```
  Course: Regression Models
  Lesson: Residuals Diagnostics and Variation

- Class: text
  Output: "Residuals, Diagnostics, and Variation. (Slides for this and other Data
  Science courses may be found at github
  https://github.com/DataScienceSpecialization/courses. If you care to use them, they
  must be downloaded as a zip file and viewed locally. This lesson corresponds to
  Regression_Models/02_04_residuals_variation_diagnostics.)"

- Class: figure
  Output: "In the accompanying figure there is a fairly obvious outlier. However
  obvious, it does not affect the fit very much as can be seen by comparing the orange
  line with the black. The orange line represents a fit in which the outlier is
  included in the data set, and the black line represents a fit in which the outlier is
  excluded. Including this outlier does not change the fit very much, so it is said to
  lack influence."
  Figure: noninfluential.R
  FigureType: new

- Class: figure
  Output: "This next figure also has a fairly obvious outlier, but in this case
  including the outlier changes the fit a great deal. The slope and the residuals of
  the orange line are very different than those of the black line. This outlier is said
  to be influential."
  Figure: influential.R
  FigureType: new

- Class: text
  Output: "Outliers may or may not belong in the data. They may represent real events
  or they may be spurious. In any case, they should be examined. In order to spot them,
  R provides various diagnostic plots and measures of influence. In this lesson we'll
  illustrate their meanings and use. The basic technique is to examine the effects of
  leaving one sample out, as we did in comparing the black and orange lines above.
  We'll use the influential outlier to illustrate, since leaving it out has clear
  effects."

- Class: cmd_question
  Output: "The influential outlier is in a data frame named out2. It has two columns,
  labeled y and x, respectively. To begin, create a model named fit using fit <- lm(y ~
  x, out2) or an equivalent expression."
  CorrectAnswer: fit <- lm(y ~ x, out2)
  AnswerTests: creates_lm_model('fit <- lm(y ~ x, out2)')
  Hint: "Enter fit <- lm(y ~ x, out2) or something equivalent at the R prompt."

- Class: cmd_question
  Output: "The simplest diagnostic plot displays residuals versus fitted values.
  Residuals should be uncorrelated with the fit, independent and (almost) identically
  distributed with mean zero. Enter plot(fit, which=1) at the R prompt to see if this
  is the case."
  CorrectAnswer: plot(fit, which=1)
  AnswerTests: omnitest(correctExpr='plot(fit, which=1)')
  Hint: Enter plot(fit, which=1) at the R prompt
  Figure: restore_1.R
  FigureType: new

- Class: mult_question
  Output: "Do the residuals appear uncorrelated with the fit?"
  AnswerChoices: Yes;No. There is a linear pattern involving all but one residual and
  the fit.
  CorrectAnswer: No. There is a linear pattern involving all but one residual and the
  fit.
  AnswerTests: omnitest(correctVal= 'No. There is a linear pattern involving all but
  one residual and the fit.')
  Hint: "There is an obvious linear relation between fit and most residuals."

- Class: mult_question
  Output: "The Residuals vs Fitted plot labels certain points with their row names or
  numbers, numbers in our case. Which of the three labeled points would you guess is
```

```
    our influential outlier?"
    AnswerChoices: 1;13;50
    CorrectAnswer: 1
    AnswerTests: omnitest(correctVal= '1')
    Hint: "It's pretty distinctive. For instance, it's far away from the others."

  - Class: cmd_question
    Output: "Our influential outlier is in row 1 of the data. To exclude it is just a
    matter using out2[-1, ] rather than out2 as data. Create a second model, named fitno
    for 'fit with no outlier', which excludes the outlier."
    CorrectAnswer: fitno <- lm(y ~ x, out2[-1, ])
    AnswerTests: creates_lm_model('fitno <- lm(y ~ x, out2[-1,])')
    Hint: "Enter fitno <- lm(y ~ x, out2[-1, ]) or something equivalent at the R prompt."

  - Class: cmd_question
    Output: "Display a Residuals vs Fitted plot for fitno. Remember to use which=1."
    CorrectAnswer: plot(fitno, which=1)
    AnswerTests: omnitest('plot(fitno, which=1)')
    Hint: Type plot(fitno, which=1) at the R prompt
    Figure: restore_2.R
    FigureType: new

  - Class: text
    Output: "This plot has none of the patterned appearance of the first. It looks as we
    would expect if residuals were independently and (almost) identically distributed
    with zero mean, and were uncorrelated with the fit."

  - Class: cmd_question
    Output: "The change which inclusion or exclusion of a sample induces in coefficents
    is a simple measure of its influence. Subtract coef(fitno) from coef(fit) to see the
    change induced by including the influential first sample."
    CorrectAnswer: coef(fit)-coef(fitno)
    AnswerTests: ANY_of_exprs('coef(fit)-coef(fitno)', 'fit$coef-fitno$coef',
    'fit$coefficients-fitno$coefficients')
    Hint: "Just enter coef(fit)-coef(fitno) at the R prompt."

  - Class: cmd_question
    Output: "dfbeta: The function, dfbeta, does the equivalent calculation for every
    sample in the data. The first row of dfbeta(fit) should match the difference we've
    just calculated. The second row is a similar calculation for the second sample, and
    so on. Since dfbeta returns a large matrix, use either head(dfbeta(fit)) or
    View(dfbeta(fit)) to examine the result."
    CorrectAnswer: head(dfbeta(fit))
    AnswerTests: ANY_of_exprs('head(dfbeta(fit))', 'View(dfbeta(fit))')
    Hint: Enter either head(dfbeta(fit)) or View(dfbeta(fit)) at the R prompt.

  - Class: text
    Output: "Comparing the first row with those below it, we see that the first sample
    has a much larger effect on the slope (the x column) than other samples. In fact, the
    magnitude of its effect is about 100 times that of any other point. Its effect on the
    intercept is not very distinctive essentially because its y coordinate is 0, the mean
    of the other samples."

  - Class: figure
    Output: "When a sample is included in a model, it pulls the regression line closer to
    itself (orange line) than that of the model which excludes it (black line.) Its
    residual, the difference between its actual y value and that of a regression line, is
    thus smaller in magnitude when it is included (orange dots) than when it is omitted
    (black dots.) The ratio of these two residuals, orange to black, is therefore small
    in magnitude for an influential sample. For a sample which is not influential the
    ratio would be close to 1. Hence, 1 minus the ratio is a measure of influence, near 0
    for points which are not influential, and near 1 for points which are."
    Figure: hatvalues.R
    FigureType: new

  - Class: cmd_question
    Output: "This measure is sometimes called influence, sometimes leverage, and
    sometimes hat value. Since it is 1 minus the ratio of two residuals, to calculate it
    from scratch we must first obtain the two residuals. The ratio's numerator (orange
```

```
    dots) is the residual of the first sample of the model we called fit. The model
    fitno, which excludes this sample, also excludes its residual, so we will have to
    calculate its value. This is easily done. We use R's predict function to calculate
    fitno's predicted value of y and subtract it from the actual value. Use the
    expression resno <- out2[1, \"y\"] - predict(fitno, out2[1,]) to do the calculation."
87    CorrectAnswer: resno <- out2[1, "y"] - predict(fitno, out2[1,])
88    AnswerTests: ANY_of_exprs('resno <- out2[1, "y"] - predict(fitno, out2[1,])', "resno
      <- out2[1, 'y'] - predict(fitno, out2[1,])")
89    Hint: Enter the expression resno <- out2[1, "y"] - predict(fitno, out2[1,]) at the R
      prompt.
90
91  - Class: cmd_question
92    Output: "Now calculate the influence of our outlier using 1-resid(fit)[1]/resno or an
      equivalent expression."
93    CorrectAnswer: 1-resid(fit)[1]/resno
94    AnswerTests: calculates_same_value('1-resid(fit)[1]/resno')
95    Hint: Enter 1-resid(fit)[1]/resno or an equivalent expression at the R prompt.
96
97  - Class: cmd_question
98    Output: "hatvalues: The function, hatvalues, performs for every sample a calculation
      equivalent to the one you've just done. Thus the first entry of hatvalues(fit) should
      match the value which you have just calculated. Since there are quite a few samples,
      use head(hatvalues(fit)) or View(hatvalues(fit)) to compare the influence measure of
      our outlier to that of some other samples."
99    CorrectAnswer: head(hatvalues(fit))
100   AnswerTests: ANY_of_exprs("head(hatvalues(fit))", "View(hatvalues(fit))")
101   Hint: Enter head(hatvalues(fit)) or View(hatvalues(fit)) at the R prompt.
102
103 - Class: text
104   Output: "Residuals of individual samples are sometimes treated as having the same
      variance, which is estimated as the sample variance of the entire set of residuals.
      Theoretically, however, residuals of individual samples have different variances and
      these differences can become large in the presence of outliers. Standardized and
      Studentized residuals attempt to compensate for this effect in two slightly different
      ways. Both use hat values."
105
106 - Class: cmd_question
107   Output: "We'll consider standardized residuals first. To begin, calculate the sample
      standard deviation of fit's residual by dividing fit's deviance, i.e., its residual
      sum of squares, by the residual degrees of freedom and taking the square root. Store
      the result in a variable called sigma."
108   CorrectAnswer: sigma <- sqrt(deviance(fit)/df.residual(fit))
109   AnswerTests: calculates_same_value('sigma <-
      sqrt(deviance(fit)/df.residual(fit))');expr_creates_var('sigma')
110   Hint: "Enter sigma <- sqrt(deviance(fit)/df.residual(fit)) or an equivalent
      expression at the R prompt."
111
112 - Class: cmd_question
113   Output: "Ordinarily we would just divide fit's residual (which has mean 0) by sigma.
      In the present case we multiply sigma times sqrt(1-hatvalues(fit)) to estimate
      standard deviations of individual samples. Thus, instead of dividing resid(fit) by
      sigma, we divide by sigma*sqrt(1-hatvalues(fit)). The result is called the
      standardized residual. Compute fit's standardized residual and store it in a variable
      named rstd."
114   CorrectAnswer: rstd <- resid(fit)/(sigma * sqrt(1-hatvalues(fit)))
115   AnswerTests: calculates_same_value('rstd <- resid(fit)/(sigma *
      sqrt(1-hatvalues(fit)))');expr_creates_var('rstd')
116   Hint: "Enter rstd <- resid(fit)/(sigma * sqrt(1-hatvalues(fit))) or an equivalent
      expression at the R prompt."
117
118 - Class: cmd_question
119   Output: "rstandard: The function, rstandard, computes the standardized residual which
      we have just computed step by step. Use head(cbind(rstd, rstandard(fit))) or
      View(cbind(rstd, rstandard(fit))) to compare the two calculations."
120   CorrectAnswer: head(cbind(rstd, rstandard(fit)))
121   AnswerTests: ANY_of_exprs('head(cbind(rstd, rstandard(fit)))', 'View(cbind(rstd,
      rstandard(fit)))')
122   Hint: "Enter head(cbind(rstd, rstandard(fit))) or View(cbind(rstd, rstandard(fit)))
      at the R prompt."
```

```
- Class: cmd_question
  Output: "A Scale-Location plot shows the square root of standardized residuals
  against fitted values. Use plot(fit, which=3) to display it."
  CorrectAnswer: plot(fit, which=3)
  AnswerTests: omnitest(correctExpr='plot(fit, which=3)')
  Hint: Enter plot(fit, which=3) at the R prompt.
  Figure: restore_3.R
  FigureType: new

- Class: cmd_question
  Output: "Most of the diagnostic statistics under discussion were developed because of
  perceived shortcomings of other diagnostics and because their distributions under a
  null hypothesis could be characterized. The assumption that residuals are
  approximately normal is implicit in such characterizations. Since standardized
  residuals adjust for individual residual variances, a QQ plot of standardized
  residuals against normal with constant variance is of interest. Use plot(fit,
  which=2) to display this diagnostic plot."
  CorrectAnswer: plot(fit, which=2)
  AnswerTests: omnitest(correctExpr='plot(fit, which=2)')
  Hint: Enter plot(fit, which=2) at the R prompt.
  Figure: restore_4.R
  FigureType: new

- Class: mult_question
  Output: "Look at the outlier's standardized residual, labeled 1 on the Normal QQ
  plot. About how many standard deviations from the mean is it?"
  AnswerChoices: About -5;About -2
  CorrectAnswer: About -5
  AnswerTests: omnitest(correctVal= 'About -5')
  Hint: This would be its position on the vertical axis.

- Class: cmd_question
  Output: "Studentized residuals, (sometimes called externally Studentized residuals,)
  estimate the standard deviations of individual residuals using, in addition to
  individual hat values, the deviance of a model which leaves the associated sample
  out. We'll illustrate using the outlier. Recalling that the model we called fitno
  omits the outlier sample, calculate the sample standard deviation of fitno's residual
  by dividing its deviance, by its residual degrees of freedom and taking the square
  root. Store the result in a variable called sigma1."
  CorrectAnswer: sigma1 <- sqrt(deviance(fitno)/df.residual(fitno))
  AnswerTests: calculates_same_value('sigma1 <-
  sqrt(deviance(fitno)/df.residual(fitno))');expr_creates_var('sigma1')
  Hint: Enter sigma1 <- sqrt(deviance(fitno)/df.residual(fitno)) or an equivalent
  expression at the R prompt.

- Class: cmd_question
  Output: "Calculate the Studentized residual for the outlier sample by dividing
  resid(fit)[1] by the product of sigma1 and sqrt(1-hatvalues(fit)[1]). There is no
  need to store this in a variable."
  CorrectAnswer: resid(fit)[1]/(sigma1*sqrt(1-hatvalues(fit)[1]))
  AnswerTests: calculates_same_value('resid(fit)[1]/(sigma1*sqrt(1-hatvalues(fit)[1]))')
  Hint: Enter resid(fit)[1]/(sigma1*sqrt(1-hatvalues(fit)[1])) or an equivalent
  expression at the R prompt.

- Class: cmd_question
  Output: "rstudent: The function, rstudent, calculates Studentized residuals for each
  sample using a procedure equivalent to that which we just used for the outlier. Thus
  rstudent(fit)[1] should match the value we calculated in the previous question. Use
  head(rstudent(fit)) or View(rstudent(fit)) to verify this and to compare the
  Studentized residual of the outlier with those of other samples."
  CorrectAnswer: head(rstudent(fit))
  AnswerTests: ANY_of_exprs('head(rstudent(fit))', 'View(rstudent(fit))',
  'rstudent(fit)')
  Hint: Enter head(rstudent(fit)) or an equivalent expression at the R prompt.

- Class: text
  Output: "Cook's distance is the last influence measure we will consider. It is
  essentially the sum of squared differences between values fitted with and without a
```

```
    particular sample. It is normalized (divided by) residual sample variance times the
    number of predictors which is 2 in our case (the intercept and x.) It essentially
    tells how much a given sample changes a model. We'll illustrate once again by
    calculating Cook's distance for the outlier."

  - Class: cmd_question
    Output: "We'll begin by calculating the difference in predicted values between fit
    and fitno, the models which respectively include and omit the outlier. This is most
    easily done by subtracting predict(fit, out2) from predict(fitno, out2). Store the
    difference in a variable named dy."
    CorrectAnswer: dy <- predict(fitno, out2)-predict(fit, out2)
    AnswerTests: calculates_ANY_value('dy <- predict(fitno, out2)-predict(fit, out2)',
    'dy <- predict(fit, out2)-predict(fitno, out2)');expr_creates_var('dy')
    Hint: Enter dy <- predict(fitno, out2)-predict(fit, out2) or an equivalent expression
    at the R prompt.

  - Class: cmd_question
    Output: "Recall that we calculated the sample standard deviation of fit's residual,
    sigma, earlier. Divide the summed squares of dy by 2*sigma^2 to calculate the
    outlier's Cook's distance. There is no need to store the result in a variable."
    CorrectAnswer: sum(dy^2)/(2*sigma^2)
    AnswerTests: calculates_same_value('sum(dy^2)/(2*sigma^2)')
    Hint: Enter sum(dy^2)/(2*sigma^2) or an equivalent expression at the R prompt.

  - Class: cmd_question
    Output: "cooks.distance: The function, cooks.distance, will calculate Cook's distance
    for each sample. Rather than verify that cooks.distance(fit)[1] is equal to the value
    just calculated, because that sort of thing must be getting tedious by now, display a
    diagnostic plot which uses Cook's distance using plot(fit, which=5)."
    CorrectAnswer: plot(fit, which=5)
    AnswerTests: omnitest(correctExpr='plot(fit, which=5)')
    Hint: Enter plot(fit, which=5) at the R prompt.

  - Class: text
    Output: "That concludes swirl's coverage of Residuals, Diagnostics, and Variation.
    The HTML5 slides for this as well as other units in the Johns Hopkins Data Science
    Specialization can be found here:
    https://github.com/DataScienceSpecialization/courses. They must be downloaded and
    viewed locally."

  - Class: mult_question
    Output: "Would you like to receive credit for completing this course on
      Coursera.org?"
    CorrectAnswer: NULL
    AnswerChoices: Yes;No
    AnswerTests: coursera_on_demand()
    Hint: ""
```