Brian Holliday

Professor Li

Intro to Data Mining

13 May 2020

Project 6 Extra Project

1. Fit the given data for model

$$X5 = \beta 1X1 + \beta 2X2 + \beta 4X4$$

Figure 1: Variation Multilinear Model Summary

```
Call:
lm(formula = y ~ . - 1, data = data.frame(A, y))

Residuals:
   Min     1Q Median     3Q    Max 
-7.588 -3.825 -1.681  1.629 40.381 

Coefficients:
   Estimate Std. Error t value Pr(>|t|)    
x1   0.7280     0.2828   2.575   0.0119 *  
x2   4.8718     0.9146   5.327 9.61e-07 ***
x4  -2.0522     0.3242  -6.329 1.50e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.036 on 77 degrees of freedom
Multiple R-squared:  0.4372,	Adjusted R-squared:  0.4152 
F-statistic: 19.94 on 3 and 77 DF,  p-value: 1.165e-09
```

We can see that for our model, we do have a model. All our coefficients are statistically significant. have an R^2 value of about 41 percent, meaning that we can account for 41 percent of the variation of the data. Since the multilinear model only accounts for 41 percent in the variation of the data, we will try the exponential model. With our fit our equation is:

$$X5 = 0.7280*x1 + 4.8718*x2+ -2.0522*x4$$

2.

Fit the given data for model

$X5 = \beta 1X1e\text{^}(\beta 2X2+\beta 4X4)$

For this regression with are going to use the exponential model

We will linearize it to run a linear regression.

x5 = B1x1e^(b2x2 + b4x4)

ln(x5/x1) = ln(b1) + b2x2 + b4x4

Figure 2: Exp Model Code

```
> x5 <- dat$x5
> x1 <- dat$x1
> x4 <- dat$x4
> x2 <- dat$x2
>
> exp_model <- lm(log(x5/x1) ~., data = data.frame(x2,x4) )
> summary(exp_model)
> x5 <- dat$x5
> x1 <- dat$x1
> x4 <- dat$x4
> x2 <- dat$x2
>
> exp_model <- lm(log(x5/x1) ~., data = data.frame(x2,x4) )
> #summary(exp_model)
>
> exp_pred <- predict(exp_model, newdata = data.frame(x2,x4,
+                                                       log(x5/x1)))
>
> RMSE <- sqrt(sum((exp_pred)^2))/sqrt(length(x5))
> RMSE
[1] 3.727413
```

After linearizing the equation, we can find the coefficient of x2, x4 and the intercept will be the ln(B1) coefficient.

Figure 3: Exp Summary

```
Call:
lm(formula = log(x5/x1) ~ ., data = data.frame(x2, x4))

Residuals:
     Min       1Q   Median       3Q      Max 
-0.40324 -0.01565  0.00115  0.02184  0.32021 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.658390   0.047801   13.77   <2e-16 ***
x2           0.994561   0.019761   50.33   <2e-16 ***
x4          -0.988633   0.004718 -209.55   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1015 on 77 degrees of freedom
Multiple R-squared:  0.9983,	Adjusted R-squared:  0.9982 
F-statistic: 2.244e+04 on 2 and 77 DF,  p-value: < 2.2e-16
```

We can see that for our model, we do have a model. All our coefficients are statistically significant. have an R^2 value of about 99 percent, meaning that we can account for 99 percent of the variation of the data.

X5 = e^ (0.658390)x1e^(0.994561x2 + -0.988633x4)

3.

Which model is better?

Although the multilinear model has a lower RMSE, the exponential model is the better model in this case. The exponential model is a near perfect fit with the R^2 score at 99 percent, therefore it can account for more of the variation in the data.