com/tensorflow/windows/cpu/tensorflow-1.0.1-cp35-cp35m-win_amd64.whl)

*From <*[*https://www.tensorflow.org/install/install\_windows*](https://www.tensorflow.org/install/install_windows)*>*

```

We try to run it aand…

(batman) C: install 
-cp35m-wi . 
-- ignore-installed 
- -L»grade 

Oh nooo… We cannot do develop using tensorflow on windows. Or can we?

If we look for documentation on tensorflow site, we will notice that

TensorFlow only supports version 3.5.x of Python on Windows.

*From <*[*https://www.tensorflow.org/install/install\_windows*](https://www.tensorflow.org/install/install_windows)*>*

And conda installed latest version of python, which is 3.6. Screw it.

Let's create another virtual env, with python 3.5 this time.

```

conda create --name superman python=3.5 anaconda

```

(batman) C: users 
C: create - -name superman pyttwn=3.5 anaconda 

Now we specified version of python directly, let's hope it will work now

C: superman 
1 
(superman) install --igrwre-installed --upgrade 
. whI 
Collecting fræ 
(14.7B) 
1 2.44B 838kB/s eta 

Yaaaah !!! It is doing something!

After a while you should see

ccessfully i 
wheel-e.29.ø 
(superman) C: 
ppdirs-I.4.3 mmy-I.12.I packaging-16.8 pyparsing-2.2.ø setuptooIs-34.4.I six-I.lø.ø 

We're on the right track,

Let's see if tensorflow works

Copypaste following to python terminal

```

**import tensorflow as tf**

**hello = tf.constant('Hello, TensorFlow!')**  
**sess = tf.Session()**  
**print(sess.run(hello))**

*From <*[*https://www.tensorflow.org/install/install\_windows*](https://www.tensorflow.org/install/install_windows)*>*

```

(s uperman ) 
Python 3.5.2 
sess 
_.cc:943] 
C : use rs 
IAnaconda 4.3. I (64-bit)l (default, 
Type "help", "copyright", "credits" or "license" 
= tf.Session() 
sess. run( hello 
ease -WI n 
• ce_type: "CPU" • ) for op: BestSpIits 
Jul 5 2Ø16, 11:41:13) v.19W 64 bit (x«h4)] on Win32 
for •re information. 
our AC 'V 
kernel. 
Ignore this part 
c : release -wi device \cpu Nos ndæ•s \tensor 
ndæStats" device_type: "CPU" • ) for unkrwwn op: CountExtremeIyRandmS 
c : device \cpu Nos \ten 
r e\fr op 
device_type: "CPU" • ) for unknown op: FinishedPbdes 
c : release -win", device \cpu Nos 
e_type: "CPU" • ) for op: 
ten k\op 
c : release \cpu Nos 
ToFIoat" device_type: "CPU"' ) for unknæm op: ReinterpretStringToFIoat 
c : release -wi device \cpu Nos ndæ•s 
ce_type: "CPU"') for unknown op: Snplelnputs 
c : release device \cpu Nos 
device_type: "CPU" • ) for unknom op: 
c : release -wi device \cpu Nos 
• ce_type: "CPU" • ) for op: TopNInsert 
c : release - device \cpu Nos 
• ce_type: "CPU"') for op: TopNRemve 
kernel. 
kernel. 
kernel. 
kernel. 
kernel. 
kernel. 
kernel. 
kernel. 
kernel. 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
cc: 943] 
• "BestSpIits" de 
• "Count Ext re—Iy 
: "FinishedPbdes" 
: devi 
• "ReinterpretStr 
• "To*lnsert" de 
• de 
: "Treeprediction 
c : hor\workspp - 
" "CPU" • ) for 
c : home\worksp 
ots" device 
o, TensorF 
our result 
• - e \cpu Nos 
ctions 
\ c pu Nos \ten k\op 
•FertileSIots 

Heep heep hurray!

Tensorflow is working.

# setting up keras

```

pip install keras

```

Collecting keras 
Keras-2.ø.3.tar.gz 
løøs I 
1 2ø4kB 685kB/s 
Collecting olefile (from pillow) 
Collecting theano keras) 
Theano-ø.9.ø.tar.gz (3. IW) 
løøs I 
| 3.1% IWkB's 
s 1 : PYY 1 n 
Building wheels for collected packages : keras, theano 
Running setup. py bdist_wheel for keras . 
stored in directory: 
Running setup. py bdist_wheel for theano 
Stored in directory: 
Successfully built keras theano 
Installing collected packages: theano, keras, olefile 
Successfully installed keras-2.ø.3 olefiIe-ø.44 theano-ø.9.ø 
(superman) C: 
(fræ h5py) 
ite-packages (fræ keras) 
WTF is that? 

Theano is yet another machine learning backend like tensorflow. Keras supports them both.

Now we need to double check which one is used on our system.

(superman) C: 
3.5.2 'Anaconda 4.3. I (64-bit)l (default, 
credits" or "license" 
invort eras 
ing TensorFIæ• backend. 
Jul 5 2Ø16, 11:41:13) 
for •re information . 

Nothing to worry about, tensorflow is used as backend, theano ignored. Which is good

Otherwise you'll need to go to edit file ~/.keras/keras.json

And make it look like

```

|  |  |
| --- | --- |
|  | {      "image\_dim\_ordering": "tf",      "epsilon": 1e-07,      "floatx": "float32",      "backend": "tensorflow"  } |

*From <*[*http://www.pyimagesearch.com/2016/11/14/installing-keras-with-tensorflow-backend/*](http://www.pyimagesearch.com/2016/11/14/installing-keras-with-tensorflow-backend/)*>*

```

Sidenote: tensorflow and theano use different ordering of images. (height, image, channel) vs ( channel, height, width)

It can and will at some point create GREAT pain in the butt. Double check dimension ordering.

One more little thing we need to install in opencv library, just because it's cool

```

conda install -c menpo opencv3

*From <*[*https://anaconda.org/menpo/opencv3*](https://anaconda.org/menpo/opencv3)*>*

```

And check that it is working

(superman) C: install 
-c rnpo opencv3 
Fetching package rtadata • • • 
Solving package . 
Package plan for installatim in environ—nt C: 
The following NN packages will be INSTALLED: 
opencv3: 3.1.ø-py35_ø 
Proceed ( [y]/n)? Y 
opencv3-3.1.ø- løø% I 
(s uperman ) 
C : use rs 
Python 3.5.2 IAnaconda 4.3. I (64-bit)l (default, 
ight", "credits" or "license" 
i.ort 
cv2 
1 
I Time: 641.84 kB/s 
Jul 5 2Ø16, 11:41:13) v.19W 
for •re information . 

Yes, you got it right. You install opencv3 and import cv2. That's python. That's why