Price Elasticity

Aravind

November 12, 2017

```
setwd("D:/R Projects")
```

Required Libraries

```
#install.packages("memisc")
library(memisc)
```

```
## Warning: package 'memisc' was built under R version 3.4.4
```

```
## Loading required package: lattice
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'memisc'
```

```
## The following objects are masked from 'package:stats':
##
##     contr.sum, contr.treatment, contrasts
```

```
## The following object is masked from 'package:base':
##
##     as.array
```

```
library(car)
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:memisc':
##
##     recode
```

Load Data & File overview

```
sales.data<-read.csv('supermarket.csv')
sapply(sales.data,class)
```

```
##         Sales    Price.Eggs       Ad.Type Price.Cookies
##     "integer"     "numeric"     "integer"     "numeric"
```

Convert Ad.Type into Factor and Print Summary

```
sales.data$Ad.Type<-as.factor(sales.data$Ad.Type)
summary(sales.data)
```

```
##      Sales         Price.Eggs    Ad.Type Price.Cookies
##  Min.   :18.00   Min.   :3.730   0:15    Min.   :4.000
##  1st Qu.:25.25   1st Qu.:4.353   1:15    1st Qu.:4.175
##  Median :28.50   Median :4.480           Median :4.325
##  Mean   :30.00   Mean   :4.426           Mean   :4.374
##  3rd Qu.:33.75   3rd Qu.:4.668           3rd Qu.:4.607
##  Max.   :46.00   Max.   :4.770           Max.   :4.810
```

From the results we find that:

Sales of eggs ranged between 18 and 46 Price of eggs ranged between $3.73 and $4.77 We showed the egg poster 15 times and the cookies poster 15 times Price of cookies ranged between $4 and $4.81

Create Models

```
m1<-lm(formula=Sales~Price.Eggs,data=sales.data)
m2<-update(m1,.~.+Ad.Type)
m3<-update(m2,.~.+Price.Cookies)
mtable(m1,m2,m3)
```

```
##
## Calls:
## m1: lm(formula = Sales ~ Price.Eggs, data = sales.data)
## m2: lm(formula = Sales ~ Price.Eggs + Ad.Type, data = sales.data)
## m3: lm(formula = Sales ~ Price.Eggs + Ad.Type + Price.Cookies, data = sales.data)
##
## ======================================================
##                       m1          m2          m3
## ------------------------------------------------------
##   (Intercept)     115.366***  101.571***  137.370***
##                   (10.041)     (9.899)    (10.834)
##   Price.Eggs      -19.286***  -16.643***  -16.118***
##                    (2.263)     (2.166)     (1.646)
##   Ad.Type: 1/0                  4.195**     4.147***
##                               (1.371)     (1.040)
##   Price.Cookies                            -8.711***
##                                            (1.901)
## ------------------------------------------------------
##   R-squared         0.722       0.793       0.886
##   adj. R-squared    0.712       0.778       0.872
##   sigma             3.924       3.444       2.611
##   F                72.644      51.838      67.140
##   p                 0.000       0.000       0.000
##   Log-likelihood  -82.550     -78.087     -69.210
##   Deviance        431.225     320.258     177.204
##   AIC             171.099     164.174     148.419
##   BIC             175.303     169.779     155.425
##   N                30          30          30
## ======================================================
```

The results are pasted below. We end up with a model "m3" that has statistically significant predictors. Our model is:

Sales of Eggs = 137.37 - (16.12)Price.Eggs + 4.15 (Ad.Type) - (8.71)Price.Cookies

We look at our R2 and see that the regression explains 88.6% of the variance in the data. We also have a low mean squared error (2.611) compared to the other models we generated.

We can actually get better results by transforming our independent and dependent variables (e.g. LN (Sales)) but this will suffice for demonstrating how we can use regressions to calculate price elasticity.

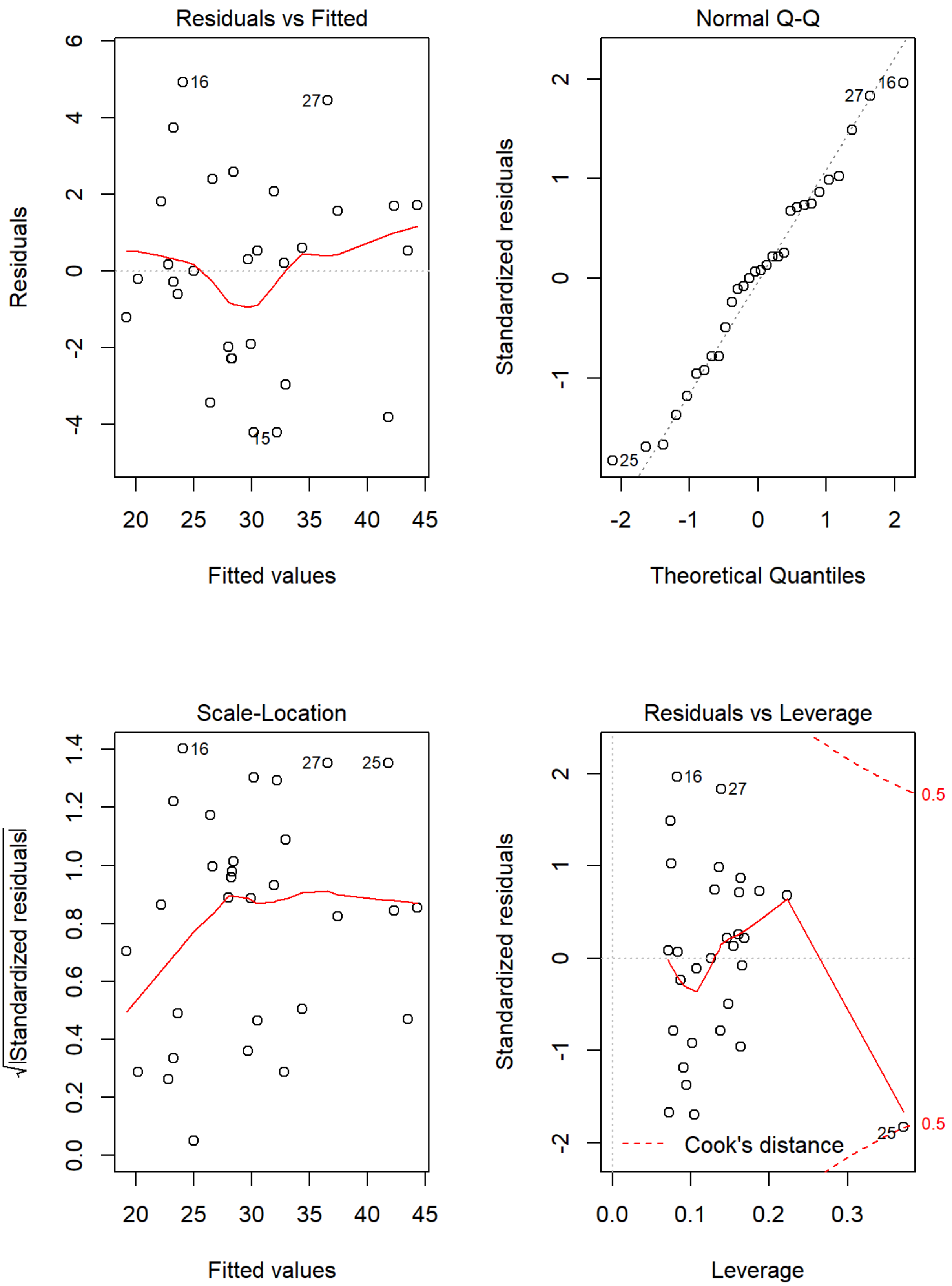
Diagnostics

We need to test for the following assumptions whenever we do regression analysis:

1. The relationship is linear
2. The errors have the same variance
3. The errors are independent of each other
4. The errors are normally distributed

Linearity Plots

```
par(mfrow=c(1,2))
plot(m3)
```



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

Check for multicollineraity

```
vif(m3)
```

```
##     Price.Eggs       Ad.Type Price.Cookies 
##       1.195107      1.189436      1.006566
```

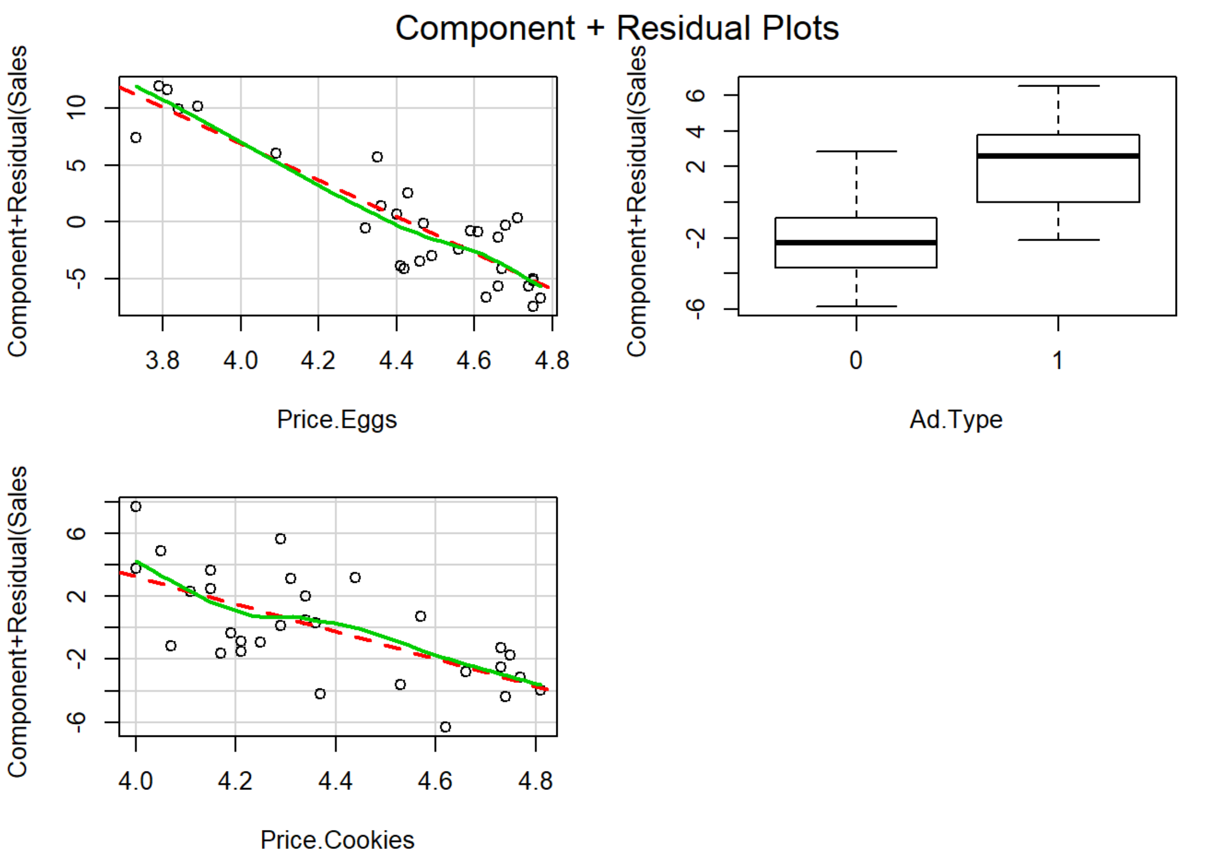
```
sqrt(vif(m3)) > 2
```

```
##     Price.Eggs       Ad.Type Price.Cookies 
##          FALSE         FALSE         FALSE
```

The code above will show if any of the variables have multicolinearity issues that could cause issues with the model’s integrity. Generally we want values less than 2, and we have values of around 1 so we are good on this front.

Diagnosis: Nonlinearity

```
crPlots(m3)
```



We see that there is definitely some issues with linearity but not to an extent that it is a cause for concern for the purpose of demonstration. So we keep calm, and move on.

Diagnosis: Non-independence of Errors We want a D-W Statistic close to 2

```
durbinWatsonTest(m3)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.05087141      1.873466    0.65
##  Alternative hypothesis: rho != 0
```

Compute Price Elasticity

Own Price Elasticity

```
PE<-as.numeric(m3$coefficients["Price.Eggs"] * mean(sales.data$Price.Eggs)/mean(sale
s.data$Sales))
PE
```

```
## [1] -2.378184
```

PE of 2.378 means that an increase in the price of eggs by 1 unit will decrease the sales by 2.38 units.

Cross Price Elasticity

```
CPEcookies<-as.numeric(m3$coefficients["Price.Cookies"] * mean(sales.data$Price.Cook
ies)/mean(sales.data$Sales))
CPEcookies
```

```
## [1] -1.269935
```

PE of 1.269 means that an increase in the price of cookies by 1 unit will decrease the sales of eggs by 1.27 units.

Interpretation

We now know that the price of eggs and price of cookies are complementary to one another in this scenario. Since you only sell too products, one explanation could be that people who come in for cookies and eggs would rather get them elsewhere if the price is too high.