TensorFlow Basics

In [1]:
```
import tensorflow as tf
```

The downloaded Jupyter Notebook says

> Make sure you are using 1.3 for exact sytnax[sic] matching!

However, the newer env file has `1.15` listed. (Note I'm in Windows at the moment.)

In [2]:
```
! type "C:\\David\\my_repos_dwb\\upgraded-waffle\\FULL_TENSORFLOW_NOTES__AND_DATA\\earlier_nonworking_env_files\\tfdl_env.yml"
```

```
name: tfdeeplearning
channels:
- defaults
dependencies:
- matplotlib=2.0.2
- numpy=1.13.1
- pandas=0.20.3
- python=3.5.4
- scikit-learn=0.19.0
- scipy==1.1.0
- pip:
  - jupyter
  - tensorflow==1.15.0
  - tensorboard==1.15.0
```

OUTPUT

```
name: tfdeeplearning
channels:
- defaults
dependencies:
- matplotlib=2.0.2
- numpy=1.13.1
- pandas=0.20.3
- python=3.5.4
- scikit-learn=0.19.0
- scipy==1.1.0
- pip:
  - jupyter
  - tensorflow==1.15.0
  - tensorboard==1.15.0
```

Yes, yes it does.

And the env file that finally worked without errors to get me here had `1.10`.

That's the one with the instructions,

* **WINDOWS SOLUTION** * Enviornment File Problems
46 upvotes
Adam . Lecture 5
. 3 years ago

A lot of people are getting enviornment file errors, even with the "updated" version

Here is a * **WORKING** * enviornment file

https://pastebin.com/2g6KGH9a (https://pastebin.com/2g6KGH9a)

1) cd to your downloaded folder of notes from this course
2) Create a new .txt file with Notepad call it `tfdl_env.txt`
3) Copy paste the data from the link above into this txt file
4) while in the directory run the command at the first line of the text file, it should be this :)

```
conda create --name tfdeeplearning --file tfdl_env.txt
```

Hope this works for you all!

Make sure you're running this txt file and not the current env file by mistake if you get an error. And just continue on with the instructions:

```
activate tfdeeplearning
```

```
jupyter notebook
```

... etc

Notes:

- you may need to run `conda activate tfdeeplearning`
- for "Section 11: Reinforcement Learning..." I am not sure if gym is installed in this env file. There is already a solution in Lecture 84 that worked for me, simply:
  - `activate tfdeeplearning`
  - `pip install gym`
    - if you get a permission error run `pip install gym --user`

In [3]:
```
! type "C:\\David\\my_repos_dwb\\upgraded-waffle\\FULL_TENSORFLOW_NOTES__AND_DATA\\tfdl_env.txt"
```

```
# $ conda create --name tfdeeplearning --file tfdl_env.txt
# platform: win-64
@EXPLICIT
https://repo.anaconda.com/pkgs/main/win-64/_tflow_select-2.3.0-mkl.conda
https://repo.anaconda.com/pkgs/main/win-64/blas-1.0-mkl.conda
https://repo.anaconda.com/pkgs/main/win-64/ca-certificates-2020.1.1-0.conda
https://repo.anaconda.com/pkgs/main/win-64/icc_rt-2019.0.0-h0cc432a_1.conda
https://repo.anaconda.com/pkgs/main/win-64/intel-openmp-2019.4-245.conda
https://repo.anaconda.com/pkgs/msys2/win-64/msys2-conda-epoch-20160418-1.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/pandoc-2.2.3.2-0.conda
https://repo.anaconda.com/pkgs/main/win-64/vs2015_runtime-14.16.27012-hf0eaf9b_2.conda
https://repo.anaconda.com/pkgs/main/win-64/winpty-0.4.3-4.conda
https://repo.anaconda.com/pkgs/main/win-64/libmklml-2019.0.5-0.conda
https://repo.anaconda.com/pkgs/msys2/win-64/m2w64-gmp-6.1.0-2.tar.bz2
https://repo.anaconda.com/pkgs/msys2/win-64/m2w64-libwinpthread-git-5.0.0.4634.697f757-2.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/mkl-2018.0.3-1.conda
https://repo.anaconda.com/pkgs/main/win-64/vc-14.1-h0510ff6_4.conda
https://repo.anaconda.com/pkgs/main/win-64/icu-58.2-ha925a31_3.conda
https://repo.anaconda.com/pkgs/main/win-64/jpeg-9b-hb83a4c4_2.conda
https://repo.anaconda.com/pkgs/main/win-64/libsodium-1.0.16-h9d3ae62_0.conda
https://repo.anaconda.com/pkgs/msys2/win-64/m2w64-gcc-libs-core-5.3.0-7.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/openssl-1.0.2u-he774522_0.conda
https://repo.anaconda.com/pkgs/main/win-64/python-3.5.4-h1357f44_23.conda
https://repo.anaconda.com/pkgs/main/win-64/tbb-2020.0-h74a9793_0.conda
https://repo.anaconda.com/pkgs/main/win-64/zlib-1.2.11-h62dcd97_4.conda
https://repo.anaconda.com/pkgs/main/win-64/astor-0.7.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/backcall-0.1.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/certifi-2018.8.24-py35_1.conda
https://repo.anaconda.com/pkgs/main/noarch/colorama-0.4.3-py_0.conda
https://repo.anaconda.com/pkgs/main/noarch/decorator-4.4.2-py_0.conda
https://repo.anaconda.com/pkgs/main/noarch/defusedxml-0.6.0-py_0.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/entrypoints-0.2.3-py35_2.conda
https://repo.anaconda.com/pkgs/main/noarch/gast-0.3.3-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/ipython_genutils-0.2.0-py35ha709e79_0.conda
https://repo.anaconda.com/pkgs/main/win-64/libpng-1.6.37-h2a8f88b_0.conda
https://repo.anaconda.com/pkgs/main/win-64/libprotobuf-3.6.0-h1a1b453_0.conda
https://repo.anaconda.com/pkgs/msys2/win-64/m2w64-gcc-libgfortran-5.3.0-6.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/markdown-2.6.11-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/markupsafe-1.0-py35hfa6e2cd_1.conda
https://repo.anaconda.com/pkgs/main/win-64/mistune-0.8.3-py35hfa6e2cd_1.conda
https://repo.anaconda.com/pkgs/main/win-64/pandocfilters-1.4.2-py35_1.conda
https://repo.anaconda.com/pkgs/main/noarch/parso-0.7.0-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/pickleshare-0.7.4-py35h2f9f535_0.conda
```

```
https://repo.anaconda.com/pkgs/main/noarch/prometheus_client-0.7.1-py_0.tar.bz2
https://repo.anaconda.com/pkgs/main/noarch/pyparsing-2.4.7-py_0.conda
https://repo.anaconda.com/pkgs/main/noarch/pytz-2020.1-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/pywin32-223-py35hfa6e2cd_1.conda
https://repo.anaconda.com/pkgs/main/noarch/qtpy-1.9.0-py_0.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/send2trash-1.5.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/simplegeneric-0.8.1-py35_2.conda
https://repo.anaconda.com/pkgs/main/win-64/sip-4.18.1-py35h6538335_2.conda
https://repo.anaconda.com/pkgs/main/win-64/six-1.11.0-py35_1.conda
https://repo.anaconda.com/pkgs/main/win-64/sqlite-3.31.1-h2a8f88b_1.conda
https://repo.anaconda.com/pkgs/main/win-64/tbb4py-2018.0.5-py35he980bc4_0.conda
https://repo.anaconda.com/pkgs/main/win-64/termcolor-1.1.0-py35_1.conda
https://repo.anaconda.com/pkgs/main/noarch/testpath-0.4.4-py_0.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/tornado-5.1.1-py35hfa6e2cd_0.conda
https://repo.anaconda.com/pkgs/main/noarch/wcwidth-0.1.9-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/webencodings-0.5.1-py35_1.conda
https://repo.anaconda.com/pkgs/main/noarch/werkzeug-1.0.1-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/win_unicode_console-0.5-py35h56988b5_0.conda
https://repo.anaconda.com/pkgs/main/win-64/wincertstore-0.2-py35hfebbdb8_0.conda
https://repo.anaconda.com/pkgs/main/win-64/zeromq-4.2.5-he025d50_1.conda
https://repo.anaconda.com/pkgs/main/win-64/absl-py-0.4.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/cycler-0.10.0-py35hcc71164_0.conda
https://repo.anaconda.com/pkgs/main/win-64/freetype-2.8-h51f8f2c_1.conda
https://repo.anaconda.com/pkgs/main/win-64/jedi-0.12.1-py35_0.conda
https://repo.anaconda.com/pkgs/msys2/win-64/m2w64-gcc-libs-5.3.0-7.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/numpy-base-1.15.2-py35h8128ebf_0.conda
https://repo.anaconda.com/pkgs/main/win-64/protobuf-3.6.0-py35he025d50_0.conda
https://repo.anaconda.com/pkgs/main/noarch/python-dateutil-2.8.1-py_0.tar.bz2
https://repo.anaconda.com/pkgs/main/win-64/pyzmq-17.1.2-py35hfa6e2cd_0.conda
https://repo.anaconda.com/pkgs/main/win-64/qt-5.6.2-vc14h6f8c307_12.conda
https://repo.anaconda.com/pkgs/main/win-64/setuptools-40.2.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/traitlets-4.3.2-py35h09b975b_0.conda
https://repo.anaconda.com/pkgs/main/noarch/bleach-3.1.4-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/grpcio-1.12.1-py35h1a1b453_0.conda
https://repo.anaconda.com/pkgs/main/noarch/jinja2-2.11.2-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/jsonschema-2.6.0-py35h27d56d3_0.conda
https://repo.anaconda.com/pkgs/main/noarch/jupyter_core-4.5.0-py_0.conda
https://repo.anaconda.com/pkgs/main/noarch/pygments-2.6.1-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/pyqt-5.6.0-py35ha878b3d_6.conda
https://repo.anaconda.com/pkgs/main/win-64/pywinpty-0.5.4-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/wheel-0.31.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/noarch/jupyter_client-5.3.3-py_0.conda
https://repo.anaconda.com/pkgs/main/noarch/nbformat-5.0.6-py_0.conda
```

```
https://repo.anaconda.com/pkgs/main/win-64/pip-10.0.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/prompt_toolkit-1.0.15-py35h89c7cb4_0.conda
https://repo.anaconda.com/pkgs/main/win-64/terminado-0.8.1-py35_1.conda
https://repo.anaconda.com/pkgs/main/win-64/ipython-6.5.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/noarch/nbconvert-5.5.0-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/ipykernel-4.10.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/jupyter_console-5.2.0-py35_1.conda
https://repo.anaconda.com/pkgs/main/win-64/notebook-5.6.0-py35_0.conda
https://repo.anaconda.com/pkgs/main/noarch/qtconsole-4.7.4-py_0.conda
https://repo.anaconda.com/pkgs/main/win-64/widgetsnbextension-3.4.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/ipywidgets-7.4.1-py35_0.conda
https://repo.anaconda.com/pkgs/main/win-64/jupyter-1.0.0-py35_7.conda
https://repo.anaconda.com/pkgs/main/win-64/matplotlib-2.0.2-py35h9bd10b2_1.conda
https://repo.anaconda.com/pkgs/main/win-64/mkl_fft-1.0.6-py35hdbbee80_0.conda
https://repo.anaconda.com/pkgs/main/win-64/mkl_random-1.0.1-py35h77b88f5_1.conda
https://repo.anaconda.com/pkgs/main/win-64/numpy-1.15.2-py35ha559c80_0.conda
https://repo.anaconda.com/pkgs/main/win-64/pandas-0.20.3-py35he2ce742_2.conda
https://repo.anaconda.com/pkgs/main/win-64/scipy-1.1.0-py35h4f6bf74_1.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorboard-1.10.0-py35he025d50_0.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-base-1.10.0-mkl_py35h81393da_0.conda
https://repo.anaconda.com/pkgs/main/win-64/scikit-learn-0.19.0-py35h3bd3ce1_2.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-1.10.0-mkl_py35h4a0f5c2_0.conda
```

OUTPUT

```
Lots of it
```

Let's make that easier to check.

In [4]:
```
! type "C:\\David\\my_repos_dwb\\upgraded-waffle\\FULL_TENSORFLOW_NOTES__AND_DATA\\tfdl_env.txt" | findstr te
nsor
```

```
https://repo.anaconda.com/pkgs/main/win-64/tensorboard-1.10.0-py35he025d50_0.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-base-1.10.0-mkl_py35h81393da_0.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-1.10.0-mkl_py35h4a0f5c2_0.conda
```

OUTPUT

```
https://repo.anaconda.com/pkgs/main/win-64/tensorboard-1.10.0-py35he025d50_0.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-base-1.10.0-mkl_py35h81393da_0.conda
https://repo.anaconda.com/pkgs/main/win-64/tensorflow-1.10.0-mkl_py35h4a0f5c2_0.conda
```

Yes, yes it does have `1.10` .

In [5]:
```
print(tf.__version__)
```

```
1.10.0
```

OUTPUT

```
1.10.0
```

Moving on

Tensors

Lecture Version

In [6]:
```
first = tf.constant("First ")
```

In [7]:
```
steps = tf.constant("Steps")
```

In [8]:
```
type(first) # a tensor object
```

Out[8]:
```
tensorflow.python.framework.ops.Tensor
```

In [9]:
```
print(first) # not printing "first"
```

```
Tensor("Const:0", shape=(), dtype=string)
```

In [10]:
```
with tf.Session() as sess:
    sess.run(first + steps)
##endof:  with tf.Session() as sess
```

In [11]:
```
# Let's save it
with tf.Session() as sess:
    result = sess.run(first + steps)
##endof:  with tf.Session() as sess
```

In [12]:
```
print(result)
```

```
b'First Steps'
```

Course Notes Version

In [13]:
```
hello = tf.constant('Hello')
```

In [14]:
```
type(hello)
```

Out[14]:
```
tensorflow.python.framework.ops.Tensor
```

In [15]:
```
world = tf.constant('World')
```

In [16]:
```
result_CNV = hello + world
```

In [17]:
```
result_CNV # we have already done an add, so we get 'add_2' instead of 'add'
```

Out[17]:
```
<tf.Tensor 'add_2:0' shape=() dtype=string>
```

In [18]:
```
type(result_CNV)
```

Out[18]:
```
tensorflow.python.framework.ops.Tensor
```

In [19]:
```
with tf.Session() as sess_CNV:
    result_CNV = sess_CNV.run(hello + world)
```

In [20]:
```
result_CNV
```

Out[20]:
```
b'HelloWorld'
```

Computations - Lecture Version

In [21]:
```
a = tf.constant(10)
```

In [22]:
```
b = tf.constant(20)
```

In [23]:
```
type(a)
```

Out[23]:
```
tensorflow.python.framework.ops.Tensor
```

In [24]:
```
a + b
```

Out[24]:
```
<tf.Tensor 'add_4:0' shape=() dtype=int32>
```

In [25]:
```
a + b
  #  Note that we get 'add_5:0' this time,
  #+ instead of 'add_4:0' last time. TF is
  #+ keeping track, somehow.
```

Out[25]:
```
<tf.Tensor 'add_5:0' shape=() dtype=int32>
```

In [26]:
```
a + b #  still keeping track of how many times you're asking for it.
      #+ No answer, b/c not in a session.
```

Out[26]:
```
<tf.Tensor 'add_6:0' shape=() dtype=int32>
```

In [27]:
```
with tf.Session() as sess:
    result = sess.run(a + b)
##endof:  with tf.Session() as sess
```

In [28]:
```
result
```

Out[28]:
```
30
```

Computations - Course Notes Version

In [29]:
```
tensor_1 = tf.constant(1)
tensor_2 = tf.constant(2)
```

In [30]:
```
type(tensor_1)
```

Out[30]:
```
tensorflow.python.framework.ops.Tensor
```

In [31]:
```
tensor_1 + tensor_2
```

Out[31]:
```
<tf.Tensor 'add_8:0' shape=() dtype=int32>
```

In [32]:
```
sess_CNV
```

Out[32]:
```
<tensorflow.python.client.session.Session at 0x238f05a7b70>
```

In [33]:
```
sess_CNV.close()
```

In [34]:
```
## DWB here
sess_CNV
```

Out[34]:
```
<tensorflow.python.client.session.Session at 0x238f05a7b70>
```

Operations

Lecture Version

In [35]:
```
const = tf.constant(10)
```

In [36]:
```
fill_mat = tf.fill((4, 4), 10)
```

In [37]:
```
myzeros = tf.zeros((4, 4))
```

In [38]:
```
myones = tf.ones((4, 4))
```

In [39]:
```
myrandn = tf.random_normal((4, 4), mean=0, stddev=1.0)
```

In [40]:
```
myrandu = tf.random_uniform((4, 4), minval=0, maxval=1)
```

In [41]:
```
myzeros
```

Out[41]:
```
<tf.Tensor 'zeros:0' shape=(4, 4) dtype=float32>
```

In [42]:
```
my_ops = [const, fill_mat, myzeros, myones, myrandn, myrandu]
```

Course Notes Version

All the same, except it has `stddev=0.5` instead of `stddev=1`

Interactive Session

He says,

> Useful for Notebook Sessions

Note that this, **Lecture Notes Version** is identical to the **Course Notes Version**

ONLY RUN THIS NEXT CELL ONCE!

In [43]:
```
# Only run this cell once
sess = tf.InteractiveSession()
```

In [44]:
```
#  With the interactive session, it's like everything
#+ is inside a `with tf.Session() as sess`

for op in my_ops:
    print(sess.run(op)) # there can also be sess.eval(op)
    print('\n')
##endof:  for op in my_ops
```

```
10


[[10 10 10 10]
 [10 10 10 10]
 [10 10 10 10]
 [10 10 10 10]]


[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]


[[1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]]


[[-0.3720909  -1.3475986  -1.278251   -1.4621896 ]
 [-1.3915237  -1.2305956  -1.4991359   1.399201  ]
 [ 0.10715739  0.4901729   0.42736274 -1.615327  ]
 [-0.5372319   1.204179    0.3970663   0.05627497]]


[[0.917171   0.7303493  0.33320653 0.40524447]
 [0.9164847  0.025437   0.9932685  0.48114038]
 [0.8789121  0.37492216 0.02805054 0.74665654]
 [0.0600822  0.19667196 0.05904555 0.1681658 ]]
```

My random matrices are different, since I don't know his seed, and I haven't been doing this right after the last lesson - I've done stuff in between, include restarting the kernel.

Matrix Multiplication

Hoping the Lecture and Course-Notes Versions are the same.

In [45]:
```
a = tf.constant([ [1,2],
                  [3,4] ])
```

In [46]:
```
a.get_shape()
```

Out[46]:
```
TensorShape([Dimension(2), Dimension(2)])
```

In [47]:
```
b = tf.constant([ [10], [100] ])
```

In [48]:
```
b.get_shape()
```

Out[48]:
```
TensorShape([Dimension(2), Dimension(1)])
```

In [49]:
```
result = tf.matmul(a, b)
```

In [50]:
```
# Lecture Version
sess.run(result) # needs the interactie session
```

Out[50]:
```
array([[210],
       [430]])
```

In [51]:
```
# Course notes version (and Lecture version)
result.eval()
```

Out[51]:
```
array([[210],
       [430]])
```

There's a demonstration of the sess.run() and the .eval() stuff he was discussing in the lecture.

That's all for now!