Non-linearity and Feature Expansion

Ryan Miller

Introduction

- Linear regressions and variants like logistic/softmax regression often lack the flexibility to learn certain patterns
  - Sometimes this bias can be beneficial due to the corresponding reduction in variance, but in other situations it can be problematic
- This lecture introduces a few strategies to add flexibility to linear models (ie: reduce their bias)

Example

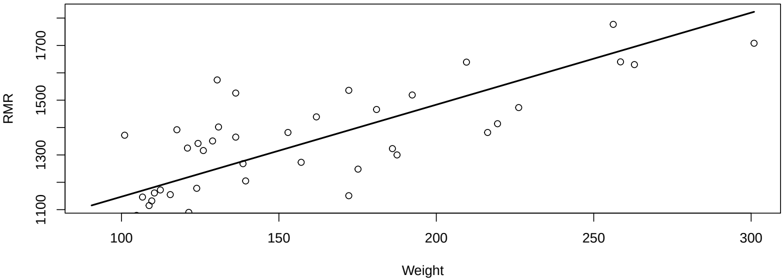
Consider a simple application aiming to predict the resting metabolic rate of an individual using their bodyweight.

weight_lbs	rate_kcal
104.79	1079
106.68	1146
108.78	1115
110.46	1161
120.96	1325
128.94	1351

The raw data might consist of a single predictor and the outcome.

Example

We might consider a simple linear model to represent this relationship:



How is it doing?

Feature Expansion

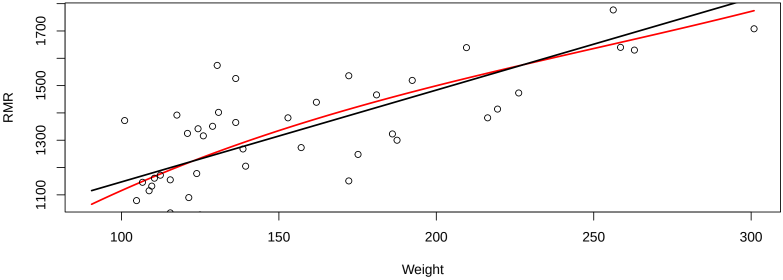
To add flexibility to our linear model, we might *expand* our single predictor using polynomials:

X1	X2	X3	rate_kcal
104.79	10980.94	1150693	1079
106.68	11380.62	1214085	1146
108.78	11833.09	1287203	1115
110.46	12201.41	1347768	1161
120.96	14631.32	1769805	1325
128.94	16625.52	2143695	1351

We're now using 3 columns to represent an individual's weight. They are: weight, weight squared, and weight cubed.

Polynomial Expansion

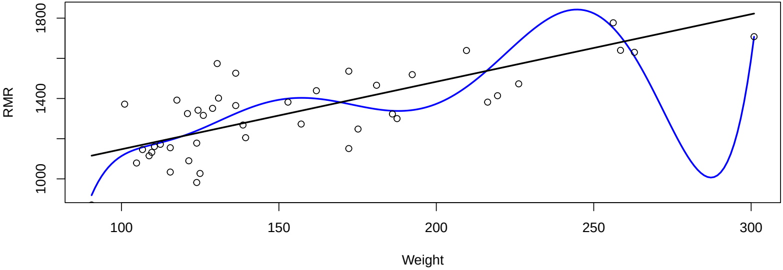
A linear regression model fit to the expanded data is shown in red:



Did feature expansion improve the accuracy of our model on the training data?

Bias-Variance Tradeoff

Because our second model estimated 3 weight parameters (and 1 bias) from the data, it has greater flexibility to represent small trends.



However, more flexibility is not always better, as the blue line above depicts and 8th degree polynomial expansion. What's wrong with this model?

Discretization

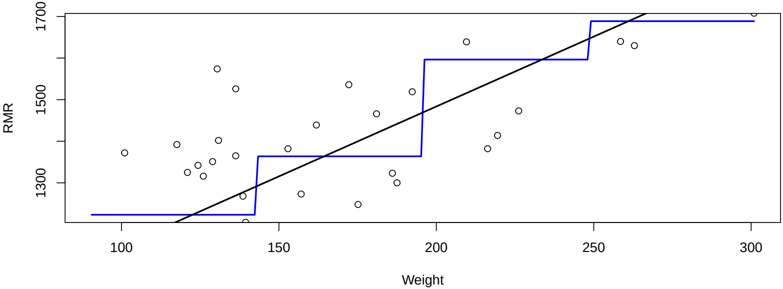
A simple alternative to polynomial expansion is *discretization*:

(90.3,143]	(143,196]	(196,248]	(248,301]	rate_kcal
1	0	0	0	1079
1	0	0	0	1146
1	0	0	0	1115
1	0	0	0	1161
1	0	0	0	1325
1	0	0	0	1351

The idea is to split a numeric predictor in to categories and represent them using one-hot encoding.

Discretization

The discretizing weight into 4 equally spaced bins yields the following model:



What are some strengths/weaknesses of this approach?

Splines

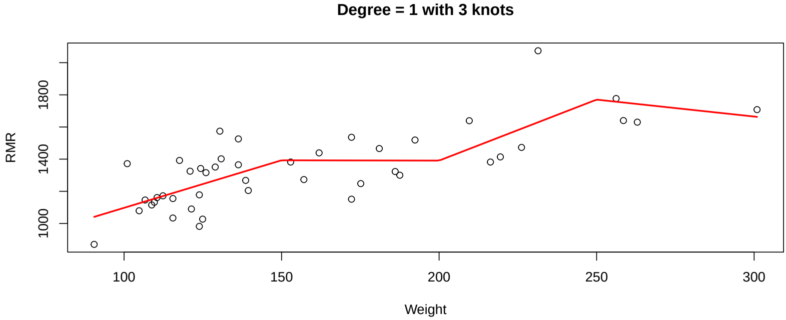
Splines are an alternative without many of the negative aspects of polynomials and descretization:

X1	X2	X3	rate_kcal
0.1768971	0.0128790	0.0003126	1079
0.1964682	0.0163547	0.0004538	1146
0.2172099	0.0206527	0.0006546	1115
0.2330538	0.0244103	0.0008523	1161
0.3175760	0.0537322	0.0030304	1325
0.3660471	0.0817907	0.0060919	1351

Basis splines, or "b-splines", use a basis matrix to represent piecewise polynomials that must connect at certain interior knots

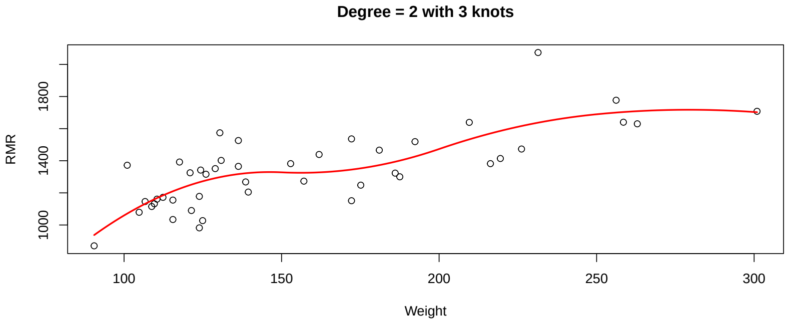
B-splines

Polynomials with degree $= 1$ are just lines, the model below demonstrates a b-spline with 3 knots and degree $= 1$:



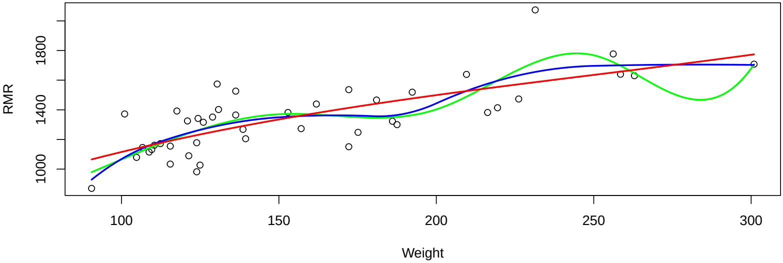
B-splines

Higher degree splines ensure smoothness by requiring continuity of derivatives up to the order degree-1 (so quadratic splines require continuity of the first derivative, or slope at the location of the knot):



B-splines

The polynomial degree and number of knots can be used to manipulate the flexibility of b-splines:



The red line uses 3rd degree polynomial expansion, the green is a 3rd degree basis spline expansion with knots at 150, 200, and 250 lbs, the blue reduces the degree to 2 and introduces additional knots at 1110 and 180 lbs.

Closing Remarks

- Splines are useful when inspection of the relationship between a predictor and the outcome shows evidence of non-linearity
  - The only scenario in which polynomials might be justified over splines are models intended to be explained to a non-technical audience
- Discretization is also generally less preferable to splines, but it might make sense in applications where clear thresholds are used
  - For example, a grade of 70% might be substantially different from a grade of 69%, but 71% or 72% might be roughly the same as a 70%