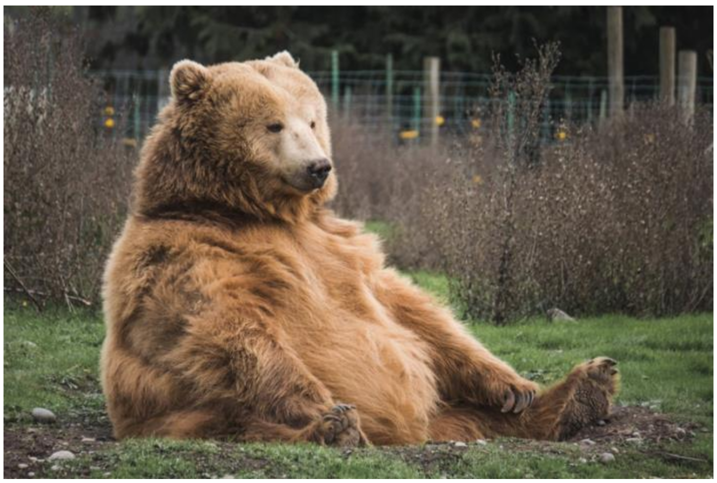
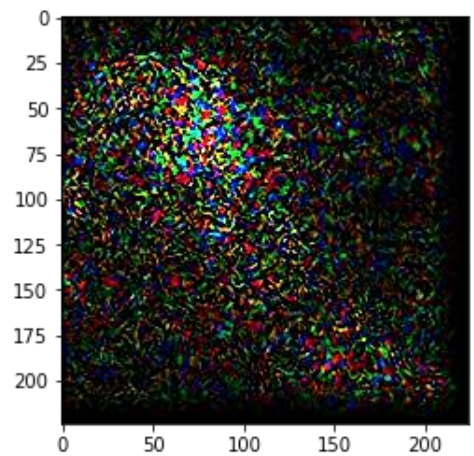
Machine Learning for the Arts
UCSD SPRING 2019
FINAL PROJECT

Neural Dynamical Systems





Malhar Jere
UC San Diego

Description:
For this project I attempted to create a new method of generating art through neural networks, specifically by using the gradients of the probability distributions generated by neural networks with respect to their input signals.

Concept:
I'd been reading 'Chaos' by James Gleick, and noticed the similarities between non-linear systems in real life (like the weather, stock market, smoke trails from cigarettes and fires, water flows, etc.) and neural networks. After a little digging, I found out that the idea of investigating neural networks as chaotic dynamical systems was relatively unexplored. I began exploring this topic for my PhD research and have tried to apply it to this class, where I attempted generating art using chaos and nonlinear systems.

A brief background on chaos and dynamical systems:

Dynamical systems is simply a term used to describe the behaviour of complex systems that evolve with time. The behaviour of these systems can be modelled with simple difference or differential equations.

The most canonical example of a dynamical system is the double pendulum. For a single pendulum, the equations of motion of the bob and string are very deterministic, have a closed form solution. Given the starting position of the pendulum, one can calculate the position at some arbitrary instance of time in the future with almost perfect accuracy. Additionally, given the position of the pendulum at some arbitrary instance of time, one can predict where it will be in the future.

This case is not as simple with the double pendulum, which consists of a pendulum attached to the bottom of a single pendulum. Assuming no friction and loss, the double pendulum traces an extremely complicated and random path that appears almost random, but is completely deterministic. At any instance of time, there is no way to predict where the pendulum might be at any future instance of time, and a tiny change in initial conditions will lead to a radically different path that the pendulum might swing. This phenomena of ordered disorder in simple systems is known mathematically as *chaos.*

Chaos primarily arises in mathematical systems that are nonlinear in nature. The nonlinear activation functions present in neural networks led me to believe that neural networks might in fact be chaotic in nature as well, which will be the focus of a good portion of my PhD thesis. It led me to attempt to use these odd, nonlinear dynamics to create art that has disorder at a micro-scale but is ordered at a larger scale.

Technique and Process:
For this project, I ran 2 experiments to demonstrate chaos and randomness to generate art from neural networks.

Experiment 1: Standalone images as seed

For this experiment, as a starting point, I calculated gradients of standalone images with respect to an empty loss vector, and accumulated the gradients in an image for numerous iterations. The above equation is modified to be the following:

$$x_{n+1} = n \times \frac{d(loss(NN(x_0)))}{dx_0}$$

In summary, this experiment is as follows:

- Start off with an image of a cat/elephant/your favourite animal
- Forward propagate the image through our neural network
- Find the loss of the output with an empty vector (filled with zeros, so we're essentially just taking the sum of the squares of the output probabilities)
- Backpropagate this loss to the input and accumulate it in a gradient vector
- Repeat the above steps repeatedly

Experiment 2: Empty images as seed

For the second experiment I actually wanted to see the evolution of the gradients as I backpropagated the gradients to my input, so I modified the above experiment slightly. I used the gradients of the current generation as the input to the next generation, and observed how this vector evolves in time.

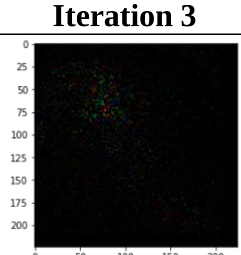
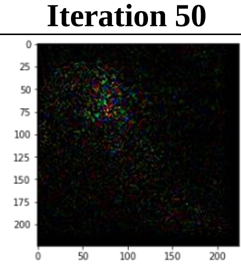
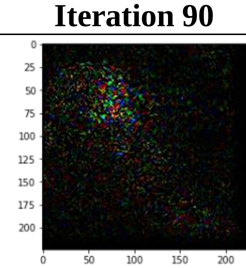
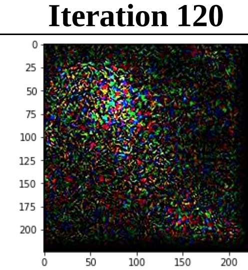
$$x_{n+1} = \frac{d(loss(NN(x_n)))}{dx_n} + x_n$$

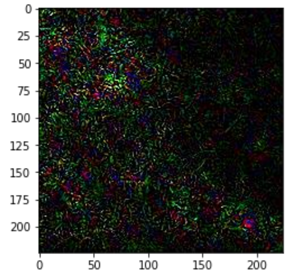
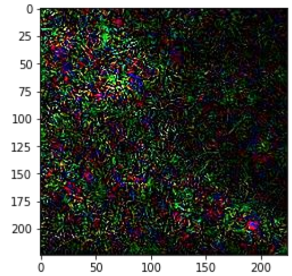
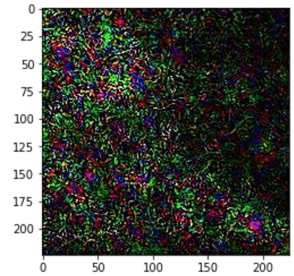
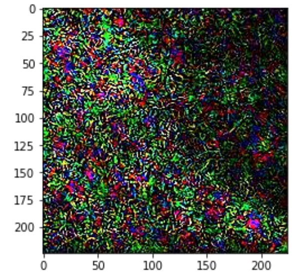
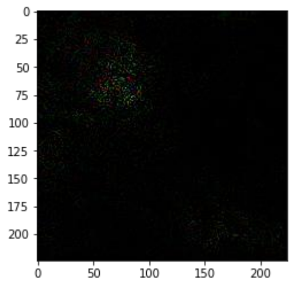
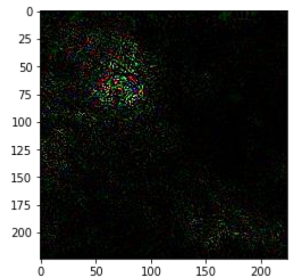
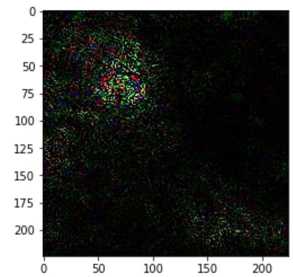
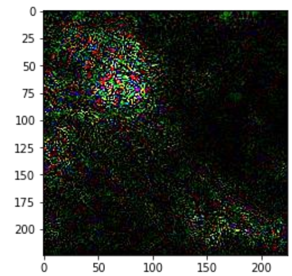
In summary, this experiment is as follows:

- Start off with a blank image
- Forward propagate the image through our neural network
- Find the loss of the output with an empty vector (filled with zeros, so we're essentially just taking the sum of the squares of the output probabilities)
- Backpropagate this loss to the input and accumulate it in a gradient vector
- Repeat steps 2 through 4 with the gradient vector as the input to the neural network

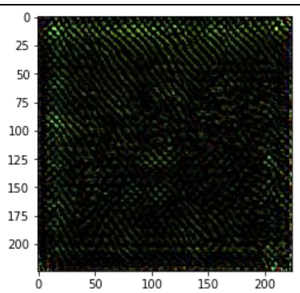
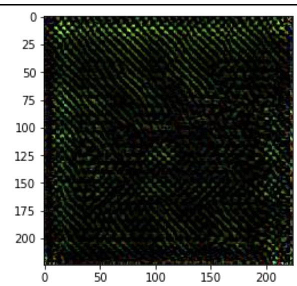
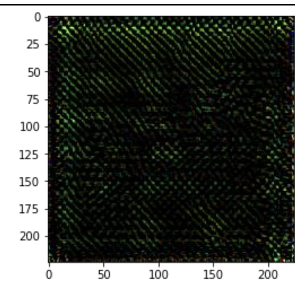
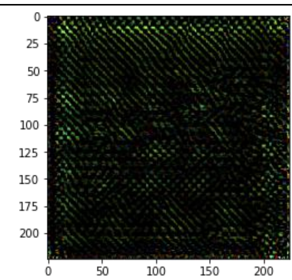
Result:

Experiment 1: I used stock images of a bear, elephant and cat to compute gradients for different neural networks across a bunch of iterations, which can be seen below (for a bear in particular).

Model	Iteration 3	Iteration 50	Iteration 90	Iteration 120
Inception v3				

ResNet18				
VGG19				

Experiment 2: I used empty images, and forward propagated the images through the neural network, found the loss with respect to the input image, backpropagated it to the input, and fed that gradient in again for the next iteration. This process was repeated several times (200 to be exact), and the results are shown below for the inception_v3 model.

Model	**Iteration 50**	**Iteration 80**	**Iteration 150**	**Iteration 200**
Inception v3				

Reflection:

The above generated images are intriguing and unique. For experiment 1, they light up in regions where the main animal is present, are unique for the neural network architecture, and appear random but ordered. For experiment 2, the images generated using the gradients appear similar but have very subtle visual differences between iterations, which might be an indication of chaotic behaviour evolving in these systems. In particular I like the second set of experimental images as it shows ordered disorder arising out of a very simple experiment.

For future experiments I'd like to run the experiment for many more iterations with a range of different types of inputs. To determine whether neural networks are truly chaotic I'd like to characterize my experiment in terms of its *Lyapunov exponent*, which is a mathematical term that helps in measuring disorder in a system. I believe this is a rich area of research and something that I’d love to continue, and I’m grateful for the opportunity to have been exposed to it.