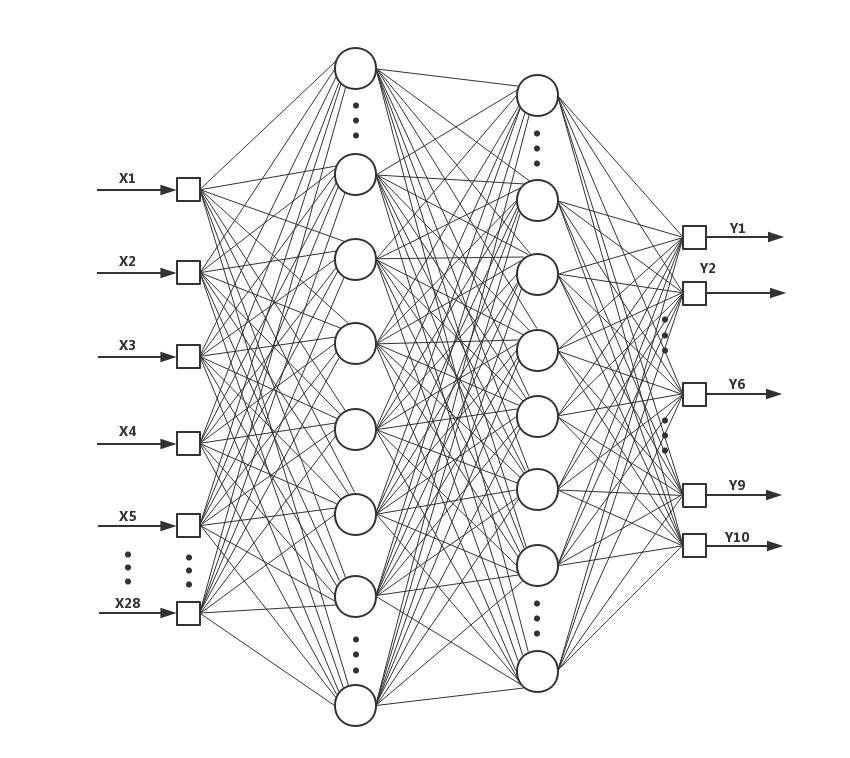
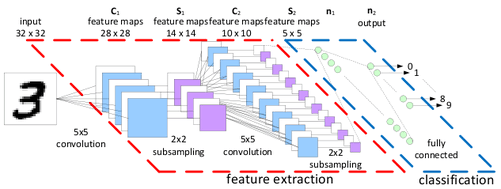
1. 程序中用python实现了神经网络对mnist数据集的分类，分别是train.py和train\_LeNet5.py,其中train.py中实现了一个3层的神经网络，采用交叉熵作为代价函数进行随机梯度下降，结构图如图1所示；而train\_LeNet5.py中使用了卷积神经网络，相比于普通的神经网络，需要训练的参数大大的增加，导致训练速度较慢，结构图如图2所示（由于卷积神经网络的结构图比较难画，所以在网上找了一张比较合适的示意图）



**图1.三层的神经网络**

****

**图2.卷积神经网络（LeNet5）**

1. 代码文件标注

运行train.py（三层神经网络）或train\_LeNet.py运行便开始训练

Inference.py：从网上获取并存储mnist数据集

Layerpart.py：定义神经网络layer的类和激活函数ReLU,Tanh等

Plugin.py：定义了一些小组件的脚本，计算Loss值，画出Loss变化的图表，随机获取一个batchsize的数据

network.py：搭建神经网络的类

lossfunction.py：定义代价函数，如交叉熵等

optimizer：进行随机梯度下降的脚本

test\_set.py：打印准确率的脚本

mnist.pkl：存放mnist数据集的字典的pickle文件

ModelParams.pkl：存放训练出来的神经网络模型的数据结构的pickle文件