STP 530
Lecture 9: Polynomial Regression Models

Yi Zheng, Ph.D.
yi.isabel.zheng@asu.edu

Arizona State University
School of Mathematical and Statistical Sciences

October 31, 2023

Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Polynomial regression models

Examples of polynomial regression models:

$$E\{Y\} = \beta_0 + \beta_1 X + \beta_{11} X^2$$

$$E\{Y\} = \beta_0 + \beta_1 X + \beta_{11} X^2 + \beta_{111} X^3$$

$$E\{Y\} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{12} X_1 X_2$$

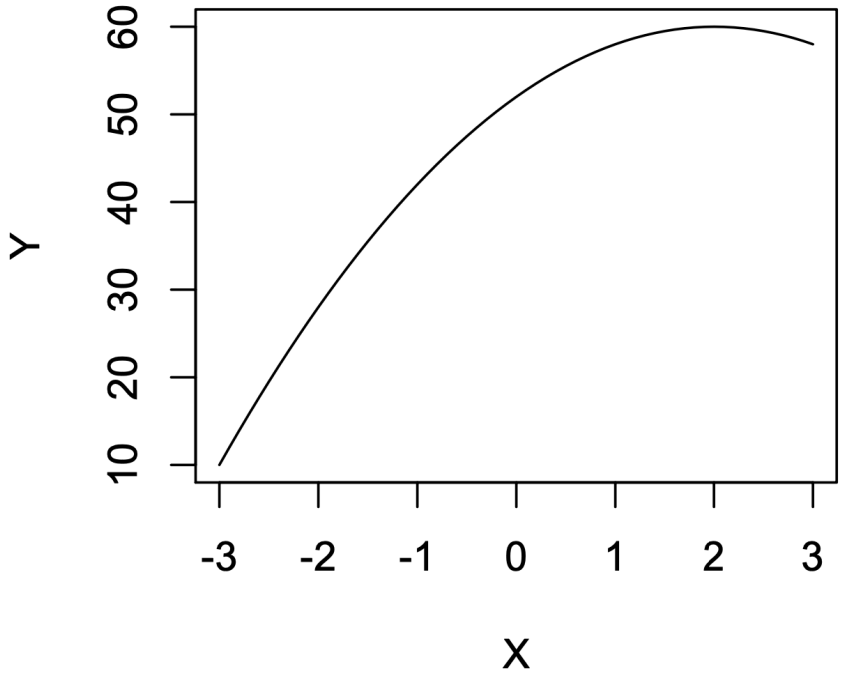
$$\begin{aligned} E\{Y\} = & \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{33} X_3^2 + \\ & \beta_{12} X_1 X_2 + \beta_{13} X_1 X_3 + \beta_{23} X_2 X_3 \end{aligned}$$

Polynomial regression models

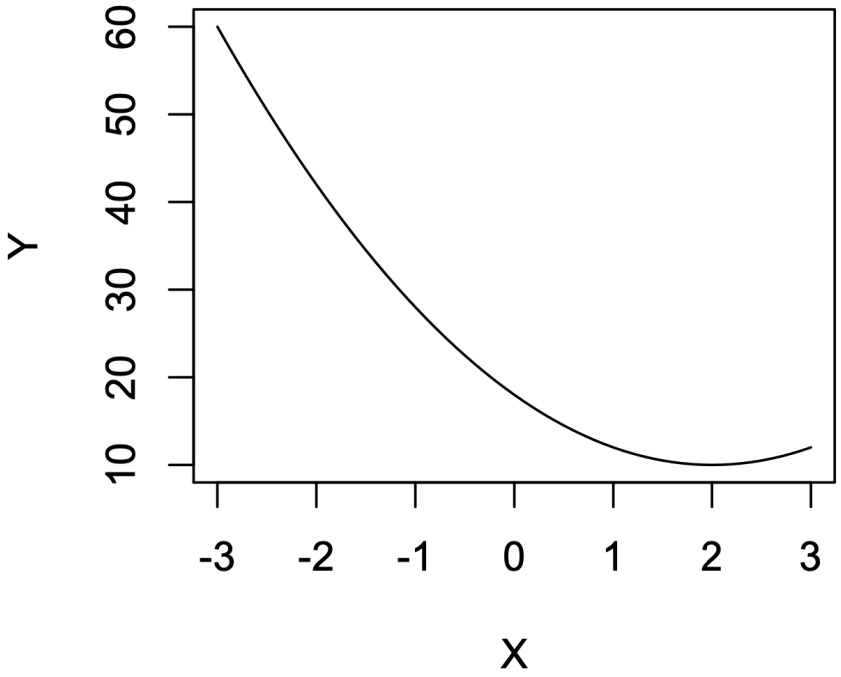
- Polynomial regression models are applicable when Y is numeric (a.k.a., quantitative).
- Higher-order terms should only be created from numeric/quantitative X's (although categorical predictors can be included in the model simultaneously).
- Polynomial regression models are among the most frequently used curvilinear response models in practice because they are handled easily as a special case of the general linear regression models.
- Polynomial regression models have two types of uses:
  1. When the true curvilinear response function is indeed a polynomial function.
  2. When the true curvilinear response function is unknown (or complex) but a polynomial function is a good approximation to the true function.

One-predictor quadratic models

$$E\{Y\} = 52 + 8X - 2X^2$$

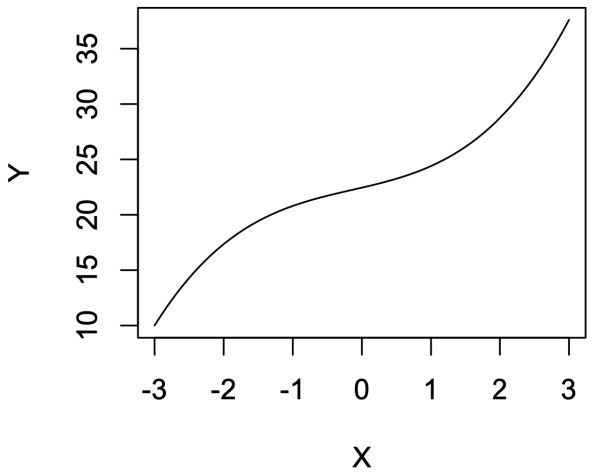


$$E\{Y\} = 18 - 8X + 2X^2$$

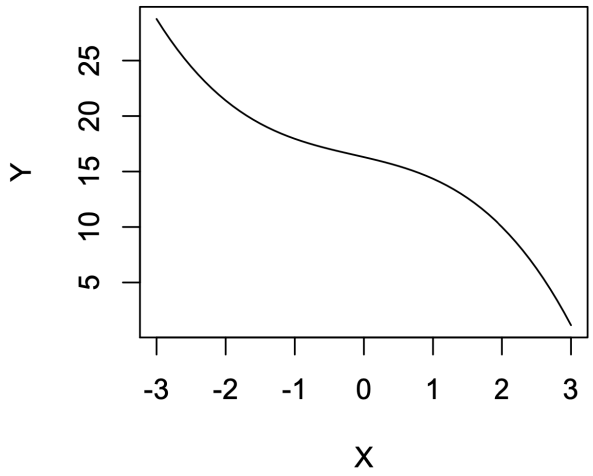


One-predictor cubic models

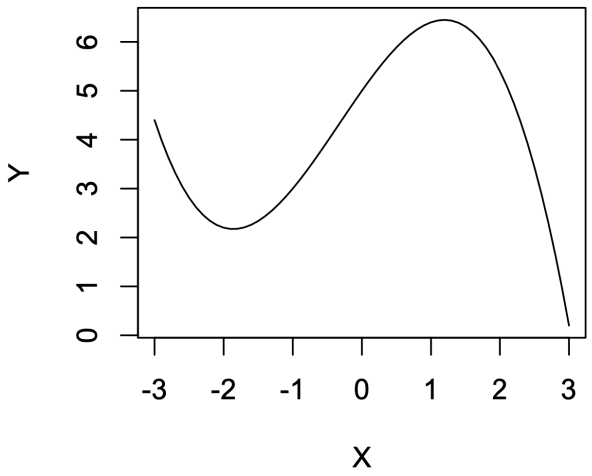
$$E\{Y\} = 22.45 + 1.45X + .15X^2 + .35X^3$$



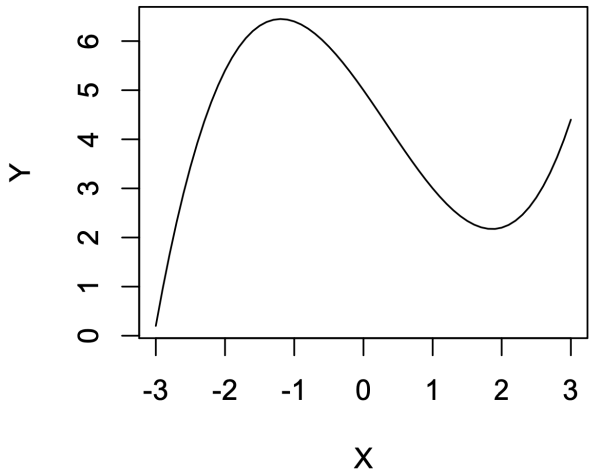
$$E\{Y\} = 16.3 - 1.45X - .15X^2 - .35X^3$$



$$E\{Y\} = 5 + 2X - .3X^2 - .3X^3$$



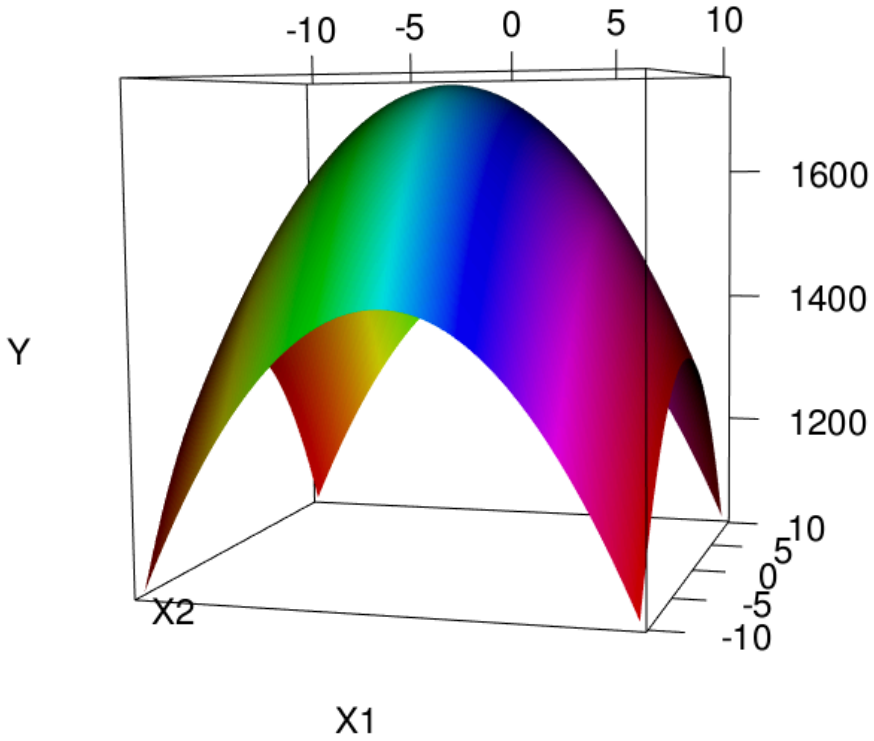
$$E\{Y\} = 5 - 2X - .3X^2 + .3X^3$$



Two-predictor quadratic models

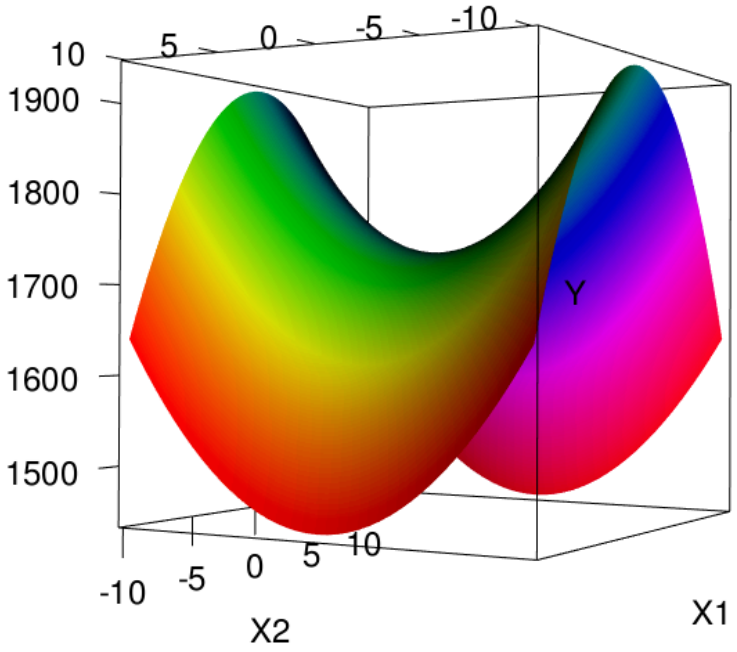
The "vaulted ceiling"

$$E\{Y\} = 1740 - 4X_1^2 - 3X_2^2$$



The "saddle"

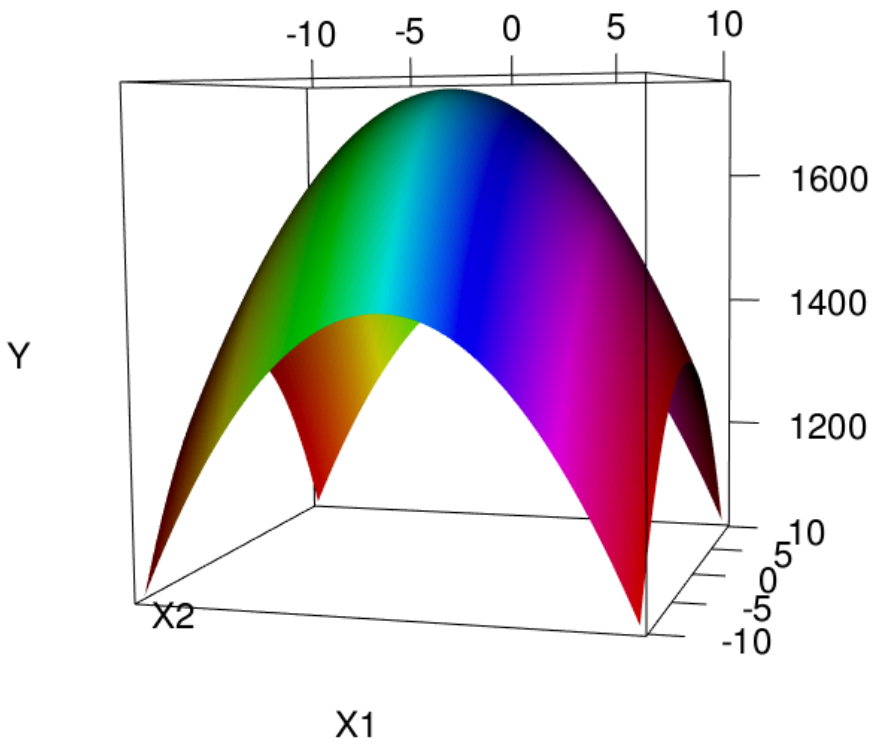
$$E\{Y\} = 1740 - 3X_1^2 + 2X_2^2$$



Two-predictor quadratic models

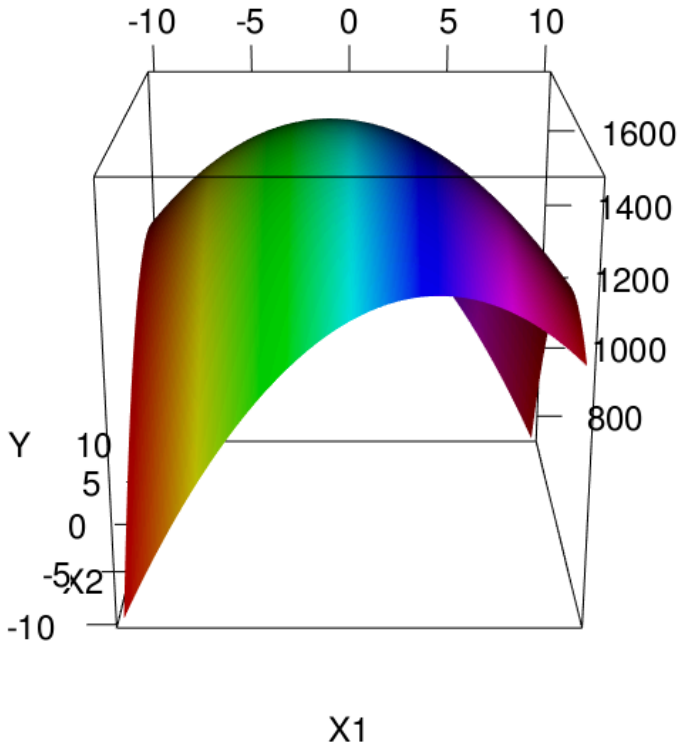
Quadratic additive model

$$E\{Y\} = 1740 - 4X_1^2 - 3X_2^2$$



Quadratic interaction model

$$E\{Y\} = 1740 - 4X_1^2 - 3X_2^2 - 3X_1X_2$$



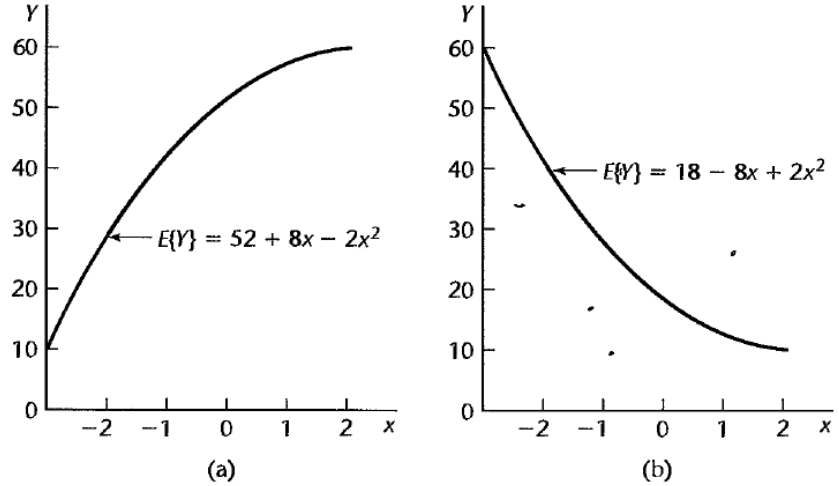
Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Special notes of using polynomial regression

- You can center the numeric predictors to alleviate multicollinearity introduced by adding higher-order terms to the model.
  - Sometimes researchers choose to standardize the variables, which not only centers the variables but also equalizes the units across variables to help with computation precision and comparison of effects.
- Be cautious about extrapolation.

FIGURE 8.1 Examples of Second-Order Polynomial Response Functions.



Special notes of using polynomial regression

- A polynomial model of sufficiently high order can always be found to fit data containing no repeat observations perfectly. So be wary of using high-order polynomials for the sole purpose of obtaining a good fit. Such regression may not show clearly the underlying pattern of data and may lead to erratic interpolations and extrapolations.

  "Distinguishing the signal from the noise requires both scientific knowledge and self-knowledge: the serenity to accept the things we cannot predict, the courage to predict the things we can, and the wisdom to know the difference." — Nate Silver

Special notes of using polynomial regression

- When fitting a polynomial regression model, fit a highest-order model first and then explore whether a lower-order model is adequate by comparing models. Try to drop the highest-order term each time.
  - **Statistical difference:** The general linear test approach
  - **Practical difference:** Compare model fit statistics, e.g., R^2, adjusted R^2

- If a polynomial term is retained, all related terms of lower-order are also retained in the model, because:
  - The lower-order terms provide the necessary flexibility for the polynomial model to fit the data in the space.
  - The lower-order terms are viewed as providing more basic information about the data being modeled, while higher order terms are viewed as providing refinements.

Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Centering predictors for polynomial regression

- **Step 1:** Center all predictors X's by subtracting the respective mean of each predictor.

$$X_{ik}^* = X_{ik} - \bar{X}_k \quad \text{for } i = 1, \cdots, n;\ k = 1, \cdots, p-1$$

- **Step 2:** Compute the polynomial terms of the centered predictor X_k^*, such as $(X_k^*)^2$, $(X_k^*)^3$, etc.
- **Step 3:** Fit a regular multiple regression model where the centered variables and the computed polynomials are included as if they are separate predictors.

$$\hat{Y} = b_0^* + b_1^* X_1^* + b_{11}^* (X_1^*)^2 + b_2^* X_2^* + b_{22}^* (X_2^*)^2 + b_{12}^* X_1^* X_2^* + \cdots$$

- **Step 4:** Transform the fitted model coefficients to express the final model in terms of the original uncentered variables. (See next page)

Step 4: Transform the fitted model coefficients back to the original scale

Assume there is one predictor X, and we center X to obtained $X_i^* = X_i - \bar{X}$. We fit a quadratic regression on X^* and obtain the fitted regression equation:

$$\hat{Y} = b_0^* + b_1^* X^* + b_{11}^* (X^*)^2$$

We can plug $X_i^* = X_i - \bar{X}$ in the above fitted regression equation:

$$\begin{aligned}
\hat{Y} &= b_0^* + b_1^* X^* + b_{11}^* (X^*)^2 \\
&= b_0^* + b_1^* (X - \bar{X}) + b_{11}^* (X - \bar{X})^2 \\
&= b_0^* + b_1^* X - b_1^* \bar{X} + b_{11}^* X^2 - 2 b_{11}^* \bar{X} X + b_{11}^* \bar{X}^2 \\
&= (b_0^* - b_1^* \bar{X} + b_{11}^* \bar{X}^2) + (b_1^* - 2 b_{11}^* \bar{X}) \cdot X + (b_{11}^*) \cdot X^2
\end{aligned}$$

The three expressions in parentheses above are the regression coefficients for the final fitted regression equation in terms of the <u>original uncentered variable</u>:

$$b_0 = b_0^* - b_1^* \bar{X} + b_{11}^* \bar{X}^2; \qquad b_1 = b_1^* - 2 b_{11}^*; \qquad b_{11} = b_{11}^*$$

Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Interpretations of the regression coefficients

We now interpret the coefficients of the transformed fitted regression equation (on the original variable scale). For example:

$$\hat{Y} = b_0 + b_1 X_1 + b_{11} X_1^2 + b_2 X_2$$

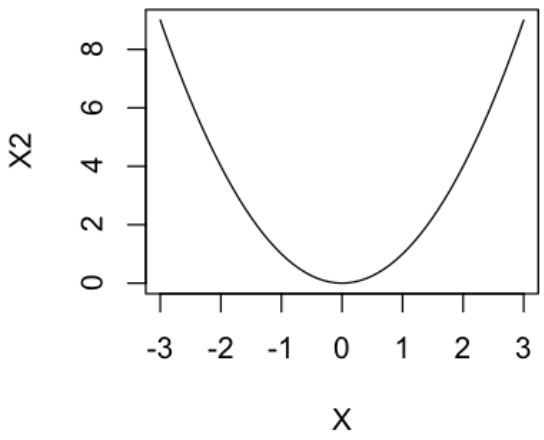
- **Intercept** (b_0). The expected Y value when all predictors equal 0.
- **Slope for X_1 (b_1):** Can you interpret it as "the change in the expected Y value for every 1-unit increase in X_1, holding all other predictors constant"?

Interpretations of the regression coefficients

$$\hat{Y} = b_0 + b_1 X_1 + b_{11} X_1^2 + b_2 X_2$$

- **Slope for X_1 (b_1):** Can you interpret it as "the change in the expected Y value for every 1-unit increase in X_1, holding all other predictors constant"?

No you cannot, because when X_1 increases by 1-unit, X_1^2 changes along. And the amount X_1^2 changes depends on the value of X_1. So "the change in the expected Y value for every 1-unit increase in X_1 holding all other predictors constant" is not b_1, nor is it $b_1 + b_{11}$. It is not a constant value and cannot be interpreted universally.



Interpretations of the regression coefficients

$$\hat{Y} = b_0 + b_1 X_1 + b_{11} X_1^2 + b_2 X_2$$

- **Intercept** b_0. The expected Y value when all predictors equal 0.
- **Slope for** X_1 **or** X_1^2 **(**b_1 **and** b_{11}**):** No universal interpretation.
- **Slope for** X_2 **(**b_2**):** Because X_2 exists solely as a linear term in the model, the universal interpretation holds:

  "The expected Y value changes by b_2 units with every 1-unit increase in X_2, holding all other predictors constant".

Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Diagnostics

All statistical inferences (hypothesis tests, confidence intervals) in multiple regression rely on the assumption:

$$\varepsilon \; iid \sim N(0, \sigma^2)$$

Hence before any statistical inference is made, model diagnostics need to be conducted and passed:

1. The regression function is linear.
2. No excessive influences by outliers.
3. The error terms have constant variance (homoskedasticity).
4. The error terms are normally distributed.
5. The error terms are independent between observations.

Hypothesis tests

The hypothesis tests in regression include:

- t-test of the individual predictors
- F-test of global utility
- F-test for comparing models ("the general linear test approach")

When the variables are linearly transformed (e.g., centered, standardized, etc.), all hypothesis tests based on the transformed data are equivalent with those based on the original data, and thus can be reported equivalently.

Confidence interval for $E\{Y_h\}$ and prediction interval for $\hat{Y}_h$

The fitted regression model on the centered scale,

$$\hat{Y} = b_0^* + b_1^* X_1^* + b_{11}^* (X_1^*)^2 + b_2^* X_2^* + b_{22}^* (X_2^*)^2 + b_{12}^* X_1^* X_2^* + \cdots$$

can be used directly to generate the confidence interval for $E\{Y_h\}$ and prediction interval for $\hat{Y}_h$ given $\mathbf{X}_h$.

But you need to first convert $\mathbf{X}_h$ to the centered scale:

$$(\mathbf{X}_h^*)' = [1, X_{h1}^*, X_{h2}^*, \cdots, X_{h,p-1}^*]$$

where

$$X_{hk}^* = X_{hk} - \bar{X}_k \quad \text{for } k = 1, \cdots, p-1$$

Confidence interval for $E\{Y_h\}$

The $1-\alpha$ confidence interval for $E\{Y_h\}$ is given by the following confidence limits:

$$\hat{Y}_h \pm t_{1-\alpha/2}(n-p) \cdot s\{\hat{Y}_h\}$$

where

$$s\{\hat{Y}_h\} = \sqrt{MSE(\mathbf{X}_h^{*\prime}(\mathbf{X}^{*\prime}\mathbf{X}^*)^{-1}\mathbf{X}_h^*)} = s\sqrt{\mathbf{X}_h^{*\prime}(\mathbf{X}^{*\prime}\mathbf{X}^*)^{-1}\mathbf{X}_h^*}$$

You can use the following R codes to find the confidence limits.

```
predict(m, newdata=?, interval="confidence", level=?)
```

Interpretation: We are $(1-\alpha)100\%$ sure that the mean value of Y for all observations with $\mathbf{X}_h$, thus the centered values of $\mathbf{X}_h^*$, falls within this interval.

Prediction interval for using $\hat{Y}_h$ to predict a single case

The $1-\alpha$ prediction interval for Y_h is given by the following prediction limits:

$$\hat{Y}_h \pm t_{1-\alpha/2}(n-p) \cdot s_{\text{pred}}\{\hat{Y}_h\}$$

where

$$s\{\hat{Y}_h\} = \sqrt{MSE(1 + \mathbf{X}_h^{*\prime}(\mathbf{X}^{*\prime}\mathbf{X}^*)^{-1}\mathbf{X}_h^*)} = s\sqrt{1 + \mathbf{X}_h^{*\prime}(\mathbf{X}^{*\prime}\mathbf{X}^*)^{-1}\mathbf{X}_h^*}$$

You can use the following R codes to find the confidence limits.

```
predict(m, newdata=?, interval="prediction", level=?)
```

Interpretation: With a $(1-\alpha)100\%$ confidence, we predict that the value of Y for the next observation with $\mathbf{X}_h$, thus the centered values of $\mathbf{X}_h^*$, falls within this interval.

Confidence intervals for regression coefficients

- The **intercept** coefficient is equivalent with $E\{Y\}$ when all predictors equal 0. Its confidence interval can be obtained in the same way as the confidence interval for $E\{Y\}$.
- When the model includes higher-order terms of a predictor, the **slope** of any single term of that predictor cannot be interpreted in isolation. So there is no point in reporting the confidence intervals of those slopes.
- If a predictor exists in the model solely in the linear form, you can interpret the **slope** for that predictor and report its confidence interval. Complication arises when the predictor was transformed (e.g., centered or standardized). In such a case, you first find the confidence limits (LL and UL) for the transformed model, and then convert the two limits back to the original scale.

Confidence intervals for regression coefficients

Step 1: Find the $1-\alpha$ confidence limits for β_k^* in the transformed model:

$$b_k^* \pm t_{1-\alpha/2}(n-p) \cdot s\{b_k^*\}$$

Step 2: If $X_k^* = aX_k + c$, transform the two confidence limits of β_k^* back to the original scale by

$$b_k = a \cdot b_k^*$$

Interpretation:

- We are $(1-\alpha)100\%$ sure that the true population parameter β_k falls within this interval.
- With a $(1-\alpha)100\%$ confidence, we estimate that expected Y value changes by somewhere between LL and UL for each 1-unit increase in X_k while holding all other predictors constant.

Outline

1. Examples of Polynomial regression models
2. Special notes of using polynomial regression models
3. Centering predictors for polynomial regression
4. Interpretations
5. Inferences
6. A case study

Textbook Section 8.1 (p.300)

A researcher studied the effects of the charge rate and temperature on the life of a new type of power cell in a preliminary small-scale experiment.

- Y: Life of the power cell, measured by the number of discharge-charge cycles that a power cell underwent before it failed.
- X_1: Charge rate, three levels (.6, 1.0, and 1.4 amperes)
- X_2: Ambient temperature, three levels (10, 20, 30 °C).

The researcher was not sure about the nature of the response function in the range of the factors studied. Hence, the researcher decided to fit the second-order polynomial regression model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{12} X_1 X_2 + \varepsilon$$

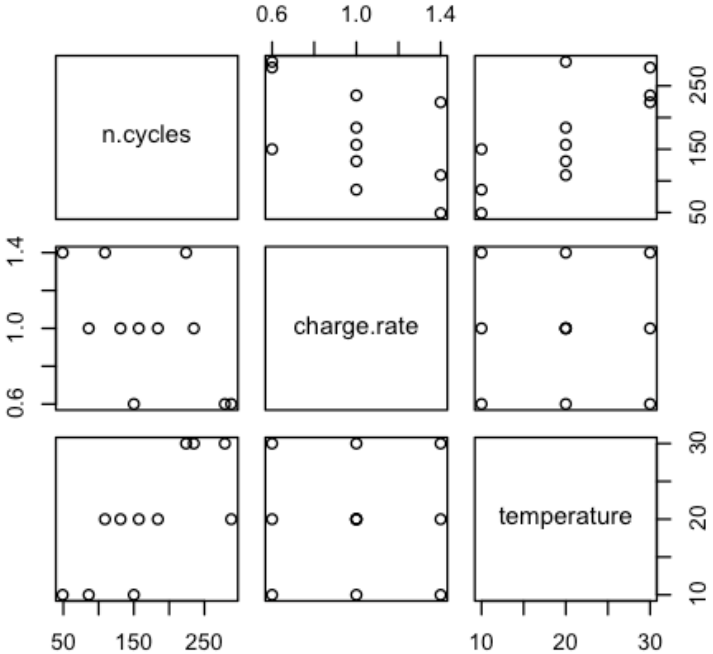
TABLE 8.1 Data—Power Cells Example.

Cell i	(1) Number of Cycles Y_i	(2) Charge Rate X_{i1}	(3) Temperature X_{i2}	(4) Coded Values x_{i1}	(5) x_{i2}	(6) x_{i1}^2	(7) x_{i2}^2	(8) $x_{i1}x_{i2}$
1	150	.6	10	−1	−1	1	1	1
2	86	1.0	10	0	−1	0	1	0
3	49	1.4	10	1	−1	1	1	−1
4	288	.6	20	−1	0	1	0	0
5	157	1.0	20	0	0	0	0	0
6	131	1.0	20	0	0	0	0	0
7	184	1.0	20	0	0	0	0	0
8	109	1.4	20	1	0	1	0	0
9	279	.6	30	−1	1	1	1	−1
10	235	1.0	30	0	1	0	1	0
11	224	1.4	30	1	1	1	1	1
		$\bar{X}_1 = 1.0$	$\bar{X}_2 = 20$					

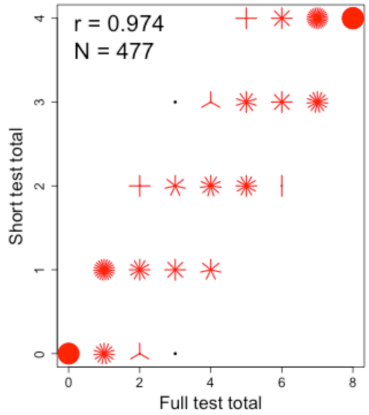
Setting adapted from: S. M. Sidik, H. F. Leibecki, and J. M. Bozek, *Cycles Till Failure of Silver-Zinc Cells with Competing Failure Modes—Preliminary Data Analysis*, NASA Technical Memorandum 815-56, 1980.

Multicollinearity?

```
> power <- read.csv("power_cell_data.csv")
> attach(power)
> table(charge.rate, temperature, useNA="ifany")
           temperature
charge.rate 10 20 30
        0.6  1  1  1
        1    1  3  1
        1.4  1  1  1
> cor(power) # All look good
              n.cycles charge.rate temperature
n.cycles     1.0000000  -0.5555349   0.7512159
charge.rate -0.5555349   1.0000000   0.0000000
temperature  0.7512159   0.0000000   1.0000000
> pairs(power)
```

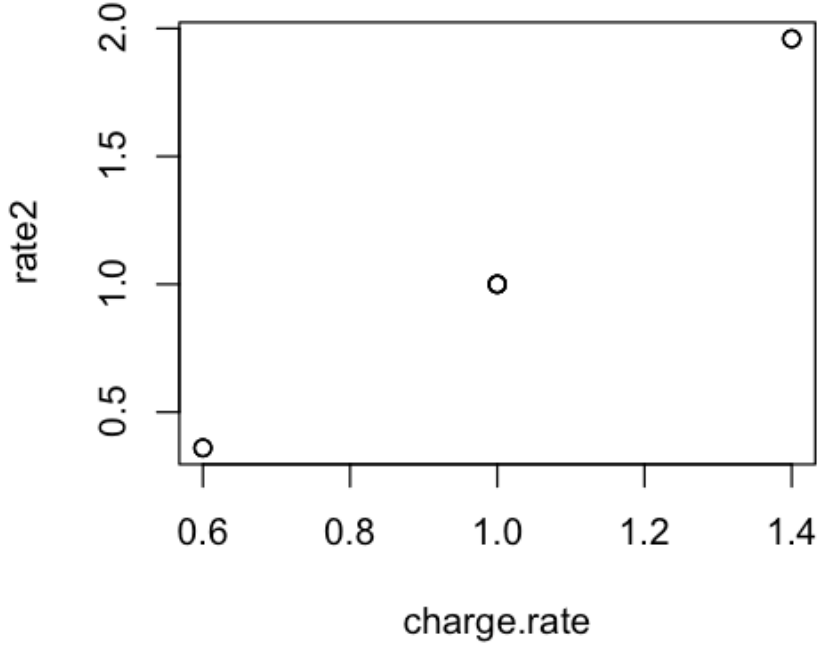


Note: When some predictors are categorical or only take a few repeated values, the plain scatterplot may not reflect the true correlation due to overlapping points. Try `sunflowerplot()` instead →.

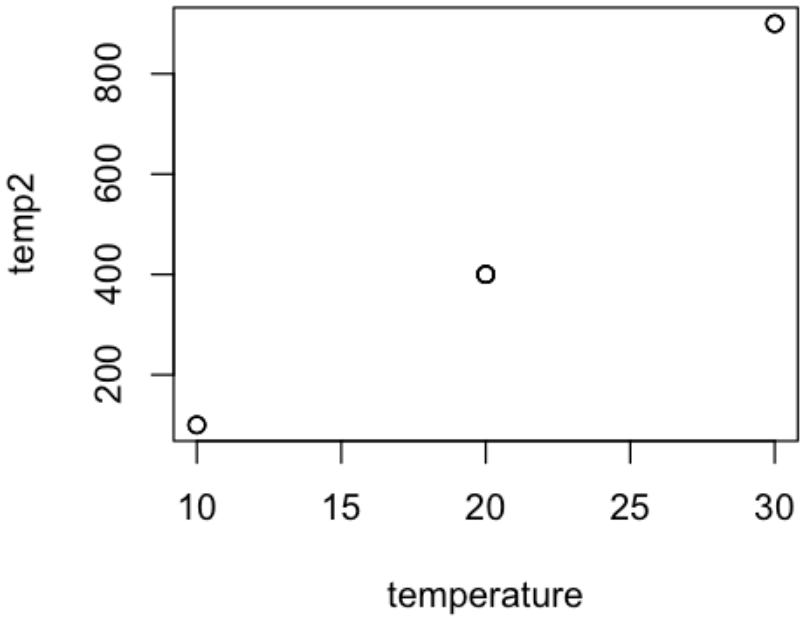


Look at the correlation between the linear and quadratic terms

```
> rate2 <- charge.rate ^ 2
> cor(charge.rate, rate2)
[1] 0.9910312
> plot(charge.rate, rate2)
```




```
> temp2 <- temperature ^ 2
> cor(temperature, temp2)
[1] 0.9860911
> plot(temperature, temp2)
```



Model output and VIF of the polynomial model on the original variables

```
> m0 <- lm(n.cycles ~ charge.rate + temperature + rate2 + temp2 + charge.rate:temperature)
> summary(m0)

Call:
lm(formula = n.cycles ~ charge.rate + temperature + rate2 + temp2 +
    charge.rate:temperature)

Residuals:
      1       2       3       4       5       6       7       8       9      10      11
-21.465   9.263  12.202  41.930  -5.842 -31.842  21.158 -25.404 -20.465   7.263  13.202

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)              337.7215   149.9616   2.252   0.0741 .
charge.rate             -539.5175   268.8603  -2.007   0.1011
temperature                8.9171     9.1825   0.971   0.3761
rate2                    171.2171   127.1255   1.347   0.2359
temp2                     -0.1061     0.2034  -0.521   0.6244
charge.rate:temperature    2.8750     4.0468   0.710   0.5092
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32.37 on 5 degrees of freedom
Multiple R-squared:  0.9135,   Adjusted R-squared:  0.8271
F-statistic: 10.57 on 5 and 5 DF,  p-value: 0.01086

> t(t(vif(m0)))
                           [,1]
charge.rate             66.21053
temperature             48.26974
rate2                   60.28708
temp2                   38.97129
charge.rate:temperature 16.37500
```

Center the predictors to alleviate multicollineairty

Additional to centering the variables, we also "semistandardize" them (a.k.a., code them) to rescale them to -1, 0, 1 for convenience.

Change rate:

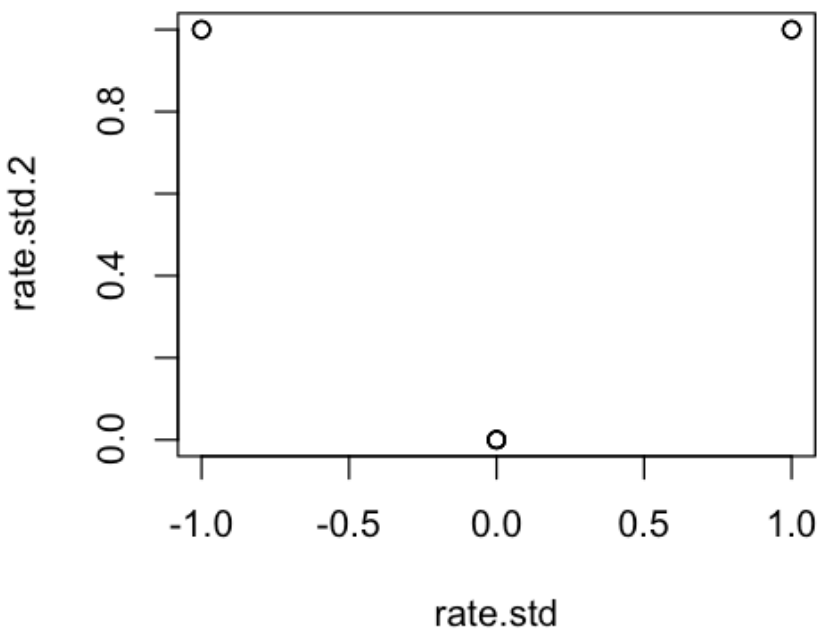
$$X_1^* = \frac{X_1 - \bar{X}_1}{.4} = \frac{X_1 - 1}{.4}$$

Ambient temperature:

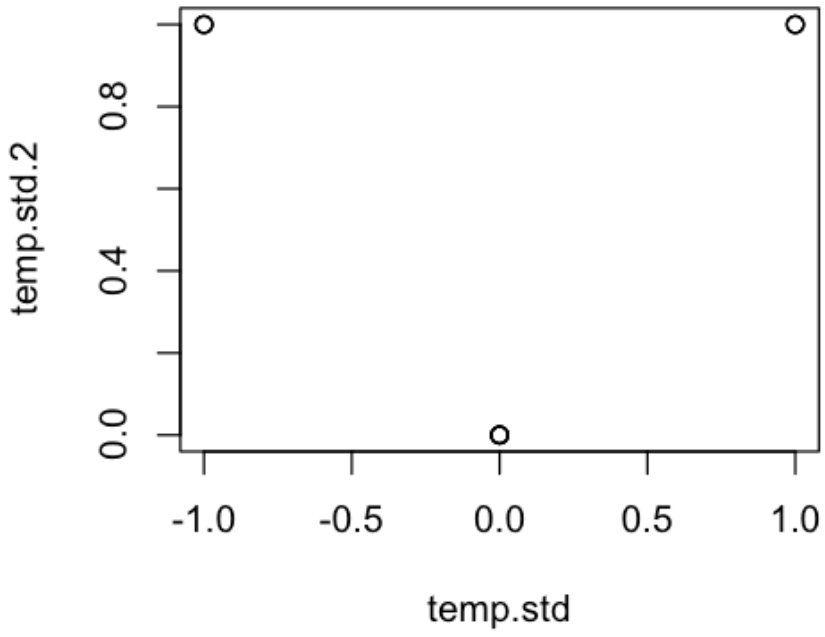
$$X_2^* = \frac{X_2 - \bar{X}_2}{10} = \frac{X_2 - 20}{10}$$

Look at the correlation between the linear and quadratic terms of the centered variables

```
> rate.std <- (charge.rate - 1) / .4
> rate.std.2 <- rate.std ^ 2
> cor(rate.std, rate.std.2)
[1] -4.042173e-16
> plot(rate.std, rate.std.2)
```




```
> temp.std <- (temperature - 20) / 10
> temp.std.2 <- temp.std ^ 2
> cor(temp.std, temp.std.2)
[1] 0
> plot(temp.std, temp.std.2)
```

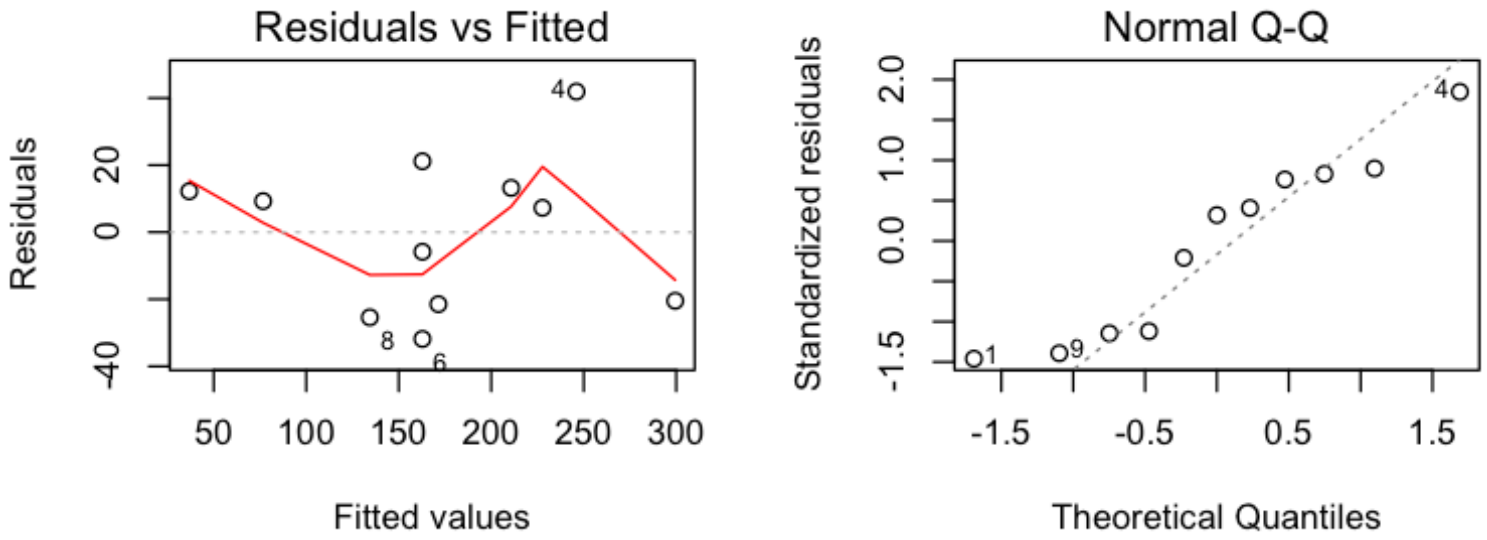


Model output and VIF of the polynomial model on the coded variables

```
> m1 <- lm(n.cycles ~ rate.std + temp.std + rate.std.2 + temp.std.2 + rate.std:temp.std)
> summary(m1)

Call:
lm(formula = n.cycles ~ rate.std + temp.std + rate.std.2 + temp.std.2 +
    rate.std:temp.std)

Residuals:
      1       2       3       4       5       6       7       8       9      10      11
-21.465   9.263  12.202  41.930  -5.842 -31.842  21.158 -25.404 -20.465   7.263  13.202

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)         162.84      16.61   9.805 0.000188 ***
rate.std            -55.83      13.22  -4.224 0.008292 **
temp.std             75.50      13.22   5.712 0.002297 **
rate.std.2           27.39      20.34   1.347 0.235856
temp.std.2          -10.61      20.34  -0.521 0.624352
rate.std:temp.std    11.50      16.19   0.710 0.509184
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32.37 on 5 degrees of freedom
Multiple R-squared:  0.9135,	Adjusted R-squared:  0.8271
F-statistic: 10.57 on 5 and 5 DF,  p-value: 0.01086

> vif(m1)
        rate.std          temp.std        rate.std.2        temp.std.2 rate.std:temp.std
        1.000000          1.000000          1.076555          1.076555          1.000000
```

Model diagnostics

Additional to checking multicollinearity, we need to conduct other model diagnostics (linearity, outliers, constant residual variance, residual normality).



- The residual pattern diagnostics are relevant because we will use the "general linear test approach" (the F-test) to decide whether we can drop some polynomial terms.
- The assumption of independence among observations is guaranteed in this case because it is a controlled replicated experiment).

Consider dropping the interaction term

```
> m1 <- lm(n.cycles ~ rate.std + temp.std + rate.std.2 + temp.std.2 + rate.std:temp.std)
> anova(m1)
Analysis of Variance Table

Response: n.cycles
                  Df Sum Sq Mean Sq F value   Pr(>F)
rate.std           1  18704   18704 17.8460 0.008292 **
temp.std           1  34202   34202 32.6323 0.002297 **
rate.std.2         1   1646    1646  1.5704 0.265552
temp.std.2         1    285     285  0.2719 0.624352
rate.std:temp.std  1    529     529  0.5047 0.509184
Residuals          5   5240    1048
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Because the interaction term is the last term in `m1` specified above, the last SSR term in the anova table gives the *extra sum of squares* of the interaction term given all other terms already in the model, $SSR(X_1^* X_2^* | X_1^*, X_2^*, (X_1^*)^2, (X_2^*)^2)$.

The corresponding p-value = .5092 is large, which means the interaction term does NOT explain a significantly larger portion of Y. So there is no need to keep it in the model.

Now we consider dropping the polynomial terms, one at a time

Note: Each time we will need to conduct model diagnostics again because the model is different. We omit those here.

Consider dropping the quadratic term of the coded temperature variable.

```
> m2 <- lm(n.cycles ~ rate.std + temp.std + rate.std.2 + temp.std.2)
> summary(m2)

Call:
lm(formula = n.cycles ~ rate.std + temp.std + rate.std.2 + temp.std.2)

Residuals:
    Min      1Q  Median      3Q     Max 
-31.965 -17.684   0.702  15.211  41.930 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   162.84      15.91  10.237 5.07e-05 ***
rate.std      -55.83      12.66  -4.410 0.004517 ** 
temp.std       75.50      12.66   5.964 0.000996 ***
rate.std.2     27.39      19.48   1.406 0.209306    
temp.std.2    -10.61      19.48  -0.544 0.605825    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 31.01 on 6 degrees of freedom
Multiple R-squared:  0.9048, Adjusted R-squared:  0.8413 
F-statistic: 14.26 on 4 and 6 DF,  p-value: 0.003204
```

```
> anova(m2)
Analysis of Variance Table

Response: n.cycles
           Df Sum Sq Mean Sq F value    Pr(>F)    
rate.std    1  18704   18704 19.4516 0.0045174 ** 
temp.std    1  34202   34202 35.5683 0.0009955 ***
rate.std.2  1   1646    1646  1.7117 0.2386411    
temp.std.2  1    285     285  0.2963 0.6058250    
Residuals   6   5769     962                      
```

Now consider dropping the polynomial term of the coded change rate variable.

```
> m3 <- lm(n.cycles ~ rate.std + temp.std + rate.std.2)
> summary(m3)

Call:
lm(formula = n.cycles ~ rate.std + temp.std + rate.std.2)

Residuals:
   Min     1Q Median     3Q    Max 
-35.50 -15.92  -1.60  12.03  49.00 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   158.60      13.15  12.059 6.15e-06 ***
rate.std      -55.83      12.01  -4.650 0.002341 ** 
temp.std       75.50      12.01   6.288 0.000409 ***
rate.std.2     24.57      17.81   1.380 0.210194    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 29.41 on 7 degrees of freedom
Multiple R-squared:  0.9001, Adjusted R-squared:  0.8573 
F-statistic: 21.02 on 3 and 7 DF,  p-value: 0.0007045
```

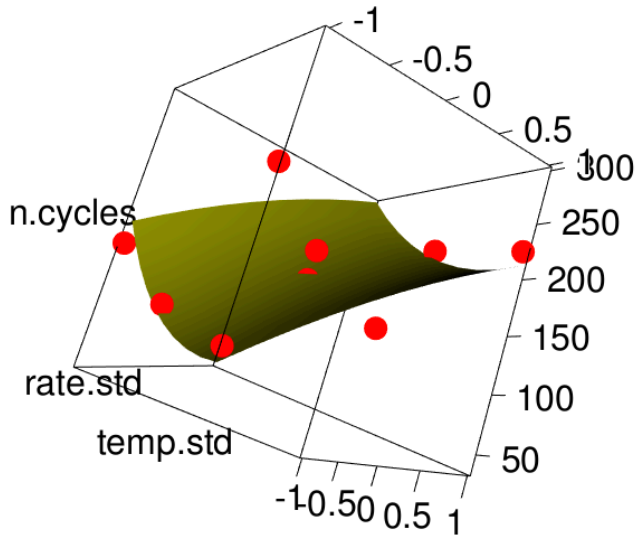
```
> anova(m3)
Analysis of Variance Table

Response: n.cycles
           Df Sum Sq Mean Sq F value    Pr(>F)    
rate.std    1  18704   18704 21.6256 0.0023410 ** 
temp.std    1  34202   34202 39.5434 0.0004087 ***
rate.std.2  1   1646    1646  1.9031 0.2101938    
Residuals   7   6054     865                      
```

No more dropping. Both linear terms contribute uniquely and significantly to the model.

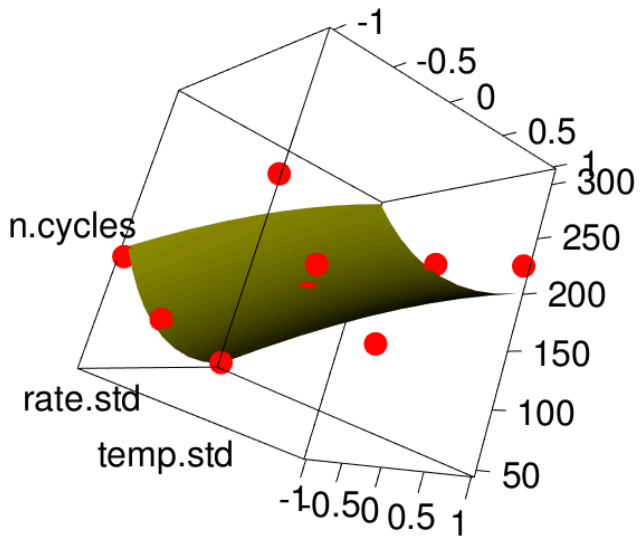
```
> m4 <- lm(n.cycles ~ rate.std + temp.std)
> summary(m4)

Call:
lm(formula = n.cycles ~ rate.std + temp.std)

Residuals:
    Min      1Q  Median      3Q     Max 
-41.000 -13.750  -7.167  10.167  60.167 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  172.000      9.354  18.387 7.88e-08 ***
rate.std     -55.833     12.666  -4.408 0.002262 ** 
temp.std      75.500     12.666   5.961 0.000338 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 31.02 on 8 degrees of freedom
Multiple R-squared:  0.8729, Adjusted R-squared:  0.8412 
F-statistic: 27.48 on 2 and 8 DF,  p-value: 0.0002606
```

```
> anova(m4)
Analysis of Variance Table

Response: n.cycles
          Df Sum Sq Mean Sq F value    Pr(>F)    
rate.std   1  18704   18704  19.432 0.0022619 ** 
temp.std   1  34202   34202  35.532 0.0003378 ***
Residuals  8   7700     963                      
```

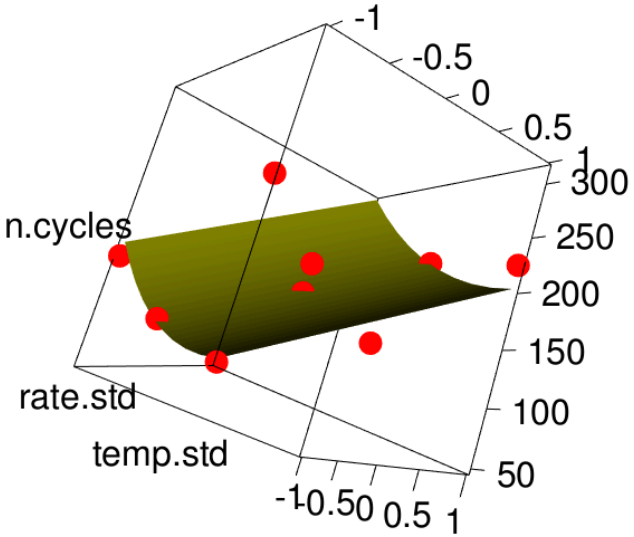
Model 1 (2 quadratics w/ interaction)



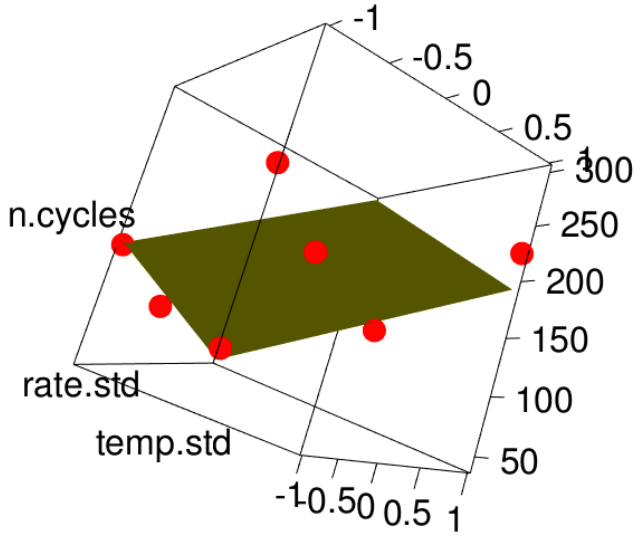
Model 2 (2 quadratics w/o interaction)

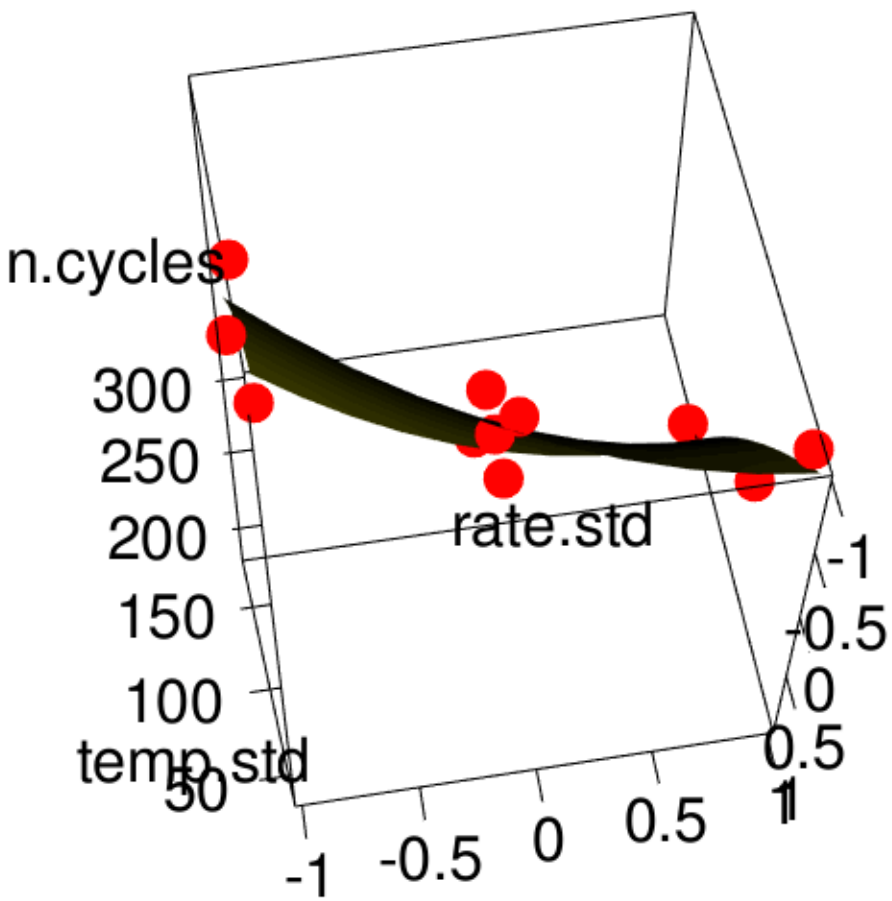


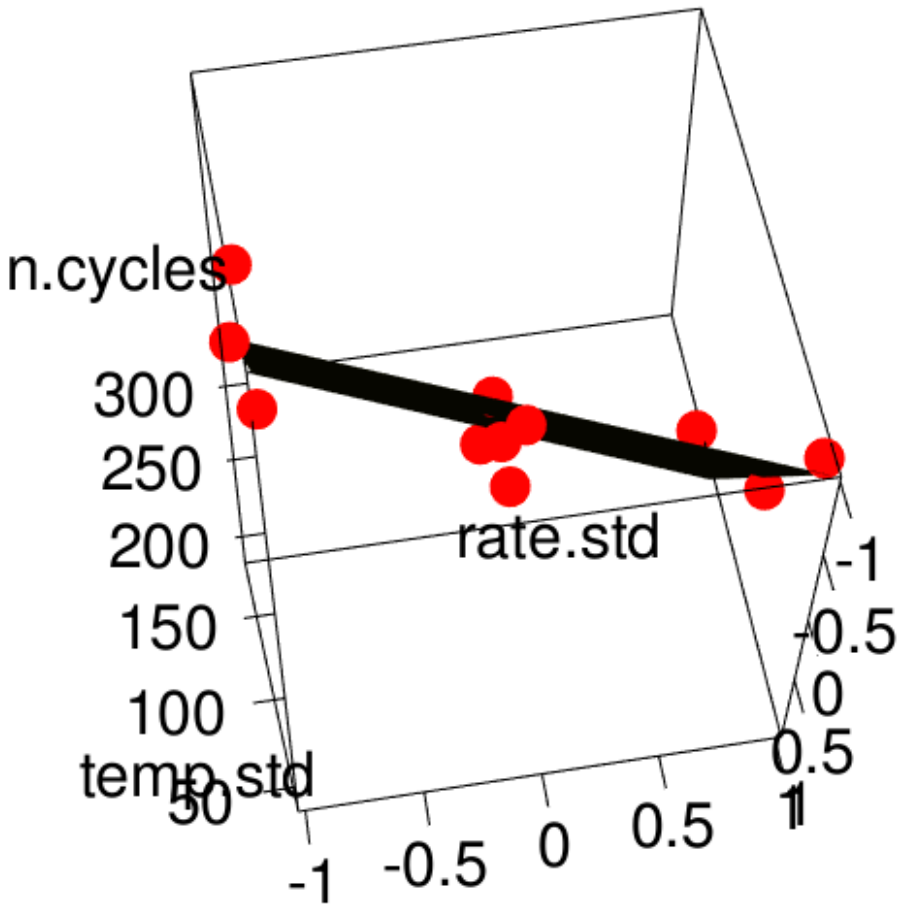
Model 3 (single quadratic)



Model 4 (two linear terms)




n.cycles
300
250
200
150
100
temp.std
50
-1
-0.5
0
0.5
1
rate.std
-1
-0.5
0
0.5
1


n.cycles
300
250
200
150
100
temp.std
50
-1
-0.5
0
0.5
1
rate.std
-1
-0.5
0
0.5
1

Last step:
Convert the fitted model from the coded scale to the original scale

Fitted model on the coded scale

$$\hat{Y} = 172 - 55.8X_1^* + 75.5X_2^*$$

Plug in the coding equations:
$X_1^* = (X_1 - \bar{X}_1)/.4 = (X_1 - 1)/.4$ and $X_2^* = (X_2 - \bar{X}_2)/10 = (X_2 - 20)/10$,
we have

$$\begin{aligned}
\hat{Y} &= 172 - 55.833X_1^* + 75.5X_2^* \\
&= 172 - 55.833\left(\frac{X_1 - 1}{.4}\right) + 75.5\left(\frac{X_2 - 20}{10}\right) \\
&= 172 - \frac{55.833}{.4}X_1 + \frac{55.833 \times 1}{.4} + \frac{75.5}{10}X_2 - \frac{75.5 \times 20}{10} \\
&= \left(172 + \frac{55.833 \times 1}{.4} - \frac{75.5 \times 20}{10}\right) - \frac{55.833}{.4}X_1 + \frac{75.5}{10}X_2 \\
&= 160.58 - 139.58X_1 + 7.55X_2
\end{aligned}$$

Interpretations

$$\widehat{n.cycles} = 160.58 - 139.58(charge.rate) + 7.55(temperature)$$

- $b_0 = 160.58$. For a charge rate of 0 ampere and ambient temperature of 0C, the power cell is expected to undergo 160.58 discharge-charge cycles before it fails.
- $b1 = -139.58$. Holding ambient temperature constant, every 1 more ampere in charge rate is expected to shorten the life of the power cell by 139.58 discharge-charge cycles.
- $b2 = 7.55$. Holding charge rate constant, raising the ambient temperature by 1 °C is expected to lengthen the life of the power cell by 7.55 discharge-charge cycles.

95% confidence interval of regression slope b_2

Step 1: From the fitted regression of the transformed data, we have $b_2^* = 75.5$ and its standard error $s\{b_2^*\} = 12.67$. With $t_{1-\alpha/2}(n-p)$ obtained by `qt(p=.975, df=8)` $= 2.3$, we have the following confidence limits of β_2^*:

$$b_k^* \pm t_{1-\alpha/2}(n-p) \cdot s\{b_k^*\} = 75.5 \pm 2.3 * 12.67 = (46.4, 104.6)$$

Step 2: Now transform the two confidence limits to the original data scale. Because X_2 was transformed by $X_2^* = (X_2 - 20)/10$, we have $b_2 = b_2^*/10$.

So the 95% confidence interval of b_2 is:

$$(46.4/10, 104.6/10) = (4.64, 10.46)$$

Confidence interval for $E\{Y_h\}$ and prediction interval for $\hat{Y}_h$

Suppose we want to predict the life for the power cell when the charge rate is 1.6 amperes and the ambient temperature is 25 °C.

To make prediction using the regression model fitted with the transformed data, we first transform the two new predictor values in the same way:

$$X_1^* = (X_1 - 1)/.4 = (1.6 - 1)/.4 = 1.5$$
$$X_2^* = (X_2 - 20)/10 = (25 - 20)/10 = .5$$

```
> # 95% confidence interval of E{Y}
> predict(m4, newdata=data.frame(rate.std=1.5, temp.std=.5),
interval="confidence", level=.95)
  fit      lwr      upr
1 126 75.02934 176.9707
>
> # 95% prediction interval of Y-hat
> predict(m4, newdata=data.frame(rate.std=1.5, temp.std=.5),
interval="prediction", level=.95)
  fit      lwr      upr
1 126 38.15654 213.8435
```

```
> # 95% confidence interval of E{Y}
> predict(m4, newdata=data.frame(rate.std=1.5, temp.std=.5),
interval="confidence", level=.95)
  fit      lwr      upr
1 126 75.02934 176.9707
>
> # 95% prediction interval of Y-hat
> predict(m4, newdata=data.frame(rate.std=1.5, temp.std=.5),
interval="prediction", level=.95)
  fit      lwr      upr
1 126 38.15654 213.8435
```

Interpretation:

- **Confidence interval for $E\{Y_h\}$:** We are 95% sure that the mean number of charge-recharge cycles for all observations with a charge rate of 1.6 amperes and an ambient temperature of 25 °C falls between 75 and 177.

- **Prediction interval for $\hat{Y}_h$** With a 95% confidence, we predict that the number of charge-recharge cycles for the next observation with a charge rate of 1.6 amperes and an ambient temperature of 25 °C falls between 38 and 214.