Rexample

Lingxiao Zhou

Bollywood Box Office Data

- Movie Budgets (X)
- Box Office Grosses (Y)

```
library(car)  # for significance tests
bbo <- read.csv("https://raw.githubusercontent.com/lingxiaozhou/STA4210Rmaterial/main/data/bollywood_bo
    header = T)
attach(bbo)
names(bbo)
```

```
## [1] "Movie"  "Gross"  "Budget"
```

```
bbo.reg1 <- lm(Gross ~ Budget)
summary(bbo.reg1)
```

```
##
## Call:
## lm(formula = Gross ~ Budget)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -131.349  -14.114   -6.371    9.195  125.236
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.9549     7.2385  -0.270    0.788
## Budget        1.2510     0.1359   9.204 1.41e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.51 on 53 degrees of freedom
## Multiple R-squared:  0.6151, Adjusted R-squared:  0.6079
## F-statistic: 84.71 on 1 and 53 DF,  p-value: 1.411e-12
```

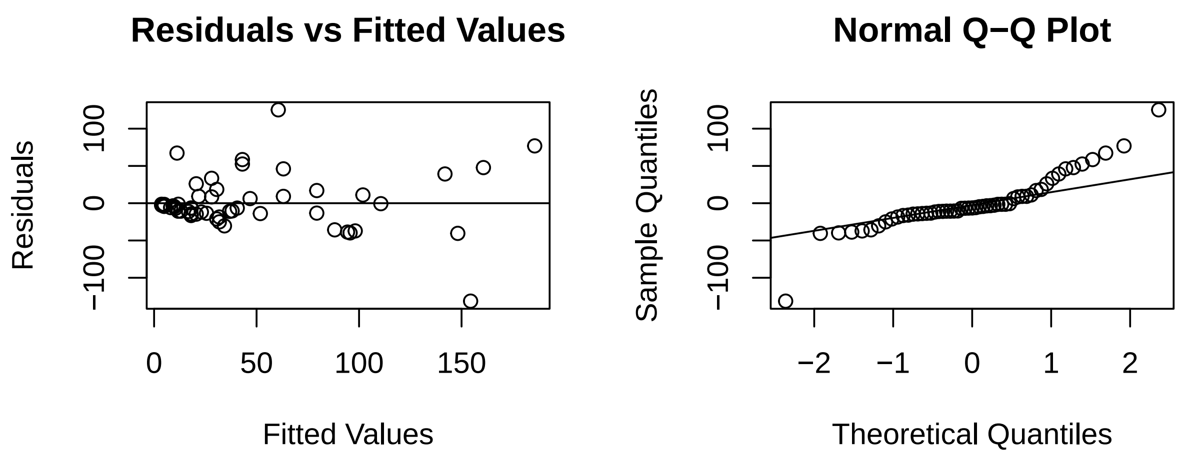
```
e1 <- residuals(bbo.reg1)
yhat1 <- predict(bbo.reg1)

par(mfrow = c(1, 2))

# Residual vs. fitted
```

```r
plot(yhat1, e1, main = "Residuals vs Fitted Values", xlab = "Fitted Values",
    ylab = "Residuals")
abline(h = 0)

# QQ plot
qqnorm(e1)
qqline(e1)
```




```r
shapiro.test(e1)   # Shapiro-Wilk
```

```
##
##  Shapiro-Wilk normality test
##
## data:  e1
## W = 0.87003, p-value = 2.627e-05
```

```r
ncvTest(bbo.reg1)   # Breusch-Pagan
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 32.22785, Df = 1, p = 1.3711e-08
```

- Diagnostic plots and the significance tests indicate that there are violations to normality and constant variance assumption.

Log transformation of Y

```r
bbo.reg2 <- lm(log(Gross) ~ Budget)
summary(bbo.reg2)
```

```
##
```

```
## Call:
## lm(formula = log(Gross) ~ Budget)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.49440 -0.79585 -0.06396  0.76221  2.42628
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.59676    0.23114   6.908 6.34e-09 ***
## Budget       0.03229    0.00434   7.439 8.86e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.166 on 53 degrees of freedom
## Multiple R-squared:  0.5108, Adjusted R-squared:  0.5016
## F-statistic: 55.34 on 1 and 53 DF,  p-value: 8.859e-10
```

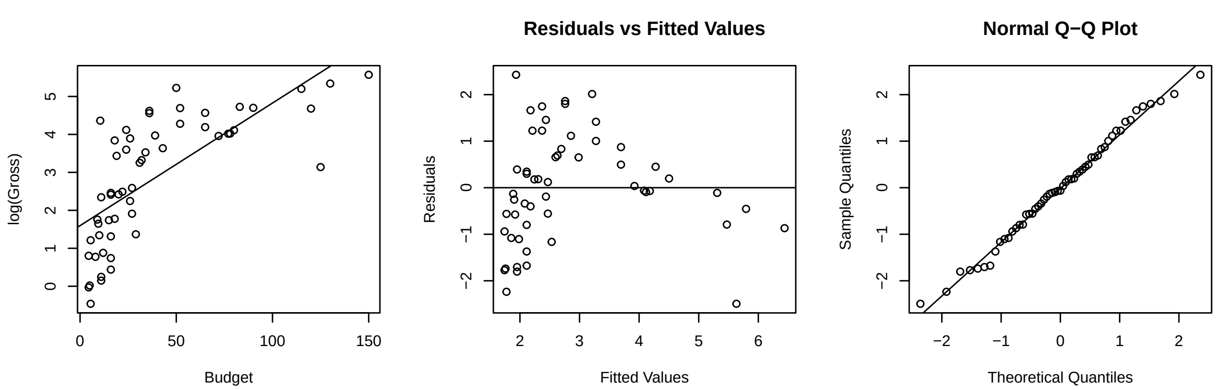
```r
e2 <- residuals(bbo.reg2)
yhat2 <- predict(bbo.reg2)

par(mfrow = c(1, 3))

# Scatter plot
plot(Budget, log(Gross))
abline(bbo.reg2)

# Residual vs. fitted
plot(yhat2, e2, main = "Residuals vs Fitted Values", xlab = "Fitted Values",
    ylab = "Residuals")
abline(h = 0)

# QQ plot
qqnorm(e2)
qqline(e2)
```




```r
shapiro.test(e2)   # Shapiro-Wilk
```

```
##
```

```
##  Shapiro-Wilk normality test
##
## data:  e2
## W = 0.98861, p-value = 0.8803
```

```r
ncvTest(bbo.reg2)  # Breusch-Pagan
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.6531736, Df = 1, p = 0.41898
```

- Based on the Shapiro-Wilk test (Normality) and the Breusch-Pagan test (Constant Variance), the new model appears to be better. However, the linearity assumption is violated based on the residuals vs. fitted plot.

Log transformation of x and Y

```r
bbo.reg3 <- lm(log(Gross) ~ log(Budget))
summary(bbo.reg3)
```

```
##
## Call:
## lm(formula = log(Gross) ~ log(Budget))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0288 -0.4361  0.0317  0.4726  2.8222
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.9038     0.4489  -4.241 8.95e-05 ***
## log(Budget)   1.4645     0.1327  11.034 2.41e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9181 on 53 degrees of freedom
## Multiple R-squared:  0.6967, Adjusted R-squared:  0.691
## F-statistic: 121.7 on 1 and 53 DF,  p-value: 2.411e-15
```

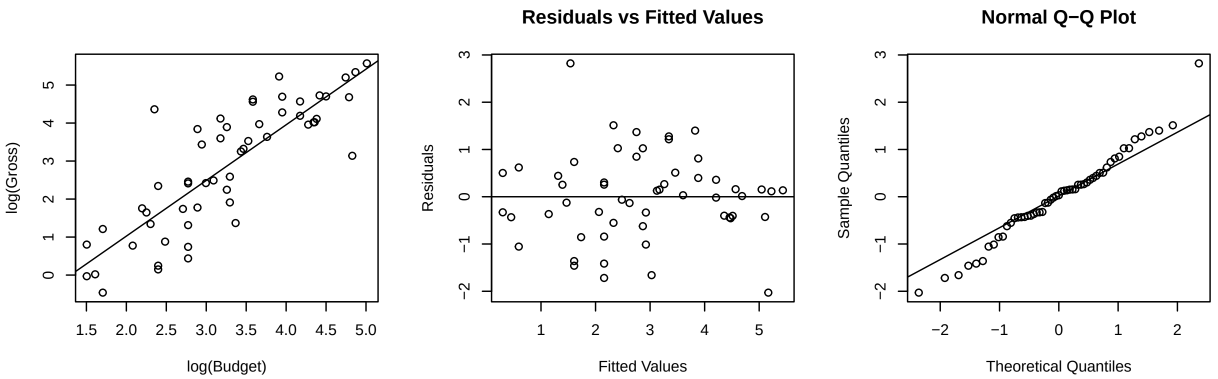
```r
e3 <- residuals(bbo.reg3)
yhat3 <- predict(bbo.reg3)
par(mfrow = c(1, 3))

# Scatter plot
plot(log(Budget), log(Gross))
abline(bbo.reg3)

# Residual vs. fitted
plot(yhat3, e3, main = "Residuals vs Fitted Values", xlab = "Fitted Values",
    ylab = "Residuals")
```

```r
abline(h = 0)

# QQ plot
qqnorm(e3)
qqline(e3)
```




```r
shapiro.test(e3)   # Shapiro-Wilk
```

```
##
##  Shapiro-Wilk normality test
##
## data:  e3
## W = 0.97997, p-value = 0.4866
```

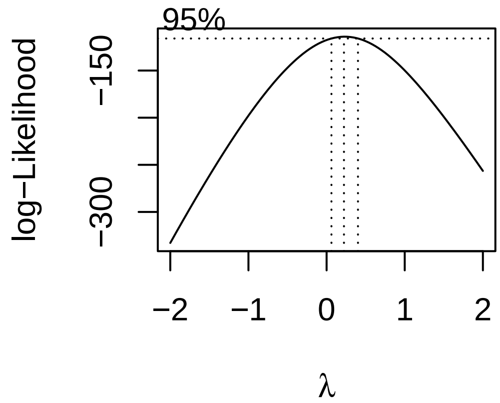
```r
ncvTest(bbo.reg3)   # Breusch-Pagan
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 1.105583, Df = 1, p = 0.29304
```

Box-Cox transformation

- $W = \begin{cases} \frac{Y^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \ln(Y) & \lambda = 0 \end{cases}$

```r
library(MASS)   # for boxcox
bbo.reg4 <- lm(Gross ~ Budget)
bc <- boxcox(bbo.reg4, plotit = T)
```




```
lambda <- bc$x[which.max(bc$y)]
lambda
```

```
## [1] 0.2222222
```

```
# fit new linear regression model using the Box-Cox
# transformation
bbo.reg.bc <- lm(((Gross^lambda - 1)/lambda) ~ Budget)
```

- The optimal λ is 0.22
- The procedure chooses a "quarter root" transformation for Y. We will not pursue that here, as we have seen that log transformations of Y and X work quite well.

Lowess

- Nonparametric method of obtaining a smooth plot of the regression relation between Y and X
- Fits regression in small neighborhoods around points along the regression line on the X axis
- Weights observations closer to the specific point higher than more distant points
- Re-weights after fitting, putting lower weights on larger residuals (in absolute value)
- Obtains fitted value for each point after "final" regression is fit
- Model is plotted along with linear fit, and confidence bands, linear fit is good if lowess lies within bands

```
par(mfrow = c(1, 2))


# find relationship between lot size (X) and work hours (Y)
toluca = read.table("http://www.stat.ufl.edu/~rrandles/sta4210/Rclassnotes/data/textdatasets/KutnerData/
    col.names = c("lotsize", "workhrs"))
plot(toluca$lotsize, toluca$workhrs, xlab = "lotsize", ylab = "workhrs")
toluca.reg = lm(workhrs ~ lotsize, data = toluca)

abline(toluca.reg, col = "darkgreen")
```

```r
x_seq = seq(20, 120, by = 1)
fitl = loess(workhrs ~ lotsize, span = 0.5, data = toluca)  # span controls the size of the neighborhoo
predl = predict(fitl, x_seq, se = TRUE)  # Get predicted y values for x_seq based on lowess model

plot(toluca)
lines(x_seq, predl$fit, lty = 1, col = "darkred")
lines(x_seq, predl$fit - 1.96 * predl$se.fit, lty = 2, col = "blue",
    lwd = 1)
lines(x_seq, predl$fit + 1.96 * predl$se.fit, lty = 2, col = "blue",
    lwd = 1)

polygon(c(x_seq, rev(x_seq)), c(predl$fit + 1.96 * predl$se.fit,
    rev(predl$fit - 1.96 * predl$se.fit)), col = "#00009933",
    border = "NA")

abline(toluca.reg, col = "darkgreen")
legend("bottomright", legend = c("Loess", "95% CB", "SLR"), col = c("darkred",
    "blue", "darkgreen"), lty = c(1, 2, 1), bty = "n")
```