

大作业

环境配置

大作业需要安装的配置：jupyter notebook, python3.7。大作业需要安装的库已经放在了requirements.txt文件中，进入你的python环境，运行

```
cd final-project
pip install -r requirements.txt
```

安装即可，强烈建议使用anaconda新建一个python3.7的虚拟环境，在环境中安装避免不必要的麻烦。创建虚拟环境的教程：<https://blog.csdn.net/lyy14011305/article/details/59500819>

除了requirements.txt中的必要库，还需要运行

```
conda install -n your-environment-name libpython
conda install -n your-environment-name -c msys2 m2w64-toolchain
```

这是编译cpython文件需要的库，因为卷积神经网络需要有效的实现，运行所需的函数都使用cpython写好了，在使用之前还需要进入setup.py所在文件夹，使用运行如下指令进行编译：

```
python setup.py build_ext --inplace
```

数据集需要下载并解压到 annp/dataset/ 文件夹下。

内容

全连接神经网络（15分）

依照 FullConnectedNetwork.ipynb 中的要求：

1. 实现affine layer的前向传播和反向传播（3分）
2. 实现ReLU激活函数的前向传播和反向传播，并在jupyter notebook上回答问题1（3分）
3. 利用你实现的affine layer和ReLU激活函数构建一个两层的全连接神经网络（3分）
4. 训练你实现的两层全连接神经网络，使测试结果的准确率达到50%以上（3分）
5. 构建多层的全连接网络，满足FullConnectedNetwork.ipynb中的测试要求（3分）

归一化（10分）

依照 BatchNormalization.ipynb 中的要求：

1. 实现batch normalization的前向传播和反向传播
2. 修改你之前实现的全连接神经网络，添加batch normalization，回答问题1
3. 探究batch normalization和batch size的关系，回答问题2

4. 实现layer normalization的前向传播和反向传播，并将layer normalization添加到你之前实现的全连接神经网络中
5. 探究layer normalization和batch size的关系，回答问题3

CNN (20分)

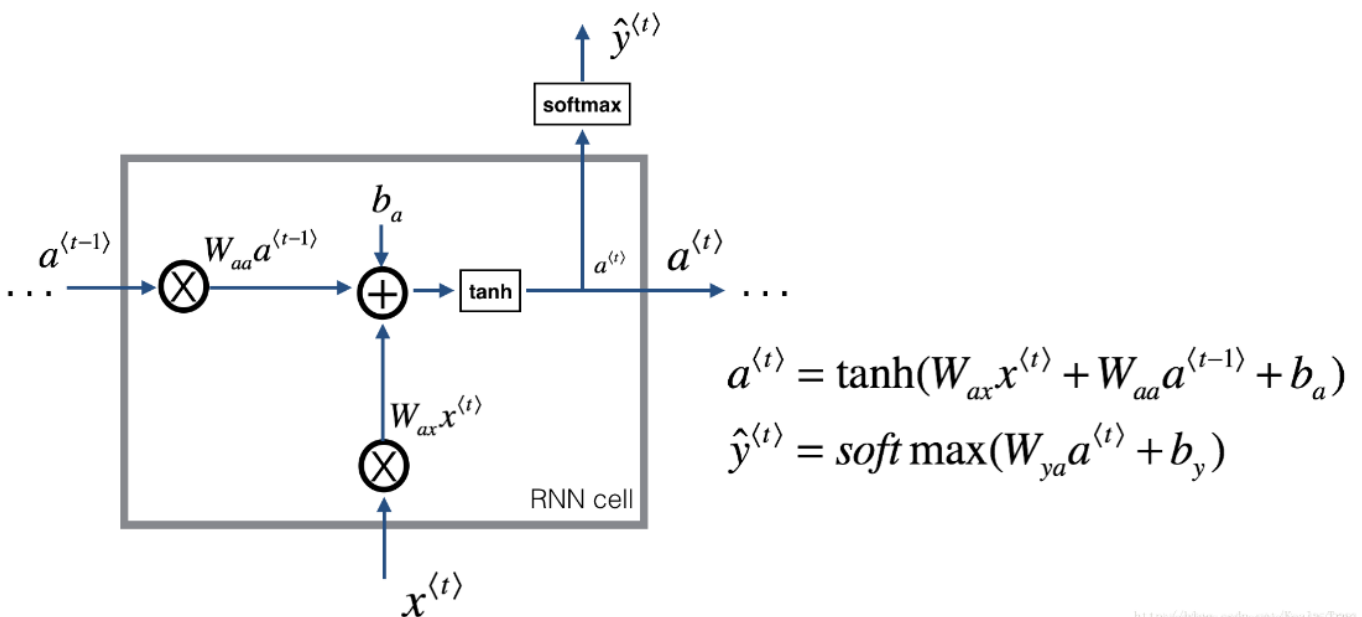
依照 ConvolutionalNetwork.ipynb 中的要求：

1. 实现CNN的前向传播和反向传播
2. 实现max pooling的前向传播和反向传播
3. 实现一个三层卷积神经网络
4. 实现spatial batch normalization

RNN (20分)

依照 RecurrentNetwork.ipynb 中的要求：

1. 使用numpy构建tanh与softmax函数
2. 构建单个RNN单元：input-hidden-output，并测试输出
3. 根据指引构建RNN网络，并测试输出
4. 根据指引构建RNN单元与RNN网络的反向传播函数，并测试输出



实现ConvNet (35分)

根据 ConvolutionalNetwork.ipynb 中 Train your best model 中的要求，利用annp文件夹中的模块实现用于分类cifar-10数据集的卷积神经网络。需要注意的是，只能用annp文件夹中的模块实现你的模型，不允许使用额外的深度学习框架，请在annp/classifiers/cnn.py中实现你的模型，在jupyter notebook对应位置实现你的训练过程，实验结果以及可视化分析。请各位同学仔细阅读annp文件夹中每个模块的用法。

实验报告

整理你实现的ConvNet，写一份实验报告描述你的模型架构，调参的过程，分析实验结果以及不同的参数对实验结果的影响，最好是对实验结果进行可视化的分析。实验报告占20分。

需要提交的文件 1. 你实现的代码，包括annp中的代码和jupyter notebook的代码。 2. 你的实验报告。大作业截至时间为第十九周周一晚上12点。将上述文件打包，命名格式为“姓名+学号.zip”发到助教邮箱：