# 《认知科学与类脑计算》

# 实验参考手册

版本：2019年3月10日

课程教师：吴强

山东大学

## 第一部分：《认知科学与类脑计算》实验大纲

### 课程背景

认知神经科学与类脑计算是新兴的交叉研究领域，它作为探究人脑或心智工作机制和新型人工智能工具的前沿性尖端学科，已经引起了全球研究者的广泛关注。

本课程较系统地介绍了认知科学的信息处理机理和类脑计算中的常用算法，这将为信息科学与人工智能相关专业同学的学习以及软件设计水平的提高提供了良好的训练过程。《认知科学与类脑计算》课程内容丰富，具有学科交叉的特点，所用到的背景知识较多，为了学生能够充分理解课程内容，设置《认知科学与类脑计算》的实践环节十分重要。通过实验实践内容的训练，训练学生的类脑机理的程序实现，提高学生编写类脑计算及大型程序的能力。

### 课程实验目的

本实验课程目的是使学生不仅能够深刻理解课堂教学内容，进一步提高运用脑机理和类脑计算算法设计能力，而且可以在大型的综合实验中的总体结构设计、数据流处理等技能方面得到充分训练。实验设置着眼于原理与应用的结合点，使学生学会如何把书本上和课堂上学到的知识用于解决实际问题，从而培养类脑计算工作所需要的动手能力。

为了帮助学生更好地学习本课程，理解和掌握算法设计所需的技术，为整个专业学习打好基础，要求运用所学知识，上机解决一些典型问题，通过分析、设计、编码、调试等各环节的训练，使学生深刻理解、牢固掌握所用到的一些技术。认知科学与类脑计算中稍微复杂一些的算法设计可能同时要用到多种技术和方法，这就要求学生在掌握基本算法的基础上，掌握分析、解决实际问题的能力。

### 课程实验要求

1、阅读实验参考书

每一次实验从阅读实验参考书开始。对于本次实验的实验目的、实验题目、实现提示以及思考题目、选做题目等应认真了解。

2、算法设计

分析实验题目，参考实现提示，进行算法设计。

3、程序设计

根据已完成的算法，用python语言进行程序设计。

4、调试和测试

将所编程序在计算机上调试通过，并选取若干组测试数据对程序进行尽可能全面的测试。

5、整理完成实验报告

实验报告一般包括下列内容：

1. 实验者姓名、学号、专业和班级，课程名称，实验日期等；
2. 本次实验的实验编号及实验名称
3. 本次实验的实验目的；
4. 本次实验的硬件及软件环境；
5. 程序结构的描述及各模块的规格说明；
6. 主要算法及其基本思想；
7. 调试过程简述（调试过程是否顺利，遇到些什么问题，如何解决的，以及上机操作所花费的时间等）；
8. 测试数据和相应输出的客观纪录，对运行结果的分析讨论。

### 课程实验环境及背景介绍

Python是一种[计算机程序设计语言](https://baike.baidu.com/item/%E8%AE%A1%E7%AE%97%E6%9C%BA%E7%A8%8B%E5%BA%8F%E8%AE%BE%E8%AE%A1%E8%AF%AD%E8%A8%80/7073760)。是一种动态的、面向对象的脚本语言，最初被设计用于编写自动化脚本(shell)，随着版本的不断更新和语言新功能的添加，越来越多被用于独立的、大型项目的开发。Python的创始人为荷兰人吉多·范罗苏姆，由于Python语言的[简洁](https://baike.baidu.com/item/%E7%AE%80%E6%B4%81)性、易读性以及可扩展性，在国外采用Python做科学计算的研究机构日益增多，一些知名大学已经采用Python来教授程序设计[课程](https://baike.baidu.com/item/%E8%AF%BE%E7%A8%8B)。Python在设计上坚持了清晰划一的风格，这使得Python成为一门易读、易维护，并且被大量用户所欢迎的、用途广泛的[语言](https://baike.baidu.com/item/%E8%AF%AD%E8%A8%80/2291095)。由于这些得天独厚的优点，Python语言在类脑计算领域得到广泛的使用。

PyTorch是一种简洁优雅且高效快速的深度学习框架。PyTorch在灵活性、易用性、速度这三个方面具有明显的优势。2017年1月，Facebook人工智能研究院（FAIR）团队在GitHub上开源了PyTorch，并迅速占领GitHub热度榜榜首。作为一个2017年才发布，具有先进设计理念的框架，PyTorch的历史可追溯到2002年就诞生于纽约大学的Torch。Torch使用了一种不是很大众的语言Lua作为接口。考虑到Python在计算科学领域的领先地位，以及其生态完整性和接口易用性，几乎任何框架都不可避免地要提供Python接口。2017年，Torch团队推出了PyTorch。PyTorch不是简单地封装Lua Torch提供Python接口，而是对Tensor之上的所有模块进行了重构，并新增了最先进的自动求导系统，成为当下最流行的动态图框架。PyTorch一经推出就立刻引起了广泛关注，并迅速在研究领域流行起来。

认知科学与类脑计算实验常用的Python包有numpy，sklearn等。NumPy是Python的一种开源的数值计算扩展。这种工具可用来存储和处理大型矩阵，比Python自身的嵌套列表（nested list structure)结构要高效的多（该结构也可以用来表示矩阵（matrix））。scikit-learn是数据挖掘与分析的简单而有效的工具，依赖于NumPy，[SciPy](https://www.baidu.com/s?wd=SciPy&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd)和matplotlib。

课程实验常用的数据库有：MNIST和CIFAR。

MNIST 数据集来自美国国家标准与技术研究所, National Institute of Standards and Technology (NIST)。训练集 (training set) 由来自 250 个不同人手写的数字构成, 其中50%是高中学生, 50%来自人口普查局(the Census Bureau)的工作人员。测试集(test set)也是同样比例的手写数字数据。训练库有60,000张手写数字图像，测试库有10,000张手写数字图像。训练库train-images.idx3-ubyte，文件大小47040016，每个图像拥有784=28×28个像素，则训练库共有784×60000 =47040000个像素；测试库t10k-images.idx3-ubyte，文件大小7840016，该测试库拥有784×10000=7840000个像素。

CIFAR-10和CIFAR-100是带有标签的数据集，都出自于规模更大的一个数据集，这个规模更大的数据集有八千万张小图片。CIFAR-10数据集由10个类的60000个32x32彩色图像组成，每个类有6000个图像。CIFAR-10数据集共有50000个训练图像和10000个测试图像。CIFAR-10数据集分为五个训练批次和一个测试批次，每个批次有10000个图像。测试批次包含来自每个类别的恰好1000个随机选择的图像。训练批次以随机顺序包含剩余图像，但一些训练批次可能包含来自一个类别的图像比另一个更多。总体来说，五个训练集之和包含来自每个类的正好5000张图像。CIFAR-10数据集的10个类分别是：飞机、汽车、鸟、猫、鹿、狗等。CIFAR-100数据集与CIFAR-10类似，它有100个类，每个类包含600个图像。每类各有500个训练图像和100个测试图像。CIFAR-100中的100个类被分成20个超类。每个图像都带有一个“精细”标签（它所属的类）和一个“粗糙”标签（它所属的超类），如：超类-水生哺乳动物；类别-海狸、海豚、水獭、海豹、鲸鱼。

### 《认知科学与类脑计算实验》考核方式

采用出勤情况、上机情况、程序质量、实验报告相结合的形式。

## 第二部分：《认知科学与类脑计算》实验步骤和实验报告规范

### 实验步骤

为了规范课程实验的流程，充分理解课堂的讲述内容，完成实验的5个步骤如下：

1. 问题分析和任务定义

实验题目的陈述比较简洁，因此，在进行设计之前，首先应该充分地分析和理解问题，明确问题要求做什么，限制条件是什么。注意：本步骤强调的是做什么，而不是怎么做，对所需完成的任务要作出明确的回答。这一步还应该为调试程序准备好测试数据。

1. 系统设计

在系统设计这一步骤中需根据问题描述划分子任务，定义主程序模块和各子模块。在编写程序之前，应该先写出各过程和函数的伪码算法。在这个过程中，要综合考虑系统功能，使得系统结构清晰、合理、简单和易于调试。作为逻辑设计的结果，应写出各个主要模块的算法，并画出模块之间的调用关系图。详细设计的结果是对主程序模块和各函数体模块规格说明作出进一步的求精，写出各函数体模块的定义，按照算法书写规范用Python语言写出过程或函数形式的算法框架。在求精的过程中，应尽量避免陷入语言细节，不必过早表述函数结构和局部变量。

1. 编码实现和静态检查

编码是把详细设计的结果进一步求精为程序设计语言程序。如何编写程序才能较快地完成调试是特别要注意的问题。要控制 if 语句连续嵌套的深度，分支过多时应考虑使用switch 语句。对函数功能和重要变量进行注释。一定要按格式书写程序，分清每条语句的层次，对齐括号，这样便于发现语法错误。

在正式调试之前，应该写出详细的程序代码，并做认真地静态检查。上机动态调试不能代替静态检查。静态检查主要有两种方法，一是用一组测试数据手工执行程序（通常应先检查单个模块）；二是通过阅读或给别人讲解自己的程序而深入全面地理解程序逻辑，在这个过程中再加入一些注解。

1. 上机准备和上机调试

上机准备包括以下几个方面：

1. 熟悉 Python语言用户手册或程序设计指导书；
2. 熟悉机器的操作系统和语言集成环境的用户手册，尤其是最常用的命令操作，以便顺利进行上机的基本活动；
3. 掌握调试工具， 考虑调试方案， 设计测试数据并手工得出正确结果。

上机调试程序时要带一本Python语言教材或手册。调试最好分模块进行，自底向上，即先调试低层过程或函数。必要时可以另写一个调用驱动程序。这种表面上麻烦的工作实际上可以大大降低调试所面临的复杂性，提高调试工作效率。调试正确后，认真整理源程序及其注释，印出带有完整注释的且格式良好的源程序清单和结果。

1. 总结和整理实验报告

### 二、实验报告规范

## 实习报告的开头应给出题目、班级、姓名、学号和完成日期，并包括以下 7个内容：需求分析、概要设计、详细设计、调试分析、测试结果、附录等。