input = T.tensor4('inputs')

对的，就是上面说的高维数组，计算机的人喜欢发明新词汇，比较好听而已。。。   
  
另外theano里面其实是这么分的  
  
from theano import tensor as T   
  
T 里面其实有scalar （一个数据点），vector (向量），matrix (矩阵），tensor3 (三维矩阵)，tensor4 （四位矩阵）。。。其实这些都落入tensor的范畴。。。   
  
另外到了四位矩阵之后就没有更高维了，当然这个是由theano在图像中的大量运用决定的，如果你有了什么更高维的问题，那么就可以为开源做贡献咯~ o(^▽^)o

作者：彭河森  
链接：https://www.zhihu.com/question/43178323/answer/98808313  
来源：知乎  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

这里的tensor是矩阵分析里面的张量，表示高维数组，或者叫多路数据。二阶张量就是矩阵，一阶张量就是向量。  
物理里面的张量，一般称为张量场。  
可以参阅张贤达《矩阵分析》红宝书，学一学基本的tucker分解和CP分解。  
  
记住一个结论，高阶SVD相对于矩阵SVD更能提取出高维数据的内在结构。

作者：晚上吃白片  
链接：https://www.zhihu.com/question/43178323/answer/99162742  
来源：知乎  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

大家都已经说的比较清楚了最近看了theano和TensorFlow的文档，对也有了新的理解，所以自己也答一下。tensor可以表示一个占位符，**包含的是表达式变量的类型的信息。**这个类型可以是数据点，向量，矩阵等等。



仅显示为一个占位符。

对的，其实用tensor这个概念只是因为theano 会在后台把python定义翻译成c语言再编译（如果我没理解错的话），然后这些维度有多少维都是hard code进去的，所以必须前台在python里面定义好

theano.tensor.scalar(*name=None*, *dtype=config.floatX*)

Return a Variable for a 0-dimensional ndarray

theano.tensor.vector(*name=None*, *dtype=config.floatX*)

Return a Variable for a 1-dimensional ndarray

theano.tensor.row(*name=None*, *dtype=config.floatX*)

Return a Variable for a 2-dimensional ndarray in which the number of rows is guaranteed to be 1.

theano.tensor.col(*name=None*, *dtype=config.floatX*)

Return a Variable for a 2-dimensional ndarray in which the number of columns is guaranteed to be 1.

theano.tensor.matrix(*name=None*, *dtype=config.floatX*)

Return a Variable for a 2-dimensional ndarray

theano.tensor.tensor3(*name=None*, *dtype=config.floatX*)

Return a Variable for a 3-dimensional ndarray

theano.tensor.tensor4(*name=None*, *dtype=config.floatX*)

Return a Variable for a 4-dimensional ndarray

cnn = lasagne.layers.InputLayer(  
 shape=(None, 3, 32, 32),  
 input\_var=input)

**Network input**

*class* lasagne.layers.InputLayer(*shape*, *input\_var=None*, *name=None*, *\*\*kwargs*)[[source]](https://github.com/Lasagne/Lasagne/blob/master/lasagne/layers/input.py#L16-L75)

This layer holds a symbolic variable that represents a network input. A variable can be specified when the layer is instantiated, else it is created.

|  |  |
| --- | --- |
| **Parameters:** | **shape** : tuple of *int* or *None* elements  The shape of the input. Any element can be *None* to indicate that the size of that dimension is not fixed at compile time.  **input\_var** : Theano symbolic variable or *None* (default: *None*)  A variable representing a network input. If it is not provided, a variable will be created. |
| **Raises:** | **ValueError**  If the dimension of *input\_var* is not equal to *len(shape)* |

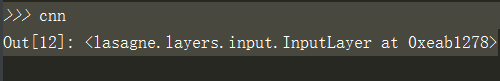
Notes

The first dimension usually indicates the batch size. If you specify it, Theano may apply more optimizations while compiling the training or prediction function, but the compiled function will not accept data of a different batch size at runtime. To compile for a variable batch size, set the first shape element to *None* instead.

Examples

>>> from lasagne.layers import InputLayer

>>> l\_in = InputLayer((100, 20))



>>> cnn

Out[12]: <lasagne.layers.input.InputLayer at 0xeab1278>