linear\_regression.R

stevehaunguyen

Sun Jun 10 05:48:32 2018

# Introduction  
##  
  
# Learning objectives:  
## Learn the R formula interface  
## Specify factor contrasts to test specific hypotheses  
## Perform model comparisons  
## Run and interpret variety of regression models in R  
  
## Set working directory  
##  
  
## It is often helpful to start your R session by setting your working  
## directory so you don't have to type the full path names to your data  
## and other files  
  
# set the working directory  
# setwd("~/Desktop/Rstatistics")  
# setwd("C:/Users/dataclass/Desktop/Rstatistics")  
  
## You might also start by listing the files in your working directory  
setwd("~/Springboard Projects/Chapter 7 - Machine Learning/linear\_regression/linear\_regression")  
getwd() # where am I?

## [1] "C:/Users/stevehaunguyen/Documents/Springboard Projects/Chapter 7 - Machine Learning/linear\_regression/linear\_regression"

list.files("dataSets") # files in the dataSets folder

## [1] "Exam.rds" "states.dta" "states.rds"

## Load the states data  
##  
  
# read the states data  
states.data <- readRDS("dataSets/states.rds")   
#get labels  
states.info <- data.frame(attributes(states.data)[c("names", "var.labels")])  
#look at last few labels  
tail(states.info, 8)

## names var.labels  
## 14 csat Mean composite SAT score  
## 15 vsat Mean verbal SAT score  
## 16 msat Mean math SAT score  
## 17 percent % HS graduates taking SAT  
## 18 expense Per pupil expenditures prim&sec  
## 19 income Median household income, $1,000  
## 20 high % adults HS diploma  
## 21 college % adults college degree

## Linear regression  
##  
  
## Examine the data before fitting models  
##  
  
## Start by examining the data to check for problems.  
  
# summary of expense and csat columns, all rows  
sts.ex.sat <- subset(states.data, select = c("expense", "csat"))  
summary(sts.ex.sat)

## expense csat   
## Min. :2960 Min. : 832.0   
## 1st Qu.:4352 1st Qu.: 888.0   
## Median :5000 Median : 926.0   
## Mean :5236 Mean : 944.1   
## 3rd Qu.:5794 3rd Qu.: 997.0   
## Max. :9259 Max. :1093.0

# correlation between expense and csat  
cor(sts.ex.sat)

## expense csat  
## expense 1.0000000 -0.4662978  
## csat -0.4662978 1.0000000

## Plot the data before fitting models  
##  
  
## Plot the data to look for multivariate outliers, non-linear  
## relationships etc.  
  
# scatter plot of expense vs csat  
plot(sts.ex.sat)



## Linear regression example  
##  
  
## Linear regression models can be fit with the `lm()' function  
## For example, we can use `lm' to predict SAT scores based on  
## per-pupal expenditures:  
  
# Fit our regression model  
sat.mod <- lm(csat ~ expense, # regression formula  
 data=states.data) # data set  
# Summarize and print the results  
summary(sat.mod) # show regression coefficients table

##   
## Call:  
## lm(formula = csat ~ expense, data = states.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -131.811 -38.085 5.607 37.852 136.495   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.061e+03 3.270e+01 32.44 < 2e-16 \*\*\*  
## expense -2.228e-02 6.037e-03 -3.69 0.000563 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 59.81 on 49 degrees of freedom  
## Multiple R-squared: 0.2174, Adjusted R-squared: 0.2015   
## F-statistic: 13.61 on 1 and 49 DF, p-value: 0.0005631

## Why is the association between expense and SAT scores /negative/?  
##  
  
## Many people find it surprising that the per-capita expenditure on  
## students is negatively related to SAT scores. The beauty of multiple  
## regression is that we can try to pull these apart. What would the  
## association between expense and SAT scores be if there were no  
## difference among the states in the percentage of students taking the  
## SAT?  
  
summary(lm(csat ~ expense + percent, data = states.data))

##   
## Call:  
## lm(formula = csat ~ expense + percent, data = states.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -62.921 -24.318 1.741 15.502 75.623   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 989.807403 18.395770 53.806 < 2e-16 \*\*\*  
## expense 0.008604 0.004204 2.046 0.0462 \*   
## percent -2.537700 0.224912 -11.283 4.21e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 31.62 on 48 degrees of freedom  
## Multiple R-squared: 0.7857, Adjusted R-squared: 0.7768   
## F-statistic: 88.01 on 2 and 48 DF, p-value: < 2.2e-16

## The lm class and methods  
##  
  
## OK, we fit our model. Now what?  
## Examine the model object:  
  
class(sat.mod)

## [1] "lm"

names(sat.mod)

## [1] "coefficients" "residuals" "effects" "rank"   
## [5] "fitted.values" "assign" "qr" "df.residual"   
## [9] "xlevels" "call" "terms" "model"

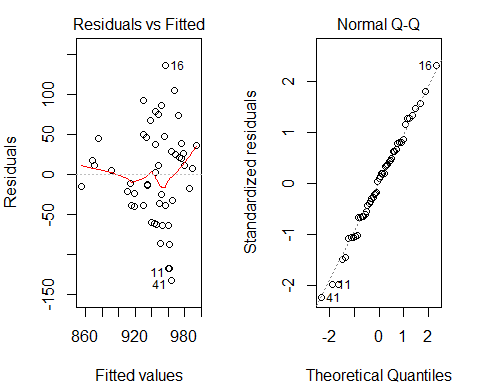
methods(class = class(sat.mod))[1:9]

## [1] "add1.lm" "alias.lm"   
## [3] "anova.lm" "case.names.lm"   
## [5] "coerce,oldClass,S3-method" "confint.lm"   
## [7] "cooks.distance.lm" "deviance.lm"   
## [9] "dfbeta.lm"

## Use function methods to get more information about the fit  
  
confint(sat.mod)

## 2.5 % 97.5 %  
## (Intercept) 995.01753164 1126.44735626  
## expense -0.03440768 -0.01014361

# hist(residuals(sat.mod))  
  
## Linear Regression Assumptions  
##  
  
## Ordinary least squares regression relies on several assumptions,  
## including that the residuals are normally distributed and  
## homoscedastic, the errors are independent and the relationships are  
## linear.  
  
## Investigate these assumptions visually by plotting your model:  
  
par(mar = c(4, 4, 2, 2), mfrow = c(1, 2)) #optional  
plot(sat.mod, which = c(1, 2)) # "which" argument optional



## Comparing models  
##  
  
## Do congressional voting patterns predict SAT scores over and above  
## expense? Fit two models and compare them:  
  
# fit another model, adding house and senate as predictors  
sat.voting.mod <- lm(csat ~ expense + house + senate,  
 data = na.omit(states.data))  
sat.mod <- update(sat.mod, data=na.omit(states.data))  
# compare using the anova() function  
anova(sat.mod, sat.voting.mod)

## Analysis of Variance Table  
##   
## Model 1: csat ~ expense  
## Model 2: csat ~ expense + house + senate  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 46 169050   
## 2 44 149284 2 19766 2.9128 0.06486 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

coef(summary(sat.voting.mod))

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 1082.93438041 38.633812740 28.0307405 1.067795e-29  
## expense -0.01870832 0.009691494 -1.9303852 6.001998e-02  
## house -1.44243754 0.600478382 -2.4021473 2.058666e-02  
## senate 0.49817861 0.513561356 0.9700469 3.373256e-01

## Exercise: least squares regression  
##  
  
## Use the /states.rds/ data set. Fit a model predicting energy consumed  
## per capita (energy) from the percentage of residents living in  
## metropolitan areas (metro). Be sure to  
## 1. Examine/plot the data before fitting the model  
## 2. Print and interpret the model `summary'  
## 3. `plot' the model to look for deviations from modeling assumptions  
  
## Select one or more additional predictors to add to your model and  
## repeat steps 1-3. Is this model significantly better than the model  
## with /metro/ as the only predictor?  
  
## Interactions and factors  
##  
  
## Modeling interactions  
##  
  
## Interactions allow us assess the extent to which the association  
## between one predictor and the outcome depends on a second predictor.  
## For example: Does the association between expense and SAT scores  
## depend on the median income in the state?  
  
 #Add the interaction to the model  
sat.expense.by.percent <- lm(csat ~ expense\*income,  
 data=states.data)   
#Show the results  
 coef(summary(sat.expense.by.percent)) # show regression coefficients table

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 1.380364e+03 1.720863e+02 8.021351 2.367069e-10  
## expense -6.384067e-02 3.270087e-02 -1.952262 5.687837e-02  
## income -1.049785e+01 4.991463e+00 -2.103161 4.083253e-02  
## expense:income 1.384647e-03 8.635529e-04 1.603431 1.155395e-01

## Regression with categorical predictors  
##  
  
## Let's try to predict SAT scores from region, a categorical variable.  
## Note that you must make sure R does not think your categorical  
## variable is numeric.  
  
# make sure R knows region is categorical  
str(states.data$region)

## Factor w/ 4 levels "West","N. East",..: 3 1 1 3 1 1 2 3 NA 3 ...

states.data$region <- factor(states.data$region)  
#Add region to the model  
sat.region <- lm(csat ~ region,  
 data=states.data)   
#Show the results  
coef(summary(sat.region)) # show regression coefficients table

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 946.30769 14.79582 63.9577807 1.352577e-46  
## regionN. East -56.75214 23.13285 -2.4533141 1.800383e-02  
## regionSouth -16.30769 19.91948 -0.8186806 4.171898e-01  
## regionMidwest 63.77564 21.35592 2.9863209 4.514152e-03

anova(sat.region) # show ANOVA table

## Analysis of Variance Table  
##   
## Response: csat  
## Df Sum Sq Mean Sq F value Pr(>F)   
## region 3 82049 27349.8 9.6102 4.859e-05 \*\*\*  
## Residuals 46 130912 2845.9   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Again, \*make sure to tell R which variables are categorical by  
## converting them to factors!\*  
  
## Setting factor reference groups and contrasts  
##  
  
## In the previous example we use the default contrasts for region. The  
## default in R is treatment contrasts, with the first level as the  
## reference. We can change the reference group or use another coding  
## scheme using the `C' function.  
  
# print default contrasts  
contrasts(states.data$region)

## N. East South Midwest  
## West 0 0 0  
## N. East 1 0 0  
## South 0 1 0  
## Midwest 0 0 1

# change the reference group  
coef(summary(lm(csat ~ C(region, base=4),  
 data=states.data)))

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 1010.08333 15.39998 65.589930 4.296307e-47  
## C(region, base = 4)1 -63.77564 21.35592 -2.986321 4.514152e-03  
## C(region, base = 4)2 -120.52778 23.52385 -5.123641 5.798399e-06  
## C(region, base = 4)3 -80.08333 20.37225 -3.931000 2.826007e-04

# change the coding scheme  
coef(summary(lm(csat ~ C(region, contr.helmert),  
 data=states.data)))

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 943.986645 7.706155 122.4977451 1.689670e-59  
## C(region, contr.helmert)1 -28.376068 11.566423 -2.4533141 1.800383e-02  
## C(region, contr.helmert)2 4.022792 5.884552 0.6836191 4.976450e-01  
## C(region, contr.helmert)3 22.032229 4.446777 4.9546509 1.023364e-05

## See also `?contrasts', `?contr.treatment', and `?relevel'.  
  
## Exercise: interactions and factors  
##  
  
## Use the states data set.  
  
setwd("~/Springboard Projects/Chapter 7 - Machine Learning/linear\_regression/linear\_regression/dataSets")  
# read the states data  
states.data <- readRDS("states.rds")   
#get labels  
states.info <- data.frame(attributes(states.data)[c("names", "var.labels")])  
#look at last few labels  
tail(states.info, 8)

## names var.labels  
## 14 csat Mean composite SAT score  
## 15 vsat Mean verbal SAT score  
## 16 msat Mean math SAT score  
## 17 percent % HS graduates taking SAT  
## 18 expense Per pupil expenditures prim&sec  
## 19 income Median household income, $1,000  
## 20 high % adults HS diploma  
## 21 college % adults college degree

## Use the states.rds data set. Fit a model predicting energy consumed per capita (energy) from the percentage of   
#residents living in metropolitan areas (metro).  
states.metro.energy <- subset(states.data, select = c("metro", "energy"))  
  
#Be sure to  
  
#Examine/plot the data before fitting the model  
summary(states.metro.energy)

## metro energy   
## Min. : 20.40 Min. :200.0   
## 1st Qu.: 46.98 1st Qu.:285.0   
## Median : 67.55 Median :320.0   
## Mean : 64.07 Mean :354.5   
## 3rd Qu.: 81.58 3rd Qu.:371.5   
## Max. :100.00 Max. :991.0   
## NA's :1 NA's :1

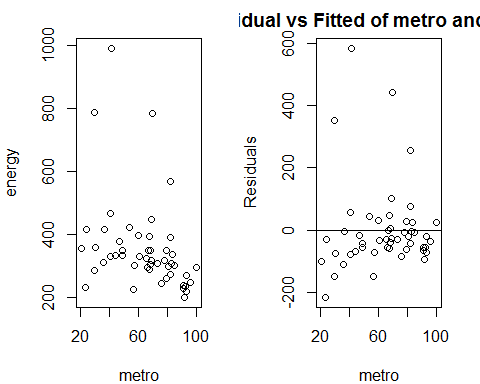
#There is an NA which will not calculate a correlation, so removing single NA from energy and metro  
states.metro.energy <- na.omit(states.metro.energy)  
cor(states.metro.energy)

## metro energy  
## metro 1.0000000 -0.3397445  
## energy -0.3397445 1.0000000

plot(states.metro.energy)  
  
#Print and interpret the model summary  
model.metro.energy <- lm(energy ~ metro, data = states.data)  
summary(model.metro.energy)

##   
## Call:  
## lm(formula = energy ~ metro, data = states.data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -215.51 -64.54 -30.87 18.71 583.97   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 501.0292 61.8136 8.105 1.53e-10 \*\*\*  
## metro -2.2871 0.9139 -2.503 0.0158 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 140.2 on 48 degrees of freedom  
## (1 observation deleted due to missingness)  
## Multiple R-squared: 0.1154, Adjusted R-squared: 0.097   
## F-statistic: 6.263 on 1 and 48 DF, p-value: 0.01578

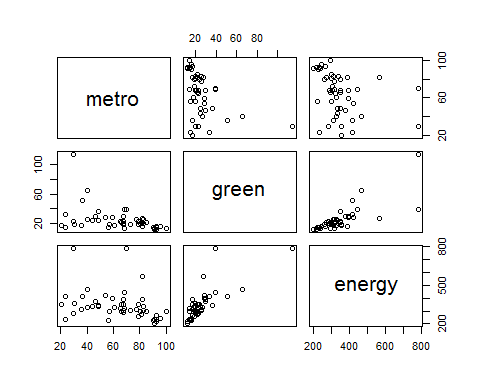
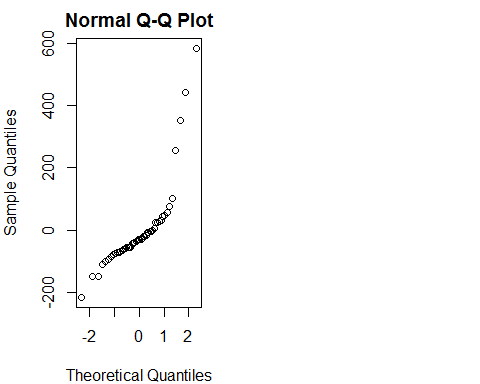
#Since probability that there is a strong relationship of energy and metro due to chance is very small,   
#we can assume there is a relationship between metro and energy. However, since the R-squared is very small,  
#metro is not the only variable that predicts energy well  
  
#plot the model to look for deviations from modeling assumptions  
model.metro.energy.resid <- resid(model.metro.energy)  
plot(states.metro.energy$metro, model.metro.energy.resid, ylab = "Residuals", xlab = "metro", main = "Residual vs Fitted of metro and energy")   
abline(0, 0)



qqnorm(model.metro.energy.resid)  
#Residual vs fitted plot does not suggest any relationship. Can assume residuals are independent  
#QQ plot with a few outliers suggest normality of residuals  
  
#Select one or more additional predictors to add to your model and repeat steps 1-3.  
states.new <- subset(states.data, select = c("metro", "green", "energy"))  
states.new <- na.omit(states.new)  
cor(states.new)

## metro green energy  
## metro 1.0000000 -0.4111107 -0.3116753  
## green -0.4111107 1.0000000 0.7706181  
## energy -0.3116753 0.7706181 1.0000000

plot(states.new)



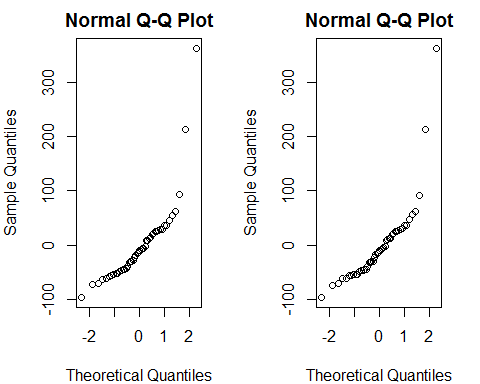
#Model with green as only independent variable  
model.new1 <- lm(energy ~ green, data = na.omit(states.data))  
summary(model.new1)

##   
## Call:  
## lm(formula = energy ~ green, data = na.omit(states.data))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -95.72 -45.64 -11.25 25.29 362.29   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 206.3554 19.9537 10.342 1.38e-13 \*\*\*  
## green 5.4641 0.6663 8.201 1.50e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 75.34 on 46 degrees of freedom  
## Multiple R-squared: 0.5939, Adjusted R-squared: 0.585   
## F-statistic: 67.26 on 1 and 46 DF, p-value: 1.497e-10

model.new.resid1 <- resid(model.new1)  
qqnorm(model.new.resid1)  
#Model with green and metro as independent variables  
model.new2 <- lm(energy ~ metro + green, data = na.omit(states.data))  
summary(model.new2)

##   
## Call:  
## lm(formula = energy ~ metro + green, data = na.omit(states.data))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -95.66 -46.12 -11.06 25.20 361.87   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 203.79651 47.64448 4.277 9.73e-05 \*\*\*  
## metro 0.03276 0.55257 0.059 0.953   
## green 5.48207 0.73892 7.419 2.45e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 76.17 on 45 degrees of freedom  
## Multiple R-squared: 0.5939, Adjusted R-squared: 0.5758   
## F-statistic: 32.9 on 2 and 45 DF, p-value: 1.565e-09

model.new.resid2 <- resid(model.new2)  
qqnorm(model.new.resid2)



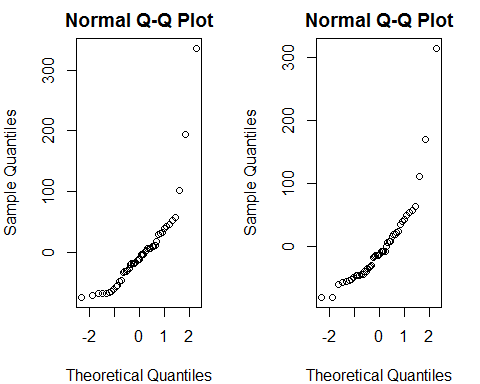
#Is this model significantly better than the model with metro as the only predictor?  
 #Using green only, rather than metro + green model, is a better model. Adjusted R-squared is relatively the same  
 #using one less independent variable meaning more simple model.  
  
  
## 1. Add on to the regression equation that you created in exercise 1 by  
## generating an interaction term and testing the interaction.  
  
#Model with green and senate as independent variables  
model.add1 <- lm(energy ~ green + senate, data = na.omit(states.data))  
summary(model.add1)

##   
## Call:  
## lm(formula = energy ~ green + senate, data = na.omit(states.data))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -73.82 -36.10 -11.41 12.75 335.77   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 275.3194 33.4857 8.222 1.64e-10 \*\*\*  
## green 4.8405 0.6790 7.129 6.57e-09 \*\*\*  
## senate -1.0577 0.4239 -2.495 0.0163 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 71.39 on 45 degrees of freedom  
## Multiple R-squared: 0.6432, Adjusted R-squared: 0.6274   
## F-statistic: 40.56 on 2 and 45 DF, p-value: 8.491e-11

model.add.resid1 <- resid(model.add1)  
qqnorm(model.add.resid1)  
  
## 2. Try adding region to the model. Are there significant differences  
## across the four regions?  
  
#Model with green, senate, and region as independent variables  
model.add2 <- lm(energy ~ green + senate + region, data = na.omit(states.data))  
summary(model.add2)

##   
## Call:  
## lm(formula = energy ~ green + senate + region, data = na.omit(states.data))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -81.13 -45.03 -13.31 20.03 314.87   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 245.8399 37.6984 6.521 7.06e-08 \*\*\*  
## green 4.9406 0.6848 7.215 7.18e-09 \*\*\*  
## senate -0.7741 0.5146 -1.504 0.140   
## regionN. East -8.4492 39.2721 -0.215 0.831   
## regionSouth 41.6295 28.0151 1.486 0.145   
## regionMidwest 1.5235 30.8505 0.049 0.961   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 70.88 on 42 degrees of freedom  
## Multiple R-squared: 0.6718, Adjusted R-squared: 0.6327   
## F-statistic: 17.19 on 5 and 42 DF, p-value: 3.107e-09

model.add.resid2 <- resid(model.add2)  
qqnorm(model.add.resid2)



#There are differences in coefficients across the four regions, but the significance of North East, South  
#and Midwest are not significant enough to be a predictor for energy