# Evaluation of Success

In order to know how our machine learning model fairs, we first need to answer the following three questions.

### A) What does success look like?

What it would look like if your machine learning model (the QuAM) answered your questions perfectly. How would you use that information? Would you trust it? Should we, trust it?

Even though no system will give you 100% correct predictions but considering this scenario will help you think and compensate for extreme conditions, like, is this model over-fitting? Is this bad? How does this affects our model? What should we do?

It will also help us evaluate our actual model. It forces us to consider what threshold of accuracy do we need to trust that prediction, to act upon that suggestion. Say, we have our prediction but what then? what could we do with that insight? and various questions like that. In turn we will have a better understanding of our problem as well as the solution and this way we can take better decisions to improve our QuAM and that's a good thing.

### B) How are we already measuring success?

It is possible to define a new metric to evaluate a process or models performance. But it's not ideal. A better situation would be, if we had some kind of system already in place to measure against. The aim of using a machine learning model to solve a problem is to increase the efficiency. So it's natural that we would like to know how our model is performing in comparison to the solution already in place (the benchmark).

### C) How are you currently measuring performance?

Does it measure what you want it to measure? How sure are you that your measurement actually evaluates something you care about. Is there another better way of evaluating that gets closer to your ultimate goal?

It's a tricky question because usually the thing we really truly care about; customer satisfaction, profitability, long-term effects on the environment, cannot be measured directly, we're almost always using a proxy.

It often happens that, the proxy we used is not really good and isn't measuring exactly what we wanted to. The only solution to this dilemma is to make conscious decisions about what proxies we're using, and recognize the potential trade-offs between them.

- What's a proxy actually?

  **Proxy** is something that was/is or can be directly measured as a stand in for something(s) that can't be measured directly.

  For example, you want to know about customer engagement with your online presence. Without highly invasive spyware watching everything a customer's reading, looking at or discussing, you can't know exactly how much those customers are talking about you. As invasive spyware is both unethical and impractical, we use proxies. That is things that we can measure ethically and practically that we're assuming have a close enough correlation to what we care about to be able to stand-in.

  In the case of customer engagement, companies often use proxies of likes, clicks, or the time the customer spends on their site. These are mostly measurable. They don't actually measure engagement directly but it's assumed that these measures are close enough to measuring engagement to be valid. These substitute evaluation measures are used every day in every industry not just for machine learning systems. For instance, your credit rating is a proxy measure of your financial responsibility and your ability to repay a loan. A course mark or a grade point average is a proxy for a student's understanding of the terms and concepts that were taught in this course or program. An IQ test is a proxy measure how well you can think or solve problems.

  There are many more examples and some proxies are more valid than others, but hopefully you get the idea. Most things we care about can't be measured directly. So we look at related things that can be measured and hope that they're good indicators of what we really care about.

  We revisit these decisions and examine these trade-offs so we don't get trapped into mistaking the metric we're currently using for the absolute ultimate goal.

So how do you find a good proxy for what you care about?

First, start simple. A simple measurement with a good enough validity is best. Particularly, when we're talking about evaluating Machine Learning models, you'll need a simple and easy-to-calculate metric. When you're first building models, you're doing a fairly coarse comparison. So improvements in this simple metric are likely to represent improvements in the real space.

As you get further along into more detailed refinements, it becomes more likely that improving your first metric involves some trade off with the others. A complex measurement that gets closer to what you care about is good to keep in mind but for initial optimization, stick with simple measurements.

It's unlikely one simple measurement captures everything you care about. So if you can find a useful proxy that someone else has established in practice that's valid, then that's great, use that. If you can't find an already validated proxy, then often you have to take your best guess. See how it works and be ready to go back and change it if you need to.

It can also be helpful to spend some time thinking about how your metrics can be gained either intentionally or unintentionally. For example, if your proxy for customer engagement is how long they're on your site, what if they just left the window open while they were doing other things. If your proxy for product quality is good online reviews, what if someone pays a whole lot of people to post either good or bad reviews of a product. So there you have it. Everything you're basing your Machine Learning system on is a proxy for something else, including the measurement of outcomes you really care about, it's especially those.

So start simple. Validate your proxies as much as you can and think about how these metrics might be changed or manipulated reducing their validity. Be aware of what your metric is and is not capturing of your true goal and put checks in place to compensate for gaps. Finally, be ready to revisit this question regularly to make sure you're not building a system that perfectly maximizes the wrong thing entirely.

▼ Related Courses/ Tutorials

- Machine Learning: Algorithms in the Real World Specialization, C01 - Introduction to Applied Machine Learning