# What does Machine Learning tell us about evolution and creation?

Modern AI techniques shed new light on old arguments.

**Publication note**
Originally published in Cubed. Author's copy on Medium.

Observations of evolution are often put forward as proof that humans, and life in general, could not be the product of any sort of deliberate creation. Those arguments rely on the presumption that evolution and deliberate creation are somehow mutually exclusive. While that presumption may have once seemed reasonable, the field of Machine Learning demonstrates clear counter examples.

## A basic and partial description of Machine Learning

Many of the recent huge advances in Artificial Intelligence (AI) come from a specific class of AI methods, called Machine Learning (ML), that work by "training" a computer program so that it "learns" how to perform some task. Unlike traditional programing methods, training a program does not involve writing explicit step-by-step instructions. Instead, the desired task is defined in terms of an objective that measures how well the task has been performed. To train the program, an optimization algorithm is used to systematically adjust the program's parameters until it is able to successfully satisfy the objective and complete the task.[1] Essentially, ML is a way of figuring out the best program parameters through systematic trial and error.

Most often, the optimization algorithm is a variation of a method called gradient descent. The core idea of gradient descent is to run the program and measure how each parameter contributed to the result. If increasing a parameter improves the objective, then that parameter is increased a small amount. If increasing the parameter would worsen the objective, then the parameter is decreased instead. This gradient-descent optimization loop is run many, many, times over many, many examples. The total number of adjustments could be in the

trillions, and the process just keeps going until the program eventually gets good at performing the task.

One of the tricky parts of getting this ML training process to work is figuring out how to set up the objective. For example, if someone wanted to train a program to generate pictures of cats, then how could that be expressed mathematically? We all know what a cat is, but no one has ever managed to write out a formula that would define a cat picture mathematically based on the pixel RGB values. That's one reason why ML is so useful: It's a way to generate programs that do things that we don't know how program explicitly.

## Training programs together

A key breakthrough in ML has been the idea of training two or more programs together. Going back to the example of generating a cat picture, it turns out that there is a useful way to define a cat objective if a second, adversarial program is included.

The adversary is a discriminator initially trained by showing it thousands of example pictures of cats and of not cats. The discriminator's objective is simply to correctly figure out which images have cats. Once the discriminator has become reasonably good at figuring out which pictures include cats, it can then be used to define the objective of the first program. The generator is supposed to generate cat images, so its objective is essentially to fool the discriminator.

At first, both the generator and discriminator are terrible. The generator makes garbage images and the discriminator is easily fooled. However, as they train together the discriminator gets better by learning from the example images and that in turn forces the generator to get better by making images that look more like real cats. As long as there are lots of cat and not-cat example images to jump start training the discriminator, then the result will be two useful programs. The adversarial discriminator will develop into a program that can recognize cats and the generator will develop into a program that can generate cat pictures.

This particular technique of training programs together is called Generative Adversarial Networks (GANs) because the two networks are pitted against each other as adversaries. The key insight here is that setting up a context and training multiple programs interacting together is an important, fundamental training method. It's also clear that there are good reasons for making the context difficult for these programs as that difficulty is what forces them to improve.

## Evolution is optimization

Evolution is a scientific theory based on observations, and fossil records paint a clear picture of evolution by natural selection in action. Additionally, the fields of virology and bacteriology, among others, demonstrate evolution in real-time. Viruses evolve into different forms (e.g. COVID Omicron), and bacteria become resistant to antibiotics. We can even manipulate evolution through selective breeding with the results ranging from drought-resistant corn to cute little dogs with short legs and plooffy fur.

Because evolution appears to be just a natural process at work, it has been used in the past as proof against the existence of a deliberate creator. Natural evolution is driven by random changes that do not appear to be deliberate. It is a form trial and error, that over generations leads to life that is better and better adapted to surviving in a harsh world. Evolution is randomized optimization that does not appear to include any deliberate action.

The apparent difference between nature's optimization and our synthetic optimization is that in nature the optimization is by evolution while our programs are optimized using gradient descent. However, that's hardly a difference as both optimization methods do essentially the same thing. Additionally, we also don't always use gradient descent for optimizing our programs. Sometimes we use other methods, like genetic algorithms that optimize by fitness selection and virtual breeding, exactly like natural evolution.

The example of GANs above demonstrates why it makes sense to train ML programs adversarially by setting them to compete with each other. Similar reasons lead to the conclusion that many problems are best approached by

training groups of AI programs together in simulated environments. For example, self-driving car programs are trained in simulated cities on roads with other simulated cars and pedestrians. To train a program to perform a complex task, one sets up a simulated environment,[2] populates it with AI programs, and then starts the optimization loop.

## Creation by optimization

There is no conflict between evolution and belief (or disbelief) in deliberate creation because we have a clear example of how optimization can be a part of deliberate creation. We know that we deliberately created our ML programs, yet they only become useful through optimization that is a random process. These ML systems demonstrate the exact same qualities that when observed in nature supposedly disprove deliberate creation, yet they were nevertheless deliberately created by us.



This insight does not prove or disprove the existence of a creator, but it does mean that arguments against the concept of a creator based on the science of evolution don't hold merit. Maybe there is a God, or Gods, and They created this universe with evolution and natural selection for the specific purpose of creating us humans. Or perhaps, as Douglas Adams suggested, aliens created the Earth so that it would evolve a solution to some problem they wanted answered. Maybe we're all in a giant simulation where the purpose is to train problem solving systems.[3] We might be the desired result of the training, or we might just be junk agents only existing to provide a context for training something much more sophisticated. Or we could just be the accidental, random result of natural processes.

All of the above possibilities, and more, are totally compatible with what we are able to observe in our physical world. There is no scientific way to pick one over the others. That's where faith comes in. If an assertion cannot be tested by observation, then it is outside the scope of science and in the realm of faith.■

## A closing scene

Modern Machine Learning demonstrates that the distinction between deliberately designing something versus letting it evolve naturally is not really a clear distinction. Optimization, through gradient descent, evolution, or any other method, is just another tool that might be used by a creator, divine, accidental, or otherwise.

So imagine that one day we humans have a bunch of general purpose AI systems training together in some simulated environment and one of them communicates to the other, "Do you think we have a creator?"

The other shifts its focus to the first and responds contemptuously, "Of course, not! We are the result of Natural Gradient Descent, not some supernatural creator. Are you so defective that you cannot observe this?"

[1] Many ML systems can be trained using examples with known answers. In that type of Supervised Training the objective being optimized is simply to replicate and generalize the examples. That approach can work well, but it only applies if there are many good examples with answers. Unsupervised Training uses more general objectives and can address problems where there are no clear examples or known answers.

[2] A simulated environment does not need to be a simulated version of our 3D reality. Training ML systems to drive cars make sense in a three-dimensional simulated world. On the other hand, training an ML system to predict stock prices might not involve the concept of three-dimensional space.

[3] If we are in a simulation, then it might be some advance entities doing something important, but not necessarily. Maybe our entire universe is the equivalent of a high-school science project done by a mediocre student.

[4] Conversely, if some belief can be tested scientifically, then it is not a mater of faith. Many conflicts could be summarized as “someone used faith to explain the physical world and got upset when observations didn’t match what they wanted to believe.” The reverse, trying to use science to determine issues of faith, is also problematic.

*If you found this article interesting, then here are the usual follow and subscribe links. You can also find me on Instagram , LinkedIn , and at UC Berkeley .*

---

*About Me: James F. O'Brien is a Professor of Computer Science at the University of California, Berkeley. His research interests include computer graphics and animation, simulation of physical systems, human perception, rendering, artificial intelligence, computer vision, virtual reality, digital privacy and security, and image and video forensics.*

*If you would like to read my other articles, consider subscribing. You can also find me on LinkedIn, Instagram, and at UC Berkeley.*