HW11

Dandong Tu

2017/11/29

8.2

8.2.1

```
library(alr4)
```

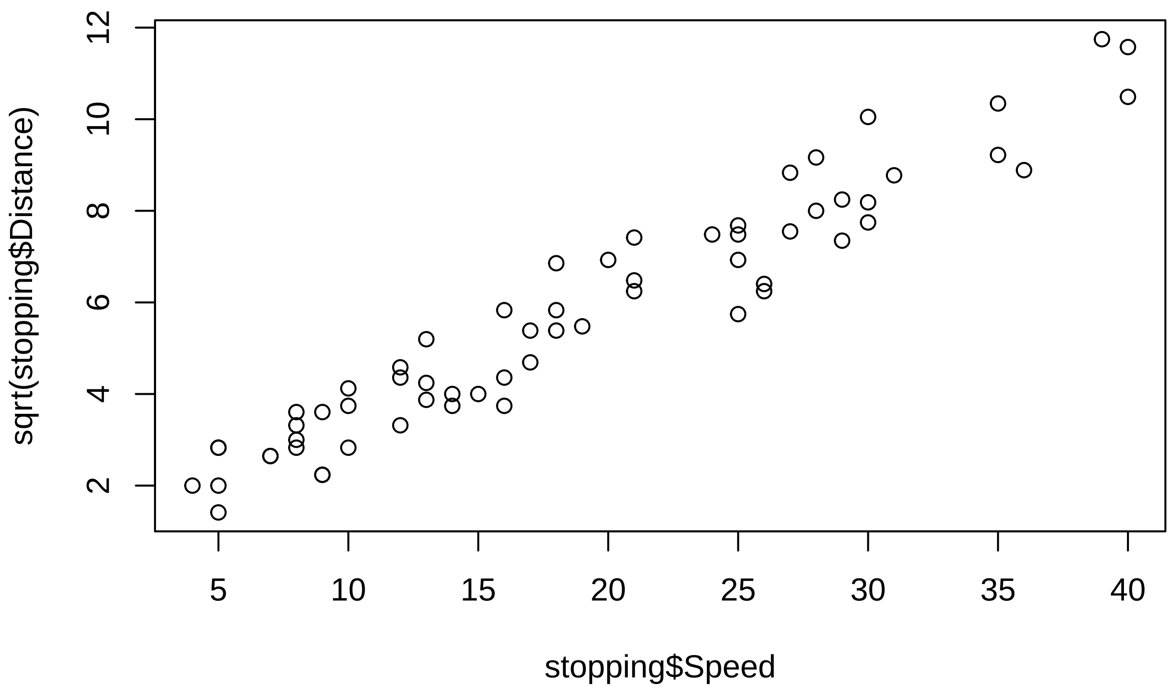
```
## Loading required package: car

## Loading required package: effects

##
## Attaching package: 'effects'

## The following object is masked from 'package:car':
##
##     Prestige
```

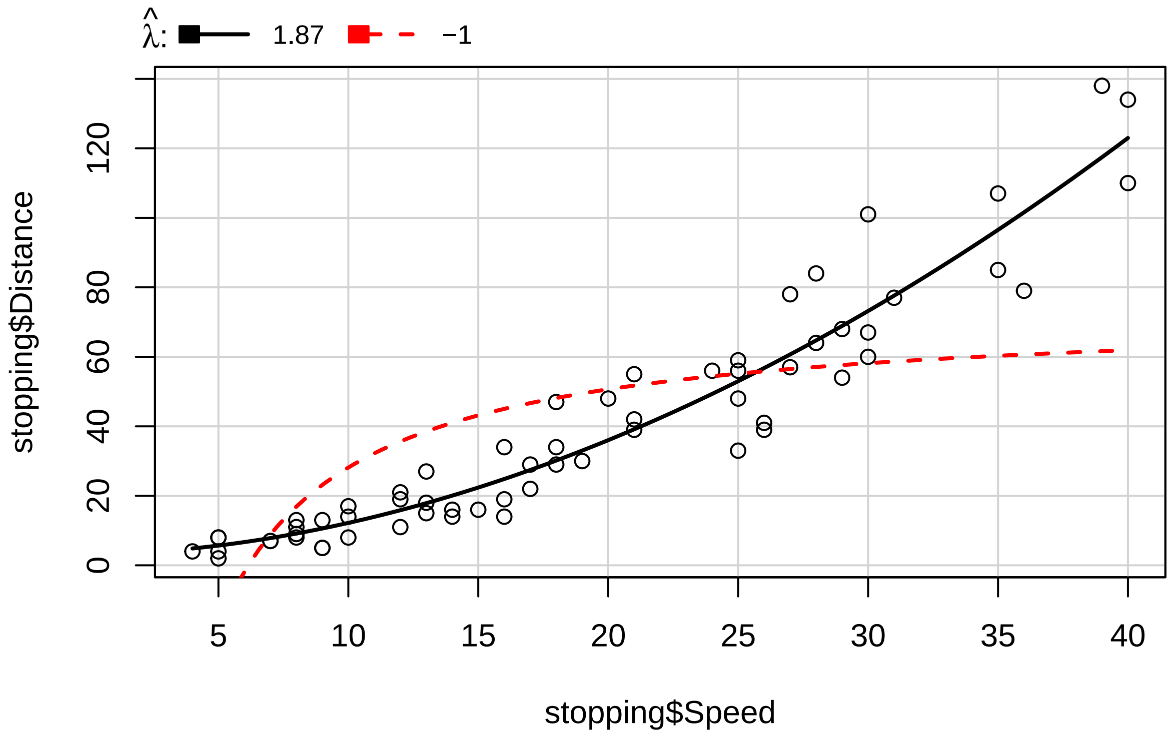
```
plot(sqrt(stopping$Distance)~stopping$Speed)
```



From the plot we see that the sqrt seems a appropriate transformation for **Distance** that can linearize this regression.

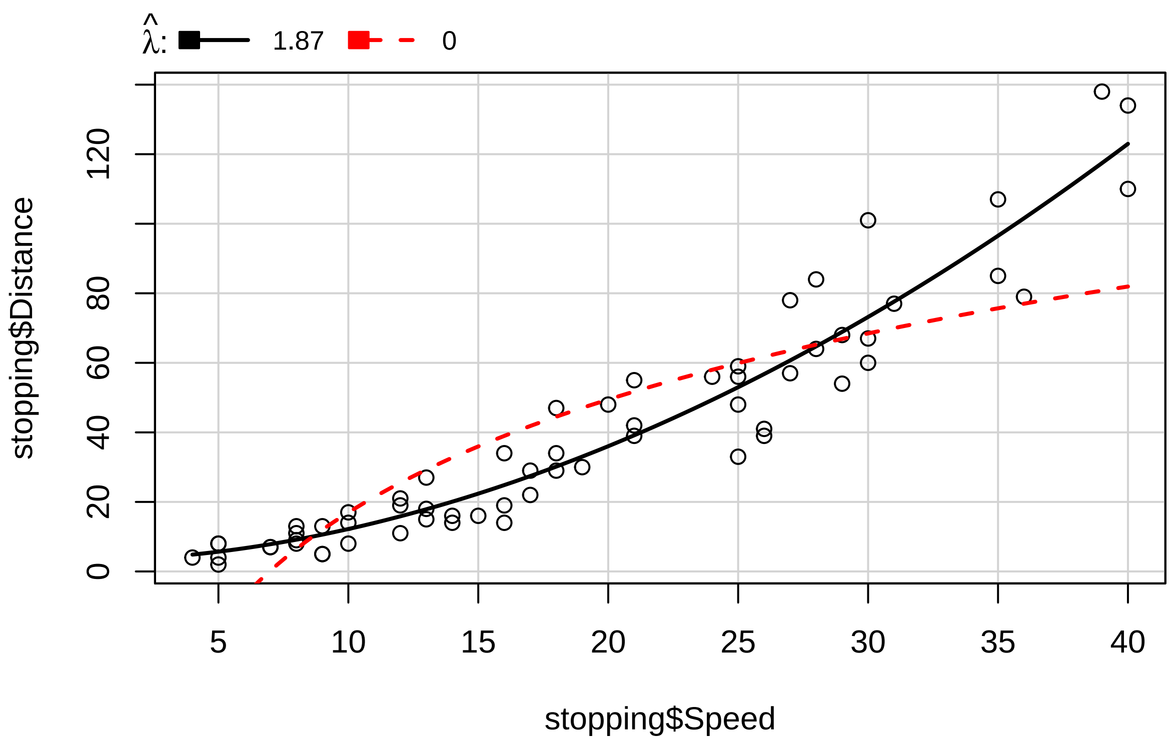
8.2.2

```
invTranPlot(stopping$Speed,stopping$Distance,-1)
```




```
##      lambda       RSS
## 1  1.868443  5823.372
## 2 -1.000000 34951.108
```

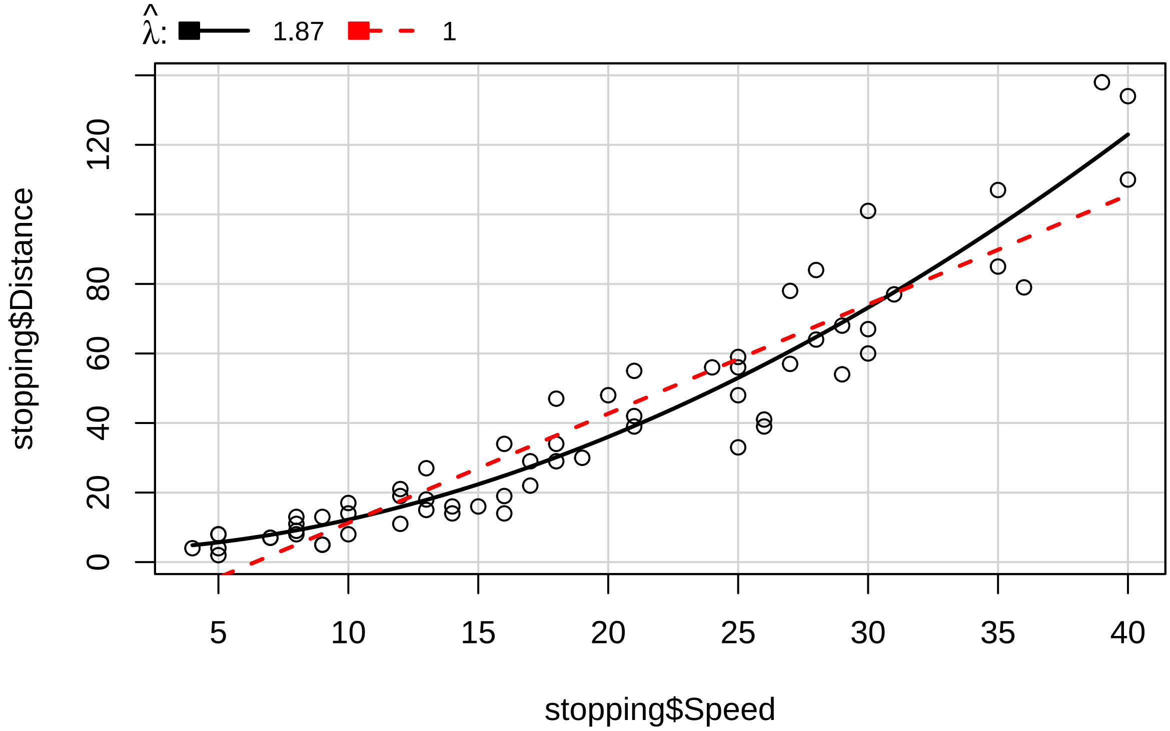
```r
invTranPlot(stopping$Speed,stopping$Distance,0)
```




```
##     lambda      RSS
## 1 1.868443 5823.372
```

```
## 2 0.000000 18844.172
```

```r
invTranPlot(stopping$Speed,stopping$Distance,1)
```

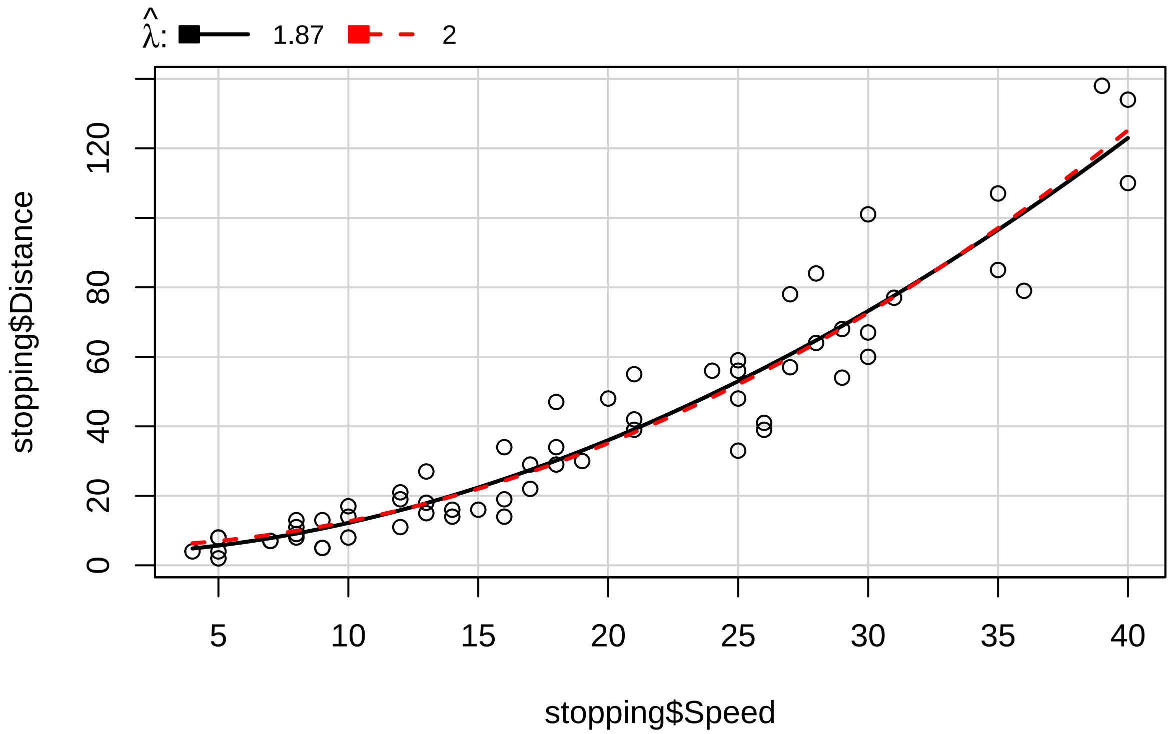



```
##      lambda      RSS
## 1 1.868443 5823.372
## 2 1.000000 8310.166
```

From the plots, it seems none of the value (-1,0,1) are adequate.

8.2.3

```r
invTranPlot(stopping$Speed,stopping$Distance,2)
```

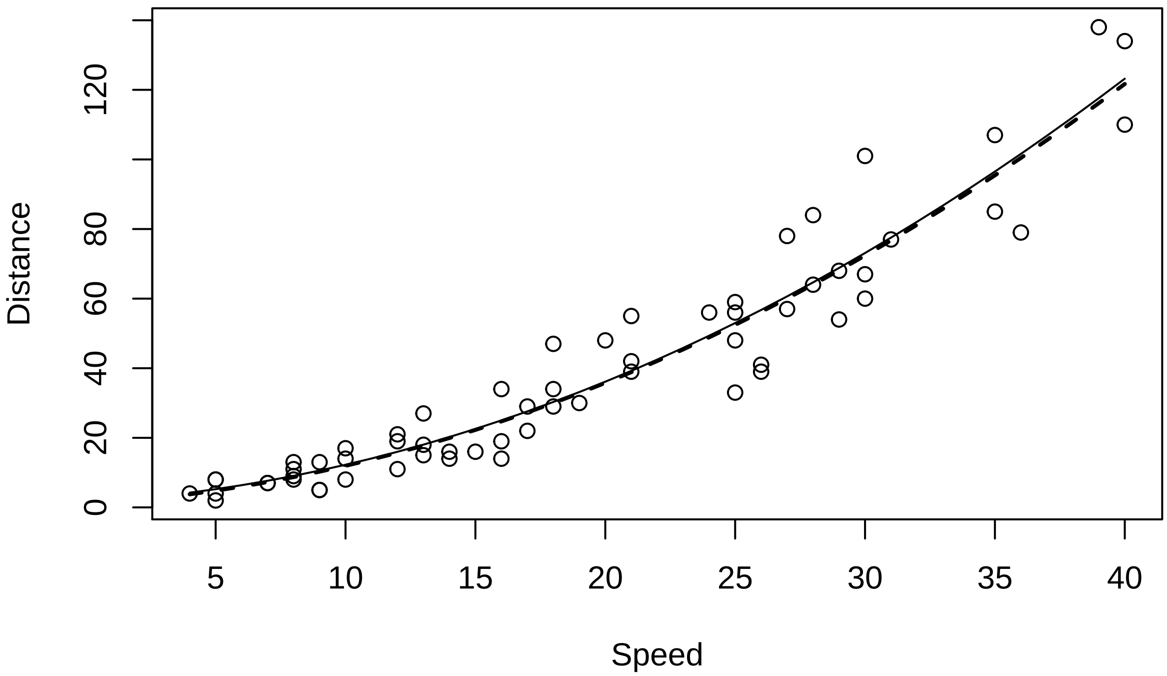



```
##      lambda      RSS
## 1 1.868443 5823.372
## 2 2.000000 5869.232
```

In the graph, the lines fit well. It seems that using lambda = 2 to transform the predictor **Speed**in problem 8.2.2 does match the data well.

8.8.4

```
w=(1/stopping$Speed)^2
m1=lm(Distance~Speed+I(Speed^2),data=stopping,weights = w)
plot(Distance~Speed,data = stopping)
lines(4:40,predict(m1,data.frame(Speed=4:40)),lty=1,lwd=1)
m11=lm(sqrt(Distance)~Speed,data = stopping)
lines(4:40,predict(m11,data.frame(Speed=4:40))^2,lty=2,lwd=2)
```

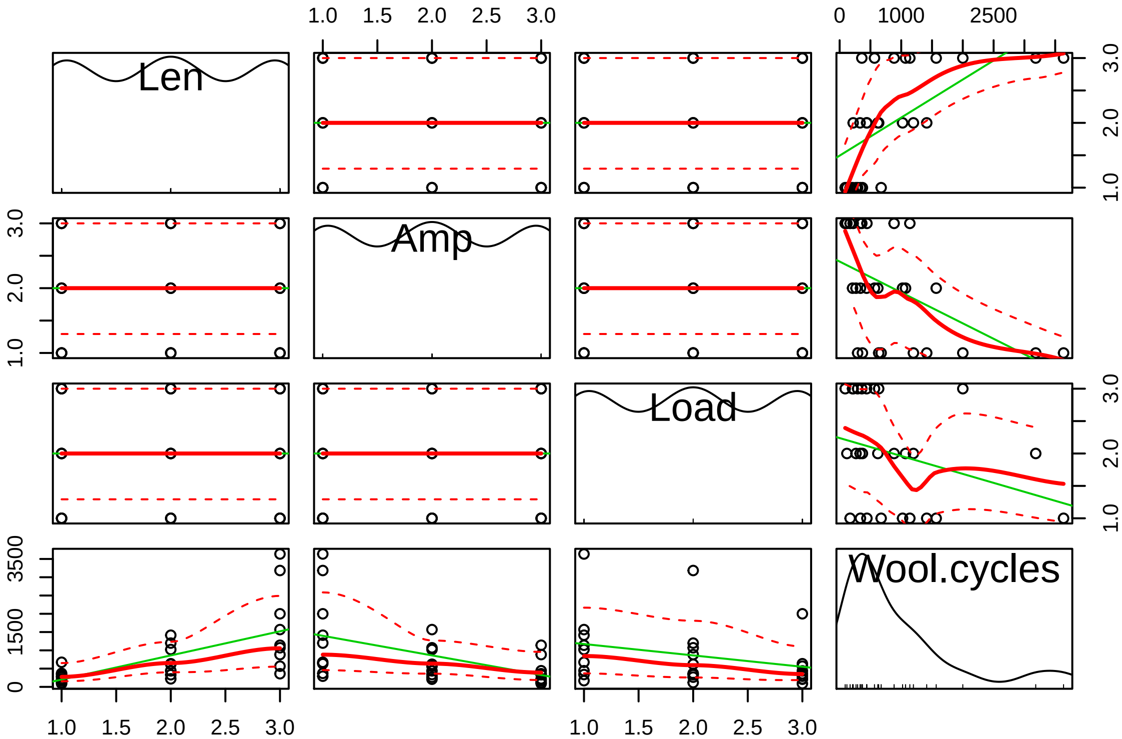


From the result, we observed that two lines are very closed and seems like lapped. The reason behind this is that take the sqrt for the **Distance** is same as take **I(Speed^2)**

8.6

8.6.1

```
Len=factor(Wool$len,ordered=FALSE)
Amp=factor(Wool$amp,ordered = FALSE)
Load=factor(Wool$load,ordered=FALSE)
Wool2=data.frame(Len,Amp,Load,Wool$cycles)
scatterplotMatrix(~Len+Amp+Load+Wool.cycles,data = Wool2)
```



The scatterplot matrix shows the relations between **Len**,**Amp**,**Load** and **cycles**

8.6.2

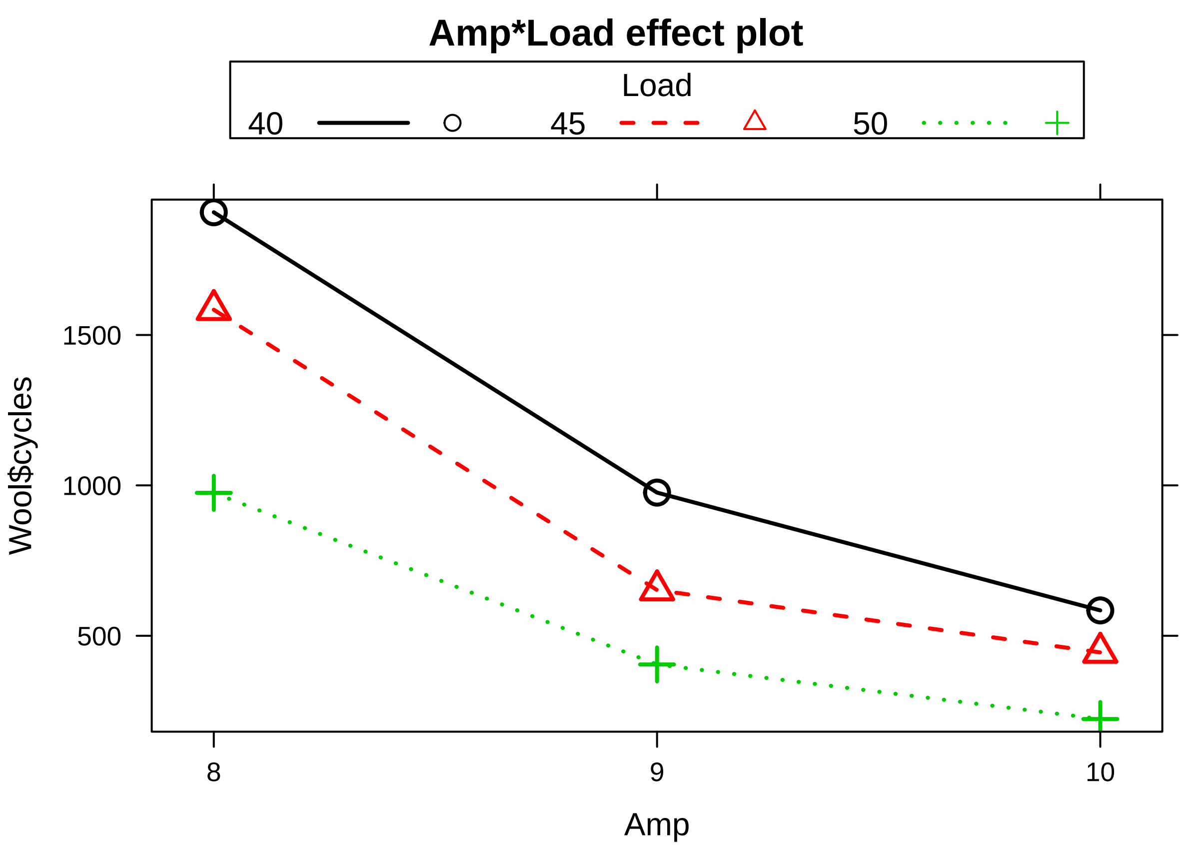
```
m2=lm(Wool$cycles~Len+Amp+Load+Len:Load+Len:Amp+Amp:Load)
summary(m2)
```

```
##
## Call:
## lm(formula = Wool$cycles ~ Len + Amp + Load + Len:Load + Len:Amp +
##     Amp:Load)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -127.593  -39.148   -9.037   58.074  117.074
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.826e+02  9.237e+01   7.390 7.69e-05 ***
## Len300           7.809e+02  1.161e+02   6.728 0.000148 ***
## Len350           2.895e+03  1.161e+02  24.946 7.13e-09 ***
## Amp9            -2.944e+02  1.161e+02  -2.537 0.034879 *
## Amp10           -5.713e+02  1.161e+02  -4.923 0.001160 **
## Load45          -2.041e+02  1.161e+02  -1.759 0.116697
## Load50          -5.077e+02  1.161e+02  -4.374 0.002368 **
## Len300:Load45 -1.003e+02  1.271e+02  -0.789 0.452782
## Len350:Load45 -2.593e+02  1.271e+02  -2.040 0.075709 .
## Len300:Load50 -3.323e+02  1.271e+02  -2.614 0.030944 *
## Len350:Load50 -9.427e+02  1.271e+02  -7.414 7.52e-05 ***
```

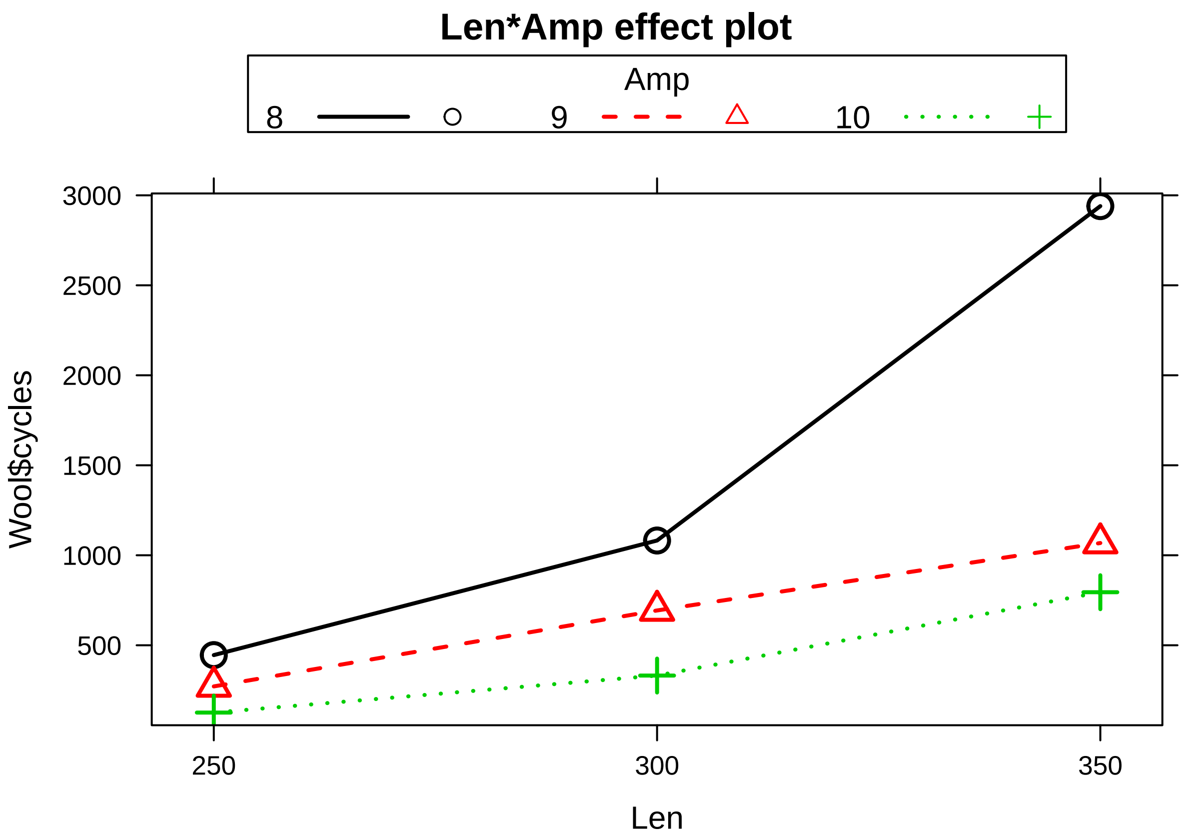
```
## Len300:Amp9   -2.147e+02  1.271e+02  -1.688 0.129813
## Len350:Amp9   -1.698e+03  1.271e+02 -13.355 9.45e-07 ***
## Len300:Amp10  -4.310e+02  1.271e+02  -3.390 0.009502 **
## Len350:Amp10  -1.826e+03  1.271e+02 -14.362 5.40e-07 ***
## Amp9:Load45   -1.600e-13  1.271e+02   0.000 1.000000
## Amp10:Load45   1.843e+02  1.271e+02   1.450 0.185155
## Amp9:Load50    3.613e+02  1.271e+02   2.842 0.021747 *
## Amp10:Load50   5.717e+02  1.271e+02   4.496 0.002012 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 110.1 on 8 degrees of freedom
## Multiple R-squared:  0.9952, Adjusted R-squared:  0.9844
## F-statistic: 92.25 on 18 and 8 DF,  p-value: 2.537e-07
```

The R^2 is 0.9952 indicates the models explain 99.52% variability of the response data. Several low p-vale indicates the regressors are statisticly significant, and the above model describes the data well.

```
print(plot(effect("Amp:Load",m2,),multiline = TRUE))
```




```
print(plot(effect("Len:Amp",m2,),multiline = TRUE))
```



The effect plot shows the effect of **AMP** in mean response in different **Len** values.

8.6.3

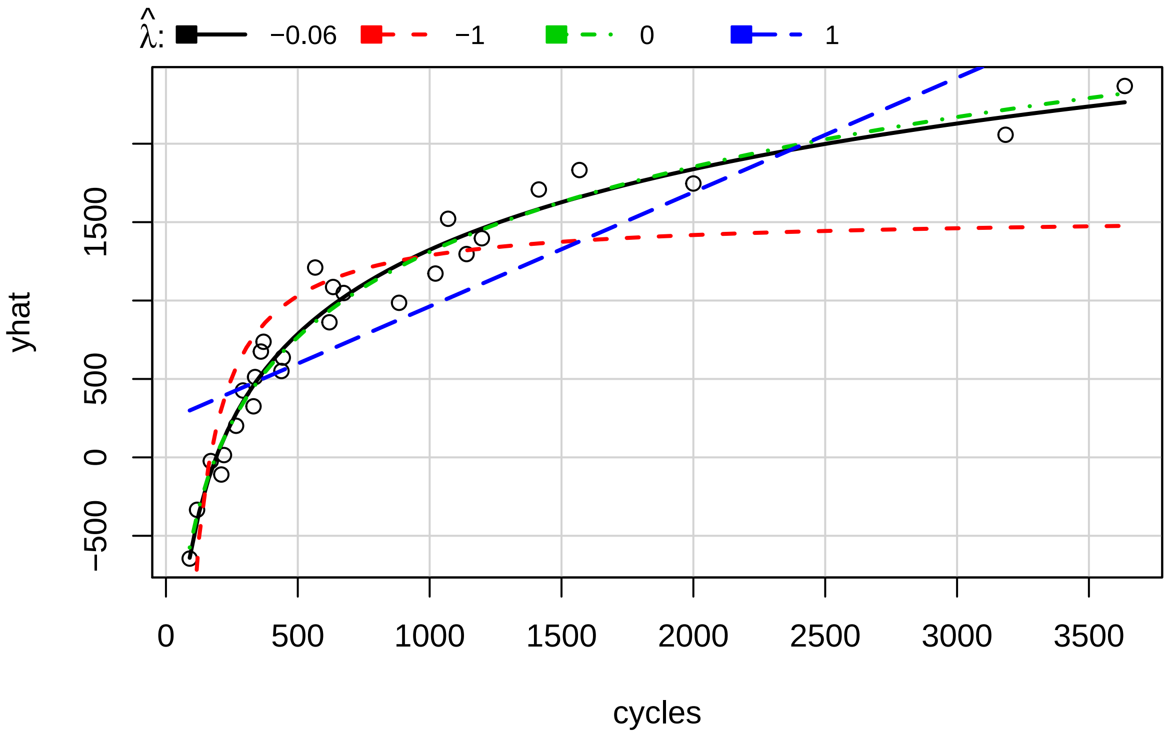
```
m3=lm(Wool$cycles~Len+Amp+Load)
summary(m3)
```

```
##
## Call:
## lm(formula = Wool$cycles ~ Len + Amp + Load)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -570.81 -308.43  -53.81  227.57 1112.63
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1203.4      246.0   4.891 8.83e-05 ***
## Len300         421.4      227.8   1.850 0.079096 .
## Len350        1320.0      227.8   5.795 1.14e-05 ***
## Amp9          -811.6      227.8  -3.563 0.001948 **
## Amp10        -1071.7      227.8  -4.705 0.000136 ***
## Load45        -262.6      227.8  -1.153 0.262611
## Load50        -621.7      227.8  -2.729 0.012918 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 483.2 on 20 degrees of freedom
## Multiple R-squared:  0.7692, Adjusted R-squared:  0.6999
## F-statistic: 11.11 on 6 and 20 DF,  p-value: 1.769e-05
```

The summary shows that the model is not adequate for these data.

```r
m4=lm(cycles~len+amp+load,data=Wool)
(inverseResponsePlot(m4))
```




```
##        lambda       RSS
## 1 -0.06052267  503066.0
## 2 -1.00000000 3457492.6
## 3  0.00000000  518854.6
## 4  1.00000000 3995721.6
```

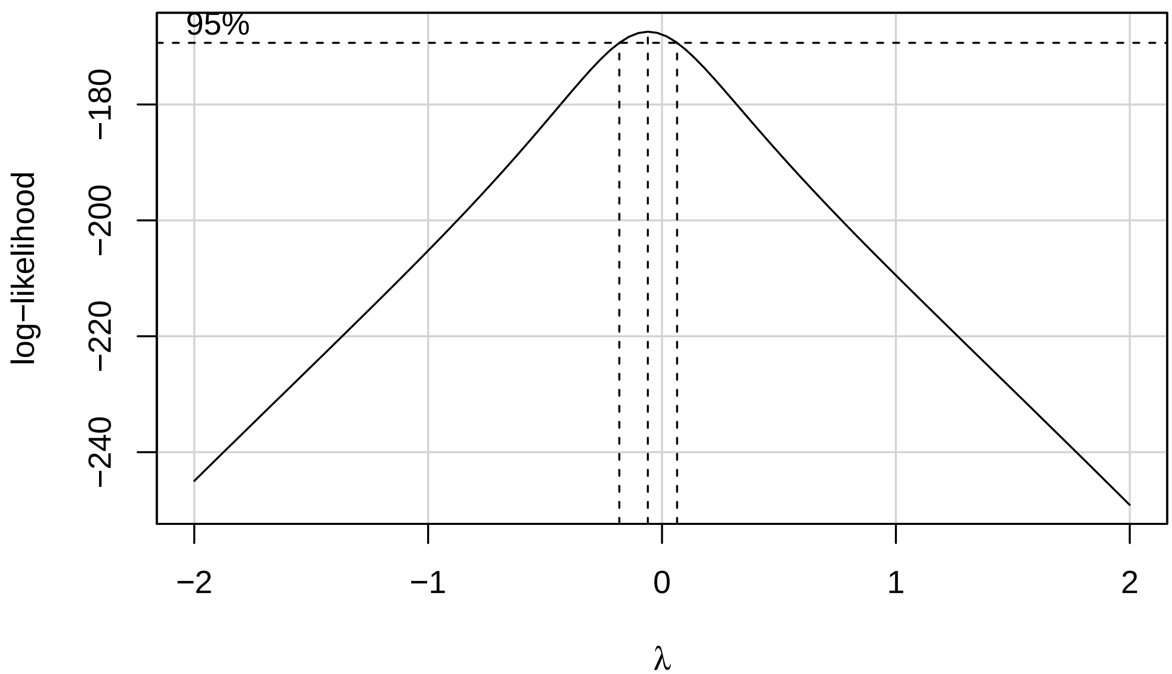
The inverseResponsePlot shows -0.06 is the best fit lambda, the value is pretty closed to 0.

```r
summary(powerTransform(m4))
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr bnd Wald Upr Bnd
## Y1   -0.0592           0      -0.1789       0.0606
##
## Likelihood ratio tests about transformation parameters
##                                LRT df      pval
## LR test, lambda = (0)  0.9213384  1 0.3371238
## LR test, lambda = (1) 84.0756559  1 0.0000000
```

The Lwr bnd to Upr Bnd is incloude value of 0. The powerTransform summary shows that the p-value for lambda=0 is large, so that we are unable to reject the null(lambda=0 which means use log transformation).

```r
boxCox(m4)
```



The Box Cox graph shows that based on a 95% confidence intervial, the lambda value of 0 is inculded. It suggests that use **Log** transformation is appropriate.

8.6.4

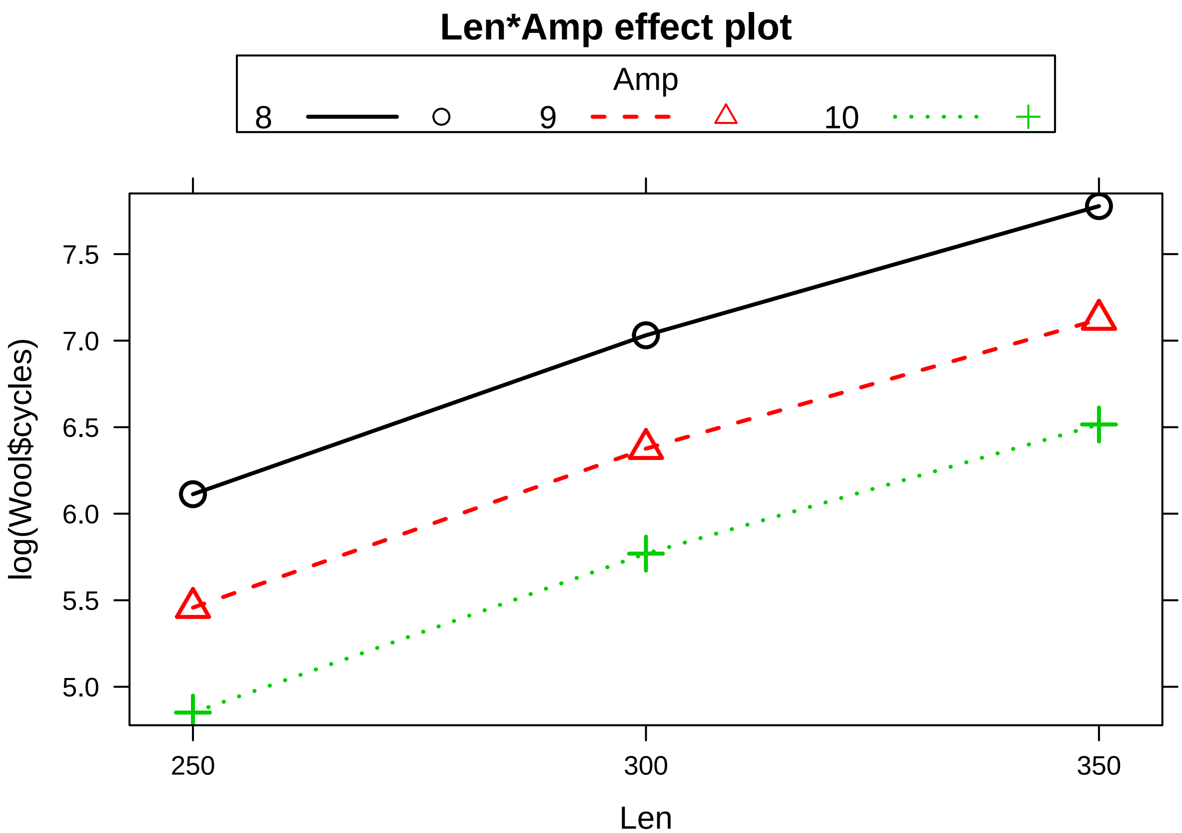
```
m5=lm(log(Wool$cycles)~Len+Amp+Load+Len:Amp+Len:Load+Amp:Load)
m6=lm(log(Wool$cycles)~Len+Amp+Load)
anova(m6,m5)
```

```
## Analysis of Variance Table
##
## Model 1: log(Wool$cycles) ~ Len + Amp + Load
## Model 2: log(Wool$cycles) ~ Len + Amp + Load + Len:Amp + Len:Load + Amp:Load
##   Res.Df     RSS Df Sum of Sq     F Pr(>F)
## 1     20 0.71742
## 2      8 0.16591 12   0.55151 2.216 0.1325
```

From the result of F-test, p-value is **0.1325** we do not have enough evidence to reject the **H0**, so we use the model with interation **m6**

```
print(plot(effect("Len:Amp",m6,),multiline = TRUE))
```

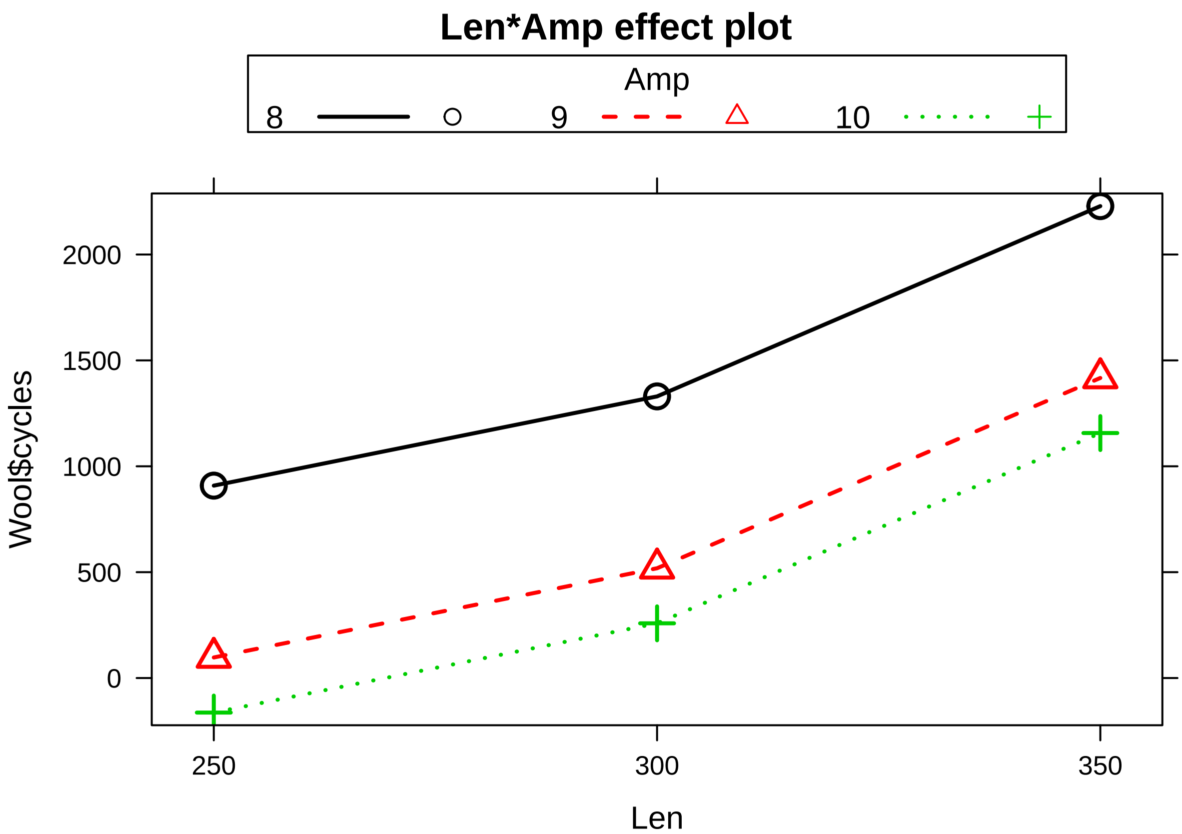
```
## NOTE: Len:Amp does not appear in the model
```



We observed three parallel lines.

```
m7=lm(Wool$cycles~Len+Amp+Load)
print(plot(effect("Len:Amp",m7,),multiline = TRUE))
```

```
## NOTE: Len:Amp does not appear in the model
```



Now comparing the result with 8.6.2, we have the similar graphs.