Regression Interaction and Prediction

PSCI 1801 · Class Handout

Dr. Marc Trussler

This handout is designed as an in-class supplement to the textbook. The material contained on this handout does not represent the full set of information you need to know for the exams. Exclusively studying off of this handout will not be sufficient for the exams.

Interaction

A polynomial says: "the effect of x changes with the value of x itself." An interaction *says: "the effect of x changes with the value of some* other *variable z."*

Model:

$$\hat{y} = \hat{\alpha} + \hat{\beta}_1 x + \hat{\beta}_2 z + \hat{\beta}_3 (x \cdot z)$$

R syntax: `lm(y ~ x*z)` *— the* `*` *automatically includes both main effects and the interaction.*

Interpreting an interaction

Take partial derivatives to get the effect of each variable:

$$\frac{\partial y}{\partial x} = \hat{\beta}_1 + \hat{\beta}_3 z$$

$$\frac{\partial y}{\partial z} = \hat{\beta}_2 + \hat{\beta}_3 x$$

The effect of x depends on z. The effect of z depends on x. Neither variable has a single "effect" — you have to state the effect at a specific value of the other.

Key things to remember

1. *The coefficient on x alone is the effect of x* when $z = 0$. *This might or might not be a meaningful quantity depending on what z is.*
2. *The interaction coefficient $\hat{\beta}_3$ is the difference in slopes across levels of z. Its hypothesis test tells you whether the effect of x meaningfully changes across z.*
3. *You must include the "main effects" (x and z separately) whenever you include an interaction. R does this automatically with `x*z`.*

Predicting with Regression

Two purposes for regression: (a) explanation *— what's the effect of x on y? (b)* prediction *— given new x values, what do we expect y to be?*

The math is the same. But prediction adds two considerations: uncertainty around the predicted value, and the danger of extrapolating outside the range of your data.

Hand-calculating predictions

$$\hat{y}_{\text{new}} = \hat{\alpha} + \hat{\beta}_1 x_{\text{new},1} + \hat{\beta}_2 x_{\text{new},2} + \ldots$$

Uncertainty around the prediction depends on covariance among the coefficients — hard to do by hand for anything beyond univariate.

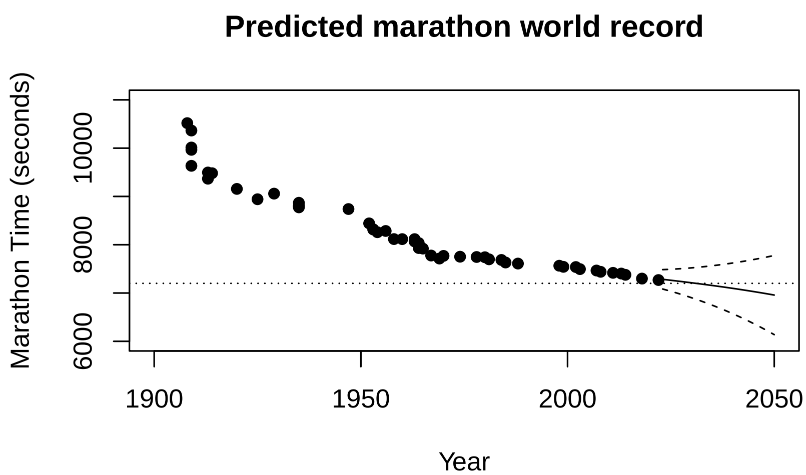
R's `predict()` function

`predict()` does the math (and standard errors) for you:

`interval = "confidence"` gives a CI around the fitted line. `interval = "prediction"` gives a (much wider) CI for an individual new observation.

Example — marathon world records

A linear fit is too crude — record improvements slow as records get faster. Fit a cubic instead and use `predict()` *for future years:*



Prediction: 2-hour barrier around 2030, with wide CIs by the late 2020s.

Out-of-Sample Prediction

If we predict on data we used to fit the model, we'll overstate accuracy — the model is trained to fit those points.

The real test of a prediction model: how well does it work on data it hasn't seen?

Train/test split

Standard approach: hold out some fraction of your data (say 30%) as a test set. *Fit the model on the remaining 70% (the* training set *). Compare predictions on the test set to their actual values.*

Root mean squared error (RMSE) on the test set is a common performance metric.

Caveats on Prediction

The model assumes current trends continue. It cannot anticipate:

- Regime shifts (technology, policy, methodology changes)
- Interventions that didn't exist in training data
- Anything outside the range of observed values

Extrapolation is always dangerous. A polynomial fit that looks perfect within-sample can go wildly wrong outside the training range.

In this course we use regression mostly for explanation — *asking whether x has an effect on y. Prediction is the same math with more warnings.*