hw4

by Jonathan Franco

Reading in the data

creating new terms as well

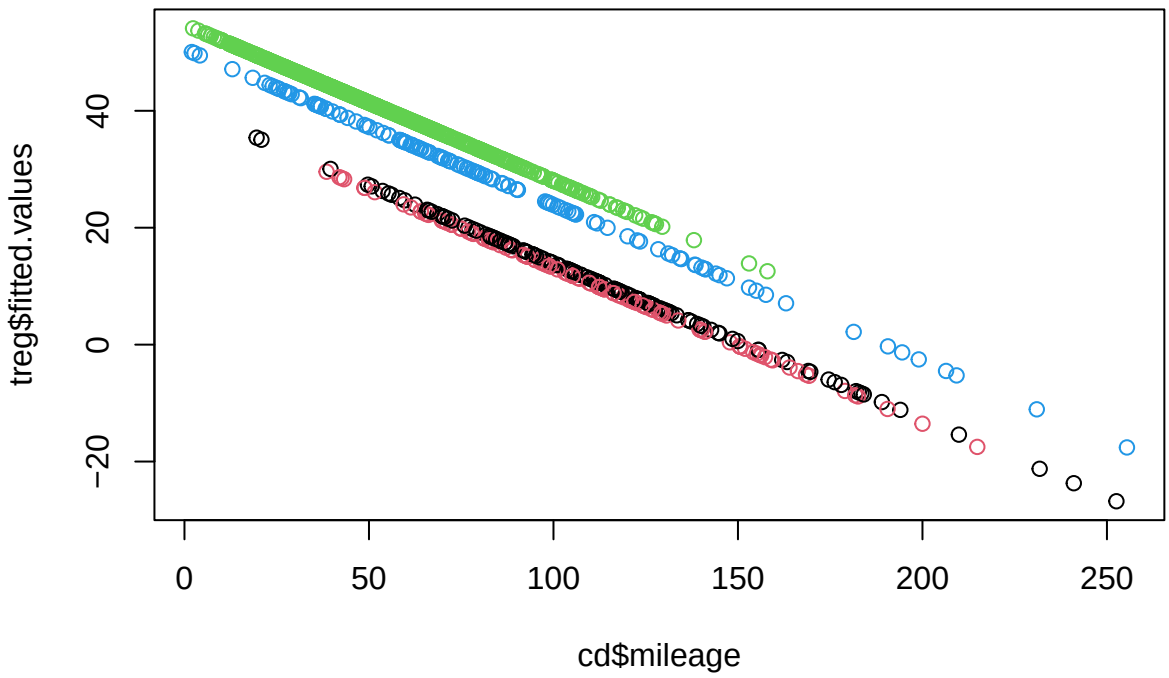
```
cd = read.csv("http://www.rob-mcculloch.org/data/susedcars.csv")
cd$price = cd$price/1000
cd$mileage = cd$mileage/1000
iifac = c(2,3,6,7)
for(i in iifac) cd[,i] = as.factor(cd[,i])
cd$mileagesq = cd$mileage^2
cd$mileagecb = cd$mileage^3
cd$mileagefr = cd$mileage^4
cd$yearsq = cd$year^2
cd$yearcb = cd$year^3
cd$yearfr = cd$year^4
cd$milexyear = cd$year * cd$mileage
head(cd)
```

```
##    price trim isOneOwner mileage year  color displacement mileagesq  mileagecb
## 1 43.995  550          f  36.858 2008 Silver          5.5  1358.512   50072.04
## 2 44.995  550          f  46.883 2012  Black          4.6  2198.016  103049.57
## 3 25.999  550          f 108.759 2007  White          5.5 11828.520 1286458.02
## 4 33.880  550          f  35.187 2007  Black          5.5  1238.125   43565.90
## 5 34.895  550          f  48.153 2007  Black          5.5  2318.711  111652.91
## 6  5.995  500          f 121.748 2002  other        other 14822.576 1804618.92
##   mileagefr  yearsq     yearcb       yearfr milexyear
## 1   1845555 4032064 8096384512 1.625754e+13  74010.86
## 2   4831273 4048144 8144865728 1.638747e+13  94328.60
## 3 139913887 4028049 8084294343 1.622518e+13 218279.31
## 4   1532953 4028049 8084294343 1.622518e+13  70620.31
## 5   5376423 4028049 8084294343 1.622518e+13  96643.07
## 6 219708745 4008004 8024024008 1.606410e+13 243739.50
```

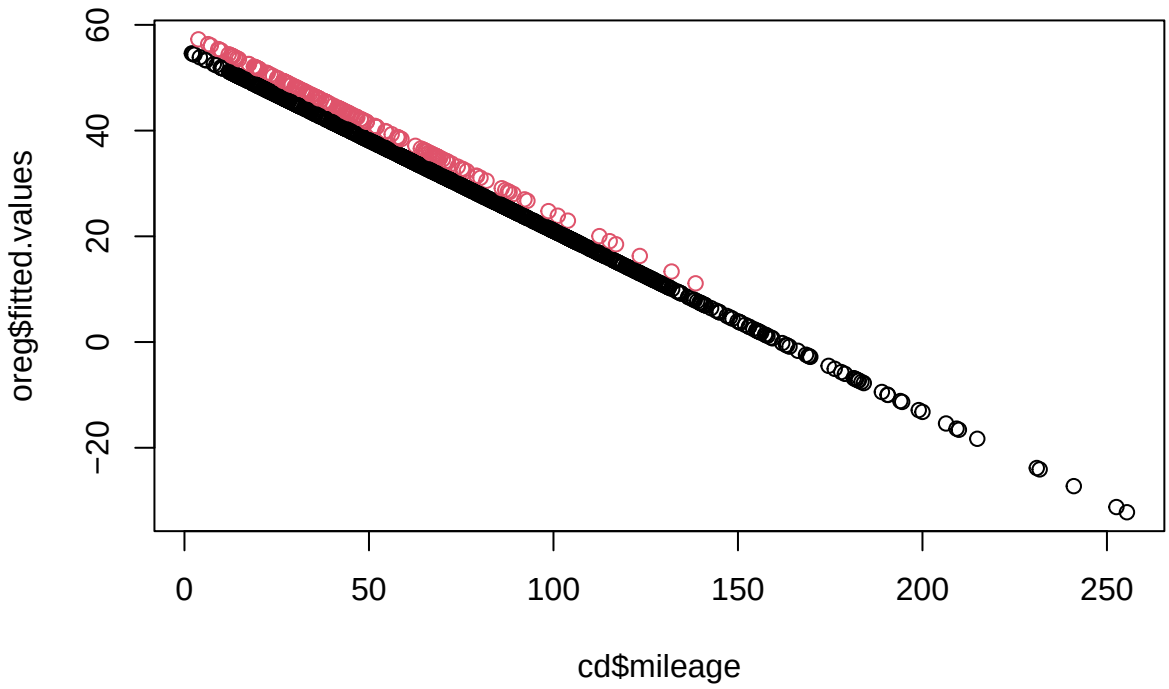
Testing categorical values

This is to see how much they could matter

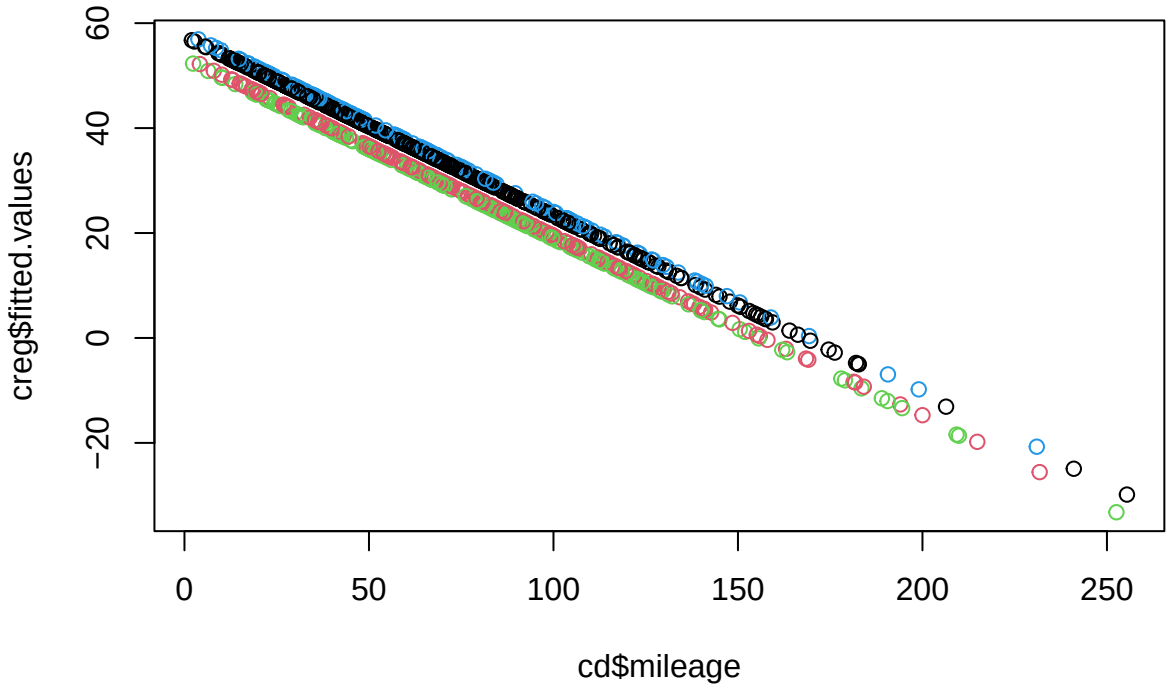
```
treg = lm(price~mileage+trim, cd)
oreg = lm(price~mileage+isOneOwner, cd)
creg = lm(price~mileage+color, cd)
dreg = lm(price~mileage+displacement, cd)
plot(cd$mileage, treg$fitted.values,col= as.integer(cd$trim))
```



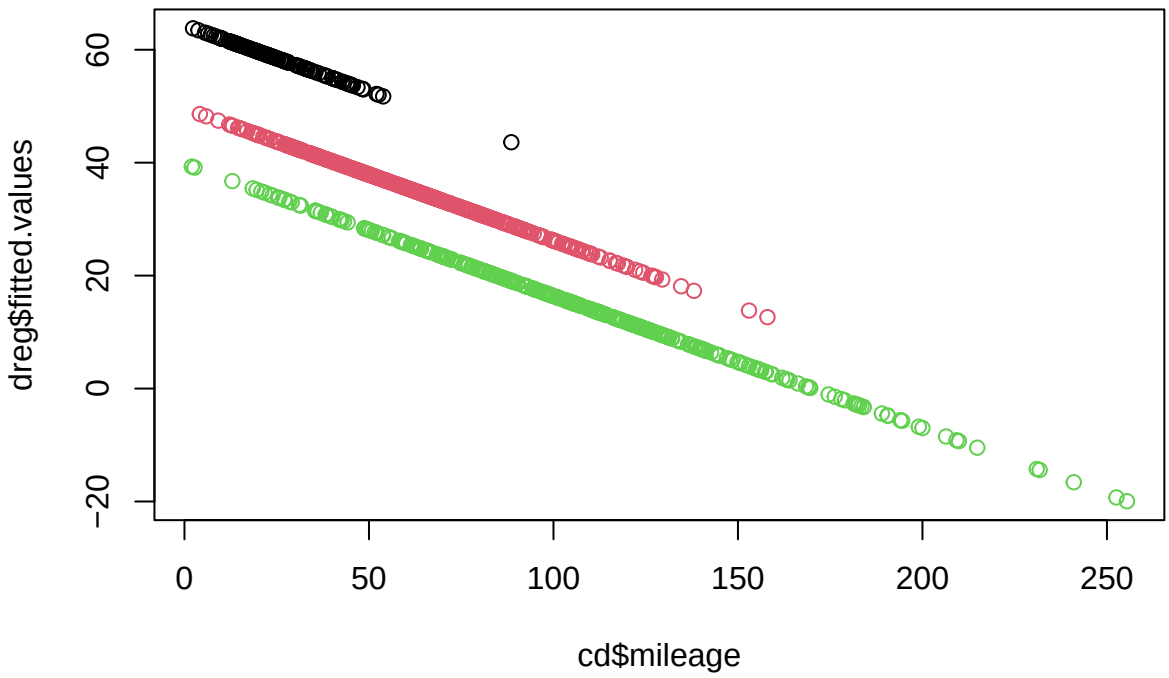

```
plot(cd$mileage, oreg$fitted.values,col= as.integer(cd$isOneOwner))
```




```
plot(cd$mileage, creg$fitted.values,col= as.integer(cd$color))
```




```
plot(cd$mileage, dreg$fitted.values,col= as.integer(cd$displacement))
```



train/test split

```
n = nrow(cd)
pin = .75
ii = sample(1:n, floor(pin*n))
cdtr = cd[ii,]
cdte = cd[-ii,]
```

making the model

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 4.1.3
```

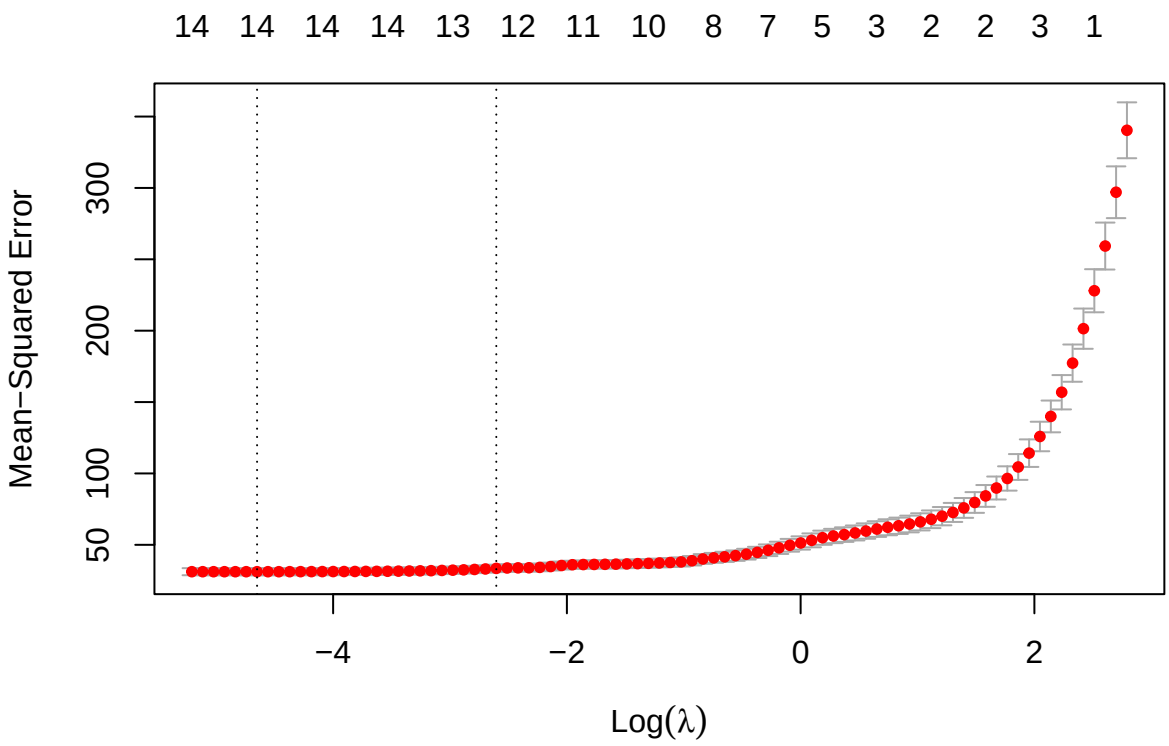
```
y = cdtr$price
x = model.matrix(price~.,cdtr)[,-1]
xtest = model.matrix(price~.,cdte)[,-1]
head(x)
```

```
##     trim500 trim550 trimother isOneOwnert mileage year colorother colorSilver
```

```
## 995       0       1         0           0 102.656 2007          0           0
## 826       0       1         0           1  26.807 2013          0           0
## 667       0       1         0           0  34.429 2010          0           0
## 41        0       1         0           0  23.143 2008          1           0
## 800       0       0         1           0  99.179 2004          0           1
## 234       0       1         0           0  73.765 2007          0           0
##     colorWhite displacement5.5 displacementother  mileagesq  mileagecb
## 995          0               1                 0 10538.2543 1081815.04
## 826          1               0                 0   718.6152   19263.92
## 667          1               1                 0  1185.3560   40810.62
## 41           0               1                 0   535.5984   12395.35
## 800          0               0                 1  9836.4740  975571.66
## 234          1               1                 0  5441.2752  401375.67
##       mileagefr  yearsq     yearcb       yearfr milexyear
## 995 111054804.5 4028049 8084294343 1.622518e+13 206030.59
## 826    516407.9 4052169 8157016197 1.642007e+13  53962.49
## 667   1405068.9 4040100 8120601000 1.632241e+13  69202.29
## 41     286865.7 4032064 8096384512 1.625754e+13  46471.14
## 800  96756221.6 4016016 8048096064 1.612838e+13 198754.72
## 234  29607476.1 4028049 8084294343 1.622518e+13 148046.36
```

Running Lasso

```
set.seed(14)
cars.gcv = cv.glmnet(x, y, type.measure = "mse", nfolds = 10, alpha = 1)
plot(cars.gcv)
```

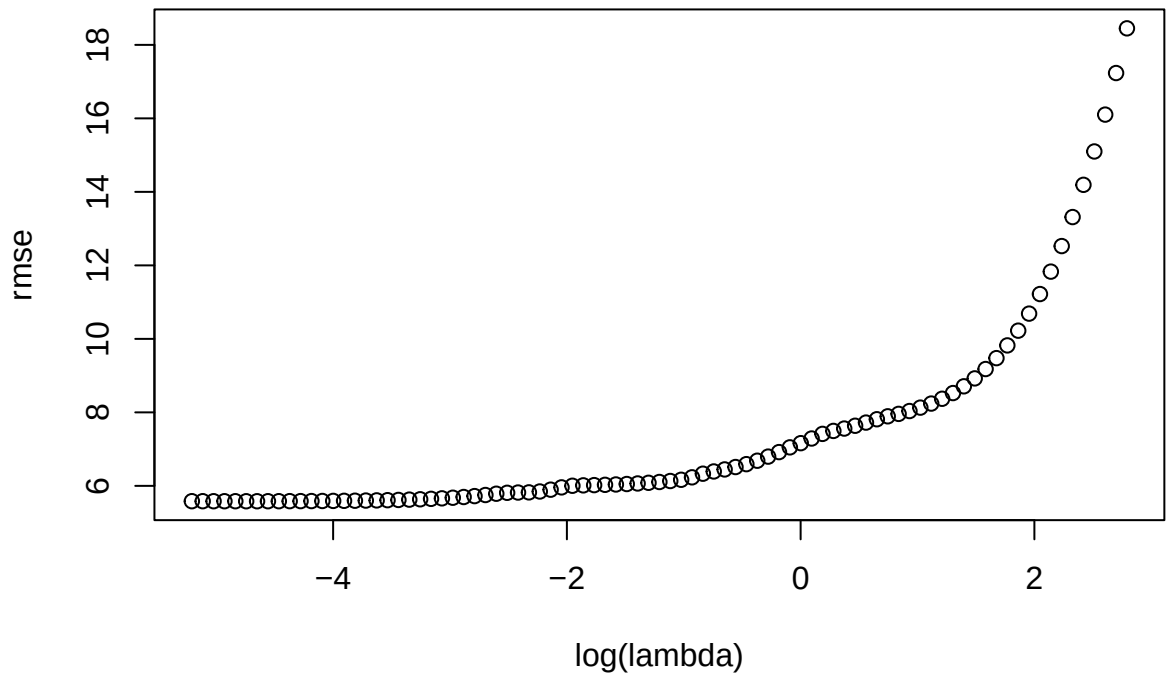


Plotting the RMSE

also showing the difference between the lmin and l1se

```
## lambda min:  0.009554867
```

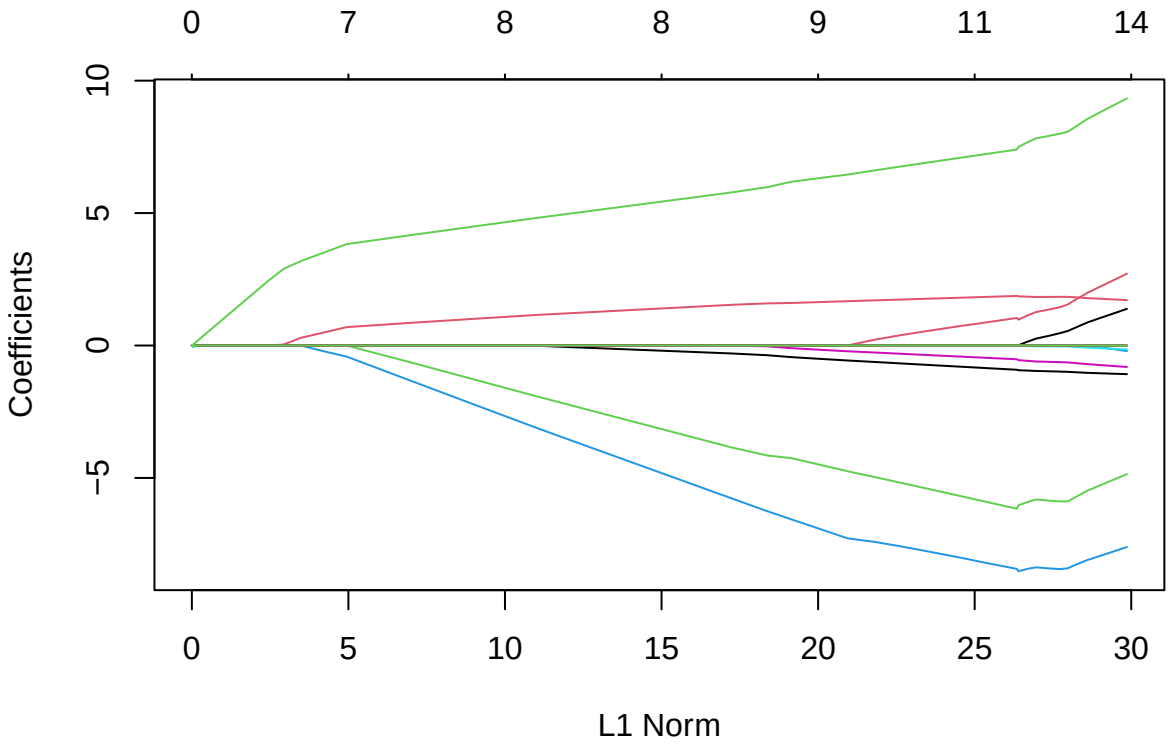
```
## lambda 1se:  0.07397987
```



Getting the predictions on the test values

using the min rmse from the plot

```
## min rmse:  5.581913
```



Finding the Out of Sample Rmse

```
## out of sample rmse:  6.042886
```