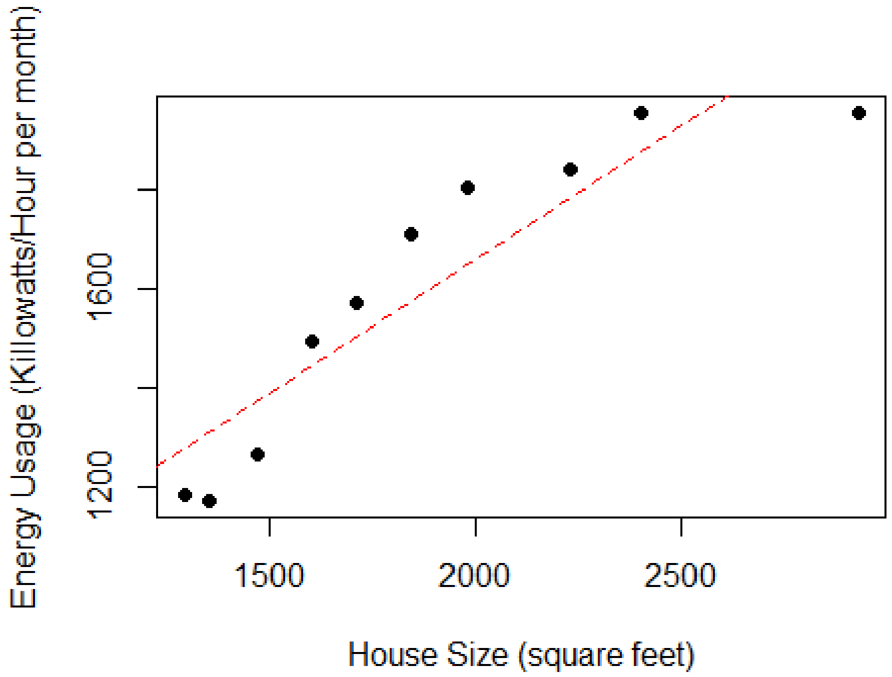
Regression_3_Extensions.R

Curvilinear Section

```
# A real estate agency collects data concerning energy usage (kilowatt/hour p
er month) and house sizes (square feet).
#install readxl package first
library(readxl)
energy <- read_excel("energy.xlsx", na="NA", col_names = TRUE)
attach(energy)
```

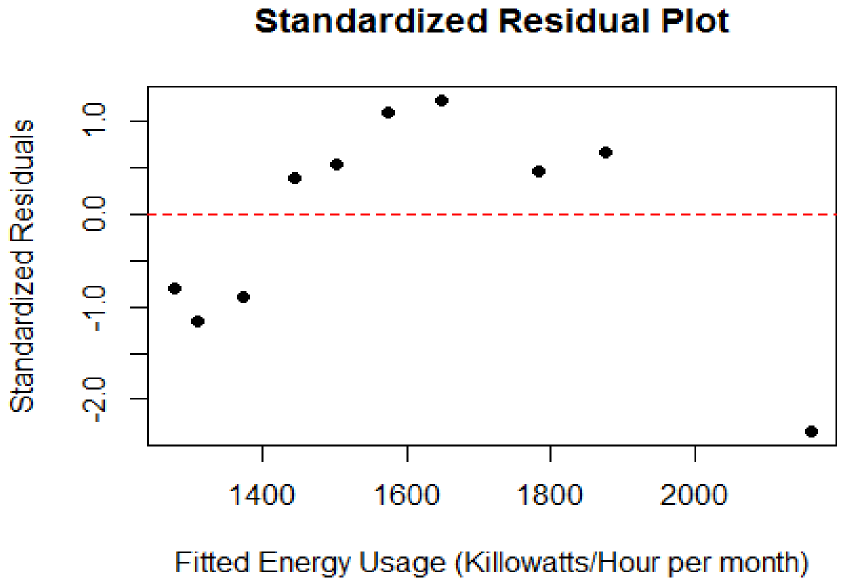
```
# scatter plot
plot(Size, Usage, pch = 16, xlab = "House Size (square feet)", ylab = "Energy
Usage (Killowatts/Hour per month)")
abline(lm(Usage ~ Size), lty=2, col="red")
```



```r
# fit the linear model
linefit <- lm(Usage ~ Size)
summary(linefit)

##
## Call:
## lm(formula = Usage ~ Size)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -208.02 -105.36   52.89   77.29  155.27
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 578.92775  166.96806   3.467 0.008476 **
## Size          0.54030    0.08593   6.288 0.000236 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 133.4 on 8 degrees of freedom
## Multiple R-squared:  0.8317, Adjusted R-squared:  0.8107
## F-statistic: 39.54 on 1 and 8 DF,  p-value: 0.0002359

# standardized residual plot - on fitted values
linefit.stres <- rstandard(linefit)
plot(linefit$fitted.values, linefit.stres, pch = 16, main = "Standardized Res
idual Plot", xlab = "Fitted Energy Usage (Killowatts/Hour per month)", ylab =
"Standardized Residuals")
abline(0,0, lty=2, col="red")
```



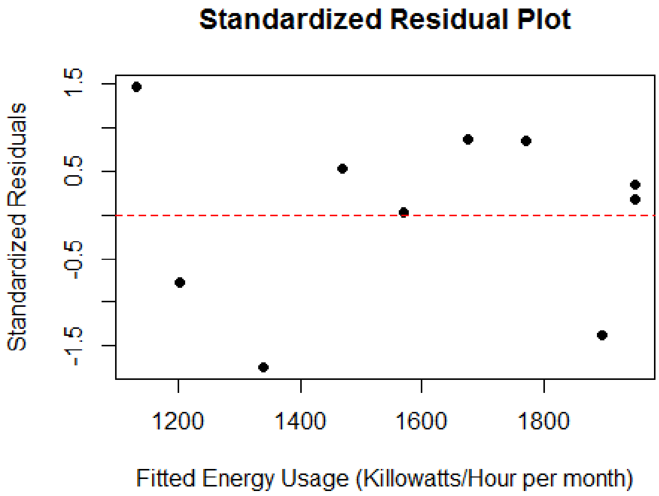
```
# fit the quadratic model
energy$SizeSqd <- Size^2
attach(energy)
```

```
## The following objects are masked from energy (pos = 3):
## 
##     Size, Usage
```

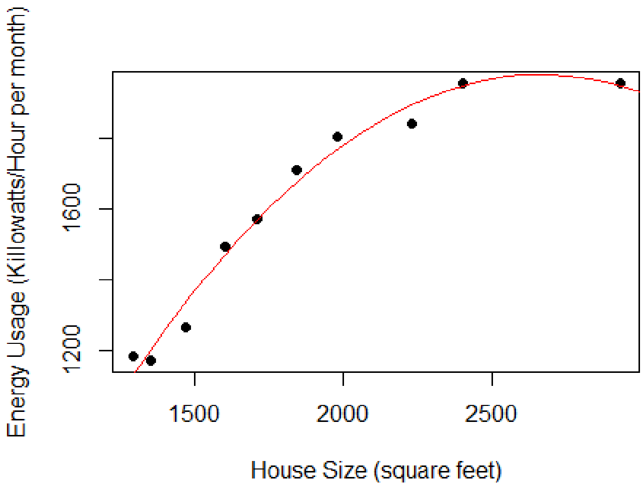
```
linefitQ <- lm(Usage ~ Size + SizeSqd)
summary(linefitQ)
```

```
## 
## Call:
## lm(formula = Usage ~ Size + SizeSqd)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -73.792 -22.426   5.886  31.689  52.436 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.216e+03  2.428e+02  -5.009 0.001550 ** 
## Size         2.399e+00  2.458e-01   9.758 2.51e-05 ***
## SizeSqd     -4.500e-04  5.908e-05  -7.618 0.000124 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 46.8 on 7 degrees of freedom
## Multiple R-squared:  0.9819, Adjusted R-squared:  0.9767 
## F-statistic: 189.7 on 2 and 7 DF,  p-value: 8.001e-07
```

```r
# standardized residual plot - on fitted values
linefitQ.stres <- rstandard(linefitQ)
plot(linefitQ$fitted.values, linefitQ.stres, pch = 16, main = "Standardized R
esidual Plot", xlab = "Fitted Energy Usage (Killowatts/Hour per month)", ylab
= "Standardized Residuals")
abline(0,0, lty=2, col="red")
```



```r
# scatter plot with fitted quadratic model curve
XvaluesQ <- seq(1000, 3000, 50)
YpredictedQ <- linefitQ$coefficients[3]*XvaluesQ^2 + linefitQ$coefficients[2]
*XvaluesQ + linefitQ$coefficients[1]
plot(Size, Usage, pch = 16, xlab = "House Size (square feet)", ylab = "Energy
Usage (Killowatts/Hour per month)")
lines(XvaluesQ, YpredictedQ, type = "l", col = "red")
```



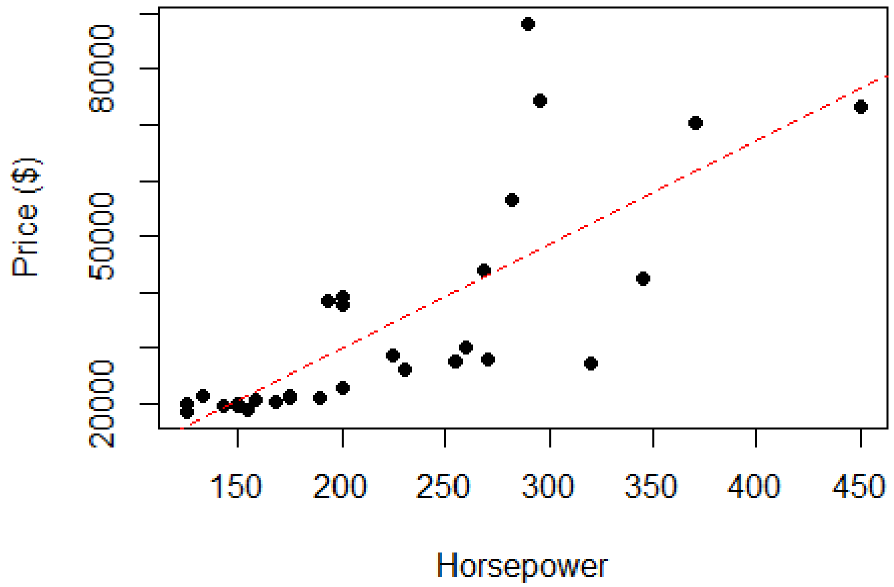
```r
detach(energy)
```

Categorical Section

```
# A company is interested in the relationship between sales price ($) and hor
sepower and type of car
#install readxl package first
library(readxl)
cars <- read_excel("cars.xlsx", na="NA", col_names = TRUE)

# convert Type to a factor variable TypeF, i.e., nominal
cars$TypeF<-factor(cars$Type)
attach(cars)

# scatter plot with Horsepower
plot(Horsepower, Price, pch = 16, xlab = "Horsepower", ylab = "Price ($)")
abline(lm(Price ~ Horsepower), lty=2, col="red")
```



```
# fit the simple linear regression model
linefitH <- lm(Price ~ Horsepower)
summary(linefitH)
```

```
##
## Call:
## lm(formula = Price ~ Horsepower)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -24972  -6649  -1218   3845  41584
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7203.81    6923.91  -1.040    0.307
## Horsepower    185.36      29.42   6.301 8.18e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12730 on 28 degrees of freedom
## Multiple R-squared:  0.5864, Adjusted R-squared:  0.5716
## F-statistic:  39.7 on 1 and 28 DF,  p-value: 8.182e-07
```

```
# fit the ANOVA
fitT <- aov(Price ~ TypeF)
summary(fitT)
##             Df    Sum Sq   Mean Sq F value   Pr(>F)    
## TypeF        4 8.160e+09 2.040e+09   18.19 4.07e-07 ***
## Residuals   25 2.803e+09 1.121e+08                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

print(model.tables(fitT, "means"))

## Tables of means
## Grand mean
## 33891.6
##  TypeF
##          1     2     3     4     5
##      20162 37554 20790 27981 61663
## rep      9     5     5     4     7

  # to check all the pairwise contrasts
TukeyHSD(fitT, conf.level = .90)

##   Tukey multiple comparisons of means
##     90% family-wise confidence level
## Fit: aov(formula = Price ~ TypeF)
## $TypeF
##             diff        lwr        upr     p adj
## 2-1  17392.5111   2010.283 32774.7394 0.0491675
## 3-1    628.3111 -14753.917 16010.5394 0.9999689
## 4-1   7819.3611  -8752.906 24391.6280 0.7350784
## 5-1  41501.3968  27603.432 55399.3617 0.0000004
## 3-2 -16764.2000 -34206.007   677.6074 0.1219798
## 4-2  -9573.1500 -28072.980  8926.6805 0.6651140
## 5-2  24108.8857   7960.910 40256.8616 0.0054346
## 4-3   7191.0500 -11308.780 25690.8805 0.8472906
## 5-3  40873.0857  24725.110 57021.0616 0.0000062
## 5-4  33682.0357  16396.660 50967.4112 0.0002746

  # SSTreatment
anova(fitT)[["Sum Sq"]][1]
## [1] 8160453274
  # SSError
anova(fitT)[["Sum Sq"]][2]
## [1] 2803444345
  # R^2
(anova(fitT)[["Sum Sq"]][1])/((anova(fitT)[["Sum Sq"]][1])+(anova(fitT)[["Sum
Sq"]][2]))
## [1] 0.7443022
  # sqrt(MSE)
sqrt(anova(fitT)[["Sum Sq"]][2]/fitT$df.residual)
## [1] 10589.51
```

```
# collinearity check between 2 predictors
fitM <- aov(Horsepower ~ TypeF)
summary(fitM)

##             Df Sum Sq Mean Sq F value   Pr(>F)    
## TypeF        4 155613   38903   30.88 2.42e-09 ***
## Residuals   25  31499    1260                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

  # R^2
(anova(fitM)[["Sum Sq"]][1])/((anova(fitM)[["Sum Sq"]][1])+(anova(fitM)[["Sum
Sq"]][2]))

## [1] 0.8316561
```

```
# fit the first-order multiple regression model (TypeF=1 as base category)
linefitHT <- lm(Price ~ Horsepower + TypeF)
summary(linefitHT)
## Call:
## lm(formula = Price ~ Horsepower + TypeF)
## Residuals:
##    Min     1Q Median     3Q    Max 
## -34013   -605    -15   1740  27929 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 15406.19    9801.66   1.572  0.12909   
## Horsepower     31.56      60.55   0.521  0.60697   
## TypeF2      15292.43    7222.61   2.117  0.04479 * 
## TypeF3         74.88    6087.81   0.012  0.99029   
## TypeF4       4565.60    8981.86   0.508  0.61587   
## TypeF5      35646.96   12469.10   2.859  0.00866 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10750 on 24 degrees of freedom
## Multiple R-squared:  0.7472, Adjusted R-squared:  0.6945 
## F-statistic: 14.18 on 5 and 24 DF,  p-value: 1.662e-06
```

```
# fit the first-order model (TypeF=2 as base category)
  # reorder the data frame with TypeF = 2 as reference
cars <- within(cars, TypeF <- relevel(TypeF, ref = 2))
linefitHTalt <- lm(cars$Price ~ cars$Horsepower + cars$TypeF)
summary(linefitHTalt)
## Call:
## lm(formula = cars$Price ~ cars$Horsepower + cars$TypeF)
## Residuals:
##    Min     1Q Median     3Q    Max
## -34013   -605    -15   1740  27929
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      30698.62   14003.10   2.192   0.0383 *
## cars$Horsepower     31.56      60.55   0.521   0.6070
## cars$TypeF1     -15292.43    7222.61  -2.117   0.0448 *
## cars$TypeF3     -15217.55    7416.54  -2.052   0.0512 .
## cars$TypeF4     -10726.83    7541.53  -1.422   0.1678
## cars$TypeF5      20354.53    9564.37   2.128   0.0438 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10750 on 24 degrees of freedom
## Multiple R-squared:  0.7472, Adjusted R-squared:  0.6945
## F-statistic: 14.18 on 5 and 24 DF,  p-value: 1.662e-06

  # return dataframe to original order with TypeF = 1 as reference
cars <- within(cars, TypeF <- relevel(TypeF, ref = 2))
```

```
# fit the first-order model (TypeF=3 as base category)
  # reorder the data frame with TypeF = 3 as reference
cars <- within(cars, TypeF <- relevel(TypeF, ref = 3))
linefitHTalt <- lm(cars$Price ~ cars$Horsepower + cars$TypeF)
summary(linefitHTalt)

##
## Call:
## lm(formula = cars$Price ~ cars$Horsepower + cars$TypeF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -34013   -605    -15   1740  27929
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      15481.07   11262.33   1.375  0.18196
## cars$Horsepower     31.56      60.55   0.521  0.60697
## cars$TypeF1        -74.88    6087.81  -0.012  0.99029
## cars$TypeF2      15217.55    7416.54   2.052  0.05125 .
## cars$TypeF4       4490.72    8877.67   0.506  0.61758
## cars$TypeF5      35572.07   11959.24   2.974  0.00659 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10750 on 24 degrees of freedom
## Multiple R-squared:  0.7472, Adjusted R-squared:  0.6945
## F-statistic: 14.18 on 5 and 24 DF,  p-value: 1.662e-06

  # return dataframe to original order with TypeF = 1 as reference
  # for switch to Type = k as reference, run the re-sort (k-1) times to undo
  # You can check the order of the values for TypeF under the Cars dataframe
in the Environment window at right to see what is happening
cars <- within(cars, TypeF <- relevel(TypeF, ref = 3))
cars <- within(cars, TypeF <- relevel(TypeF, ref = 3))


# This process can be repeated for using the other types as the base
# fit the second-order multiple regression model with interaction term (TypeF
=1 as base category)
linefitHT2 <- lm(Price ~ Horsepower * TypeF)
summary(linefitHT2)

##
## Call:
## lm(formula = Price ~ Horsepower * TypeF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -34112   -589    -16   1845  27644
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          14297.93   27059.84   0.528    0.603
## Horsepower              38.92     177.70   0.219    0.829
## TypeF2               12524.07   49700.81   0.252    0.804
## TypeF3                1049.22   61294.99   0.017    0.987
## TypeF4               -2814.08  105012.83  -0.027    0.979
## TypeF5               38832.25   38634.75   1.005    0.327
## Horsepower:TypeF2       10.49     260.44   0.040    0.968
## Horsepower:TypeF3       -6.56     370.83  -0.018    0.986
## Horsepower:TypeF4       26.09     436.96   0.060    0.953
## Horsepower:TypeF5      -13.54     195.28  -0.069    0.945
##
## Residual standard error: 11770 on 20 degrees of freedom
## Multiple R-squared:  0.7475, Adjusted R-squared:  0.6338
## F-statistic: 6.577 on 9 and 20 DF,  p-value: 0.0002325
```

```r
# fit the second-order model with interaction term (TypeF=5 as base category)
  # reorder the data frame with TypeF = 5 as reference
cars <- within(cars, TypeF <- relevel(TypeF, ref = 5))
linefitHTalt <- lm(cars$Price ~ cars$Horsepower * cars$TypeF)
summary(linefitHTalt)
```

```
##
## Call:
## lm(formula = cars$Price ~ cars$Horsepower * cars$TypeF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -34112   -589    -16   1845  27644
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    53130.176  27575.515   1.927   0.0683 .
## cars$Horsepower                   25.385     80.961   0.314   0.7571
## cars$TypeF1                   -38832.247  38634.751  -1.005   0.3268
## cars$TypeF2                   -26308.179  49983.438  -0.526   0.6044
## cars$TypeF3                   -37783.025  61524.387  -0.614   0.5461
## cars$TypeF4                   -41646.327 105146.893  -0.396   0.6962
## cars$Horsepower:cars$TypeF1       13.535    195.278   0.069   0.9454
## cars$Horsepower:cars$TypeF2       24.027    206.899   0.116   0.9087
## cars$Horsepower:cars$TypeF3        6.975    335.401   0.021   0.9836
## cars$Horsepower:cars$TypeF4       39.629    407.323   0.097   0.9235
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11770 on 20 degrees of freedom
## Multiple R-squared:  0.7475, Adjusted R-squared:  0.6338
## F-statistic: 6.577 on 9 and 20 DF,  p-value: 0.0002325
```

```r
  # return dataframe to original order with TypeF = 1 as reference
cars <- within(cars, TypeF <- relevel(TypeF, ref = 5))
cars <- within(cars, TypeF <- relevel(TypeF, ref = 5))
cars <- within(cars, TypeF <- relevel(TypeF, ref = 5))
cars <- within(cars, TypeF <- relevel(TypeF, ref = 5))


# clean up
detach(cars)
```