

Visual Recognition using Deep Learning

2025 Spring, Homework 1

Release Date: 2025/03/05 12:00

Homework 1

- Deadline: **23:59, 03/26 (Wed), 2025**
- **Participate the competition (80%): Image Classification Problem**
 - Participant the competition on the CodaBench and get the highest score as possible. (70%)
 - Code reliability & quality (10%)
- **Report and code (20%): Document your method and findings.**
 - Report
 - **In PDF format and written in English.** (5pt penalty)
 - Introduction to your method (e.g., data pre-processing, model architecture, hyper-parameters)
 - Innovative ideas or additional experiments to further improve the model.
 - Code
 - Zip your code (.py) alone with report - Submit to E3.
 - You should also put your code on your GitHub repository and provide the link in the report.

Links

- [Link to the dataset](#)
- [Link to the competition](#)

How to participate the competition and do submission

1. Register an account on [CodaBench](#)
 - a. When registering the account, please use your **studentID** as the **UserName**
2. After you click the competition link, go to My Submissions, and join the competition

Get Started Phases **My Submissions** Results Forum

You have not yet registered for this competition.

To participate in this competition, you must accept its specific [terms and conditions](#). This competition **does not** require approval, once you register, you will immediately be able to participate.

☐ I accept the terms and conditions of the competition.

Register

Get Started Phases My Submissions Results Forum ?

Participation
Timeline
Terms
Files

Download	Phase	Task	Type	Size
solution @ 03-01-2025 16:45	Testing Phase	Main classification Task	Solution	108.97 KB

An “example” submission

How to participate the competition and do submission

3. Submit your results and don't forget to “Add to Leaderboard”

4. Don't forget to check your results can be found on the leaderboard

Get Started Phases **My Submissions** Results Forum ?

Testing Phase

Number of submissions used for the day
1 out of 10

Number of total submissions used
1 out of 99

Submission upload

Submit as: ?
Yourself

[File upload area highlighted with a red box]

No limitation for the name of the “zip” file;
however, inside the zip file, your result file
must be named “prediction.csv”

名稱	類型
solution.zip	Microsoft Excel 逗點分隔值檔案

ID #	File name	Date	Status	Score
239199	solution.zip	2025-03-02 01:15	Finished	0.00

[Add to Leaderboard button highlighted with a red box]

Get Started Phases My Submissions **Results** Forum

Testing Phase

Filter Leaderboard by Columns

Task	Results				Main classification Task
#	Participant	Entries	Date	ID	Prediction score (Public)
1	strong-baseline	1	2025-03-02 01:28	239205	0.94
2	nycuvilabguess	1	2025-03-02 01:15	239199	0.0

Coding Environment

- Recommendation: Python 3.9 or higher
- Tips
 - We recommend you to use **virtual environments** when implementing your homework assignments.
 - Here are some popular virtual environment management tools
 - [Poetry](#)
 - [Conda](#)
 - [Virtualenv](#)

Numpy & PyTorch

- Numpy Tutorial: [Link](#)
- PyTorch Tutorial: [Link](#)
 - Free to use any modules and functions

Task and Dataset

- Task: Image classification
- Inputs
 - RGB images
 - Training / Validation: 21,024; Test: 2,344
- Target
 - Corresponding object category id of the image
 - 100 categories
- **Additional Requirements / Limitations** (15 pts penalty, each)
 - No external data (i.e., data from other sources) allowed.
 - Model size (#parameters) should less than 100M. (ways to check your #parms: [Ref](#))
 - Also, you can **only** use **ResNet (e.g., ResNet18/34/...)** as the model backbone in this task. Modification to the backbone is allowed, elaborate your modification in the report may help you get good report score.
 - Not a requirement: pretrained weights is allowed.



Grading Policy - Report (20%)

- Format: PDF, written in English. (-5pts if not followed)
- Sections that you should include
 - **Introduction** to the task and core idea of your method
 - **Method**: Describe how you pre-process the data; what is your model architecture, and hyperparameters, etc.
 - **Results**: your findings / model performance (e.g., training curve, confusion matrix, etc.)
 - **References**: Your method references (paper / Github sources, must include if you use any.)

We encourage you to stand on the shoulders of giants - only clone and run it is not enough.

5pts

- **Additional experiments** to explore better performance
 - Simply tuning the hyper-parameters doesn't count (e.g., batch-size, LR, different optimizers)
 - Hint: Try to add/remove some layers, use different loss functions, etc.
- You should 1) include your hypothesis (why you do this), 2) How this may (or may not) work, and 3) The experiment results and their implications.

1. [PEP8](#)
2. [Google Python Style](#)

Grading Policy - Code Reliability (10%)

1. Please follow the PEP8 instructions and lint your code.
2. Push your code to the GitHub
 - It should contains a README.md to introduce this work (And your StudentID).



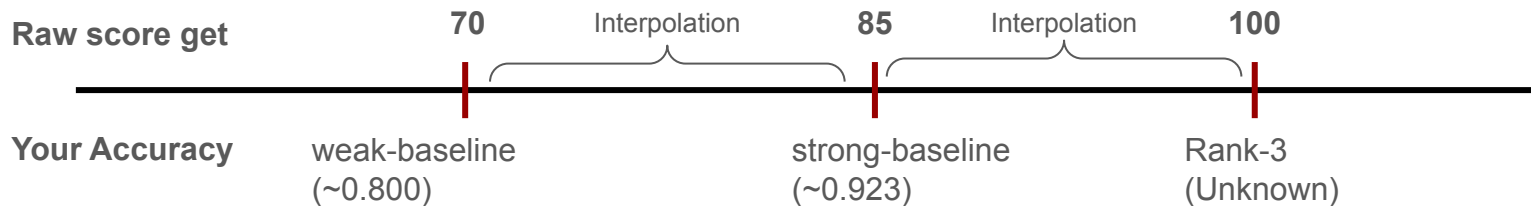
An example: README.md

Grading Policy - Competition (70%)

We will use private (hidden) leaderboard to evaluate the performance (the distribution is similar for data in public and private set.) The public leaderboard is for you as reference.

Your score (competinon):

- Less than weak-baseline ($\text{Acc} < \text{w.baseline}$): $S = 0$
- Between weak-baseline and strong baseline ($\text{Acc} \geq \text{w.baseline} \ \& \ \text{Acc} < \text{s.baseline}$):
 $(70 + (X - \text{w.baseline}) / (\text{s.baseline} - \text{w.baseline}) * (85 - 70)) * 0.7$
- Between strong-baseline and Rank3: $(85 + (X - \text{s.baseline}) / (\text{Acc.rank3} - \text{s.baseline}) * (100 - 85)) * 0.7$
- $\text{Rank1,2,3} = 100 * 0.7$



Submission

- Compress your code and report into a **.zip file** and submit it to E3.
 - Don't forget to push your code to GitHub. And your GitHub link should be written in the report.
- Report should be written in English.
- <STUDENT ID>_HW1.zip
 - codes (.py, folders, etc)
 - <STUDENT ID>_HW1.pdf (NO .doc, .docx or others format)
- Don't put the data (e.g. x.jpg / train.csv / test.csv) and model checkpoints into submission file (-5 if not followed)

Other rules

- **Late Policy**: A penalty of **20 points** per additional late day. (-20pt / delayed.day)
 - For example, If you get 90 points but delay for two days, your will get only 50 points!
- **No Plagiarism**: You should complete the assignment by yourself. Students engaged in plagiarism will be penalized heavily. Super serious penalty.
 - e.g. -100pt for the assignment or failed this course, etc
 - Report to academic integrity office

FAQs

- Can I use any library/package/framework from GitHub or other resources?
 - **Yes**, we encourage you to learn how to leverage existing knowledge on your own task
 - e.g., Github of published works and model zoo from Torchvision
 - Focus on how to step forward from them - That's why part of scores comes from your competition ranks
 - You **should not copy-and-paste from your classmates** (Plagiarism)
- How to handle the GPU Out-of-Memory (OOM) issue?
 - Easy answer - Make your batch size smaller or make your model smaller.
 - Advanced methods: Try to figure it out by yourself. (Many online resources and AI-assistance)

FAQs

- If I don't have my own GPU – Use Google Colab
 - It should be 12 hours, please check [this discussion in the stackoverflow](#)
 - And some tricks [here](#) may make it longer.
- If you have other questions, ask on [E3 forum](#) first! We will reply as soon as possible.

It's your turn! Have Fun!



Essential

Googling the Error Message