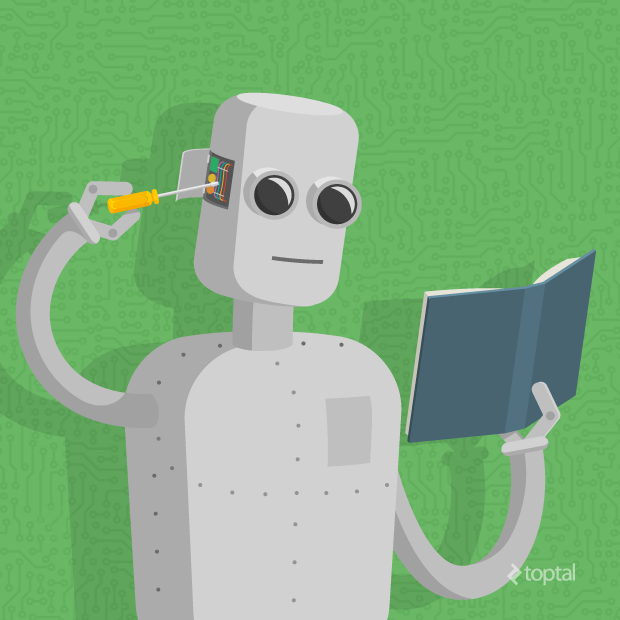
# mltest：在一个函数调用中自动测试神经网络模型



所以我在上一篇文章中得到了很多积极的反馈。有几个人直接给我发消息说，他们在自己的代码中发现了一个bug，推荐了一些测试，真是太棒了！但这些问题仍然太常见，忘记编写测试和首先编写bug一样容易。我们需要一个更好、更自动化的解决方案。

这就是为什么我们要介绍

过来看！

完成。由于几乎没有设置，我们现在正在测试几个不同的常见机器学习问题。

要安装它，只需运行：

函数调用mltest.test\_suite（…）是这个库的主要功能部件。它运行几个测试，包括：

### 一。变量变化

对人们帮助最大的测试是变量变化测试。基本上，运行train-op并确保列表或范围内的所有变量都被修改。

### 2。变量不变

还可以确保只有作用域或列表中的变量才是更改的变量，其他变量不会更改。这对于GAN训练是非常有用的，因为生成器和鉴别器通常有不同的训练操作。

### 三。登录范围

我会犯的一个常见错误是在logits输出中添加非线性。这通常会导致很多问题，一旦它击中softmax。我们的test\_suite（）会自动检查logits层的值是否高于或低于0。这也许不是检测它的最佳方法，但它有助于缩小问题的范围。

您还可以将要检查的范围设置为您想要的任何范围。假设您希望在logits上有一个tanh，您可以设置一个检查以确保logits的所有值都在（-1，1）中。

### 四。输入依赖项

一个常见的问题是，有时人们会忘记将网络的各个分支连接在一起。不管是忘记将两个张量相加，还是忘记为某个分支调用函数调用。一个网络仍然可以训练和收敛不良，只有部分输入，所以它是重要的，以确保您的所有输入值都依赖于训练操作。

Of course, any of these tests can be turned off manually with flags in test\_suite(). See the code for documentation on how to do this.

### mltest setup

Another useful feature is mltest.setup().

This call will automatically reset the default tensorflow graph and set tensorflow’s, numpy’s, and python’s random seeds. It’s very easy to forget to seed your random values, and can cause a massive headache when trying to recreate bugs.

This suite is still in beta, so if you have any requests or notice any bugs, please add an issue on Github! Also, I accept pull requests. ;)