Interpretability

There is no mathematical definition of interpretability. A (non-mathematical) definition I like by Miller (2017)[10] is: **Interpretability is the degree to which a human can understand the cause of a decision**. Another one is: **Interpretability is the degree to which a human can consistently predict the model's result** [11]. The higher the interpretability of a machine learning model, the easier it is for someone to comprehend why certain decisions or predictions have been made. A model is better interpretable than another model if its decisions are easier for a human to comprehend than decisions from the other model. I will use both the terms interpretable and explainable interchangeably. Like Miller (2017), I think it makes sense to distinguish between the terms interpretability/explainability and explanation. I will use "explanation" for explanations of individual predictions. See the section about explanations to learn what we humans see as a good explanation.

Importance of Interpretability

If a machine learning model performs well, **why do not we just trust the model** and ignore **why** it made a certain decision? "The problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks." (Doshi-Velez and Kim 2017 [12])

Let us dive deeper into the reasons why interpretability is so important. When it comes to predictive modeling, you have to make a trade-off: Do you just want to know **what** is predicted? For example, the probability that a customer will churn or how effective some drug will be for a patient. Or do you want to know **why** the prediction was made and possibly pay for the interpretability with a drop in predictive performance? In some cases, you do not care why a decision was made, it is enough to know that the predictive performance on a test dataset was good. But in other cases, knowing the 'why' can help you learn more about the problem, the data and the reason why a model might fail. Some models may not require explanations because they are used in a low-risk environment, meaning a mistake will not have serious consequences, (e.g. a movie recommender system) or the method has already been extensively studied and evaluated (e.g. optical character recognition). The need for interpretability arises from an incompleteness in problem formalization (Doshi-Velez and Kim 2017), which means that for certain problems or tasks it is not enough to get the prediction (the **what**). The model must also explain how it came to the prediction (the **why**), because a correct prediction only partially solves your original problem. The following reasons drive the demand for interpretability and explanations (Doshi-Velez and Kim 2017 and Miller 2017).

[10]Miller, Tim. "Explanation in artificial intelligence: Insights from the social sciences." arXiv Preprint arXiv:1706.07269. (2017).

[11]Kim, Been, Rajiv Khanna, and Oluwasanmi O. Koyejo. "Examples are not enough, learn to criticize! Criticism for interpretability." Advances in Neural Information Processing Systems (2016).

[12]Doshi-Velez, Finale, and Been Kim. "Towards a rigorous science of interpretable machine learning," no. Ml: 1–13. http://arxiv.org/abs/1702.08608 (2017).

Human curiosity and learning: Humans have a mental model of their environment that is updated when something unexpected happens. This update is performed by finding an explanation for the unexpected event. For example, a human feels unexpectedly sick and asks, "Why do I feel so sick?". He learns that he gets sick every time he eats those red berries. He updates his mental model and decides that the berries caused the sickness and should therefore be avoided. When opaque machine learning models are used in research, scientific findings remain completely hidden if the model only gives predictions without explanations. To facilitate learning and satisfy curiosity as to why certain predictions or behaviors are created by machines, interpretability and explanations are crucial. Of course, humans do not need explanations for everything that happens. For most people it is okay that they do not understand how a computer works. Unexpected events makes us curious. For example: Why is my computer shutting down unexpectedly?

Closely related to learning is the human desire to **find meaning in the world**. We want to harmonize contradictions or inconsistencies between elements of our knowledge structures. "Why did my dog bite me even though it has never done so before?" a human might ask. There is a contradiction between the knowledge of the dog's past behavior and the newly made, unpleasant experience of the bite. The vet's explanation reconciles the dog owner's contradiction: "The dog was under stress and bit." The more a machine's decision affects a person's life, the more important it is for the machine to explain its behavior. If a machine learning model rejects a loan application, this may be completely unexpected for the applicants. They can only reconcile this inconsistency between expectation and reality with some kind of explanation. The explanations do not actually have to fully explain the situation, but should address a main cause. Another example is algorithmic product recommendation. Personally, I always think about why certain products or movies have been algorithmically recommended to me. Often it is quite clear: Advertising follows me on the Internet because I recently bought a washing machine, and I know that in the next days I will be followed by advertisements for washing machines. Yes, it makes sense to suggest gloves if I already have a winter hat in my shopping cart. The algorithm recommends this movie, because users who liked other movies I liked also enjoyed the recommended movie. Increasingly, Internet companies are adding explanations to their recommendations. A good example is the Amazon product recommendation, which is based on frequently purchased product combinations:

Recommended products when buying some paint from [Amazon](https://www.amazon.com/Colore-Acrylic-Paint-Set-12/dp/B014UMGA5W/). Visited on December 5th 2012.

In many scientific disciplines there is a change from qualitative to quantitative methods (e.g. sociology, psychology), and also towards machine learning (biology, genomics). The **goal of science** is to gain knowledge, but many problems are solved with big datasets and black box machine learning models. The model itself becomes the source of knowledge instead of the data. Interpretability makes it possible to extract this additional knowledge captured by the model.

Machine learning models take on real-world tasks that require **safety measures** and testing. Imagine a self-driving car automatically detects cyclists based on a deep learning system. You want to be 100% sure that the abstraction the system has learned is error-free, because running over cyclists is quite bad. An explanation might reveal that the most important learned feature is to recognize the two wheels of a bicycle, and this explanation helps you think about edge cases like bicycles with side bags that partially cover the wheels.

By default, machine learning models pick up biases from the training data. This can turn your machine learning models into racists that discriminate against protected groups. Interpretability is a useful debugging tool for **detecting bias** in machine learning models. It might happen that the machine learning model you have trained for automatic approval or rejection of credit applications discriminates against a minority. Your main goal is to grant loans only to people who will eventually repay them. The incompleteness of the problem formulation in this case lies in the fact that you not only want to minimize loan defaults, but are also obliged not to discriminate on the basis of certain demographics. This is an additional constraint that is part of your problem formulation (granting loans in a low-risk and compliant way) that is not covered by the loss function the machine learning model was optimized for.

The process of integrating machines and algorithms into our daily lives requires interpretability to increase **social acceptance**. People attribute beliefs, desires, intentions and so on to objects. In a famous experiment, Heider and Simmel (1944) [13] showed participants videos of shapes in which a

[13]Heider, Fritz, and Marianne Simmel. "An experimental study of apparent behavior." The American Journal of Psychology 57 (2). JSTOR: 243–59. (1944).

circle opened a "door" to enter a "room" (which was simply a rectangle). The participants described the actions of the shapes as they would describe the actions of a human agent, assigning intentions and even emotions and personality traits to the shapes. Robots are a good example, like my vacuum cleaner, which I named "Doge". If Doge gets stuck, I think: "Doge wants to keep cleaning, but asks me for help because it got stuck." Later, when Doge finishes cleaning and searches the home base to recharge, I think: "Doge has a desire to recharge and intends to find the home base." I also attribute personality traits: "Doge is a bit dumb, but in a cute way." These are my thoughts, especially when I find out that Doge has knocked over a plant while dutifully vacuuming the house. A machine or algorithm that explains its predictions will find more acceptance. See also the chapter on explanations, which argues that explanations are a social process.

Explanations are used to **manage social interactions**. By creating a shared meaning of something, the explainer influences the actions, emotions and beliefs of the recipient of the explanation. For a machine to interact with us, it may need to shape our emotions and beliefs. Machines have to "persuade" us, so that they can achieve their intended goal. I would not fully accept my robot vacuum cleaner if it did not explain its behavior to some degree. The vacuum cleaner creates a shared meaning of, for example, an "accident" (like getting stuck on the bathroom carpet ... again) by explaining that it got stuck instead of simply stopping to work without comment. Interestingly, there may be a misalignment between the goal of the explaining machine (create trust) and the goal of the recipient (understand the prediction or behavior). Perhaps the full explanation for why Doge got stuck could be that the battery was very low, that one of the wheels is not working properly and that there is a bug that makes the robot go to the same spot over and over again even though there was an obstacle. These reasons (and a few more) caused the robot to get stuck, but it only explained that something was in the way, and that was enough for me to trust its behavior and get a shared meaning of that accident. By the way, Doge got stuck in the bathroom again. We have to remove the carpets each time before we let Doge vacuum.

Doge, our vacuum cleaner, got stuck. As an explanation for the accident, Doge told us that it needs to be on an even surface.

Machine learning models can only be **debugged and audited** when they can be interpreted. Even in low risk environments, such as movie recommendations, the ability to interpret is valuable in the research and development phase as well as after deployment. Later, when a model is used in a product, things can go wrong. An interpretation for an erroneous prediction helps to understand the cause of the error. It delivers a direction for how to fix the system. Consider an example of a husky versus wolf classifier that misclassifies some huskies as wolves. Using interpretable machine learning methods, you would find that the misclassification was due to the snow on the image. The classifier learned to use snow as a feature for classifying images as "wolf", which might make sense in terms of separating wolves from huskies in the training dataset, but not in real-world use.

If you can ensure that the machine learning model can explain decisions, you can also check the following traits more easily (Doshi-Velez and Kim 2017):

- Fairness: Ensuring that predictions are unbiased and do not implicitly or explicitly discriminate against protected groups. An interpretable model can tell you why it has decided that a certain person should not get a loan, and it becomes easier for a human to judge whether the decision is based on a learned demographic (e.g. racial) bias.
- Privacy: Ensuring that sensitive information in the data is protected.
- Reliability or Robustness: Ensuring that small changes in the input do not lead to large changes in the prediction.
- Causality: Check that only causal relationships are picked up.

- Trust: It is easier for humans to trust a system that explains its decisions compared to a black box.

When we do not need interpretability.

The following scenarios illustrate when we do not need or even do not want interpretability of machine learning models.

Interpretability is not required if the model **has no significant impact**. Imagine someone named Mike working on a machine learning side project to predict where his friends will go for their next holidays based on Facebook data. Mike just likes to surprise his friends with educated guesses where they will be going on holidays. There is no real problem if the model is wrong (at worst just a little embarrassment for Mike), nor is there a problem if Mike cannot explain the output of his model. It is perfectly fine not to have interpretability in this case. The situation would change if Mike started building a business around these holiday destination predictions. If the model is wrong, the business could lose money, or the model may work worse for some people because of learned racial bias. As soon as the model has a significant impact, be it financial or social, interpretability becomes relevant.

Interpretability is not required when the **problem is well studied**. Some applications have been sufficiently well studied so that there is enough practical experience with the model and problems with the model have been solved over time. A good example is a machine learning model for optical character recognition that processes images from envelopes and extracts addresses. There is years of experience with these systems and it is clear that they work. In addition, we are not really interested in gaining additional insights about the task at hand.

Interpretability might enable people or programs to **manipulate the system**. Problems with users who deceive a system result from a mismatch between the goals of the creator and the user of a model. Credit scoring is such a system because banks want to ensure that loans are only given to applicants who are likely to return them, and applicants aim to get the loan even if the bank does not want to give them one. This mismatch between the goals introduces incentives for applicants to game the system to increase their chances of getting a loan. If an applicant knows that having more than two credit cards negatively affects his score, he simply returns his third credit card to improve his score, and organizes a new card after the loan has been approved. While his score improved, the actual probability of repaying the loan remained unchanged. The system can only be gamed if the inputs are proxies for a causal feature, but do not actually cause the outcome. Whenever possible, proxy features should be avoided as they make models gameable. For example, Google developed a system called Google Flu Trends to predict flu outbreaks. The system correlated Google searches with flu outbreaks – and it has performed poorly. The distribution of search queries changed and Google Flu Trends missed many flu outbreaks. Google searches do not cause the flu. When people search for symptoms like “fever” it is merely a correlation with actual flu outbreaks. Ideally, models would only use causal features because they would not be gameable.