**Machine Learning Explainability vs Interpretability: Two concepts that could help restore trust in AI**

We explain the key differences between explainability and interpretability and why they're so important for machine learning and AI, before taking a look at several techniques and methods for improving machine learning interpretability.

It doesn’t take a data scientist to work out that the machine and deep learning algorithms built into automation and artificial intelligence systems lack transparency. It also doesn’t take a great deal of detective work to see that many of these systems contain an imprint of the unconscious biases of the engineers that helped to develop them.

Arguably, in the midst of what The Economist termed a [*techlash*](https://www.economist.com/international/2017/08/10/internet-firms-face-a-global-techlash), this lack of transparency has only (ironically) become more visible. While many of the incidents that have contributed towards the techlash are as much issues caused by a mixture of corporate self-interest and an alarming absence of governance and accountability, there’s no escaping the fact that the practice of data science and machine learning engineering naturally find their way hooked onto some of the year’s biggest business and politics stories.

It’s in this context that the concepts of explainability and interpretability [have taken on new urgency](https://hub.packtpub.com/improve-interpretability-machine-learning-systems/). It’s likely that they are only going to become more important in 2019, as discussions around the ethics of artificial intelligence continues.

But what are explainability and interpretability? And how what do they actually mean for those of us doing data mining, analysis, science in 2019?

**The difference between machine learning explainability and interpretability**

In the context of machine learning and artificial intelligence, explainability and interpretability are often used interchangeably. While they are very closely related, it’s worth unpicking the differences, if only to see how complicated things can get once you start digging deeper into machine learning systems.

Interpretability is about the extent to which a cause and effect can be observed within a system. Or, to put it another way, it is the extent to which you are able to *predict* what is going to happen, given a change in input or algorithmic parameters. It’s being able to look at an algorithm and go *yep, I can see what’s happening here.*

Explainability, meanwhile, is the extent to which the internal mechanics of a machine or deep learning system can be explained in human terms. It’s easy to miss the subtle difference with interpretability, but consider it like this: interpretability is about being able to discern the mechanics without necessarily knowing why. Explainability is being able to quite literally explain what is happening.

Think of it this way: say you’re doing a science experiment at school. The experiment might be interpretable insofar as you can see what you’re doing, but it is only really explainable once you dig into the chemistry behind what you can see happening.

That might be a little crude, but it is nevertheless a good starting point for thinking about how the two concepts relate to one another.

**Why are explainability and interpretability important in artificial intelligence and machine learning?**

If 2018’s techlash has taught us anything, it’s that although technology can certainly be put to dubious usage, there are plenty of ways in which it can produce poor - discriminatory - outcomes with no intention of causing harm.

As domains like healthcare look to deploy artificial intelligence and deep learning systems, where questions of accountability and transparency are particularly important, if we’re unable to properly deliver improved interpretability, and ultimately explainability, in our algorithms, we’ll seriously be limiting the potential impact of artificial intelligence. Which would be a shame.

But aside from the legal and professional considerations that need to be made, there’s also an argument that improving interpretability and explainability are important even in more prosaic business scenarios. Understanding how an algorithm is actually working can help to better align the activities of data scientists and analysts with the [key questions and needs of their organization](https://hub.packtpub.com/ux-designers-can-teach-machine-learning-engineers-start-model-interpretability/).

**Techniques and methods for improving machine learning interpretability**

While questions of transparency and ethics may feel abstract for the data scientist on the ground, there are, in fact, a number of practical things that can be done to improve an algorithm’s interpretability and explainability.

**Algorithmic generalization**

The first is to improve generalization. This sounds simple, but it isn’t that easy. When you think most machine learning engineering is applying algorithms in a very specific way to uncover a certain desired outcome, the model itself can feel like a secondary element - it’s simply a means to an end. However, by shifting this attitude to consider the overall health of the algorithm, and the data on which it is running, you can begin to set a solid foundation for improved interpretability.

**Pay attention to feature importance**

This should be obvious, but it’s easily missed. Looking closely at the way the various features of your algorithm have been set is a practical way to actually engage with a diverse range of questions, from business alignment to ethics. Debate and discussion over how each feature should be set might be a little time-consuming, but having that tacit awareness that different features have been set in a certain way is nevertheless an important step in moving towards interpretability and explainability.

**LIME: Local Interpretable Model-Agnostic Explanations**

While the techniques above offer practical steps that data scientists can take, LIME is an actual method developed by researchers to gain greater transparency on what’s happening inside an algorithm. The researchers explain that LIME can explain “the predictions of any classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction.”

What this means in practice is that the LIME model develops an approximation of the model by testing it out to see what happens when certain aspects within the model are changed. Essentially it’s about trying to recreate the output from the same input through a process of experimentation.

**DeepLIFT (Deep Learning Important Features)**

DeepLIFT is a useful model in the particularly tricky area of deep learning. It works through a form of backpropagation: it takes the output, then attempts to pull it apart by ‘reading’ the various neurons that have gone into developing that original output.

Essentially, it’s a way of digging back into the feature selection inside of the algorithm (as the name indicates).

**Layer-wise relevance propagation**

Layer-wise relevance propagation is similar to DeepLIFT, in that it works backwards from the output, identifying the most relevant neurons within the neural network until you return to the input (say, for example, an image). If you want to learn more about the mathematics behind the concept, [this post by Dan Shiebler is a great place to begin](http://danshiebler.com/2017-04-16-deep-taylor-lrp/).

**Adding complexity to tackle complexity: can it improve transparency?**

The central problem with both explainability and interpretability is that you’re adding an additional step in the development process. Indeed, you’re probably adding multiple steps. From one perspective, this looks like you’re trying to tackle complexity with even greater complexity.

And to a certain extent, this is true. What this means in practice is that if we’re to get really serious about interpretability and explainability, there needs to be a broader cultural change in the way in which data science and engineering is done, and how people *believe* it should be done.

That, perhaps, is the really challenging part.

**Bio**: [Richard Gall](https://www.linkedin.com/in/richard-gall-5a6a235a/?originalSubdomain=uk) is the Editorial Content Product Manager at Packt. Through quality messaging, focused on the needs of the communities and customers Packt helps, Richard works to drive both direct and inbound revenues and improve the reach and relevance of the Packt brand within today's tech landscape.