Homework 7

Emily Lu

6/4/2019

1. The data set mantel in the alr4 package has a response Y and three predictors X1, X2 and X3, apply the forward selection and backward elimination algorithms, using AIC as a criterion function. Also, find AIC and BIC for all possible models and compare results. Which appear to be the active regressors?

Forward Selection procedure adds variables one at a time until the chosen information criterion cannot be decreased anymore.

```r
library(alr4)
```

```
## Loading required package: car

## Loading required package: carData

## Loading required package: effects

## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```r
attach(mantel)

# With AIC
mod0 <- lm(Y ~ 1, data = mantel)
modfull <- lm(Y ~., data = mantel)

step(mod0, scope = list(lower = mod0, upper = modfull),
     direction = 'forward')
```

```
## Start:  AIC=9.59
## Y ~ 1
##
##         Df Sum of Sq     RSS     AIC
## + X3     1   20.6879  2.1121 -0.3087
## + X1     1    8.6112 14.1888  9.2151
## + X2     1    8.5064 14.2936  9.2519
## <none>               22.8000  9.5866
##
## Step:  AIC=-0.31
## Y ~ X3
##
##         Df Sum of Sq    RSS      AIC
## <none>               2.1121 -0.30875
## + X2     1  0.066328 2.0458  1.53172
## + X1     1  0.064522 2.0476  1.53613

##
## Call:
## lm(formula = Y ~ X3, data = mantel)
##
## Coefficients:
## (Intercept)           X3
```

```
##      0.7975       0.6947
```

```r
# With BIC
n <- length(mantel$Y)
step(mod0, scope = list(lower = mod0, upper = modfull),
     direction = 'forward', k = log(n), trace = 0)
```

```
##
## Call:
## lm(formula = Y ~ X3, data = mantel)
##
## Coefficients:
## (Intercept)           X3
##      0.7975       0.6947
```

With the forward selection for both AIC and BIC, the final model only has X3 as an active regressor.

Backward Elimination:

```r
# With AIC
step(modfull, scope = list(lower = mod0, upper = modfull),
     direction = 'backward')
```

```
## Start:  AIC=-285.77
## Y ~ X1 + X2 + X3

## Warning: attempting model selection on an essentially perfect fit is
## nonsense

##        Df Sum of Sq    RSS      AIC
## - X3    1    0.0000 0.0000 -287.749
## <none>              0.0000 -285.768
## - X1    1    2.0458 2.0458    1.532
## - X2    1    2.0476 2.0476    1.536
##
## Step:  AIC=-287.75
## Y ~ X1 + X2

## Warning: attempting model selection on an essentially perfect fit is
## nonsense

##        Df Sum of Sq    RSS      AIC
## <none>               0.000 -287.749
## - X2    1    14.189 14.189    9.215
## - X1    1    14.294 14.294    9.252

##
## Call:
## lm(formula = Y ~ X1 + X2, data = mantel)
##
## Coefficients:
## (Intercept)           X1           X2
##       -1000            1            1
```

```r
# With BIC
step(modfull, scope = list(lower = mod0, upper = modfull),
     direction = 'backward', k = log(n), trace = 0)
```

```
## Warning: attempting model selection on an essentially perfect fit is
## nonsense
```

```
## Warning: attempting model selection on an essentially perfect fit is
## nonsense
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2, data = mantel)
##
## Coefficients:
## (Intercept)           X1           X2
##       -1000            1            1
```

With the backward elimination for both AIC and BIC, X1 and X2 appears to be the active regressors.

2. In an unweighted regression problem with n = 54, p = 4, the results included $\hat{\sigma}$ = 4.0$ and the following statistics for four of the cases:

e_i	h_{ii}
1.000	0.900
1.732	0.750
9.000	0.250
10.295	0.185

For each of these four cases, compute r_i, D_i, and t_i. Test each of the four cases to be an outlier. Make a qualitative statement about the influence of each case on the analysis.

```r
ei <- c(1, 1.732, 9, 10.295)
hii <- c(.9, .75, .25, .185)
ri <- ei[1]/(4*sqrt(1-hii[1]))
Di <- ri[1]^2*hii[1]/4 * 1/(1-hii[1])
ti <- ri[1]*sqrt((49)/(50-ri[1]^2))

for(i in c(2:4)){
  r <- ei[i]/(4*sqrt(1-hii[i]))
  ri <- c(ri, r)
  D <- ri[i]^2*hii[i]/4 * 1/(1-hii[i])
  Di <- c(Di, D)
  t <- ri[i]*sqrt((49)/(50-ri[i]^2))
  ti <- c(ti, t)}
```

```
## For i = 1, 2, 3, 4:
```

```
## r_i =  0.7905694 0.866 2.598076 2.850937
```

```
## D_i =  1.40625 0.562467 0.5625 0.4612424
```

```
## t_i =  0.7875615 0.8637988 2.765393 3.084061
```

Based on the standardized residuals, r_i, and unstandardized residuals, t_i, 3^{rd} and 4^{th} are flagged as potential outliers since r_3, r_4, t_3, $t_4 > 2$. Futhermore, Cook's distance measure, D_i, which summarizes how much all the fitted values changes when the i^{th} observation is deleted, flags 1^{st} case to be almost certainly influential.

3. The lathe1 data set from the alr4 package contains the results of an experiment on characterizing the life of a drill bit in cutting steel on a lathe. Two factors were varied in the

experiment, Speed and Feed rate. The response is Life, the total time until the drill bit fails, in minutes. The values of Speed and Feed in the data have been coded by computing

$$\textbf{Speed} = \frac{\textbf{Actual speed in feet per minute} - 900}{300}$$

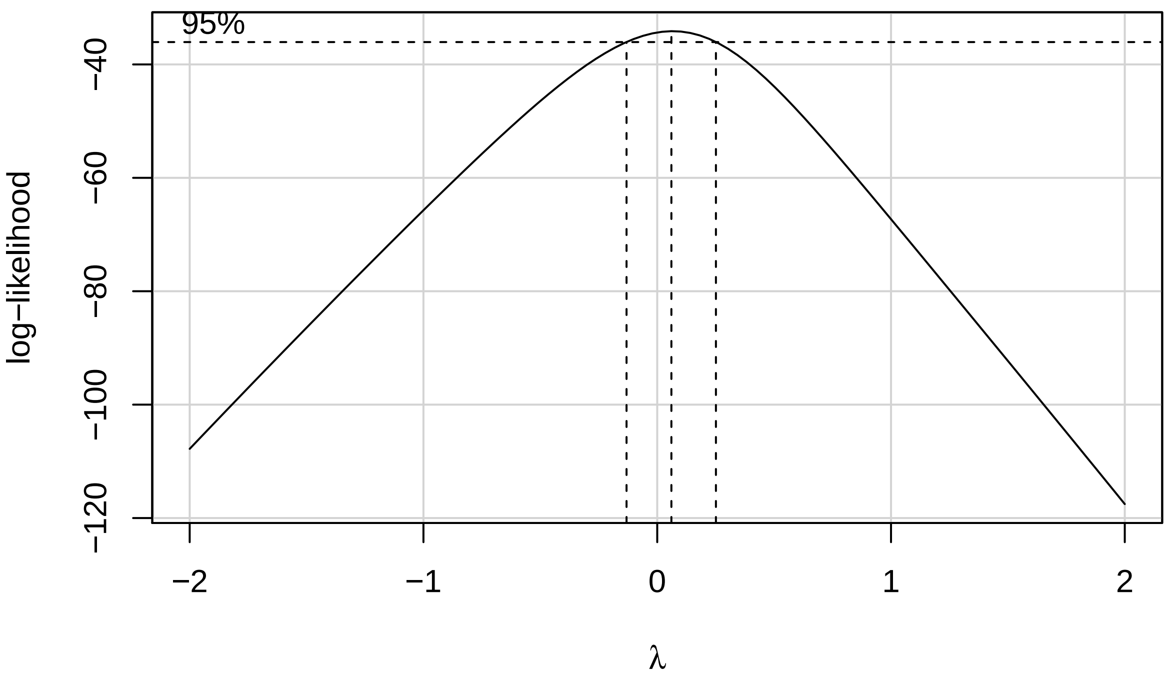
$$\textbf{Feed} = \frac{\textbf{Actual feed rate in thousandths of an inch per revolution} - 13}{6}$$

a. Starting with the full second-order model

$$E[\textbf{Life}|\textbf{Speed, Feed}] = \beta_0 + \beta_1\textbf{Speed} + \beta_2\textbf{Feed} + \beta_{11}\textbf{Speed}^2 + \beta_{22}\textbf{Feed}^2 + \beta_{12}\textbf{Speed*Feed},$$

use the Box–Cox method to show that an appropriate scale for the response is the logarithmic scale.

```r
library(alr4)
attach(lathe1)
model <- lm(Life ~ Speed + Feed + I(Speed^2) + I(Feed^2) + Speed:Feed)
boxCox(model)
```




```r
# We can confirm with the summary function:
summary(powerTransform(model))
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1    0.0659           0      -0.1227       0.2546
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                               LRT df    pval
## LR test, lambda = (0) 0.4569984  1 0.49903
##
## Likelihood ratio test that no transformation is needed
```

```
##                              LRT df       pval
## LR test, lambda = (1) 66.30774  1 3.3307e-16
```

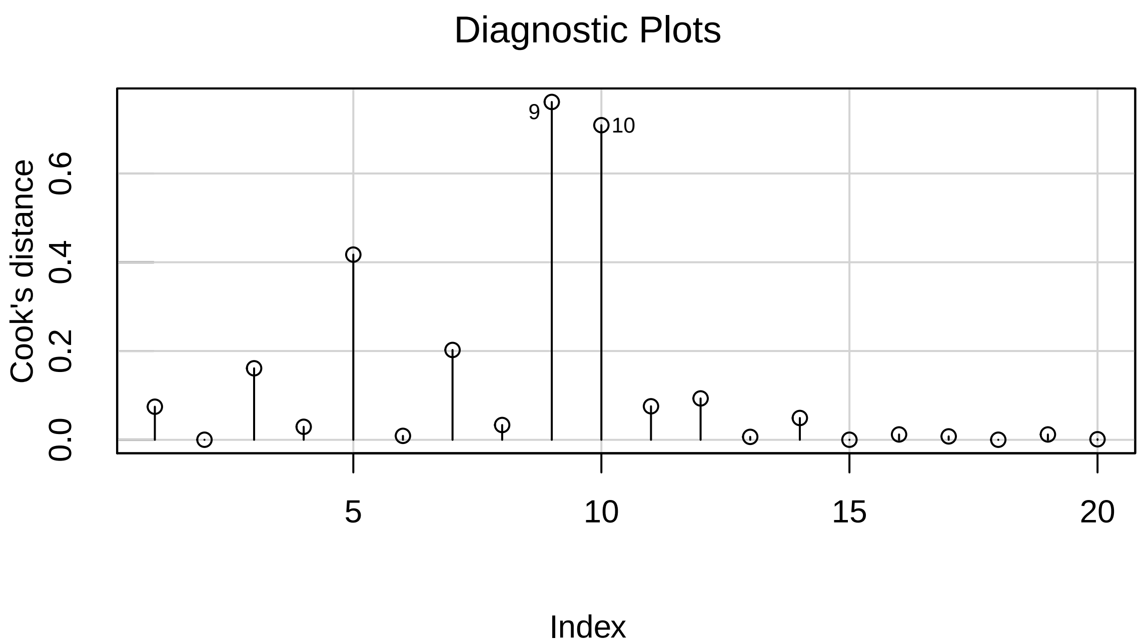
From the Box-Cox method, we see that λ includes 0; therefore, an appropriate scale for the response is the logarithmic scale.

b. Find the two cases that are most influential in the fit of the quadratic mean function for log(Life), and explain why they are influential. Delete these points from the data, refit the quadratic mean function, and compare with the fit with all the data.

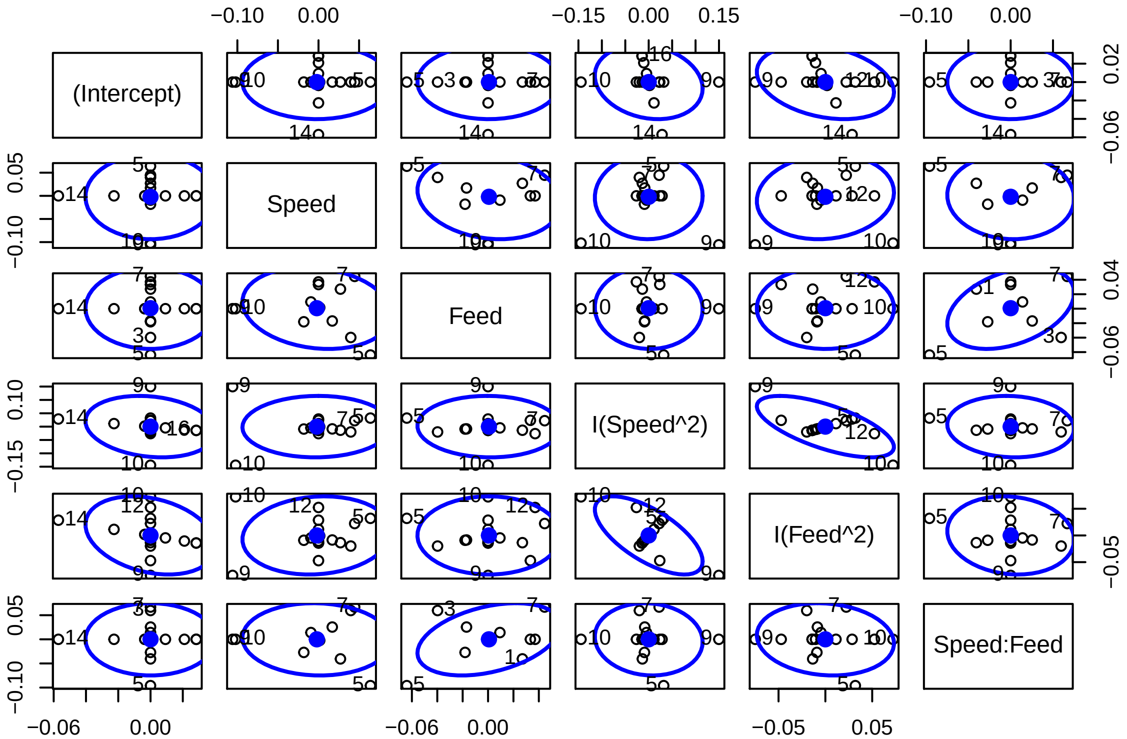
```
ftmodel <- lm(log(Life) ~ Speed + Feed + I(Speed^2) + I(Feed^2) + Speed*Feed)

# find influential cases
ft.cooks <- cooks.distance(ftmodel)
which(ft.cooks > 4/(length(Life)-2-1))
```

```
##  5  9 10
##  5  9 10
```

```
influenceIndexPlot(ftmodel, vars = 'Cook',
                   id = list(location = "avoid", n = 2, cex = 0.7))
```




```
# how these influential points affect the estimated coeffecient
betahat.not.i <- influence(ftmodel)$coefficients
panel.fun <- function(x, y, ...){
  points(x, y, ...)
  dataEllipse(x, y, plot.points=FALSE, levels=c(.90))
  showLabels(x, y, labels=rownames(lathe1),
             method="mahal", n=4) }
pairs(betahat.not.i, panel=panel.fun)
```




```r
# delete influential points from data
noInfluence <- lathe1[c(1:8,11:20),]

# refitted quadratic mean function
full.lathe1 <- lm(log(Life) ~ Speed + Feed + I(Speed^2) + I(Feed^2) + Speed:Feed,
                  data = noInfluence)
```

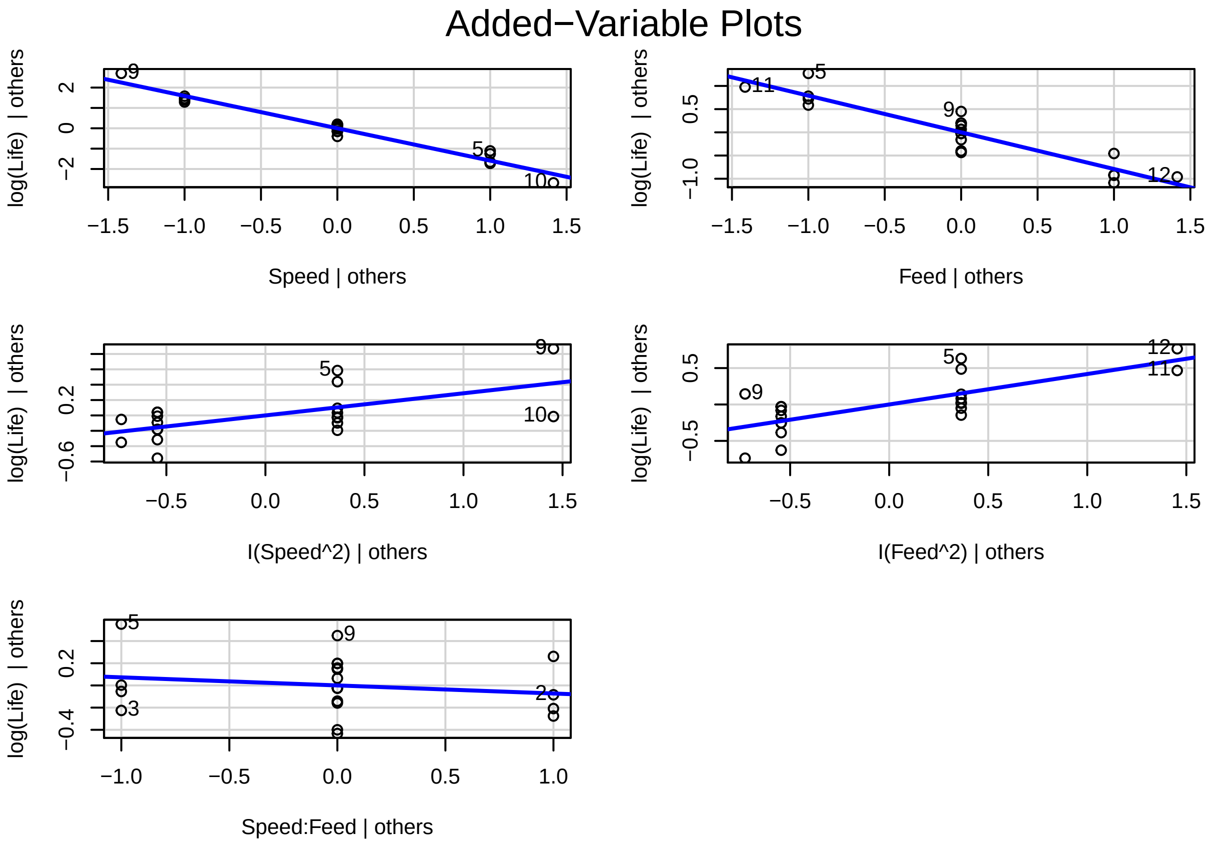
Data with Influential Points Vs. Data with No Influential Points

```r
# influential points included:
summary(ftmodel)
```

```
##
## Call:
## lm(formula = log(Life) ~ Speed + Feed + I(Speed^2) + I(Feed^2) +
##     Speed * Feed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43349 -0.14576 -0.02494  0.16748  0.47992
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.18809    0.10508  11.307 2.00e-08 ***
## Speed       -1.58902    0.08580 -18.520 3.04e-11 ***
## Feed        -0.79023    0.08580  -9.210 2.56e-07 ***
## I(Speed^2)   0.28808    0.10063   2.863 0.012529 *
## I(Feed^2)    0.41851    0.10063   4.159 0.000964 ***
## Speed:Feed  -0.07286    0.10508  -0.693 0.499426
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.2972 on 14 degrees of freedom
## Multiple R-squared:  0.9702, Adjusted R-squared:  0.9596
## F-statistic: 91.24 on 5 and 14 DF,  p-value: 3.551e-10
```

```r
avPlots(ftmodel)
```

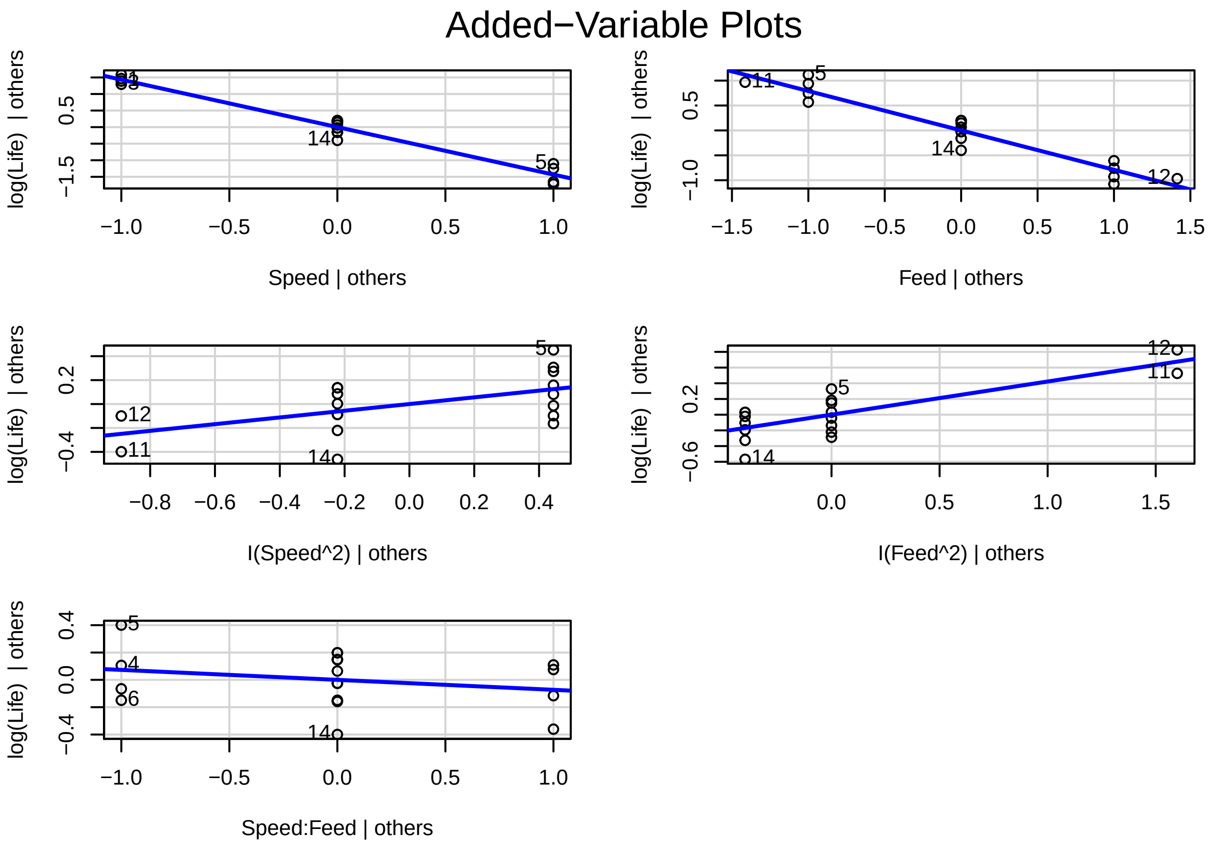



```r
# no influential points:
summary(full.lathe1)
```

```
##
## Call:
## lm(formula = log(Life) ~ Speed + Feed + I(Speed^2) + I(Feed^2) +
##     Speed:Feed, data = noInfluence)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39963 -0.14660  0.00387  0.14917  0.32783
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.18809    0.08241  14.417 6.11e-09 ***
## Speed        -1.43300    0.08241 -17.388 7.10e-10 ***
## Feed         -0.79023    0.06729 -11.743 6.15e-08 ***
## I(Speed^2)    0.28022    0.12363   2.267 0.042700 *
## I(Feed^2)     0.42244    0.09217   4.583 0.000629 ***
## Speed:Feed   -0.07286    0.08241  -0.884 0.394025
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.2331 on 12 degrees of freedom
## Multiple R-squared:  0.9759, Adjusted R-squared:  0.9658
## F-statistic: 97.07 on 5 and 12 DF,  p-value: 2.804e-09
```

```r
avPlots(full.lathe1)
```



The two most "influential in the fit of the quadratic mean function function for log(Life)" cases are 9 and 10. These cases are influential because they are outliers, i.e. do not follow the general trend among most of the points. After removing the influential cases, there seems to be not much of a difference.