Polynomial Effects

2020-01-17

Preparation

In this set of notes, you will learn one method of dealing with nonlinearity. Specifically, we will look at the including polynomial effects into a model. To do so, we will use the *mn-schools.csv* dataset (see the data codebook) to examine if (and how) academic "quality" of the student-body (measured by SAT score) is related to institutiional graduation rate.

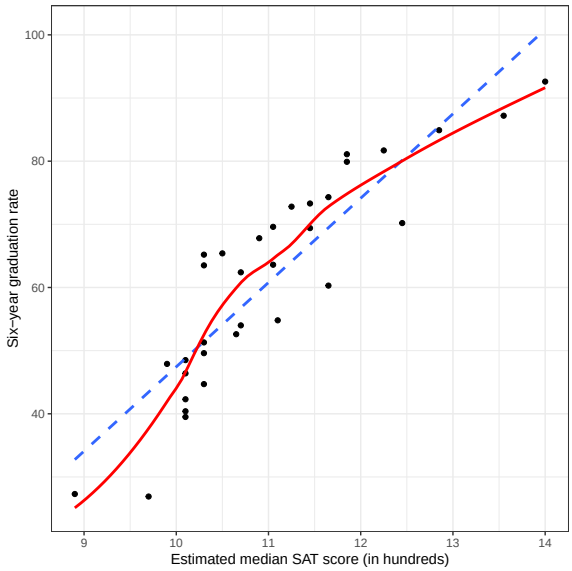
```
# Load libraries
library(broom)
library(corrr)
library(educate) #Need version 0.1.0.1
library(tidyverse)

# Read in data
mn = read_csv(file = "~/Documents/github/epsy-8252/data/mn-schools.csv")
```

Examine Relationship between Graduation Rate and SAT Scores

As always, we begin the analysis by graphing the data.

```
# Scatterplot
ggplot(data = mn, aes(x = sat, y = grad)) +
    geom_point() +
    geom_smooth(method = "lm", se = FALSE, linetype = "dashed") +
  geom_smooth(method = "loess", se = FALSE, color = "red") +
    theme_bw() +
  xlab("Estimated median SAT score (in hundreds)") +
  ylab("Six-year graduation rate")
```

Figure 1
Six-Year Graduation Rate as a Function of Median SAT Score



Note. The regression (blue, dashed line) and the loess (red, solid line) smoothers are also displayed.

The loess smoother suggests that the relationship between SAT scores and graduation rate may be non-linear. Nonlinearity implies that a the effect of SAT on graduation rates is not constant across the range of SAT scores; for colleges with lower values of SAT (say SAT < 1100) the effect of SAT has a rather high, positive effect (steep slope), while for colleges with higher values of SAT (≥ 1100) the effect of SAT is positive and moderate (the slope is less steep). Another way of saying this is that for schools with lower SAT scores, a one-unit difference in SAT is associated with a larger change in graduation rates than the same one-unit change for schools with higher SAT values.

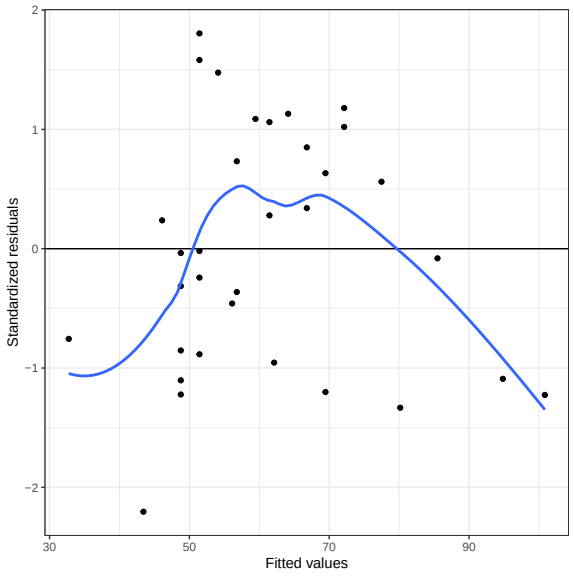
Residual Plot: Another Way to Spot Nonlinearity

Sometimes, the nonlinear relationship is difficult to detect from the scatterplot of Y versus X. Often it helps to fit the linear model and then examine the assumption of linearity in the residuals. It is sometimes easier to detect nonlinearity in the plot of the residuals versus the fitted values.

```
# Fit linear model
lm.1 = lm(grad ~ 1 + sat, data = mn)

# Obtain residuals
out = augment(lm.1)

# Examine residuals for linearity
ggplot(data = out, aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  geom_smooth(se = FALSE) +
  theme_bw() +
  xlab("Fitted values") +
  ylab("Standardized residuals")
```

Figure 2

Standardized Residuals versus the Fited Values for a Model Regressing Six-Year Graduation Rates on Median SAT Scores



Note. The line $Y = 0$ (black) and the loess smoother (blue) are also displayed.

This plot suggests that the assumption of linearity may be violated. This plot suggests that the average residual is not zero at each fitted value. For low fitted values it appears as though the average residual may be less than zero, for moderate fitted values it appears as though the average residual may be more than zero, and for high fitted values it appears as though the average residual may be less than zero.

Notice that the pattern displayed in the residuals is consistent with the pattern of the observed data in the initial scatterplot (Figure 1). If we look at the data relative to the regression smoother we see that there is not even vertical scatter around this line. At low and high SAT scores the observed data tends to be below the regression line (the regression is over-estimating the average graduation rate), while for moderate SAT scores the observed data tends to be above the regression line (the regression is under-estimating the average graduation rate).

Polynomial Effects

One way of modeling non-linearity is by including polynomial effects. In regression, polynomial effects are predictors that have a power greater than one. For example, x^2 (quadratic term), or x^3 (cubic term). Note that

$$x^2 = x \times x.$$

So the quadratic term, x^2 is a product of x times itself. Recall that products are how we express interactions. Thus the quadratic term of x^2 is really the interaction of x with itself. To model this, we simply (1) create the product term, and (2) include the product term and all constituent main-effects in the regression model.

```
# Create quadratic term in the data
mn = mn %>%
  mutate(
    sat_quadratic = sat * sat
  )
```

```
# View data
head(mn)
```

```
# A tibble: 6 x 7
     id name                              grad public   sat tuition sat_quadratic
  <dbl> <chr>                            <dbl>  <dbl> <dbl>   <dbl>         <dbl>
1     1 Augsburg College                  65.2      0  10.3    39.3          106.
2     3 Bethany Lutheran College          52.6      0  10.6    30.5          113.
3     4 Bethel University, Saint Paul,~  73.3      0  11.4    39.4          131.
4     5 Carleton College                  92.6      0  14      54.3          196
5     6 College of Saint Benedict         81.1      0  11.8    43.2          140.
6     7 Concordia College at Moorhead     69.4      0  11.4    36.6          131.
```

```
# Fit model
lm.2 = lm(grad ~ 1 + sat + sat_quadratic, data = mn)

# Model-level output
glance(lm.2)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.835         0.824  7.02      76.0 1.81e-12     3  -110.  227.  233.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Coefficient-level output
tidy(lm.2)
```

```
# A tibble: 3 x 5
  term          estimate std.error statistic  p.value
  <chr>            <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)    -366.      98.6       -3.71 0.000831
2 sat              62.7     17.3        3.63 0.00104
3 sat_quadratic    -2.15     0.751     -2.86 0.00756
```

Since this is an interaction model, we start by examining the interaction term; the quadratic coefficient. The statistical evidence for this term indicates that the empirical data are inconsistent with the hypothesis of no quadratic effect ($p = .008$). This suggests that the quadratic term explains variation in six-year graduation rates above and beyond the linear term. To see this, we can compare the model-level output from the quadratic model to that of the linear effect model.

```
# Model-level output (linear effect)
glance(lm.1)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.790         0.783  7.79      117. 4.94e-12     2  -114.  233.  238.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Model-level output (linear and quadratic effects)
glance(lm.2)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.835         0.824  7.02      76.0 1.81e-12     3  -110.  227.  233.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

The linear-effect model explains 79.0% of the variation in six-year graduation rates. When we also include the quadratic effect of SAT to the model, the explained variation increases to 83.5%. Although the improvement of 4.5% seems modest, the p-value for the quadratic effect ($p = .008$) suggests that this difference in explained variation is more than we expect because of sampling error.

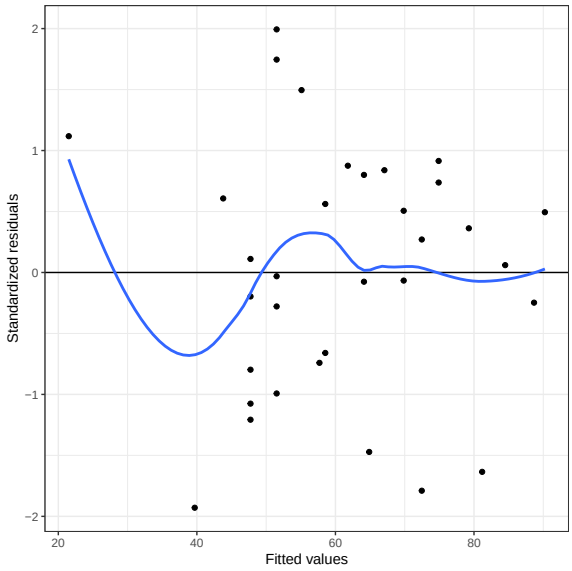
Are the Assumptions More Tenable for this Model?

More important than whether the p-value is small, is whether including the quadratic effect improved the assumption violation we noted earlier. To evaluate this, we will examine a plot of the standardized residuals versus the fitted values for the quadratic model.

```
# Obtain residuals
out_2 = augment(lm.2)

# Examine residuals for linearity
ggplot(data = out_2, aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  geom_smooth(se = FALSE) +
  theme_bw() +
  xlab("Fitted values") +
  ylab("Standardized residuals")
```

Figure 3

Standardized Residuals versus the Fited Values for a Model Regressing Six-Year Graduation Rate on Linear and Quadratic Effects of Median SAT Scores



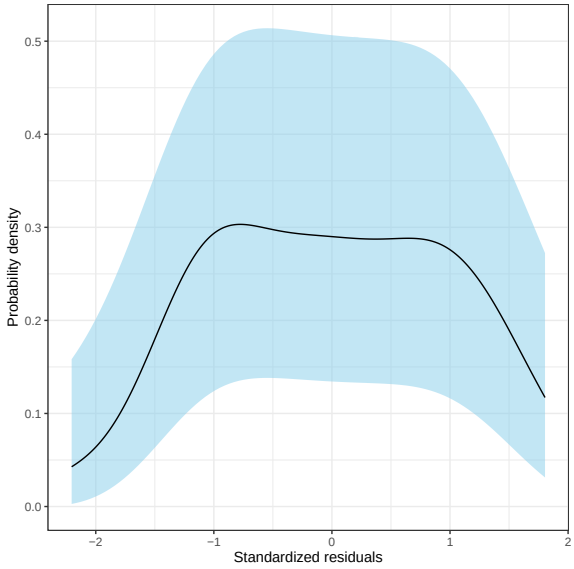
Note. The line $Y = 0$ (black) and the loess smoother (blue) are also displayed.

Note that the residuals in this plot are far better behaved, indicating much more consistency with the assumption that the average residual is zero at each fitted value. This is the evidence that we would use to justify retaining the quadratic effect in the model.

We can also examine the assumption of residual normality for the model. We will use the `stat_density_confidence()` function from the **educate** package to create a confidence envelope for a normal reference distribution. The density of the marginal residuals seems consistent with the assumption of normality for this model.

Figure 4

Density Plot of the Standardized Residuals for a Model Regressing Six-Year Graduation Rate on Linear and Quadratic Effects of Median SAT Scores



Note. The pointwise confidence envelope for a normal reference distribution (blue shaded area) is also displayed.

Interpretation of a Polynomial Term

How do we interpret the quadratic effect of SAT? First, we will write out the fitted model.

$$\widehat{\text{Graduation Rate}}_i = -366.34 + 62.72(\text{SAT}_i) - 2.15(\text{SAT}_i^2)$$

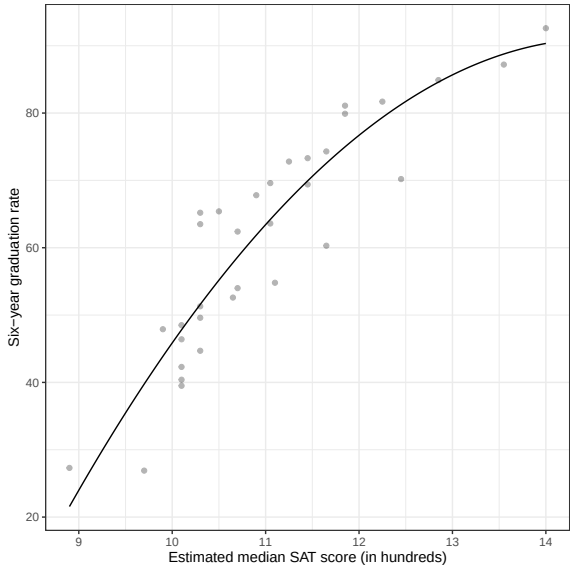
Since the quadratic term is an interaction, we can interpret this term as we do any other interaction, namely that the effect of median SAT score on six-year graduation rates depends on the magnitude of median SAT score. To better understand the nature of this effect we will plot the fitted equation and use the plot to aid our interpretation.

Graphical Interpretation

To plot the fitted equation, we use the `stat_function()` layer to add the fitted curve. (Note that we can no longer use `geom_abline()` since the addition of the polynomial effect implies that a line is no longer suitable.) This layer takes the argument `fun=` which describes a function that will be plotted as a line. To describe a function, use the syntax `function(){...}`. Here we use this syntax to describe the function of `x` (which in the `ggplot()` global layer is mapped to the `sat` variable). The fitted equation is then also written in terms of `x` and placed inside the curly braces. (Note that it is best to be more exact in the coefficient values—don't round—when you create this plot as even minor differences can grossly change the plot.)

```
# Scatterplot
ggplot(data = mn, aes(x = sat, y = grad)) +
    geom_point(alpha = 0.3) +
    stat_function(fun = function(x) {-366.34 + 62.72*x - 2.15 * x^2} ) +
    theme_bw() +
  xlab("Estimated median SAT score (in hundreds)") +
  ylab("Six-year graduation rate")
```

Figure 5
Scatterplot of Six-Year Graduation Rate versus Median SAT Score with Prediction Curve



This plot helps us interpret the nature of the relationship between median SAT scores and graduation rates. The effect of median SAT score on graduation rate depends on SAT score (definition of an interaction). For schools with low SAT scores, the effect of SAT score on graduation rate is positive and fairly high. For schools with high SAT scores, the effect of SAT score on graduation rate remains positive, but it has a smaller effect on graduation rates; the effect diminishes.

Algebraic Interpretation

From algebra, you may remember that the coefficient in front of the quadratic term (-2.2) informs us of whether the quadratic is an upward-facing U-shape, or a downward-facing U-shape. Since our term is negative, the U-shape is downward-facing. This is consistent with what we saw in the plot. What the algebra fails to show is that, within the range of SAT scores in our data, we only see part of the entire downward U-shape.

This coefficient also indicates whether the U-shape is skinny or wide. Although "skinny" nd "wide" are only useful as relative comparisons. Algebraically, the comparison is typically to a quadratic coefficient of 1, which is generally not useful in our interpretation. The intercept and coefficient for the linear term help us locate the U-shape in the coordinate plane (moving it right, left, up, or down from the origin). You could work all of this out algebraically. You can see how different values of these coefficients affect the curve on Wikipedia.

What can be useful is to find where the minimum (upward facing U-shape) or maximum (downward facing U-shape) occurs. This point is referred to as the *vertex* of the parabola and can be algebraically determined. To do this we determine the x-location of the vertex by

$$x_{\text{Vertex}} = -\frac{\hat{\beta}_1}{2 \times \hat{\beta}_2}$$

where, $\hat{\beta}_1$ is the estimated coefficient for the linear term and $\hat{\beta}_2$ is the estimated coefficient for the quadratic term. The y coordinate for the vertex can then be found by substituting the x-coordinate into the fitted equation. For our example,

$$x_{\text{Vertex}} = -\frac{62.72}{2 \times -2.15} = 14.58$$

and

$$y_{\text{Vertex}} = -366.34 + 62.72(14.58) - 2.15(14.58^2) = 91.08$$

This suggests that at a median SAT score of 1458 we predict a six-year graduate rate of 91.08. This x-value also represents the value at which the direction of the effect changes. In our example recall that for higher values of SAT the effect of SAT on graduation rate was diminishing. This is true for schools with median SAT scores up to 1458. For schools with higher SAT scores the effect of SAT score on graduation rate would theoretically be negative, and would be more negative for higher values.

This is all theoretical as our data only includes median SAT scores up to 1400. Everything past that value (including the vertex) is extrapolation. Extrapolation is exceedingly sketchy when we start fitting non-linear models. For example, do we really think that the average graduation rate for schools with a median SAT scores higher than 1458 would actually be smaller than for schools at 1458? It is more likely that the effect just flattens out.

Fit the Polynomial Term using the I() Function

We can also use a method of fitting polynomial terms directly in the `lm()` function. To do this, we create the polynomial directly in the model using the `I()` function. When you use the `I()` function to create the polynomial term, you do not need to create a new column of squared values in the data set. Here we fit the same quadratic model using this method of fitting the model. (Reminder: You cannot use the colon notation (`:`) to fit a polynomial term in the model.)

```
# Fit model using I() function
lm.2 = lm(grad ~ 1 + sat + I(sat ^ 2), data = mn)

# Model-level output
glance(lm.2)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.835         0.824  7.02      76.0 1.81e-12     3  -110.  227.  233.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Coefficient-level output
tidy(lm.2)
```

```
# A tibble: 3 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)  -366.      98.6       -3.71 0.000831
2 sat            62.7     17.3        3.63 0.00104
3 I(sat^2)       -2.15     0.751     -2.86 0.00756
```

Adding Covariates

We can also include covariates in a polynomial model (to control for other predictors), the same way we do in a linear model, by including them as additive terms in the `lm()` model. Below we include the `public` dummy-coded predictor to control for the effects of sector.

```
# Fit model
lm.3 = lm(grad ~ 1 + sat + I(sat^2) + public, data = mn)

# Model-level output
glance(lm.3)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.897         0.886  5.64      84.1 2.05e-14     4  -102.  214.  221.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Coefficient-level output
tidy(lm.3)
```

```
# A tibble: 4 x 5
  term        estimate std.error statistic   p.value
  <chr>          <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept)  -384.      79.4       -4.84 0.0000398
2 sat            67.0     13.9        4.81 0.0000425
3 I(sat^2)       -2.37     0.606     -3.91 0.000507
4 public         -9.12     2.19      -4.17 0.000251
```

Based on these results, the quadratic effect of SAT on graduation rates seems to persist, even after controlling for differences in sector ($p = 0.0005$). This means that after controlling for sector differences, the effect of median SAT score on graduation rate depends on the level of the median SAT score. Alternatively, the effect of median SAT score on graduation rate is different for different median SAT scores.

There also seems to be an effect of sector on graduation rate after controlling for the linear and quadratic effects of SAT ($p = 0.003$). This effect suggests that, after controlling for the linear and quadratic effects of median SAT score, public schools have a graduation rate that is 9.1 percentage points lower than private schools, on average. Again, we can plot the fitted model to aid interpretation. To do so, we determine the fitted equations for both public and private schools.

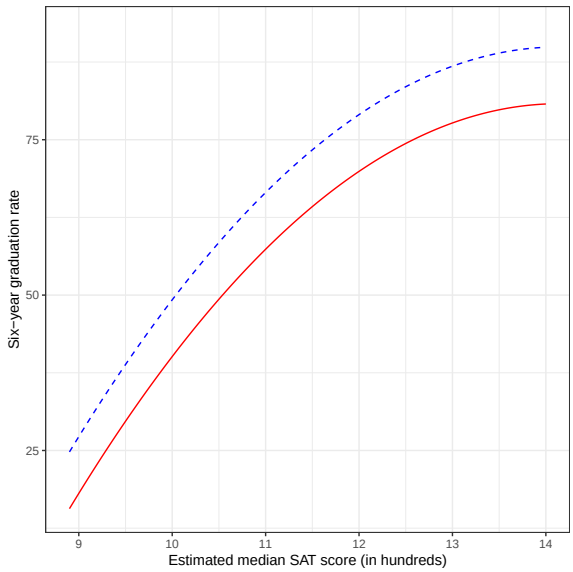
$$\textbf{Public}: \ \widehat{\text{Graduation Rate}}_i = -384.16 + 67.04(\text{SAT}_i) - 2.37(\text{SAT}_i^2) - 9.12(0)$$
$$= -384.16 + 67.04(\text{SAT}_i) - 2.37(\text{SAT}_i^2)$$

$$\textbf{Public}: \ \widehat{\text{Graduation Rate}}_i = -384.16 + 67.04(\text{SAT}_i) - 2.37(\text{SAT}_i^2) - 9.12(1)$$
$$= -393.29 + 67.04(\text{SAT}_i) - 2.37(\text{SAT}_i^2)$$

We can then plot each fitted curve by including each in a separate `stat_function()` layer.

```
# Plot of the fitted model
ggplot(data = mn, aes(x = sat, y = grad)) +
    geom_point(alpha = 0) +
    stat_function(
      fun = function(x) {-384.16 + 67.04*x - 2.37 * x^2},
      color = "blue",
      linetype = "dashed"
      ) +
  stat_function(
    fun = function(x) {-393.29 + 67.04*x - 2.37 * x^2},
    color = "red",
    linetype = "solid"
    ) +
    theme_bw() +
  xlab("Estimated median SAT score (in hundreds)") +
  ylab("Six-year graduation rate")
```

Figure 6
Six-Year Graduation Rate as a Function of Median SAT Score for Private (Red, Solid Line) and Public (Blue, Dashed Line) Institutions



The plot shows the linear and quadratic effect of median SAT scores on graduation rate. In general, after controlling for sector differences, the effect of median SAT on graduation rates is positive (institutions with higher median SAT scores tend to have higher graduation rates. But, this effect diminishes for institutions with increasingly higher SAT scores. The main effect of sector is also visualized since private schools have higher graduation rates, on average, than public schools for all levels of median SAT score. This difference, regardless of median SAT score, is constantly that public schools have a lower graduation rate than private schools by 9.12 percentage points.

Interactions with Polynomial Terms

We can also fit interactions between predictors in polynomial models. In this example, an interaction between median SAT score and sector would indicate that the fitted curves for the public and private institutions are not parallel. In other words, the effect of median SAT score on graduation rates does not have the same positive, diminishing effect for both public and private institutions.

With polynomial models, there are many ways a potential interaction can play out. In our example, sector can interact with the linear effect of median SAT score or the quadratic effect of median SAT score. Remember that if we include an interaction, we need to include all lower order effects as well. In a polynomial model this means that if we include an interaction with a higher order polynomial term, we also needto include interactions with the lower order polynomial terms.

For us this implies that if we want to include an interaction between sector and the quadratic effect of median SAT score, we also need to include the interaction between sector and the linear effect of median SAT score. Because of this there are two potential interaction models we can fit.

```
# Interaction between sector and linear effect of SAT
lm.4 = lm(grad ~ 1 + sat + I(sat^2) + public + public:sat, data = mn)

# Interaction between sector and linear and quadratic effects of SAT
lm.5 = lm(grad ~ 1 + sat + I(sat^2) + public + public:sat + public:I(sat^2), data = mn)
```

We now have three candidate models describing the effects of median SAT scores and sector to graduation rates. In increasing levels of complexity these are:

1. Main effects model
2. Interaction effect between linear SAT term and sector
3. Interaction effect between quadratic SAT term and sector AND linear SAT term and sector

Which should we adopt? One exploratory approach to evaluating the potential interaction effects are to start with the model with the most complexity and pare it down by removing higher order terms that do not improve the explanatory power of the model.

The metric we use to evaluate explanatory power of the model (at least so far) isthe model-level R^2 value. For our three candidate models the R^2 values can be obtained from the `glance()` output.

```
# Main-effects model
glance(lm.3)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.897         0.886  5.64      84.1 2.05e-14     4  -102.  214.  221.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Interaction model (linear term)
glance(lm.4)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.906         0.893  5.48      67.5 5.71e-14     5  -100.  213.  222.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

```
# Interaction model (linear and quadratic terms)
glance(lm.5)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
1     0.907         0.890  5.55      52.9 4.22e-13     6  -100.  214.  225.
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

The main effects model explains 89.7% of the variation in graduation rates. The interaction model that includes an interaction between the linear SAT term and sector explains an additional 0.9% of the variation, and including an additional interaction between the quadratic SAT term and sector improves this explanation by another 0.1%.

Thus, the most complex model explains the most variation in the sample data. This is always the case so long as the predictors have any amount of correlation with the outcome. The bigger question is whether that improvement we see in the explanation of variability is more than what we would expect because of sampling error. To determine this we need to carry out an inferential analysis of the improvement in explained variation.

F-Test for Nested Models

One method of evaluating the interaction terms is to use an F-test to test the change in R^2. This test requires that of the two models being compared, one is nested within the other. Consider the following two models:

$$\textbf{Model 1}: \; Y_i = \beta_0 + \beta_1(X_{1i}) + \beta_2(X_{2i}) + \epsilon_i$$
$$\textbf{Model 2}: \; Y_i = \beta_0 + \beta_1(X_{1i}) + \beta_2(X_{2i}) + \beta_3(X_{1i})(X_{2i}) + \epsilon_i$$

Model 1 is said to be *nested in* Model 2 as it is a subset of Model 2. In general a model is nested in another model if setting a subset of the coefficients of the more complex model to zero results in the simpler model. In this example, Setting β_3 in Model 2 (the more complex model) to 0 results in Model 1 (the simpler model).

In our graduation rate example, the three candidate models are:

$$\textbf{Model 3}: \; \text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \epsilon_i$$
$$\textbf{Model 4}: \; \text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \beta_4(\text{SAT}_i)(\text{Public}_i) + \epsilon_i$$
$$\textbf{Model 5}: \; \text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \beta_4(\text{SAT}_i)(\text{Public}_i) + \beta_5(\text{SAT}_i^2)(\text{Public}_i) + \epsilon_i$$

Here Model 3 is nested within Model 4, which is, in turn, nested in Model 5. Because of this we can use an F-test to compare the change in R^2 between the different models. Because we are evaluating the *change in* R^2, sometimes this test is referred to as the *Delta F-test*, or symbollically, the ΔF-test.

In general, the null hypothesis that is tested when we compare nested models is:

$$H_0 : \rho^2_{\text{Complex Model}} = \rho^2_{\text{Simple Model}}$$

or

$$H_0 : \rho^2_{\text{Complex Model}} - \rho^2_{\text{Simple Model}} = 0$$

Recall that

$$R^2 = \frac{SS_{\text{Total}} - SS_{\text{Error}}}{SS_{\text{Total}}}$$

Recall further that the Sum of Squared Total does not change from model-to-model (so long as we use the same outcome and data to fit the model). Thus if we compare the R^2 values for two models, the only thing that would differ in the equation for their resulting R^2 values is the SSE. In the actual test, the SSE is what is compared between the two nested models.

To carry out the test in practice, we include both models in the `anova()` function. For example to compare the main effects model (`lm.3`) to the interaction model (linear term; `lm.4`), we use the following syntax.

```
anova(lm.3, lm.4)
```

```
Analysis of Variance Table

Model 1: grad ~ 1 + sat + I(sat^2) + public
Model 2: grad ~ 1 + sat + I(sat^2) + public + public:sat
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     29 923.80
2     28 842.23  1    81.567 2.7117 0.1108
```

The SSE, or Residual Sum of Squares, for each candidate model is shown in the `RSS` column. Based on these values the interaction model fits better than the main effects model as it has a lower SSE value. How much better is the fit of this model? The difference between these two SSE values is 81.57.

The residual degrees of freedom are a measure of each model's complexity. Smaller values indicate more complexity. Comparing these values for the two candidate models, the interaction model ($df = 28$) is more complex than the main effects model ($df = 29$). How much more complex is the interaction model than the main effects model? The difference between these two df values is 1, which indicates that the interaction model has one additional parameter (β_4) than the main effects model.

The F-test works by intially quantifying the relationship between the relative increase in explanation and the relative increase in complexity. This relative relationship is quantified as an F-statistic where,

$$F = \frac{\text{SSE}_{\text{Simple Model}} - \text{SSE}_{\text{Complex Model}}}{\text{SSE}_{\text{Complex Model}}} \times \frac{df_{\text{Complex Model}}}{df_{\text{Simple Model}} - df_{\text{Complex Model}}}$$

For our example,

$$F = \frac{923.80 - 842.23}{842.23} \times \frac{28}{29 - 28} = 2.71$$

The F-test evaluates this F-statistic using an F-distribution with numerator df equal to the difference in complexity and denominator df equal to the df in the more complex model.

$$F(1, 28) = 2.71$$

In this case, the resulting p-value is 0.1108. This suggests that the empirical data are consistent with the null hypothesis that there is no difference in the explained variation in the population of graduation rates between Model 3 and Model 4. In other words, the added complexity of the interaction model does not yield any additional explanation in the population.

We could also compare the main effects model to the model with interaction effects between sector and both the linear and quadratic effects of median SAT score using the F-test since Model 3 is also nested within Model 5.

```
# Compare Model 1 to Model 3
anova(lm.3, lm.5)
```

```
Analysis of Variance Table

Model 1: grad ~ 1 + sat + I(sat^2) + public
Model 2: grad ~ 1 + sat + I(sat^2) + public + public:sat + public:I(sat^2)
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     29 923.80
2     27 830.93  2     92.87 1.5088 0.2392
```

The results of this test, $F(2, 27) = 1.51$, $p = .239$, also suggest that the empirical data are consistent with the null hypothesis that there is no difference in the explained variation in the population of graduation rates between Model 3 and Model 5. In other words, the added complexity of the two interaction terms does not yield any additional explanation in the population.

Based on these sets of results, we would adopt the main effects model as the "best" model of the three candidate models.

Model Selection Using a Set of Hierarchical Nested Models

The ΔF-test is often used to aid model selection. This is only useful when the candidate models under consideration form a hierarchy of nested models. In our example, we have a natural hierarchy of candidate models we may want to consider:

Model 1 : $\text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \epsilon_i$

Model 2 : $\text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \epsilon_i$

Model 3 : $\text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \epsilon_i$

Model 4 : $\text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \beta_4(\text{SAT}_i)(\text{Public}_i) + \epsilon_i$

Model 5 : $\text{Graduation Rate}_i = \beta_0 + \beta_1(\text{SAT}_i) + \beta_2(\text{SAT}_i^2) + \beta_3(\text{Public}_i) + \beta_4(\text{SAT}_i)(\text{Public}_i) + \beta_5(\text{SAT}_i^2)(\text{Public}_i) + \epsilon_i$

Sometimes this comparison of sets of nested models is referred to as *hierarchical regression*. Be careful with this terminology as it is easy to confuse this with *hierarchical linear modeling* (HLM) which is based on an entirely different statistical model. The `anova()` function can be used to compare a hierarchical set of nested models.

```
anova(lm.1, lm.2, lm.3, lm.4, lm.5)
```

```
Analysis of Variance Table

Model 1: grad ~ 1 + sat
Model 2: grad ~ 1 + sat + I(sat^2)
Model 3: grad ~ 1 + sat + I(sat^2) + public
Model 4: grad ~ 1 + sat + I(sat^2) + public + public:sat
Model 5: grad ~ 1 + sat + I(sat^2) + public + public:sat + public:I(sat^2)
  Res.Df     RSS Df Sum of Sq       F    Pr(>F)
1     31 1882.36
2     30 1478.12  1    404.24 13.1351 0.0011851 **
3     29  923.80  1    554.32 18.0118 0.0002314 ***
4     28  842.23  1     81.57  2.6504 0.1151381
5     27  830.93  1     11.30  0.3673 0.5495565
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

When there are more than two models inputted into the `anova()` function, each model is compared to the model in the line previous to it in the output. For example here Model 2 (represented in the second line of output) is compared to Model 1 (represented in the first line of output). These results are provided in the more complex model's line of output. For example, the results of comparing Model 2 to Model 1 are:

$$F(1, 30) = 13.14,\ p = .001$$

The results (F- and p-values) in the third line compare Model 3 (represented in the third line of output) to Model 2 (represented in the second line of output). These results are:

$$F(1, 29) = 18.01,\ p = .0002$$

There are two methods to select the "best" model from this output. The first method is to start with the simplest model (Model 1) and work forward to compare this with the next most complex model. If the more complex model adds explanation beyond what is expected because of sampling error we adopt it and continue the evaluation process by comparing it to the next most complex model. If not, we adopt the simpler model as the "best" candidate model.

The second method of model selection begins by comparing the most complex model to the the second most complex model (backwards approach). If the most complex model adds explanation beyond what is expected because of sampling error we adopt it as "best" candidate model. If not, we adopt the simpler model and then compre it to the next most complex model. We continue until we have identified the "best" candidate model

Which of these methods you use is up to you, but you need to identify which you will use prior to seeing any fitted results. Here I will illustrate the backwards approach to identifying the "best" candidate model. Comparing Model 5 to Model 4, we adopt Model 4 as the data are consistent with there being no imrovement in explanation to accompany the added complexity of Model 5. Becuase of this, we eliminate (or drop) the parameter associated with the interaction term between the quadratic SAT term and sector.

We would then compare Model 4 to Model 3; adopting Model 3 using the same rationale, again eliminating the interaction term associated with the linear SAT term and sector. When we compare Model 3 to Model 2, however, the small p-value suggests that we should adopt Model 3. The additional explnation (relative to the model's increased complexity) is more than we would expect because of sampling variation.

Using this backwards approach, we would again adopt the main effects model as the "best" model of the three candidate models. In this example, using the forward approach we would end up selecting the same model as the "best" candidate model. This does not always happen.