TASessionWeek5

CSCI E-106 Staff

10/10/2019

```
library(ggplot2)
library(MASS)
```

Question 4.27

Refer to the SENIC data set in Appendix C.1 and Project 1.45. Consider the regression relation of average length of stay to infection risk.

a. Obtain Bonferroni joint confidence intervals for β_0 and β_1, using a 90 percent family confidence coefficient.

```
#Read data into a data frame
dts2 <- read.table(url("http://www.stat.purdue.edu/~minzhang/525-Spring2018/Datasets_files/APPENC01.txt
colnames(dts2) <- c("ID", "LoS", "A", "IR", "RCR", "RCXR", "NB",
                    "MSA", "R", "ADC", "NoN", "AFS")
```

```
#Regression relation of average length of stay to infection risk
mdl2 <- lm(LoS~IR, dts2)
summary(mdl2)
```

```
##
## Call:
## lm(formula = LoS ~ IR, data = dts2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0587 -0.7776 -0.1487  0.7159  8.2805
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.3368     0.5213  12.156  < 2e-16 ***
## IR            0.7604     0.1144   6.645 1.18e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.624 on 111 degrees of freedom
## Multiple R-squared:  0.2846, Adjusted R-squared:  0.2781
## F-statistic: 44.15 on 1 and 111 DF,  p-value: 1.177e-09
```

```
B <- qt(1-.10/(2*2), mdl2$df.residual)

print("Joint confidence interval for b_0 is from")
```

```
## [1] "Joint confidence interval for b_0 is from"
```

```
coef(mdl2)[1] - coef(summary(mdl2))[1, 2]*B
```

```
## (Intercept)
##    5.303844
```

```r
print("to")
```

```
## [1] "to"
```

```r
coef(mdl2)[1] + coef(summary(mdl2))[1, 2]*B
```

```
## (Intercept)
##    7.369729
```

```r
print("Joint confidence interval for b_1 is from")
```

```
## [1] "Joint confidence interval for b_1 is from"
```

```r
coef(mdl2)[2] - coef(summary(mdl2))[2, 2]*B
```

```
##        IR
## 0.5336442
```

```r
print("to")
```

```
## [1] "to"
```

```r
coef(mdl2)[2] + coef(summary(mdl2))[2, 2]*B
```

```
##        IR
## 0.9871976
```

b. A researcher has suggested that β_0 should be approximately 7 and β_1 should be approximately 1. Do the joint confidence intervals in part (a.) support this expectation?

The joint confidence intervals in part (a.) do not support this view.

c It is desired to esimate the expected hospital stay for persons with infection risks $X = 2, 3, 4, 5$ with family confidence coefficient .95. Which procedure, the Working-Hotelling or the Bonferroni, is more efficient here?

```r
W <- sqrt(2*qf(.95, 2, 111))
B <- qt(.99375, 111)
W
```

```
## [1] 2.481152
```

```r
B
```

```
## [1] 2.539061
```

Working-Hotelling is tighter, i.e., more efficient.

d Obtain the family of interval estimates required in part (c), using the more efficient procedure. Interpret your confidence intervals.

```r
Xh <-data.frame(IR = 2:5)

pred2 <- predict(mdl2,newdata = Xh, se.fit = TRUE)

print(paste0("Family confidence interval for ",Xh[1,1]  ," is from:"))
```

```
## [1] "Family confidence interval for 2 is from:"
```

```r
pred2$fit[1] - pred2$se.fit[1]*W
```

```
##        1
## 7.088991
```

```r
print("to")
```

```
## [1] "to"
```

```r
pred2$fit[1] + pred2$se.fit[1]*W
```

```
##        1
## 8.626266
```

```r
print(paste0("Family confidence interval for ",Xh[2,1]  ," is from:"))
```

```
## [1] "Family confidence interval for 3 is from:"
```

```r
pred2$fit[2] - pred2$se.fit[2]*W
```

```
##        2
## 8.077961
```

```r
print("to")
```

```
## [1] "to"
```

```r
pred2$fit[2] + pred2$se.fit[2]*W
```

```
##        2
## 9.158137
```

```r
print(paste0("Family confidence interval for ",Xh[3,1]  ," is from:"))
```

```
## [1] "Family confidence interval for 4 is from:"
```

```r
pred2$fit[3] - pred2$se.fit[3]*W
```

```
##        3
## 8.986242
```

```r
print("to")
```

```
## [1] "to"
```

```r
pred2$fit[3] + pred2$se.fit[3]*W
```

```
##        3
## 9.770698
```

```r
print(paste0("Family confidence interval for ",Xh[4,1]  ," is from:"))
```

```
## [1] "Family confidence interval for 5 is from:"
```

```r
pred2$fit[4] - pred2$se.fit[4]*W
```

```
##        4
## 9.717885
```

```r
print("to")
```

```
## [1] "to"
```

```r
pred2$fit[4] + pred2$se.fit[4]*W
```

```
##       4
## 10.5599
```

(Textbook 3.17) Sales growth.

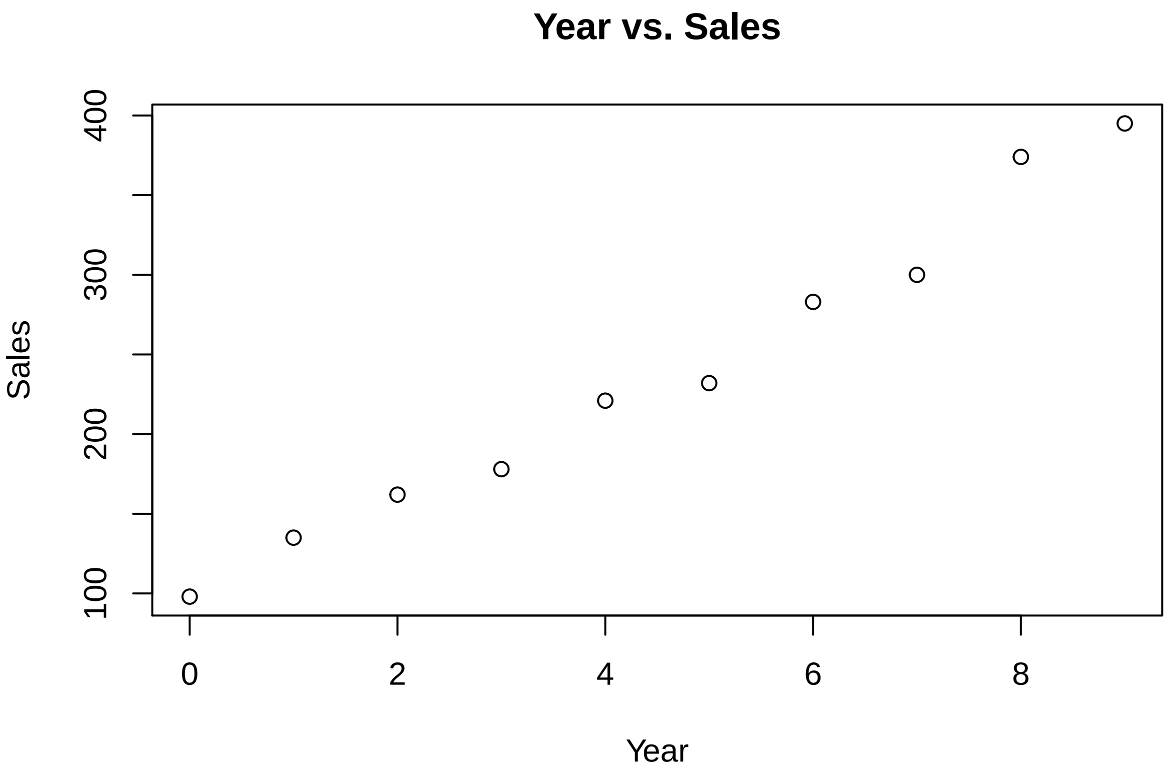
A marketing researcher studied annual sales of a product that had been introduced 10 years ago. The data are as follows, where X is the year (coded) and Y is sales in thousands of units:

Please use dataset titled: ***CH03PR17.txt***

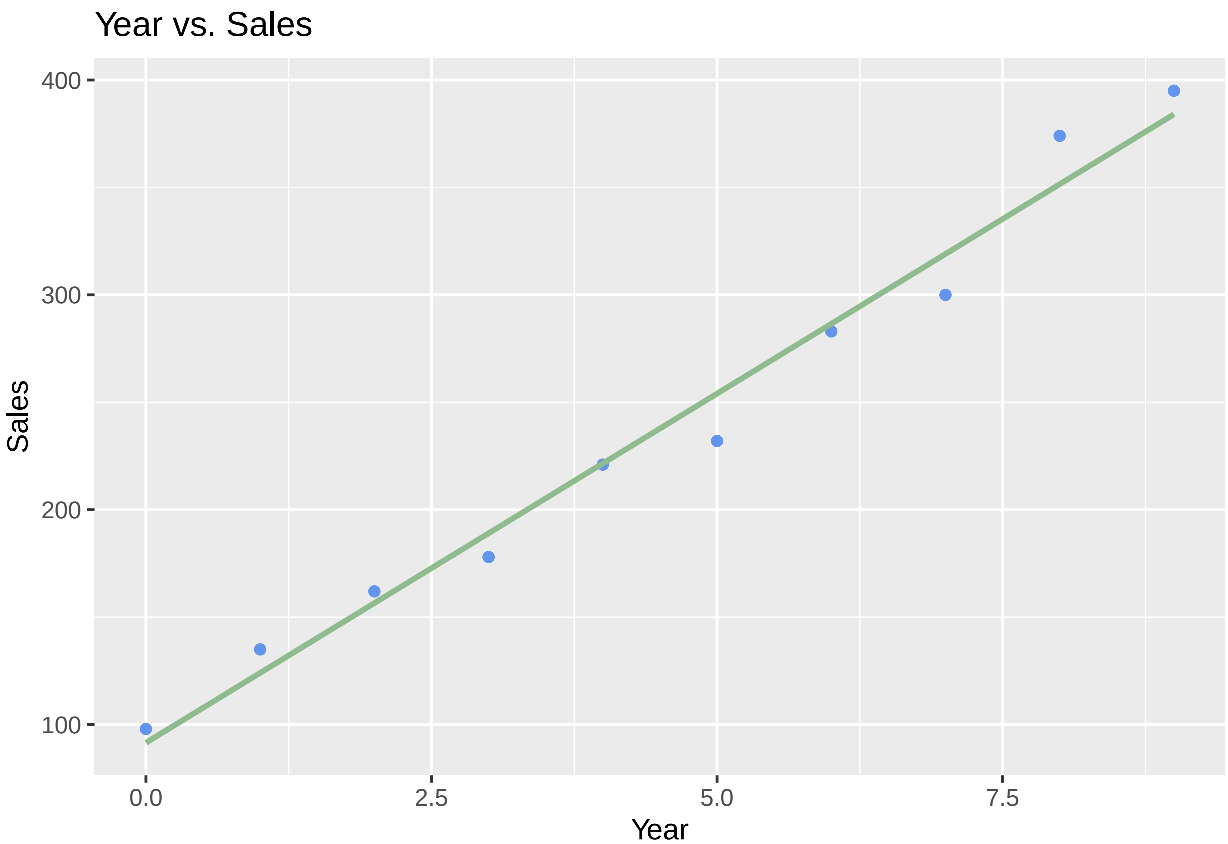
a. Prepare a scatter plot of the data. Does a linear relation appear adequate here?

Solution Below

```
df_sales = read.table(url("http://users.stat.ufl.edu/~rrandles/sta4210/Rclassnotes/data/textdatasets/Ku
                      col.names=c("sales", "year"))

lmFit317 = lm(sales~year, data=df_sales)

# Method #1
plot(df_sales$year, df_sales$sales, xlab="Year", ylab="Sales", main="Year vs. Sales")
```




```
# Method #2
plot_317a = ggplot(data=df_sales, aes(x=year, y=sales)) +
        geom_point(color="cornflowerblue") +
        geom_smooth(color="darkseagreen", method="lm", se=FALSE) +
        labs(title="Year vs. Sales",
                 x="Year", y="Sales")

plot_317a
```



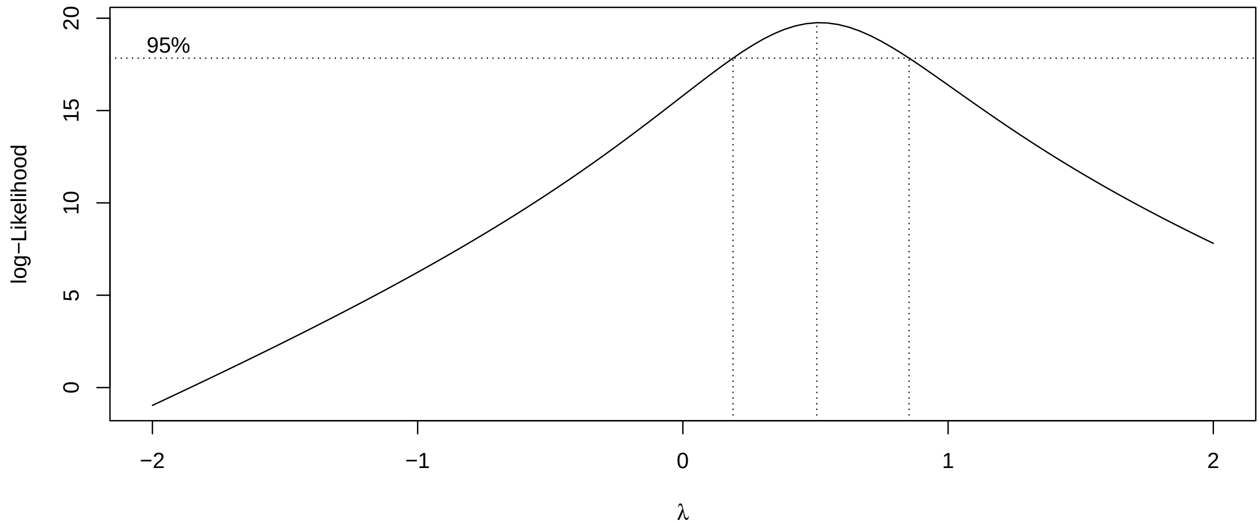
Interpretation

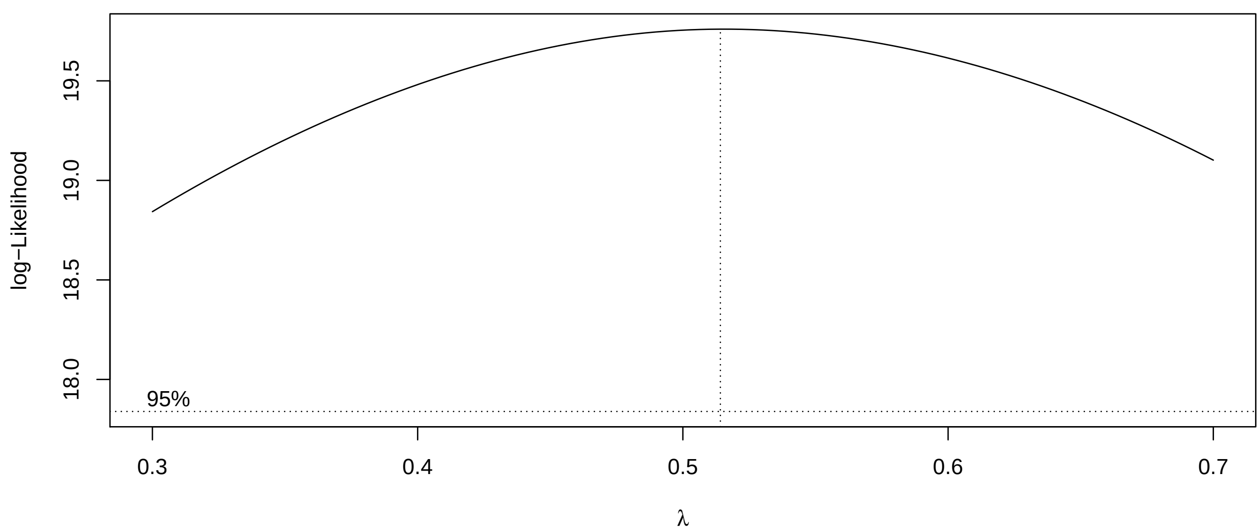
Creating a scatter plot of the data and analyzing it, it does appear there is a linear relationship between year and sales.

b. Use the Box-Cox procedure and standardization (3.36) to find an appropriate power transformation of Y. Evaluate SSE for $\lambda = .3, .4, .5, .6, .7$. What transformation of Y is suggested?

Solution Below

```
# mfrow argument takes in a vector specifying layout for subsequent displays of figures
par(mfrow=c(2,1))
boxcox(lmFit317)
boxcox(lmFit317, lambda=c(0.3,0.4,0.5,0.6,0.7))
```





Interpretation

The Box-Cox procedure identified $\lambda = 0.5$ as the best power transformation. Referring back to page 135, $\lambda = 0.5$ which suggests a square-root transformation.

c. Use the transformation $Y' = \sqrt{Y}$ and obtain the estimated linear regression function for the transformed data.

Solution Below

```
df_sales = cbind(df_sales, sqrt(df_sales$sales))
colnames(df_sales)[3] = "salesTrans"

lmFit317b = lm(salesTrans~year, data=df_sales)
summary(lmFit317b)
```

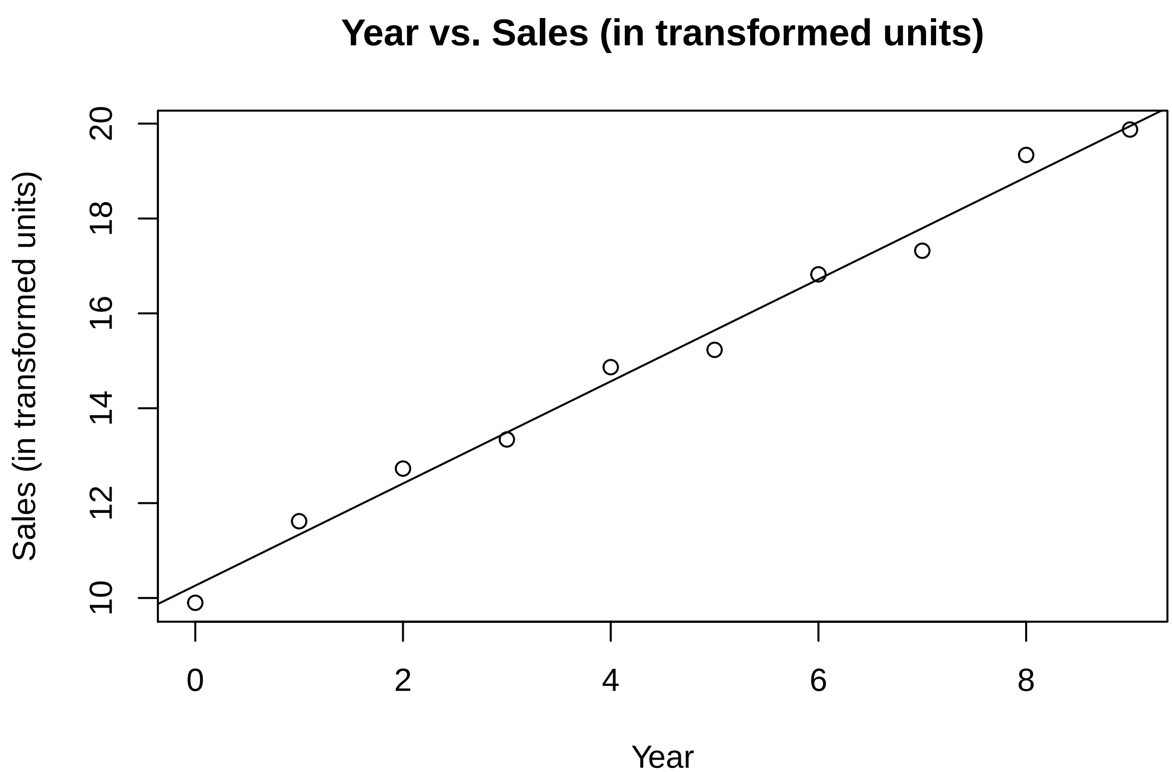
```
##
```

```
## Call:
## lm(formula = salesTrans ~ year, data = df_sales)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.47447 -0.30811  0.01549  0.29541  0.46781
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.26093    0.21290   48.20 3.80e-11 ***
## year         1.07629    0.03988   26.99 3.83e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3622 on 8 degrees of freedom
## Multiple R-squared:  0.9891, Adjusted R-squared:  0.9878
## F-statistic: 728.4 on 1 and 8 DF,  p-value: 3.826e-09
```

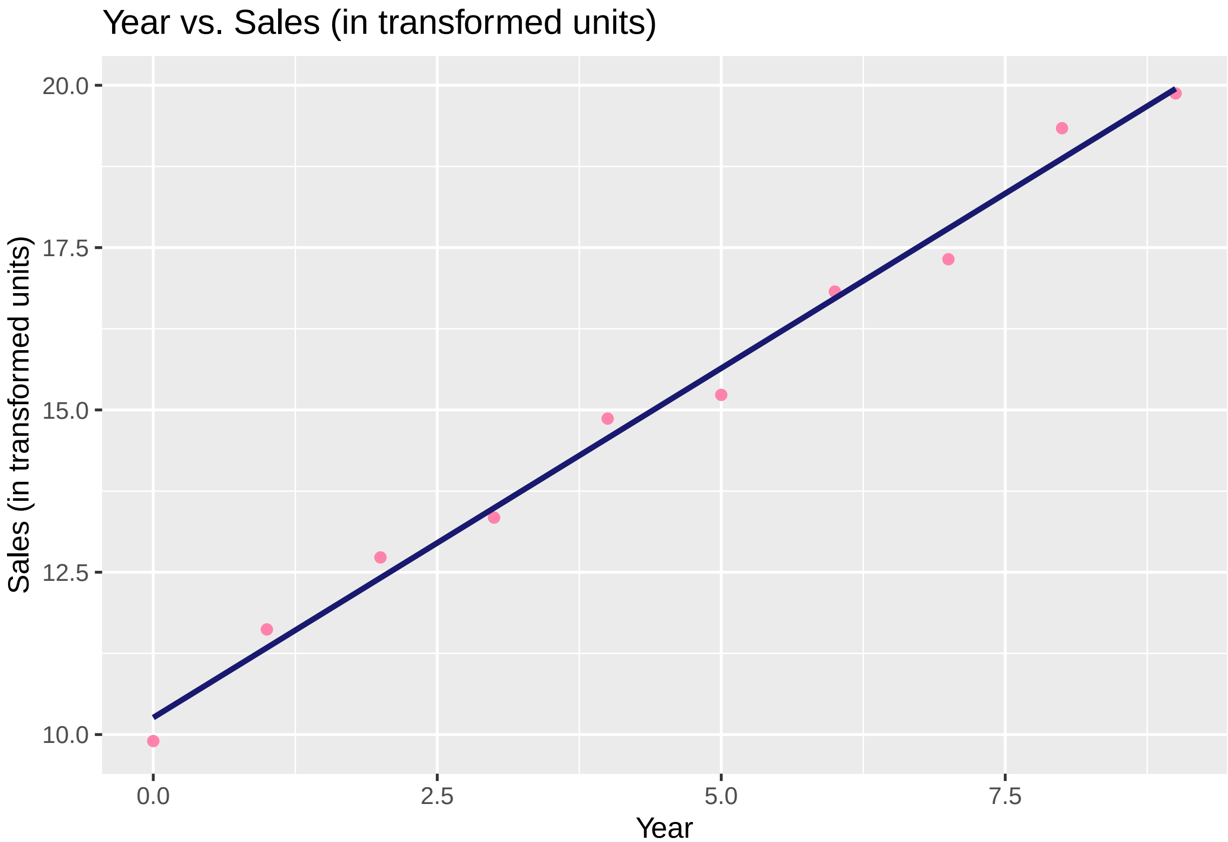
d. Plot the estimated regression line and the transformed data. Does the regression line appear to be a good fit to the transformed data?

Solution Below

```r
# Method #1
plot(df_sales$year, df_sales$salesTrans,
     xlab="Year", ylab="Sales (in transformed units)",
     main="Year vs. Sales (in transformed units)")
abline(lmFit317b)
```



```r
# Method #2
plot_317b = ggplot(data=df_sales, aes(x=year, y=salesTrans)) +
        geom_point(color="palevioletred1") +
        geom_smooth(color="midnightblue", method="lm", se=FALSE) +
        labs(title="Year vs. Sales (in transformed units)",
                 x="Year", y="Sales (in transformed units)")

plot_317b
```



Interpretation

Assessing the plot(s), it looks like a linear regression model is a great fit. Looking at the summary, we see that the r-sqaured value is 0.98.

e. Obtain the residuals and plot them against the fitted values. Also prepare a normal probability plot. What do your plots show?

Solution Below

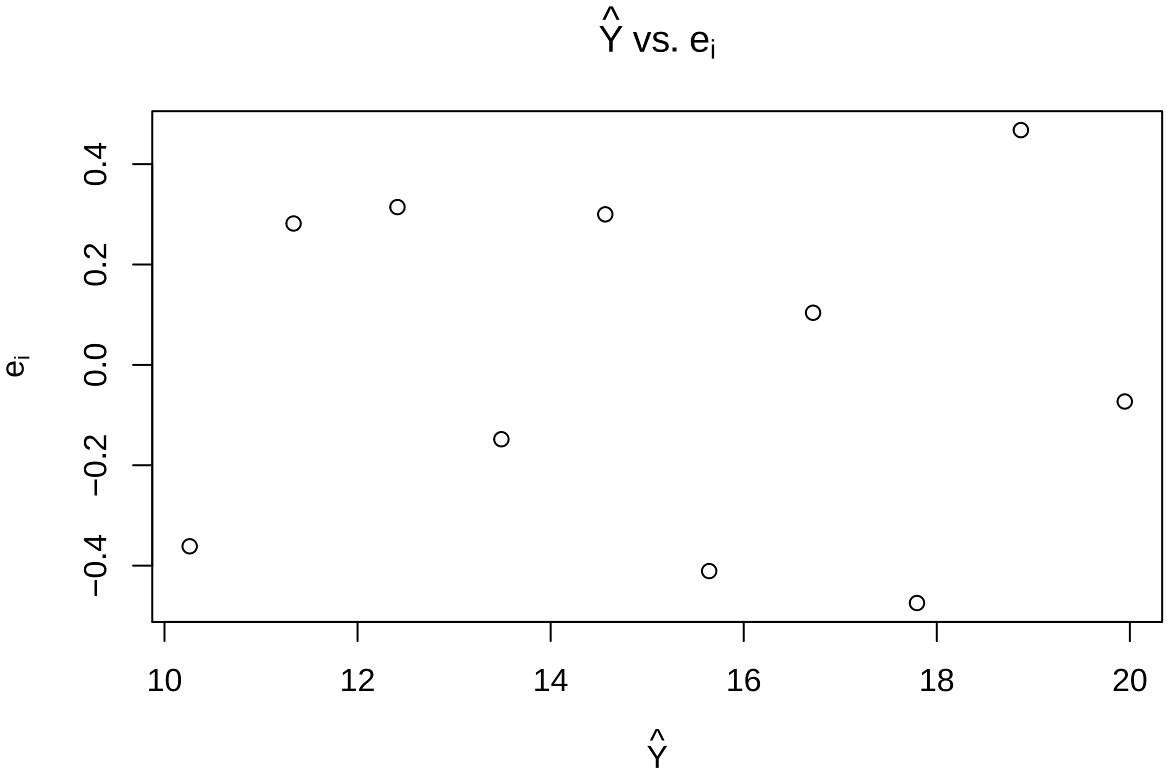
```r
ei= resid(lmFit317b)
print(ei)
```

```
##           1           2           3           4           5           6
## -0.36143656  0.28172678  0.31440703 -0.14814273  0.29997018 -0.41084412
##           7           8           9          10
##  0.10392174 -0.47446579  0.46781397 -0.07295049
```

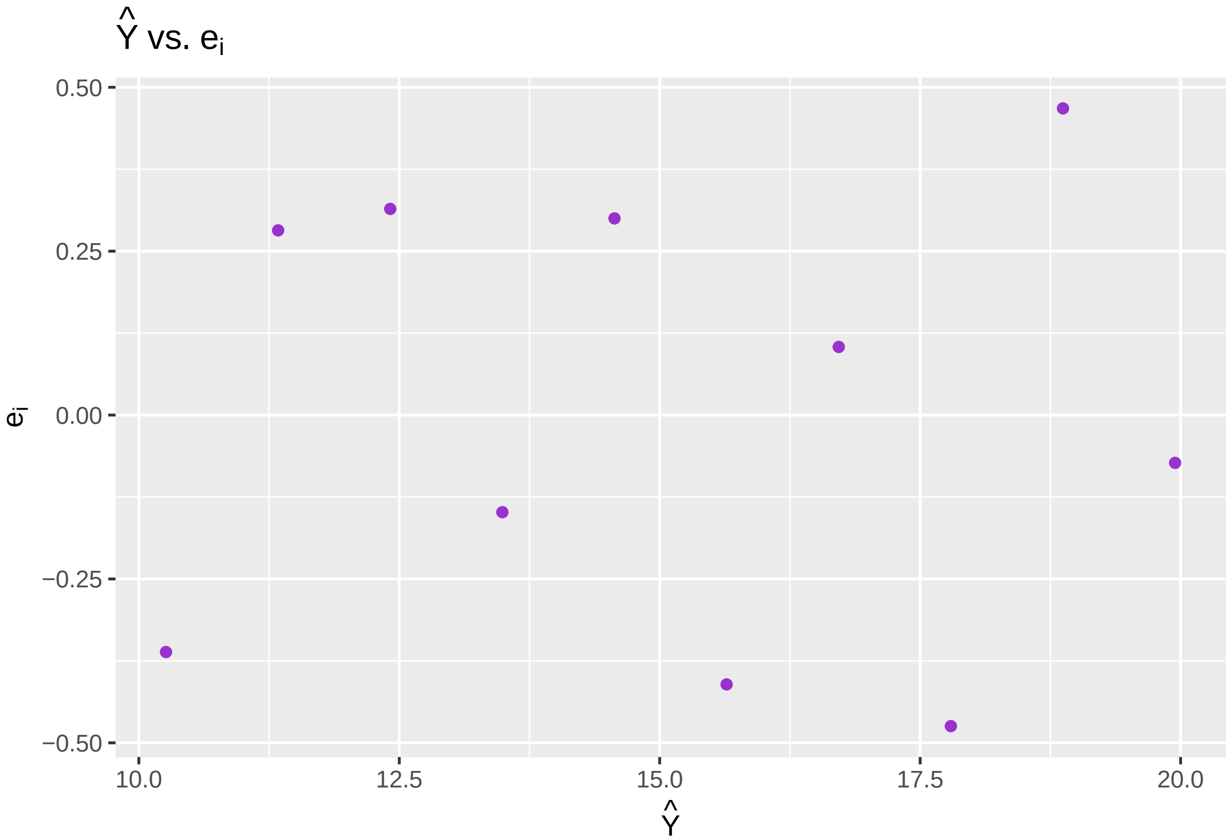
```r
yhat = fitted.values(lmFit317b)
print(yhat)
```

```
##        1        2        3        4        5        6        7        8
## 10.26093 11.33722 12.41352 13.48981 14.56610 15.64239 16.71868 17.79497
##        9       10
## 18.87127 19.94756
```

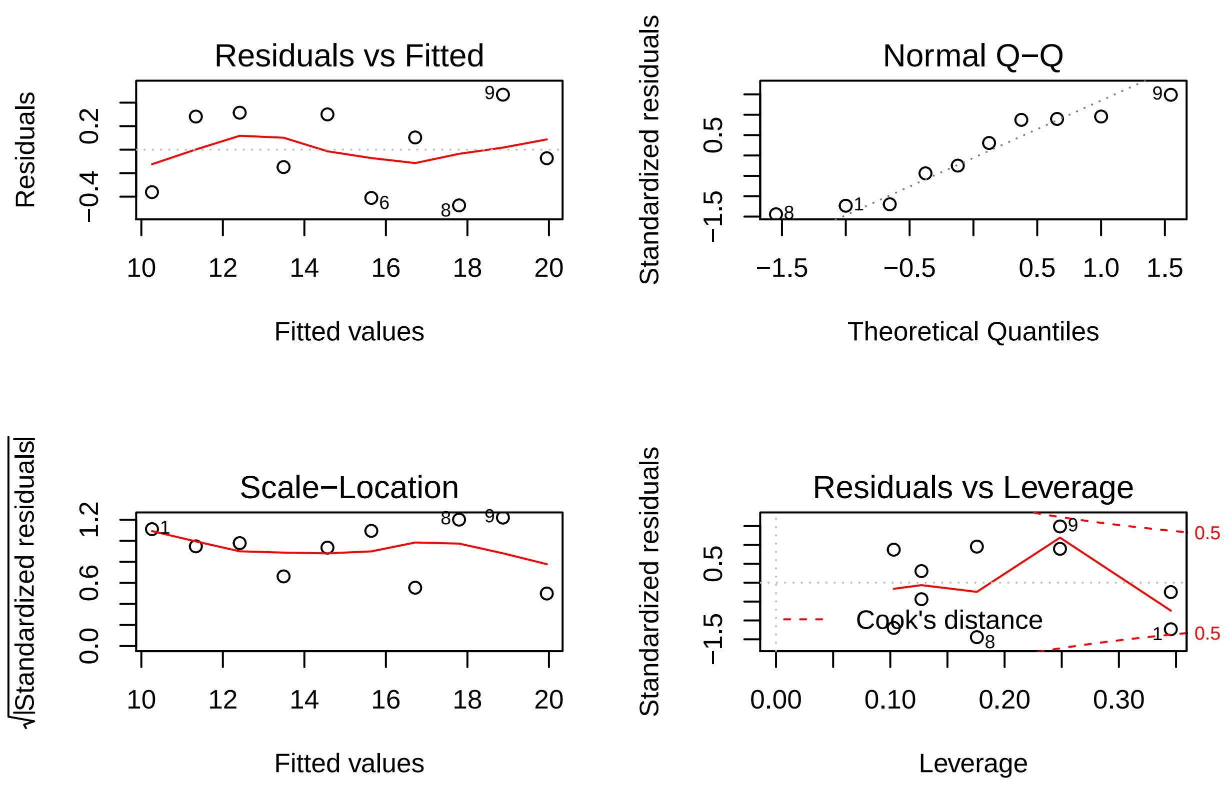
```
# Method #1
plot(yhat, ei, xlab=expression(hat(Y)), ylab=expression("e"[i]),
        main=expression(hat(Y)~"vs."~"e"[i]))
```




```
# Method #2
plot_317e = ggplot(mapping=aes(x=yhat, y=ei)) +
        geom_point(color="darkorchid") +
        labs(title=expression(hat(Y)~"vs."~"e"[i]),
                 x=expression(hat(Y)), y=expression("e"[i]))

plot_317e
```




```r
par(mfrow=c(2,2))
plot(lmFit317b)
```



Interpretation

Residuals vs Fitted plot shows if residuals have non-linear patterns. Equally spread residuals around a horizontal line without distinct patterns, which suggests there aren't non-linear relationships.

Normal Q-Q plot shows if residuals are normally distributed. QQ plot indicates "S" shape, which shows heavy tails. This suggests the data have more extreme values than would be expected if they truly came from a Normal distribution.

Spread-Location plot shows if residuals are spread equally along the ranges of predictors and how we can check the assumption of equal variance (homoscedasticity). A horizontal line suggests equally (randomly) spread points.

Residuals vs Leverage plot helps us to find influential cases (i.e., subjects) if any exists. Plot shows no influential cases, as we can barely see Cook's distance lines (a red dashed line) because all cases are well inside of the Cook's distance lines.

f. Express the estimated regression function in the original units.

Solution Below

Interpretation

Since the Box-Cox suggested $\lambda = 0.5$ for transformation (i.e., the square root of the original data), the back-transformation for the original units involves squaring the transformed data.