The Mythical Pegasus and Gravitar

Deep Learning and Reinforcement Learning Summative Assignment

Michaelmas Term, 2020

Introduction

The assignment is split into two parts: (1) Deep Learning (50 marks) and (2) Reinforcement Learning (50 marks) accordingly, for a combined total of 100 marks available.

In short, you are to build a generative model that creates unique and diverse images of winged horses that all look like a Pegasus, the mythical divine horse in Greek mythology. This is usually depicted as pure white, with clear outspread wings. The second part of the assignment is to train a deep reinforcement learning agent to play the Atari 2600 game Gravitar, published in 1983. This is one of the most difficult Atari games where you have to fly to different planets as you get pulled towards deadly stars by gravity, shoot enemies, conserve and pick up fuel with a tractor beam that also deflects bullets.

For the Pegasus part, you are to write a short scientific paper for the method, results, and limitations in a provided LaTeX template that closely follows parts of the ICLR 2021 conference style guidelines. Whereas for the Gravitar part, you are to provide a video of the agent playing the game, code, and log file (no report is required). These files must all be zipped together like this:

```
submission.zip
   pegasus-paper.pdf
   pegasus-code.ipynb (or .py)
   gravitar-code.ipynb (or .py)
   gravitar-video-episode-1210-score-85.mp4
   gravitar-log.txt
```

To assist in this, the following templates are provided to build on:

- [Pegasus Paper LaTeX Template] - *login with durham email, on 'overleaf.com' click 'make a copy' to edit*
- [Google Colab Pegasus Starter Code]
- [Google Colab Gravitar Starter Code]

Part 1: The Mythical Pegasus

Using the CIFAR-10 and/or STL-10 datasets, you are to train a deep generative model to synthesise a unique batch of 64 images that all looked like winged horses (of any colour). There are no such images of winged horses in these two datasets. You should then select, from the batch, the single best image that looks most like a white Pegasus. The paper should use the provided LaTeX template, where you are to write up the methodology, results, and limitations of your approach alongside a short abstract.

The report should be written like an academic paper, where mathematical notation should try to follow the ICLR 2021 guidelines (see the template for more information). This means your discussions should be short, clear, and concise —*less is more*. Where appropriate, it is recommended to include a high-level architectural diagram in the paper to help explain your approach.

You can use any architecture that you like, and you can use any sampling strategy. For example you could train an autoencoder, and condition it on a birds and horses which have been manually identified from the dataset. These datasets mostly have annotated class labels available for the images, which you can use to help identify the horses and birds. This will give two latent codes via the encoder network, that you could linearly interpolate between to give the final outputs via the decoder network. Alternatively, you

could train a GAN and randomly sample a batch of 64 images from the model. These two ideas are not necessarily the best solution. Also, there are penalties and bonuses that will influence your design:

Use any adversarial training (e.g. GAN) method	-4 marks
Your best Pegasus is non-white (e.g. brown or black)	-2 marks
Nearly all winged horses in the batch are white	$+1$ mark
Train only on CIFAR-10	-2 marks
Train with STL-10 resized to 48x48 pixels	$+1$ mark
Train with STL-10 resized to 64x64 pixels	$+1$ mark
Train with STL-10 at the full 96x96 pixels	$+3$ marks
Manually edit (paint) any images or outputs	-50 marks
Train on any other data outside of the datasets	-50 marks
Use or modify someones code without referencing it	-50 marks
Use pre-trained weights from another model	-50 marks
Every page over 4 pages in the paper (excluding references)	-5 marks

Table 1: Penalties and bonuses accumulate, and are added onto the final mark.

Please state at the end of the paper the total bonus or penalty you are expecting, based on this table. For example if you successfully train a GAN to produce a white Pegasus from a batch of mixed-colour winged horses, using both data from CIFAR-10 and from STL-10 at the full resolution, you can expect to receive: -4 marks (as its a GAN), then 1+1+3 = +5 marks for STL-10 at 96x96 resolution, for a total bonus of +1 mark. If you submit a paper that is 7 pages long, you will receive an additional -15 mark penalty.

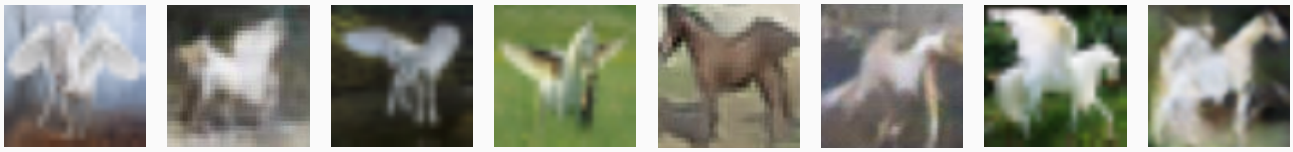


Figure 1: Generating a realistic and recognisable Pegasus is very difficult within the dataset constraints, therefore outstanding attempts with unimpressive results can still get good marks. These are some excellent results when training only on CIFAR-10.

Pegasus paper marking scheme

The paper will be marked as follows:

- **[20 marks]** Scientific quality and mathematical rigor for the paper and solution
  - Communication, application, and presentation of the underpinning mathematical theory
  - Architectural design, sophistication, appropriateness, and novelty
  - Clarity, simplicity and frugality of both the scientific writing and the implementation
- **[10 marks]** Recognisability of the single best output
  - Can I tell this is an image of a Pegasus?
  - How much do I need to stretch my imagination to see a winged horse?
- **[10 marks]** Realism of the sampled batch of model outputs
  - Are the generated images blurry?
  - Do the objects in them have realistic shapes or textures? Do they look real?
- **[10 marks]** Uniqueness of the sampled batch of model outputs
  - How different are the images from their nearest neighbours in the dataset?
  - How diverse are the samples within the batch of 64 provided?
  - Do all the winged horses look the same? Is there any mode collapse?

Part 2: Gravitar

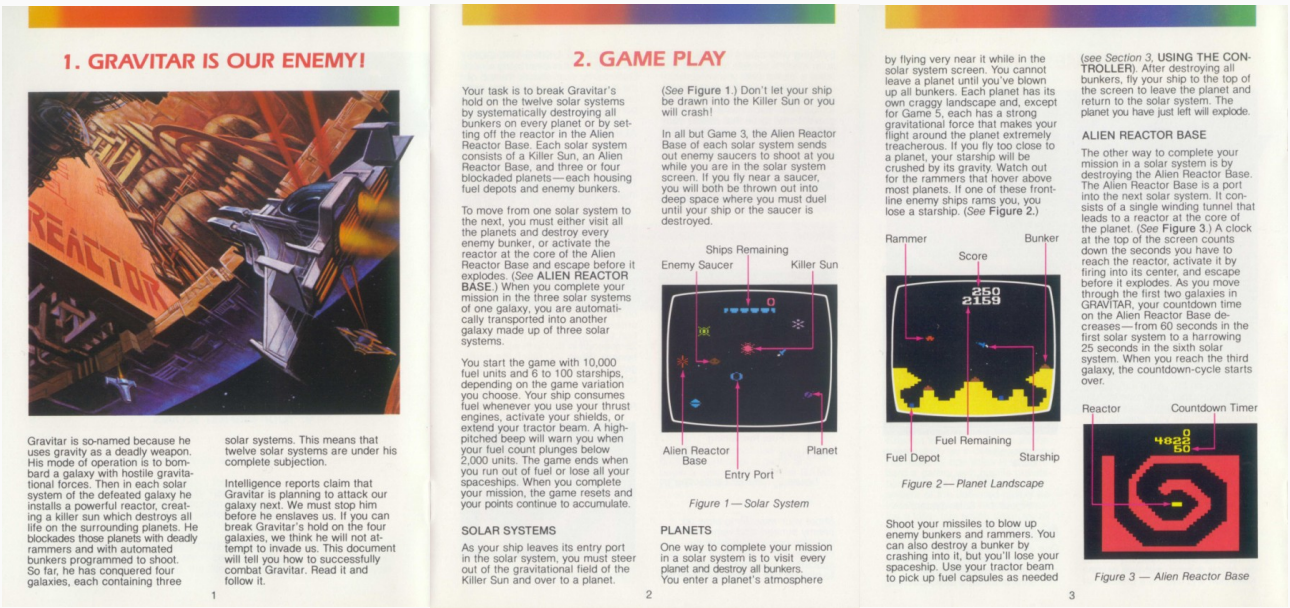
Gravitar is a notoriously difficult Atari 2600 game, with complex controls and changing environments where you always have to navigate away from gravity. This is what the beautiful Atari 2600 console looked like, which was released in 1977. It had an 8-bit 1.19 MHz CPU with just 128 bytes of RAM!

Figure 2: Atari 2600 console

Figure 3: Gravitar cartridge

I recommend that before you begin, you have a go at playing Gravitar online to understand the game mechanics. Try to visit a couple of planets and use the tractor beam (down key) to pick up some more fuel. My score is 7,250 - can you do any better?

- [Play Gravitar with an online Atari 2600 emulator] spacebar to start/fire and move with arrow keys
- [More information, including the full manual and cartridge images]



1. GRAVITAR IS OUR ENEMY!

Gravitar is so-named because he uses gravity as a deadly weapon. His mode of operation is to bombard a galaxy with hostile gravitational forces. Then in each solar system of the defeated galaxy he installs a powerful reactor, creating a killer sun which destroys all life on the surrounding planets. He blockades those planets with deadly rammers and with automated bunkers programmed to shoot. So far, he has conquered four galaxies, each containing three solar systems. This means that twelve solar systems are under his complete subjection.

Intelligence reports claim that Gravitar is planning to attack our galaxy next. We must stop him before he enslaves us. If you can break Gravitar's hold on the four galaxies, we think he will not attempt to invade us. This document will tell you how to successfully combat Gravitar. Read it and follow it.

1

2. GAME PLAY

Your task is to break Gravitar's hold on the twelve solar systems by systematically destroying all bunkers on every planet or by setting off the reactor in the Alien Reactor Base. Each solar system consists of a Killer Sun, an Alien Reactor Base, and three or four blockaded planets—each housing fuel depots and enemy bunkers.

To move from one solar system to the next, you must either visit all the planets and destroy every enemy bunker, or activate the reactor at the core of the Alien Reactor Base and escape before it explodes. (See ALIEN REACTOR BASE.) When you complete your mission in the three solar systems of one galaxy, you are automatically transported into another galaxy made up of three solar systems.

You start the game with 10,000 fuel units and 6 to 100 starships, depending on the game variation you choose. Your ship consumes fuel whenever you use your thrust engines, activate your shields, or extend your tractor beam. A high-pitched beep will warn you when your fuel count plunges below 2,000 units. The game ends when you run out of fuel or lose all your spaceships. When you complete your mission, the game resets and your points continue to accumulate.

SOLAR SYSTEMS

As your ship leaves its entry port in the solar system, you must steer out of the gravitational field of the Killer Sun and over to a planet. (See Figure 1.) Don't let your ship be drawn into the Killer Sun or you will crash!

In all but Game 3, the Alien Reactor Base of each solar system sends out enemy saucers to shoot at you while you are in the solar system screen. If you fly near a saucer, you will both be thrown out into deep space where you must duel until your ship or the saucer is destroyed.

Ships Remaining
Enemy Saucer
Killer Sun
Alien Reactor Base
Planet
Entry Port

Figure 1—Solar System

PLANETS

One way to complete your mission in a solar system is to visit every planet and destroy all bunkers. You enter a planet's atmosphere by flying very near it while in the solar system screen. You cannot leave a planet until you've blown up all bunkers. Each planet has its own craggy landscape and, except for Game 5, each has a strong gravitational force that makes your flight around the planet extremely treacherous. If you fly too close to a planet, your starship will be crushed by its gravity. Watch out for the rammers that hover above most planets. If one of these frontline enemy ships rams you, you lose a starship. (See Figure 2.)

2

Rammer
Bunker
Score
250
2159
Fuel Remaining
Fuel Depot
Starship

Figure 2—Planet Landscape

Shoot your missiles to blow up enemy bunkers and rammers. You can also destroy a bunker by crashing into it, but you'll lose your spaceship. Use your tractor beam to pick up fuel capsules as needed (see Section 3, USING THE CONTROLLER). After destroying all bunkers, fly your ship to the top of the screen to leave the planet and return to the solar system. The planet you have just left will explode.

ALIEN REACTOR BASE

The other way to complete your mission in a solar system is by destroying the Alien Reactor Base. The Alien Reactor Base is a port into the next solar system. It consists of a single winding tunnel that leads to a reactor at the core of the planet. (See Figure 3.) A clock at the top of the screen counts down the seconds you have to reach the reactor, activate it by firing into its center, and escape before it explodes. As you move through the first two galaxies in GRAVITAR, your countdown time on the Alien Reactor Base decreases—from 60 seconds in the first solar system to a harrowing 25 seconds in the sixth solar system. When you reach the third galaxy, the countdown-cycle starts over.

Reactor
Countdown Timer
4822
50

Figure 3 — Alien Reactor Base

3

Task

Your task is to build and train a deep reinforcement learning agent to obtain the highest score possible, using the provided starter code linked on the first page. OpenAI gym provides two Gravitar environments '**Gravitar-ram-v0**' and '**Gravitar-v0**' respectively. You can use either of them, but pay attention to the size of the input array shapes as documented here and here.

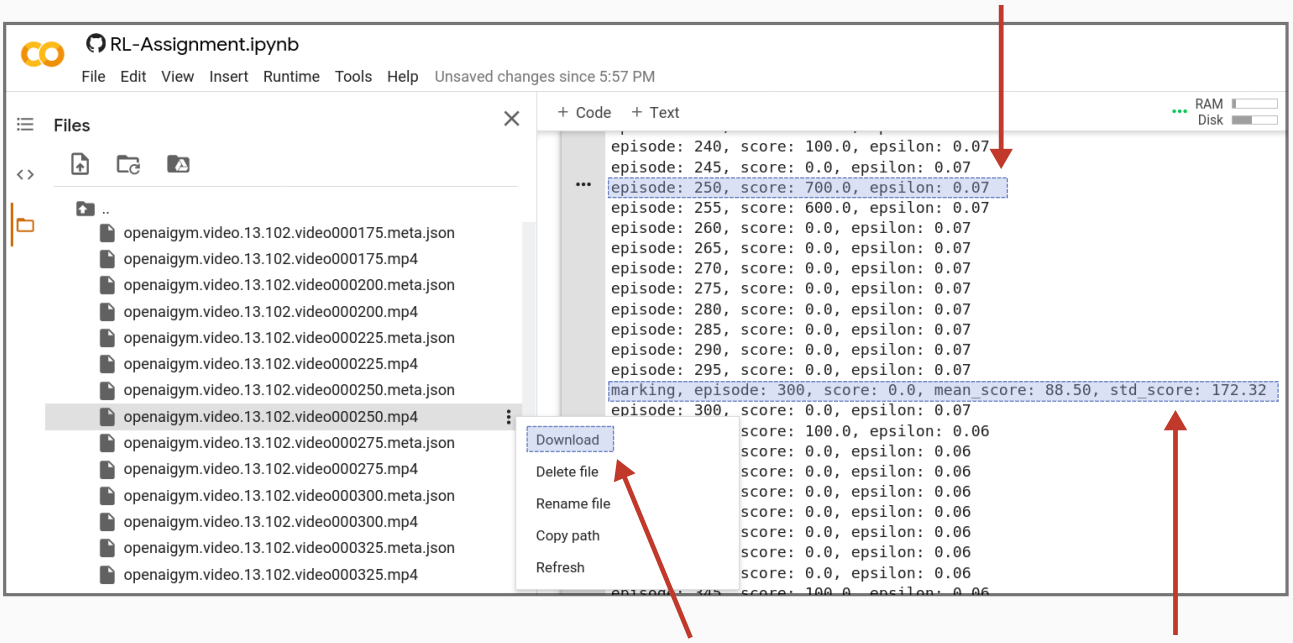
You can use any reinforcement learning algorithm that you like, including ones not covered in the lectures.

Submission

The following figure explains how and what to submit. You can print whatever data you like into the log, but you must keep the string which starts with 'marking'. This string prints the mean and standard deviation of the score over every 100 episodes. You must not change this. The submitted video should show the highest score that you are able to successfully record. Recording a video is quite slow, so you may wish

to do it every 25 or more episodes. Increasing this interval may slightly improve training performance, but you may miss recording a good episode.

This is the best episode with a recorded video



Therefore download episode 250 and rename it **gravitar-video-episode-250-score-700.mp4**

Make sure these lines are printed
Copy the full log to **gravitar-log.txt**

The code submission must be written with clarity and minimalism. You can submit an .ipynb or .py file. You do not need to follow PEP-8 or any departmental guidelines for code quality with this assignment. Try to keep comments to a minimum, as good code should speak for itself. If you have tried many impressive designs and ideas, you may briefly explain these in a large comment (or text in .ipynb if you prefer) at the bottom of the submission, but do not expect this to significantly influence marking of your final solution.

Anti-tampering

There will be statistical analysis of the submitted log files. Any log files with outliers in the convergence behaviour will have their associated code retrained multiple times using SLURM scripts on NCC, learning a distribution over its expected convergence behaviour. If the likelihood of the submitted log data shows evidence of tampering, a departmental investigation will be conducted. If interested, this method for detecting tampering is similar to parts of the solution used to solve Sploosh Kaboom in Zelda Windwaker.

Marking

The code, video, and log submission will be marked as follows:

- **[30 marks]** Convergence and score
  - How efficiently does your agent learn?
  - Does it just get lucky after lots and lots of training?
  - How often does it get a good score?
- **[10 marks]** Sophistication and appropriateness of the solution
  - How well have you applied the relevant theory to the problem?
  - How hackish is your implementation, or is it robust and well-designed?
  - Have you just cited and pasted code, or is their evidence of comprehension with further study and novel design extending beyond the lecture materials?
- **[10 marks]** Intuition of video
  - Does the video look like the agent is just randomly getting lucky after lots of episodes?

- Is it chasing and hunting down the enemies efficiently?
- Is it navigating gravity gracefully, or is it like it's powered by an infinite improbability drive?
- Has it learnt any surprising behaviours?

PyTorch Training

This assignment can be completed entirely using Google Colab, or you may wish to register for an account and train on NCC: `http://ncc.clients.dur.ac.uk/` (only available on the internal university network). Users without remote server experience or job queuing system experience (such as SLURM) are recommended to continue to use Google Colab, which is just as fast for PyTorch training. If using NCC, please carefully read the documentation and respect other users on the job queuing system.

Closing Comment

I hope that you enjoy this coursework. If you are struggling, please join one of the weekly zoom meetings to ask questions and discuss any issues, such as with programming or relevant theory.

Questions and Answers

It is strongly recommended to read these common questions and answers carefully.

I've spent a million hours training these models and I still need more time. This assignment is impossible!	This is always the problem with Deep Learning and Reinforcement Learning. Some of our models take many months to train, so you need to choose your experiments very carefully, with only well-tested code where you have evidence it will converge given more time. Also, in some ways, yes these tasks are impossible to get to perfection. That is the nature of these ill-defined learning problems, but i'm not looking for perfection for full marks. When marking, I will be looking carefully at the expected converging behaviour of the method, rather than directly at the best scores/images.
I just read this interesting new paper which is very different to techniques from the course. Can I use it?	Yes! This is exactly the kind of thing you should be doing. Study the paper and reimplement it, but make sure you cite it in the report, and you must also reference it in the comments at the top of the code.
I found code online which looks similar to what I need. Can I use it?	Yes, but you must cite the code in both the written report and in the comments at the top of the code. I will then carefully cross-reference this code with your implementation to see how well you have adapted their code. If you have simply copied and pasted without evidence of experimentation or tailoring, do not expect to get a good grade even if your results are good. Whereas if I see evidence of original interpretation, novel comprehension and application of the theory in the lectures, even if the experimental results are not as strong, you can get very high marks. However if you pass-off other people's code as your own, and 'forget' to cite their code, or work together with other students, you will very likely get caught (see the submission Plagiarism and Collusion section on DUO to read about the tools used to detect this). This incurs a very severe departmental penalty.

Table 2: Common questions and answers

“	What do you mean by manual image editing? Can I write code to do X?
💬	The answer to this is ‘within reason’. If you manually paint wings on the inputs or outputs, you will get zero. If you write code to manually set the pixels which look like wings, you’ll also get zero. If you start using the OpenCV library to enhance and brighten the images etc, then technically yes this is acceptable, but do not expect to get additional marks for this, as this is a Deep Learning assignment and not Computer Vision or Image Processing.
“	Isn’t the best strategy to just copy the state-of-the-art?
💬	Actually no, the state-of-the-art for these tasks are large distributed models (which use large numbers of GPUs scaled horizontally). You don’t have these kinds of resources available to you, so you will need a different strategy. The state-of-the-art models also have been highly tuned to squeeze every last bit of juice out of them. If you copy this code directly, as mentioned in a previous answer, you won’t get a good grade. Also I will be marking based on the convergence behaviour, not just the final score. If I find evidence that a poor quality model has been trained with large numbers of GPUs for several thousand pounds on AWS, it will not be awarded higher marks than an excellent model that has only been trained for a few hours on a single Google Colab GPU – even if it has a higher score.
“	To farm the STL-10 bonuses, can I train on CIFAR-10, then do one optimisation step at the full resolution on STL-10, so I can say i’ve trained on it?
💬	This won’t make significant improvement to the output quality, so I won’t count the bonuses for this.
“	Do I have to train on all the data?
💬	No, you can train on parts of CIFAR-10 or STL-10, or combinations of them in any way you like.
“	I have this amazing/funny/great video. The score is not as good, but it’s so surprising! Can I submit it too?
💬	Yes, you can upload an additional outtake video named accordingly to this: **gravitar-outtake-episode-100-score-50.mp4**
“	What are pre-trained weights?
💬	This is where you would use a model that someone else has trained on different data, e.g. a GAN trained on ImageNet, and then transfer the weights (transfer learning) for the new task. While this would generate beautiful high-resolution winged horses, this is not allowed.
“	Can I use Keras, Theano, Tensorflow, Caffe, Chainer, or X to develop the coursework instead?
💬	No, unless you already are very experienced with PyTorch and want to use this assignment to learn another framework such as JAX. If this is the case, you must email me and provide evidence that you already have significant PyTorch experience, as it is the industry standard for DL research.

Table 2: Common questions and answers

“	I’m really struggling and feeling overwhelmed by all of this. The maths is too complicated and I don’t know where to begin.
💬	Okay, first don’t panic, get a wet towel and wipe your face :) Now open up clean versions of the Colab Starter Code and try running it. Next try to make some small modifications to the models. If you get errors, read them slowly, Google them. Start by trying to improve the existing code – why not try making it residual to start with? Try to get a feel for its limitations. When you’re confident enough, try to implement something a little bit more sophisticated that you think will do better. Here you need to be careful, don’t try too much at once. Try to add just the smallest bit of extra functionality you can, then test it and check if it works. Build up slowly and incrementally until you have the confidence to try more advanced ideas.
“	Is Google Colab free to use?
💬	Yes, although when Google is experiencing heavy load they impose limits (which vary) on their resources. Therefore to get the most out of Colab, save your model and optimisation parameters as in the example code, and close your tabs when you are done. This is because Google prioritises users who have used less resources.
“	What Gravitar score do I need for a 1st? Will this Pegasus give me a 1st?
💬	I can’t answer this, as the marking ‘function’ has too many parameters.
“	Does the page limit include references?
💬	No, your references section can start on the 5th page without penalty. You can have an unlimited number of references.
“	Do I have to use the LaTeX template for the paper?
💬	Yes, this is for the experience of writing a conference paper in this field and it will hopefully encourage slower and more concise writing, alongside mathematical rigor with professional typesetting.
“	Help! I’ve never used or been taught LaTeX!
💬	After you’ve logged into overleaf.com and cloned the paper template (click the copy icon from the overleaf.com home page, then edit the copied version), just modify some of the text and press Ctrl+S to compile. If you want to achieve something specific, Google it. You can then download and submit the compiled .PDF directly from overleaf. The rest will come naturally.

Table 2: Common questions and answers