[MINITUTORIAL]

Interactions – and why forecasters should care

Stephan Kolassa

In forecasting, we often want to take dynamics or predictors into account. These can be "time series-like," such as yearly or day-of-week seasonality or trends, or they can be "causal" or "external" drivers, like promotions, prices, or the weather. Causal methods, from various flavors of regression over Deep Learning to tree-based methods, can account for all of these.

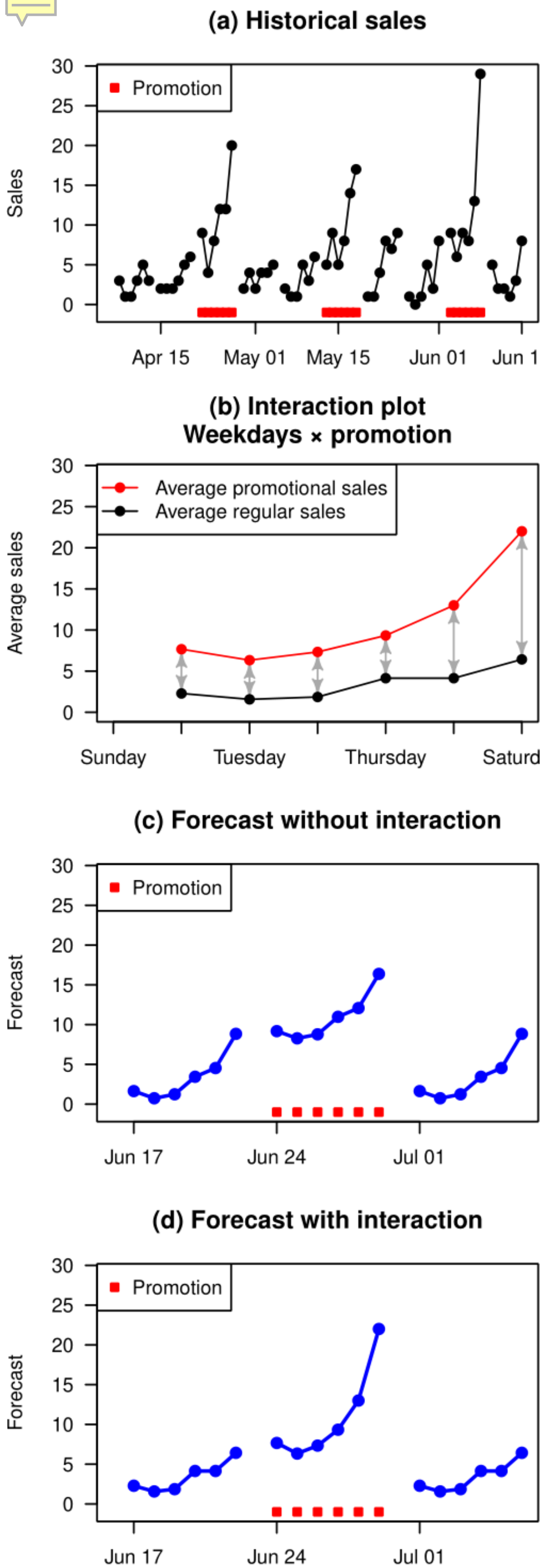
In the simplest model, explanation and forecast is fully additive. "Baseline sales are 10. If it's Saturday, add 5. If there is a promotion, add 15." In a regression model, the predictors are known as *main effects*. The impact of the day-of-week effect is completely separate from the promotion effect. Put differently, whenever we forecast a Saturday, our uplift of 5 is independent of whether there is a promotion or not, and the other way around: whenever we forecast for a promotion, we add an uplift of 15, regardless of the day-of-week. Thus, in this "additive" scenario, a promotion on a Saturday would mean an uplift of 20.

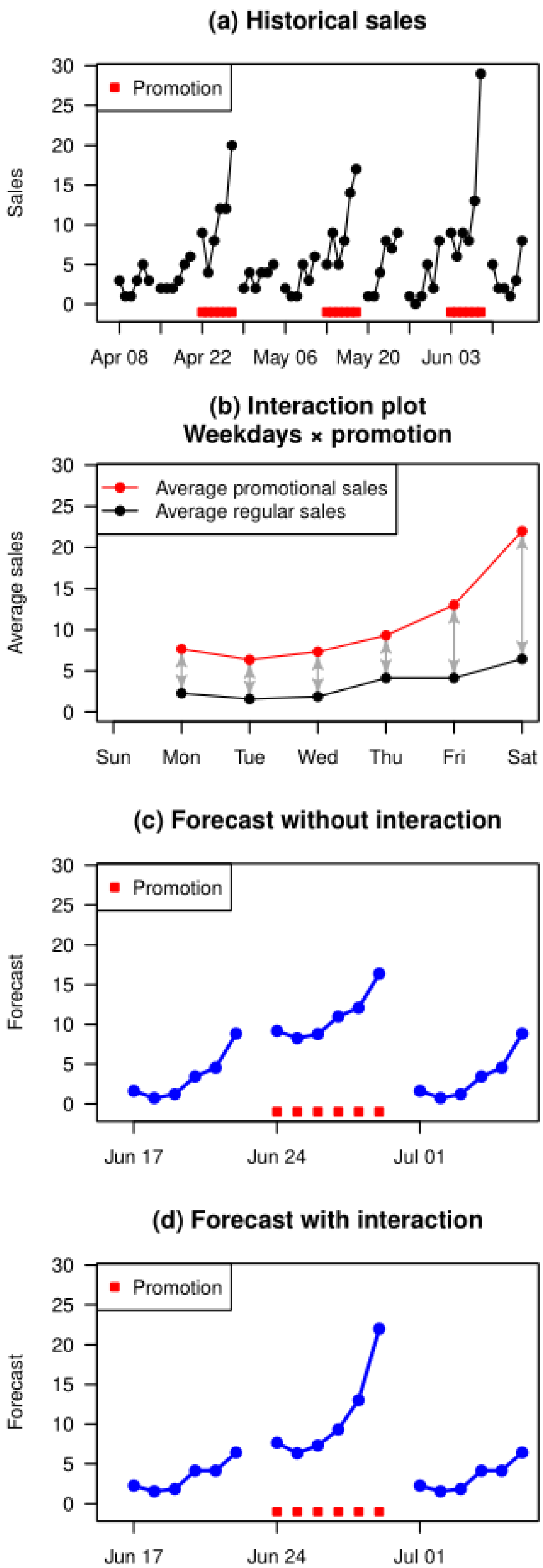
The additive scenario, however, may be too simplistic. Predictors *interact*. A promotion will usually have a higher impact on a day that has higher sales in the first place. We can also say that the impact of one predictor (promotion) depends on the value of a different predictor (day-of-week). In regression modeling parlance, this is called an *interaction effect*. Other examples of interactions can be between time of year and promotions (promotions on ice cream have a higher impact in summer than in winter) or between different promotions (promotion A has a higher – or lower! – impact if we are simultaneously running promotion B).

Panel (a) in **Figure 1** shows simulated daily sales with a promotion. (We are assuming a closed store on Sunday, which is common across much of Europe and here allows us to distinguish between weeks more easily.) At first glance, we see that the promotion seems to result in an uplift across the week. Panel (b) shows average sales plotted against day-of-week separately by promotion – an *interaction plot*. This deeper look reveals that the promotional uplift is larger on Friday and Saturday. Panel (c) shows a forecast for three weeks, one of them with a promotion, where we only model the main effects, but not the interaction. We see the same day-of-week pattern forecasted for each week, just shifted up by a constant (additive) amount during the promotion. Finally, panel (d) shows the forecast from a model *with* an interaction. The forecast for the promotional week is no longer a simple additive shift of the nonpromotional forecast; rather, the entire shape changes. The difference between (c) and (d) is typical for forecasts that exclude or include interactions.

How do we actually model interactions? In a regression (or a regression with ARIMA errors), we need to model them explicitly. In standard statistical software, like R or Python's statsmodels, we do this by combining predictors using a "*" sign rather than a "+" sign (the "+" sign only models main effects). Under the hood, this tells our software to take componentwise products of the relevant design matrix columns and append the result as new columns. In contrast, Machine Learning (ML) methods like neural networks and tree-based methods will account for interactions automatically.

Figure 1. Historical sales and the impact of interactions.


(a) Historical sales
Promotion
Sales
Apr 15
May 01
May 15
Jun 01
Jun 1
(b) Interaction plot
Weekdays × promotion
Average promotional sales
Average regular sales
Average sales
Sunday
Tuesday
Thursday
Saturd
(c) Forecast without interaction
Promotion
Forecast
Jun 17
Jun 24
Jul 01
(d) Forecast with interaction
Promotion
Forecast
Jun 17
Jun 24
Jul 01


(a) Historical sales
Promotion
Sales
0 5 10 15 20 25 30
Apr 08 Apr 22 May 06 May 20 Jun 03
(b) Interaction plot
Weekdays × promotion
Average promotional sales
Average regular sales
Average sales
0 5 10 15 20 25 30
Sun Mon Tue Wed Thu Fri Sat
(c) Forecast without interaction
Promotion
Forecast
0 5 10 15 20 25 30
Jun 17 Jun 24 Jul 01
(d) Forecast with interaction
Promotion
Forecast
0 5 10 15 20 25 30
Jun 17 Jun 24 Jul 01

One drawback of modeling interactions is that because of the much more flexible model, we need much more data to fit it well. This aligns well with the dominant philosophy of ML methods as “global” – meaning that we fit the model to many time series at once, rather than separately to each series. Incidentally, note that models like linear regression can also be fitted in a global way by using pooled regression or mixed effects models; and that panel regression is also by definition global. Thus, interactions between predictors can be very naturally included in such models. Another point to keep in mind is that an interaction model is a bit harder to explain and to communicate.

In takeaway, the next time you have to forecast something with multiple predictors, consider whether there might be interaction effects. And, if so, be prepared to model them.

[PHOTO & BIO on file]