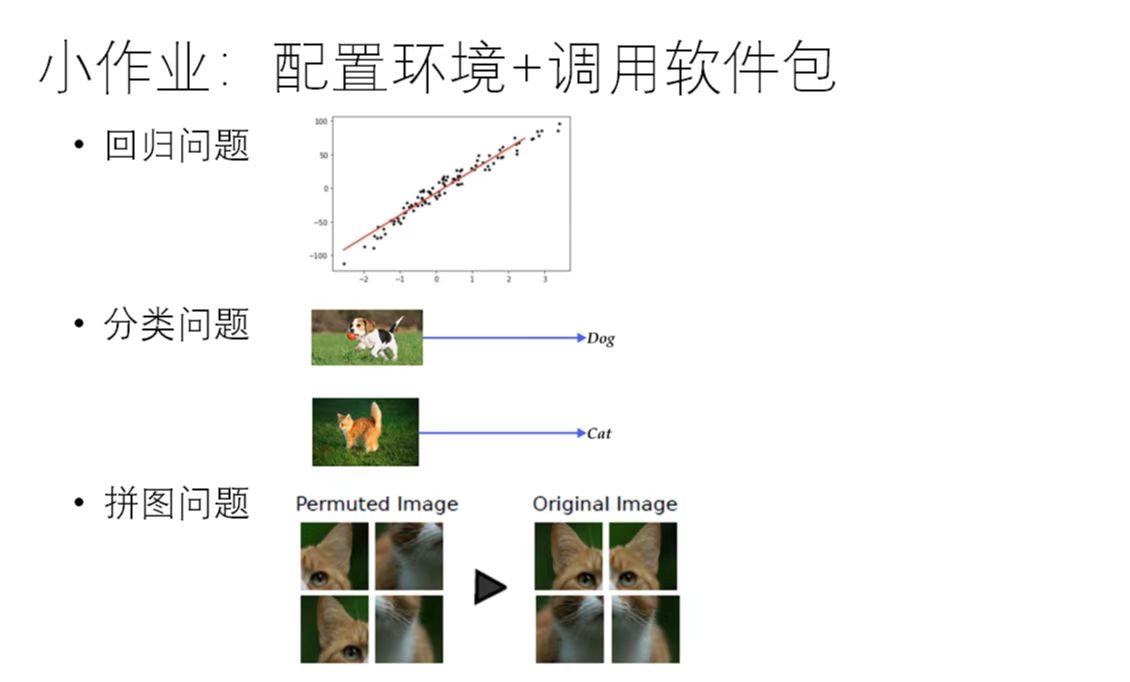
深度学习课程作业拟定

* **小作业（30分）**



本课程的小作业旨在是同学们熟悉并入门国产的开源深度学习框架（Jittor, Paddle, Mindspore），包括但不限于安装框架，配置深度学习环境，设计实现简单的深度学习模型，按照问题需求对模型进行修改等。目前计划安排小作业三次，难度逐步增加。同学们需要在DDL前将三个小作业相关的文件（包括报告和代码）一起打包上传到Canvas上。（超过DDL十天以内，晚交会扣除一定的分数，每晚交一天扣一分，作业超过十天则不再收。）

**第一次小作业：回归问题（15分）**

目标：

1. 安装国产深度学习框架 Jittor，Mindspore 或 Paddle （三选一，推荐Jittor）
2. 配置工作环境
3. 训练一个简单的模型，目标是拟合一个多项式函数

安装教程：

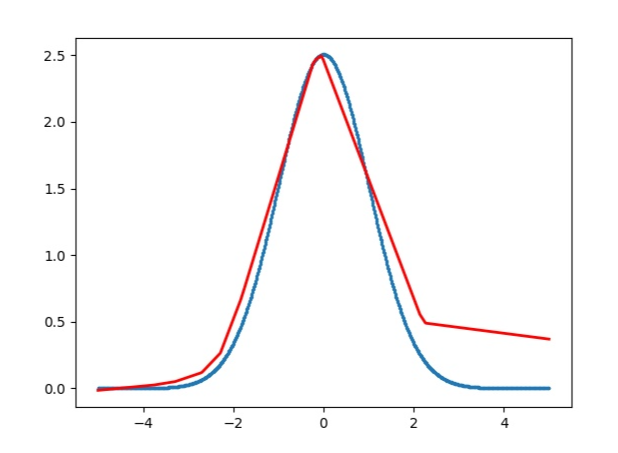
1. Jittor（计图）：<https://cg.cs.tsinghua.edu.cn/jittor/download/>

2. Mindspore（昇思） : <https://www.mindspore.cn/install>

3. Paddle（飞桨）: <https://www.paddlepaddle.org.cn/install/quick>

任务：

自行给定一个函数f(x)(例如正态分布函数，三次函数)，自行确定定义域x的范围，在这一区间内随机采样1000个点，其中800个数据点作为训练集，200个数据点作为测试集。

自行设计一个深度学习模型（例如多层感知机）尝试拟合f(x)，并将结果作图以供展示。下面展示了一个函数拟合的例子：

提交作业时上传全部的python代码，与拟合的效果图。在DDL前发送至canvas上。

**第二次小作业：分类问题（15分）**

目标：

1. 下载并学习使用CIFAR-10数据集
2. 搭建深度学习训练框架 Jittor，Mindspore 或 Paddle （三选一，推荐Jittor），并设计深度学习模型
3. 在给定的要求下改进模型

CIFAR-10数据集下载：https://www.cs.toronto.edu/~kriz/cifar.html

任务：

在国产深度学习框架（Jittor，Mindspore 或 Paddle （三选一，推荐Jittor）中，利用神经网络解决CIFAR-10图像分类这一任务。我们不会对得到的精度做出要求，可以选择任意网络进行实现。实现初步的图像分类任务后，我们额外增加了一个关于数据分布不均匀情况下的任务，具体要求如下：

1. 构建神经网络（具体网络不限，如CNN等）在CIFAR-10数据集上训练测试（自行划分训练集测试集），并汇报结果。
2. 将训练集进行重新划分，具体做法为所有类别为（0, 1, 2, 3, 4）的图像仅保留10%，剩余部分不变。
3. 在新的训练集上训练神经网络，这时结果相较于原先会有下降，那么，提出可能的解决方案，尝试改进你的模型。解决方案从各种角度出发都可以，言之有理即可。
4. 将你在新训练集上的分析，模型设计改进，连同实验结果一起，写成pdf报告的形式，在DDL前与代码一起上传到Canvas上。（报告限制pdf两页）

训练集切分方式示例：

if i % 10 != 0:

mask = (labels >= 5)

else:

mask = (labels >= 0)

labels = labels[mask]

images = images[mask]

* 大作业（70分）

本次深度学习课程大作业是针对基于深度学习的图匹配算法进行研究与探索。图匹配（Graph Matching）是机器学习、计算机科学、运筹优化等学科中的一项基础性问题，图匹配通过构建两个图相似度关系的优化问题，求解图和图之间的匹配关系，进而辅助下游应用。自2018年以来，作为一个新兴的研究方向，国内外学者已经在TPAMI、ICML、NeurIPS、ICLR、CVPR、ICCV、ECCV、EMNLP、ACL、AAAI、IJCAI等顶会、顶刊上发表了数十篇深度学习图匹配相关的学术论文。本次深度学习课程大作业基于“视觉”与“图”两大深度学习的重点研究方向，内容设计意在让同学完整、系统地了解深度学习代码中训练、测试、模型搭建、数据集加载等必备功能，掌握如何编写一份常见的深度学习训练测试代码。

以深度图匹配作为主题，本课程的大作业共有3个任务可供选择：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 任务简述 | 深度学习框架 | 考核内容 | 满分 |
| 任务A | 复现开源的深度图匹配算法 | pytorch | 1. 实验报告（50%） 2. Presentation（20%） | 80% |
| 任务B | 使用国产框架完成深度学习训练、测试 | jittor/paddle/mindspore | 1. 源代码（40%） 2. 实验报告（40%） 3. Presentation（20%） | 100% |
| 任务C | 开放性科研探索【待定】 | tensorflow/pytorch/jittor/paddle/mindspore | 1. 源代码（40%） 2. 小论文（40%） 3. Presentation（20%） | 100% |

# 任务A：复现已开源的深度图匹配算法

## A.1任务简述

本任务为基础任务，旨在让同学了解如何采用已深度学习模型的预测。阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Introduction and Guidelines”章节，学习图匹配的基础知识，了解图匹配的主要步骤。阅读在线文档的“Get Started”章节，在深度学习实验平台上安装软件包并配置PyTorch环境。阅读神经网络图匹配算法的[文档介绍](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.neural_solvers.html)，同时以原论文作为参考，了解对应的算法原理。基于在线文档中的例子（[使用深度学习进行图像匹配](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html#sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)），下载并运行代码，复现在线文档中的测试结果。撰写至少4页的实验报告，内容包括配置环境、对算法的理解、训练测试结果、遇到的问题和解决方式等；准备一个3分钟的pre，在15-16周课堂上进行一次展示。

## A.2 主要步骤

1. 阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Introduction and Guidelines”章节

2. 阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Get Started”章节，在深度学习实验平台上安装软件包并配置PyTorch环境

3. 阅读神经网络图匹配算法的[文档介绍](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.neural_solvers.html)，仔细阅读论文并深入理解PCA-GM, IPCA-GM和CIE算法

4. 运行实验，复现[样例](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html" \l "sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)中的测试结果（网页最下方有代码下载链接）

5. 撰写报告，准备展示

## A.3 评分标准

实验报告（50%）：实验过程描述清晰准确，对算法的理解深刻、有独特见解。

Presentation（20%）：对实验、算法的描述清晰准确。

# 任务B：使用国产框架完成深度学习训练、测试

## B.1任务简述

本任务旨在通过亲手实践，培养同学搭建完整的深度学习训练、测试框架的能力。

首先，需完成任务A中的内容，了解模型结构和目标任务：阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Introduction and Guidelines”章节，学习图匹配的基础知识，了解图匹配的主要步骤。阅读在线文档的“Get Started”章节，在深度学习实验平台上安装软件包并配置PyTorch环境。阅读神经网络图匹配算法的[文档介绍](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.neural_solvers.html)，参考原论文一起阅读，了解算法原理。基于在线文档中的例子（[使用深度学习进行图像匹配](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html#sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)），下载并运行代码，复现在线文档中的测试结果。

其次，需基于国产深度学习框架，编写完整的深度学习训练、测试代码。一份完整的深度学习代码需要包含以下功能：

* 数据集加载模块
* 定义模型结构
* 训练代码
* 测试代码

具体地，关于如何编写这样一份深度学习代码，深度学习框架的在线文档中通常会给出详细的例子：[jittor](https://cg.cs.tsinghua.edu.cn/jittor/tutorial/2021-2-4-00-00-jittor_class_1/)，[paddle](https://www.paddlepaddle.org.cn/documentation/docs/zh/guides/beginner/quick_start_cn.html)，[mindspore](https://www.mindspore.cn/tutorial/zh-CN/r0.6/quick_start/quick_start.html)。由于三个框架的pygmtools工具开发进度不一，对应的任务要求亦有差别。下面以最完整的要求（Jittor框架）作为整体介绍：

本次大作业推荐完成WillowObjectClass数据集上的训练、测试任务。

关于**数据集加载**，pygmtools提供了WillowObjectClass的数据接口。使用提示：在你选择的深度学习框架下，编写对应的数据集加载模块；创建一个[Benchmark对象](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.benchmark.Benchmark.html" \l "pygmtools.benchmark.Benchmark)，在类内通过调用get\_data成员函数获取训练数据或测试数据。

关于**模型结构**，建议从pygmtools提供的[神经网络图匹配模型中](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.neural_solvers.html)任选一个。在线文档中的[深度学习图像匹配例子](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html#sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)展示了如何搭建一个由CNN模块和图匹配模块组成的神经网络。

关于**训练和测试**，训练和测试功能可结合[深度学习图像匹配例子](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html#sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)与国产深度学习框架的例子（[jittor](https://cg.cs.tsinghua.edu.cn/jittor/tutorial/2021-2-4-00-00-jittor_class_1/)，[paddle](https://www.paddlepaddle.org.cn/documentation/docs/zh/develop/practices/cv/convnet_image_classification.html)，[mindspore](https://www.mindspore.cn/tutorial/zh-CN/r0.6/quick_start/quick_start.html)）编写。在收集得到匹配结果后，通过调用[Benchmark对象](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.benchmark.Benchmark.html" \l "pygmtools.benchmark.Benchmark)的eval成员函数，即可得到匹配精度。

三种不同框架需完成的主要任务总结如下：

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Jittor** | **PaddlePaddle** | **MindSpore** |
| 改写一个模型（包括转换PyTorch预训练模型） | × | × | √ |
| 数据集加载 | √ | √ | √ |
| 训练功能 | √ | √ | × |
| 测试功能 | √ | √ | √ |

撰写至少4页的实验报告，内容可包括配置环境、对算法的理解、国产框架的使用感受（与PyTorch对比）、训练测试结果、遇到的问题和解决方式等；准备一个3分钟的pre，在15-16周课堂上进行一次展示。

## B.2 主要步骤

1. 阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Introduction and Guidelines”章节

2. 阅读[在线文档](https://pygmtools.readthedocs.io/en/latest/index.html)的“Get Started”章节，在深度学习实验平台上安装软件包并配置PyTorch环境

3. 阅读神经网络图匹配算法的[文档介绍](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.neural_solvers.html)，仔细阅读论文并深入理解PCA-GM, IPCA-GM和CIE算法，选择其中一个模型作为大作业内容

4. 运行实验，复现[样例](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html" \l "sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)中的测试结果（网页最下方有代码下载链接，可点击下载）

5. 在你选择的深度学习框架下（[jittor](https://cg.cs.tsinghua.edu.cn/jittor/assets/docs/jittor.dataset.html" \l "jittor.dataset.Dataset)或[paddle](https://www.paddlepaddle.org.cn/documentation/api/paddle/io/DataLoader_cn.html)或[mindspore](https://www.mindspore.cn/docs/api/zh-CN/r1.5/api_python/dataset/mindspore.dataset.RandomSampler.html" \l "mindspore.dataset.RandomSampler)），编写对应的数据集加载模块，创建一个[Benchmark对象](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.benchmark.Benchmark.html" \l "pygmtools.benchmark.Benchmark)，在类内通过调用get\_data成员函数获取训练数据或测试数据

6. 基于在线文档中的[深度学习图像匹配例子](https://pygmtools.readthedocs.io/en/latest/auto_examples/pytorch/plot_deep_image_matching.html" \l "sphx-glr-auto-examples-pytorch-plot-deep-image-matching-py)，编写模型代码，编写训练和测试代码

7. 收集得到匹配结果后，通过调用Benchmark对象的eval成员函数，得到匹配精度

8. 撰写报告，准备展示

## B.3 评分标准

源代码（40%）：代码清晰、注释丰富，变量、函数等命名合理。汇报的精度需要可以复现，无法复现的代码将会被惩罚。代码中允许使用Numpy，但所有PyTorch接口需要改写为国产框架。

实验报告（40%）：实验过程描述清晰准确，对算法的理解深刻、有独特见解。如果你在实验中尝试并解决了某些挑战（包括但不限于：实现了比基础要求更多的功能，比较不同优化器的性能，学习率scheduling，适配了更多的数据集，支持了更多的模型，通过超参数调优提升了精度，发现并修正/绕过了开源代码的bug），请在实验报告中标出，评分时将会充分考虑。

Presentation（20%）：对实验、算法的描述清晰准确。

# 任务C：开放性科研探索

## C.1 任务简述：

作为一个新兴的研究方向，深度学习领域还有许多有趣的研究方向等待大家探索。我们鼓励开放性的科研探索作为深度学习课程的大作业，选择该方向的同学需要在向助教提交一份科研提案，篇幅不限，其中需要包括拟使用的技术、拟解决的问题、计划达成的效果。助教审核通过后即视为成功选择该任务。使用[Latex模板](https://cvpr2022.thecvf.com/sites/default/files/2021-10/cvpr2022-author_kit-v1_1-1.zip)，撰写至少6页（英文、双栏）的小论文，包含背景介绍（Introduction）、相关工作（Related Work）、提出的方法（Our Approach）、实验结果（Experiment）等部分；准备一个3钟的pre，在15-16周课堂上进行一次展示。

## C.2 主要步骤

1.向助教提交一份科研提案，篇幅不限，其中需要包括拟使用的技术、拟解决的问题、计划达成的效果

2. 落实你的科研提案，开展理论和实验研究

3. 使用[Latex模板](https://cvpr2022.thecvf.com/sites/default/files/2021-10/cvpr2022-author_kit-v1_1-1.zip)，撰写至少6页（英文、双栏）的小论文

4. 撰写报告，准备展示

## C.3 评分标准

源代码（40%）：代码清晰、注释丰富，变量、函数等命名合理。汇报的精度需要可以复现，无法复现的代码将会被惩罚。

小论文（40%）：英语表达流利、行文逻辑清晰、技术新颖、对于图匹配等科学问题有自己的洞察。文章图文并茂，实验结果清晰、消融实验合理。我们以工作量作为评分基础，在技术和写作方面参照机器学习领域CCF-A类会议审稿的标准评价论文。

Presentation（20%）：对实验、算法的描述清晰准确，能够准确地描述新算法、新技术。