Recognizing the System Behavior of Seemingly Linear Models

Jonny Morell
Jamorell@jamorell.com

Contents

There is a great deal of conversation in our field about the need to move from linear-based to system-based evaluation. The discussion then proceeds to conversations about what system-based evaluation is and how to go about doing it. The reasoning is one of transformation from a "linear" to a "system" state. There is something missing in this direction of reasoning, namely, what "linear" means. This language leads us astray because many models that people disparage as "linear" are in fact deeply system based.

I think the difficulty begins with a definition of "linear" and then proceeds to the imposition of this definition on the models that we use in evaluation. After various AI queries and reading dictionary definitions, I'll proceed with a definition of linearity as:

> A structured, step-by-step process that follows a straight, logical, and sequential path along a cause-and-effect sequence.

Non-linear Behavior of Seemingly Linear Models

The above definition leads us to believe that linear models are not system based. I see two difficulties. First, why can't a system be sequential? More important, Complexity Science tells us that even seemingly linear systems can exhibit complex behavior. I'll illustrate this with two examples, one involving feedback and one involving causal relationships between program outcomes. Let's begin with **Error! Reference source not found.**, an exceedingly simple toy model.

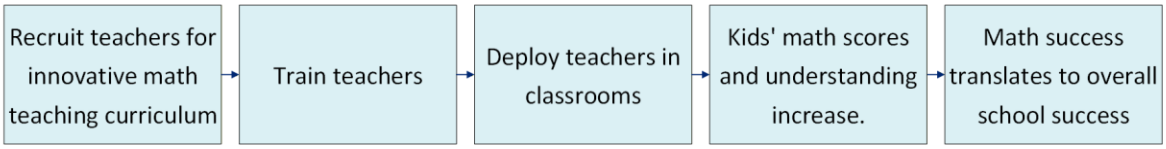

Figure 1: Toy Example Math Program with no Feedback

Figure 1 is truly linear.

- If we recruit teachers, then we can train them.
- If we train the teachers, then we can deploy them in classrooms.
- If we deploy the trained teachers in classrooms, then kid's math scores will improve.'
- If kids' math scores improve, then there will be generalized educational benefit.

Adding a Single Feedback Process

Many elaborations of **Error! Reference source not found.** spring to mind, but I'll pick one. If math scores improve, the success of the program will become known, and it will become easier to recruit teachers (Figure 2).

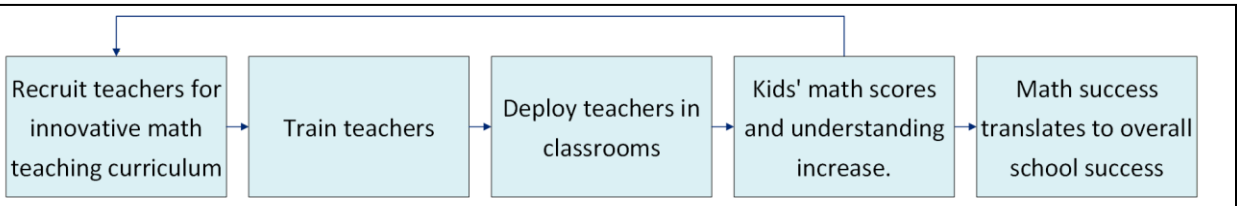

Figure 2: Toy Example Math Program with Feedback

Figure 2 is still mostly linear, but the addition of that one puny little feedback loop can induce nonlinear behavior. Any feedback loop can. In this case, program success might disproportionately increase the number of teachers who want to join the program. Evaluators might not recognize the implications of their feedback loops for program theory or for their methodologies, but they should. By adding even a single feedback loop they are constructing system-based complexity-inducing models.

Causal Relationships Among Outcomes

Ponder the outcomes. We see an arrow from math success to school success. But that arrow says nothing about the shape of the relationship. To my mind, casual inspection of that arrow carries the implication that there is a 1:1 relationship between change in the first outcome and change in the second. A moment's reflection is all that is needed to conjure other possibilities. For instance, I can think of all kinds of reasons why math success shows only a minimal (or no) influence on school success for quite a while, and then the effect becomes dramatic. There is nothing linear about that, but glancing at the model certainly conveys a sense of linearity. It would be nice if we hypothesized the nature of that relationship in advance. So doing would be a

worthwhile contribution to program theory. It would also inform our data analysis plans. But the fact that we do not specify the pattern of the relationship only conceals the complex system driven nonlinearity that can crop up in what looks like a linear model.

System-based Analysis of Models that Appear to be Linear

My contention is that all models, even those that seem linear, are in fact system models and can be treated as such. But what does it mean to treat them as such? The answer is that the evaluation design can reflect a chosen system view. If it is a system, evaluate it as a system. Here are two examples, again drawing on the toy model shown in Figure 2.

Example 1: Stock and Flow

A stock and flow analysis of the system depicted in Figure 2 would entail analysis of means and variances over time for:

- The rate at which teachers are recruited.
- The number waiting to be trained.
- Retention rates for teachers during training. It's reasonable to expect some to drop out.
- The number of trained teachers who accept teaching jobs in classrooms. Not all will.
- The number of teachers who stay in their jobs. Some will not like the work or have other reasons to go.

Compare what we can learn by adding this system-based view to a traditional evaluation. The traditional evaluation would tell us how each of the elements in the causal chain behaved. Was teacher training successful? Did kids' math scores improve? And so on. But a stock and flow-based systems evaluation would tell us much else. For instance, it would tell us not just that teacher training succeeded in terms of teacher skills, but also about the movement of teachers into and out of training, over time, as the program proceeds. That data would be useful for understanding the program. But the possibility and value of having that data would only be revealed by taking a system view of a model that we originally derided as merely linear, and in need of being replaced by a system-based model. In fact that linear model was always a system model.

Example 2: Extending the Stock and Flow to Complex System Behavior

Evaluators could look at the joint distribution of all those means and variances to get an overall picture of how that program operates, how it "moves around" in a program behavior space. Or put in complex systems terms, what is the attractor space that characterizes the program as a whole? Such an exercise would reinterpret the stock and flow data in complex systems terms. Each variable (drawn from the stock and flow data) would be considered as a dimension in a state space. The joint distribution would represent the likelihood of the system being in any particular part of that space. Such knowledge would provide insight on the limits and likely status of the program over time.

But as with the stock and flow example, the value of having that data would only be revealed by taking a system view of a model that we originally dismissed as merely linear, and in need of being replaced by a system-based model. In fact that linear model was always a system model.

Minimal Added Burden for Data Collection and Methodology

There is a real but manageable evaluation burden in evaluating programs in system terms. Consider the math education example. Doing a system-based analysis would require collecting data that would be already available or collected with minimal extra effort. The feedback between kids' math scores and teacher recruitment? Most of the data on recruitment and math scores would be collected anyway. The only additional data requirement would involve information on whether that hypothesized feedback process was indeed operating. Collecting that additional data does not seem like much of an extra effort.

As for the means and variances – any well-constructed evaluation would have provisions to use the administrative and monitoring data that would reside in any well-run program's information systems. It would be necessary to extract that data in usable form, and I don't want to minimize the difficulties in doing this. But the data are there, and evaluators have plenty of experience in extracting it.

In Sum

When models look linear, we think that they do not capture system behavior. This is not true, or at least it is often not true. It may be appropriate to construct models that depict seemingly linear processes. That's fine as long as we recognize the system behavior that can be contained in those models.