Linear Regression Models

Mid-semester practical exam

2021-10-13

# DATA SET

The data set used has generic data, with fields the response variable and regressors giving no indication of the underlying purpose of the data collection. Out of the available 9 regressors, I will be choosing only 5 for the purpose of this assignment.

setwd("~/Documents/Study/computerScience/programming/r/data/")  
d = read.csv("justSomeData.csv")[c(1:6)]  
head(d)

## y x1 x2 x3 x4 x5  
## 1 25.9 4.9176 1 3.472 0.998 1  
## 2 29.5 5.0208 1 3.531 1.500 2  
## 3 27.9 4.5429 1 2.275 1.175 1  
## 4 25.9 4.5573 1 4.050 1.232 1  
## 5 29.9 5.0597 1 4.455 0.988 1  
## 6 30.9 5.8980 1 5.850 1.240 1

# FULL MODEL BEST FIT AND RESIDUAL ANALYSIS

We will be finding the best fit model for all the regressors together and the response. We will also confirm the normality assumption (i.e. residuals of the model are normally distributed).

## FITTED MODEL

model = lm(y~., d)  
summary(model)

##   
## Call:  
## lm(formula = y ~ ., data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.434 -2.168 0.061 2.077 3.941   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.2547 3.4140 2.711 0.0154 \*  
## x1 2.2315 0.7655 2.915 0.0101 \*  
## x2 6.7510 3.9733 1.699 0.1087   
## x3 0.4895 0.4716 1.038 0.3147   
## x4 -1.2846 3.9116 -0.328 0.7469   
## x5 1.5499 1.2723 1.218 0.2408   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.889 on 16 degrees of freedom  
## Multiple R-squared: 0.8309, Adjusted R-squared: 0.7781   
## F-statistic: 15.73 on 5 and 16 DF, p-value: 1.106e-05

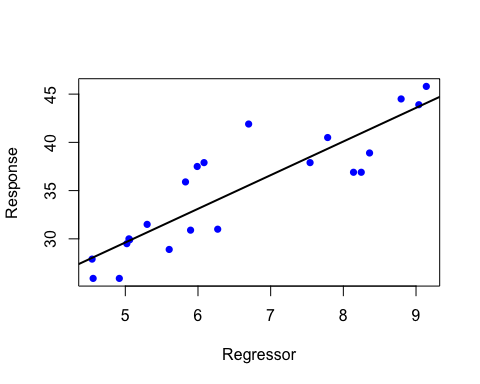
Hence, we get the fitted model for all 5 regressors as

**y = 9.2547 + 2.2315x1 + 6.7510x2 + 0.4895x3 - 1.2846x4 + 1.5499x5**

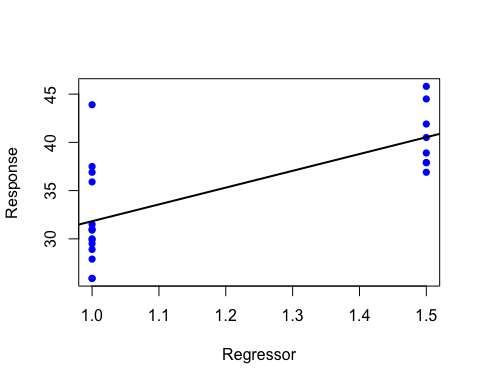
## SCATTER PLOTS FOR VARIOUS REGRESSORS

# Function for doing this...  
scatterPlot = function(x, y)  
{  
 plot(x, y,  
 type = "p",  
 xlab = "Regressor",  
 ylab = "Response",  
 col = "blue",  
 pch = 16,  
 )  
 # To draw the fitted regression line.  
 abline(lm(y~x, data.frame(x, y)), lwd = 2)  
}  
# Loop for function calls...  
for(x in d[c(2:6)]){scatterPlot(x, d$y)}

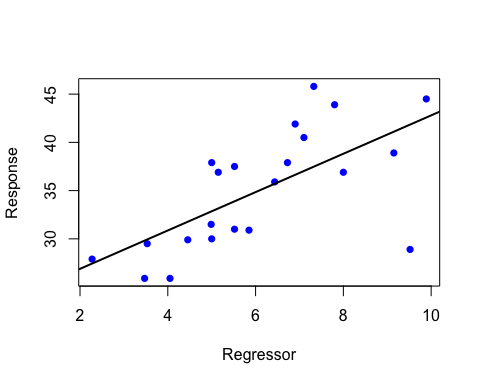
y ~ x1...



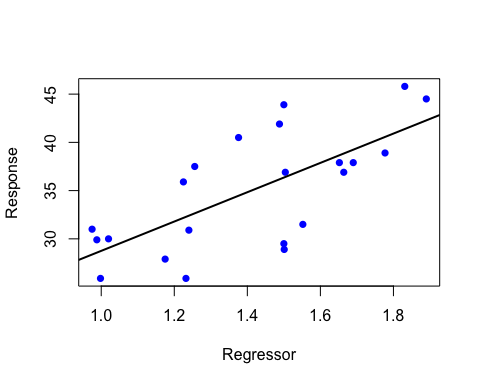
y ~ x2...



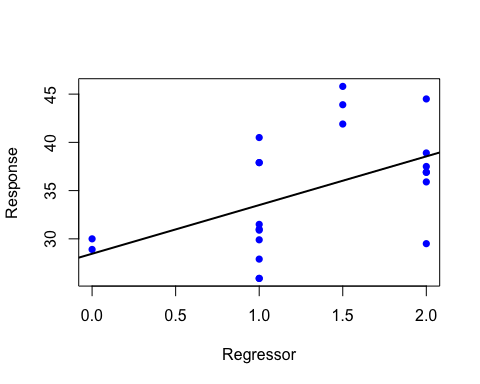
y ~ x3...



y ~ x4...



y ~ x5...



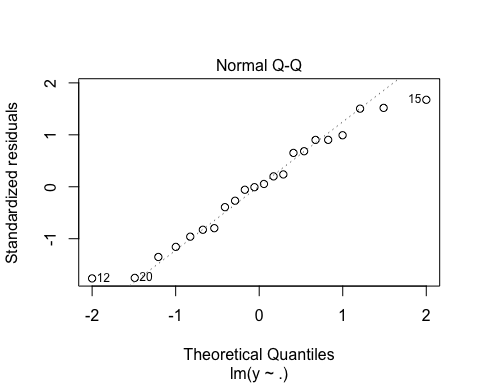
From the above plots, we can see that most regressors seem to have a roughly linear relationship with the regressor. x2 and x5 seem to be variables that could be categorical, but they have not been handled as such, since

1. It is not known for sure whether they are, since only numerical values are given
2. It is beyond the scope of this assignment

## NORMALITY ASSUMPTION VERIFICATION

We will be mainly concerned with the Q-Q residual plot, which helps verify the normality of the error terms.

plot(model)



Q-Q plot shows ideal probability distribution of points around the regression line, hence we may conclude that

* The standardized residuals are normally distributed
* The variables are linearly associated (at least largely)

# BEST FIT MODEL THROUGH VARIABLE SELECTION

## FULL MODEL

model = lm(y~., d)

This is the model with all the available regressors. This will be used as a source for the regressors, when we create the best fitting model.

summary(model)

##   
## Call:  
## lm(formula = y ~ ., data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.434 -2.168 0.061 2.077 3.941   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.2547 3.4140 2.711 0.0154 \*  
## x1 2.2315 0.7655 2.915 0.0101 \*  
## x2 6.7510 3.9733 1.699 0.1087   
## x3 0.4895 0.4716 1.038 0.3147   
## x4 -1.2846 3.9116 -0.328 0.7469   
## x5 1.5499 1.2723 1.218 0.2408   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.889 on 16 degrees of freedom  
## Multiple R-squared: 0.8309, Adjusted R-squared: 0.7781   
## F-statistic: 15.73 on 5 and 16 DF, p-value: 1.106e-05

Hence, we get the fitted model for all 5 regressors as

**y = 9.2547 + 2.2315x1 + 6.7510x2 + 0.4895x3 - 1.2846x4 + 1.5499x5**

## STEP FUNCTION

Chooses the best regression model using the AIC stepwise variable selection algorithm. Best in this context means the model that is has the regressors and coefficients that best explain or match the responses, given the data. Hence, it is not only best fitting for a given set of regressors, it is also best fitting among all possible models using the available regressors.

(**AIC** => Akaike’s information criterion. It compares the quality of a set of statistical models to each other)

## USAGE

# step(object, scope, scale = 0,  
# direction = c("both", "backward", "forward"),  
# trace = 1, keep = NULL, steps = 1000, k = 2)

Argument “**object**” is an object representing a model of an appropriate class (mainly “lm” and “glm”). This is used as the initial model in the stepwise search (variable selection) for the best regressors for modelling the given response. Initial model implies the model with the response and an initial set of regressors and coefficients on top of which more regressors will be added. Typically, it is a model with only the response, intercept and error term.

Argument “**scope**” defines the range of models examined in the stepwise search. It holds the model or models containing the different regressors that may be selected for the final model returned by the function. This option could contain a single model, or two models “lower” and “upper”, wherein the regressors in the lower model are a subset of the regressors in the upper model. In the case of “lower” and “upper” models, the step function performs a stepwise search for every model from the lower to the upper (and the models in between, with respect to the regressors used).

Argument “**direction**” the mode of stepwise search, can be one of “both”, “backward”, or “forward”, with a default of “both”. If the scope argument is missing the default for direction is “backward”. Forward implies that we start with less regressors, and keep adding and fitting more.

## MAKING THE INITIAL MODEL

This acts as the starting point for the selection process. Regressors that are selected are added onto this model initially.

initialModel = lm(y~1, data = d)  
# 1 as the regressor implies the constant term, whose coefficient is the intercept.  
summary(initialModel)

##   
## Call:  
## lm(formula = y ~ 1, data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.095 -5.071 1.405 3.655 10.805   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34.995 1.308 26.77 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.133 on 21 degrees of freedom

## PERFORMING THE STEPWISE SEARCH

step(initialModel, direction = "forward", scope = formula(model))

## Start: AIC=80.78  
## y ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + x1 1 616.67 173.16 49.390  
## + x2 1 386.12 403.71 68.012  
## + x4 1 385.20 404.63 68.062  
## + x3 1 339.51 450.32 70.416  
## + x5 1 199.58 590.25 76.369  
## <none> 789.83 80.777  
##   
## Step: AIC=49.39  
## y ~ x1  
##   
## Df Sum of Sq RSS AIC  
## + x2 1 22.9619 150.20 48.260  
## <none> 173.16 49.390  
## + x4 1 7.8167 165.34 50.374  
## + x5 1 5.6693 167.49 50.657  
## + x3 1 3.2496 169.91 50.973  
##   
## Step: AIC=48.26  
## y ~ x1 + x2  
##   
## Df Sum of Sq RSS AIC  
## <none> 150.20 48.260  
## + x5 1 7.6678 142.53 49.107  
## + x3 1 4.2757 145.92 49.625  
## + x4 1 0.3311 149.87 50.212

##   
## Call:  
## lm(formula = y ~ x1 + x2, data = d)  
##   
## Coefficients:  
## (Intercept) x1 x2   
## 9.321 2.923 5.550

From the function’s results, we see that x1 and x2 are the best regressors for our response y, with the given coefficients leading to the best fitting model possible for the data and available regressors.

#

## FINAL MODEL (BEST FIT MODEL)

bestModel = lm(y~x1+x2, data = d)  
summary(bestModel )

##   
## Call:  
## lm(formula = y ~ x1 + x2, data = d)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.8510 -2.0998 0.0266 1.3604 5.1215   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.3207 3.1373 2.971 0.00785 \*\*   
## x1 2.9232 0.5162 5.663 1.85e-05 \*\*\*  
## x2 5.5497 3.2563 1.704 0.10462   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.812 on 19 degrees of freedom  
## Multiple R-squared: 0.8098, Adjusted R-squared: 0.7898   
## F-statistic: 40.46 on 2 and 19 DF, p-value: 1.418e-07

Hence, we get the best fit model (using forward selection) as

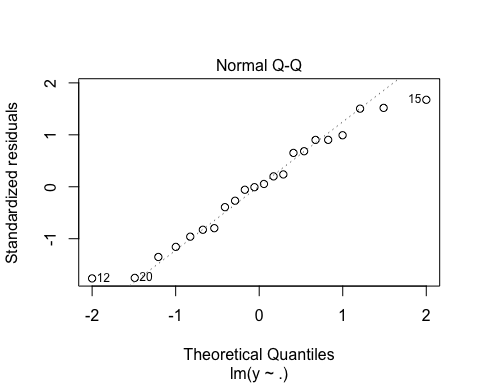
**y = 9.3207 + 2.9232x1 + 5.5497x2**

Compared to the full model, whose adjusted R-squared value is 0.7781 i.e. 77.81%, the best fit model using forward selection has adjusted R-squared value 0.7898 i.e. 78.98, showing a marginal increase in the amount of variation in the data explained by the model, indicating a better fit.

## NORMALITY ASSUMPTION VERIFICATION

plot(model)

We will be mainly concerned with the Q-Q residual plot, which helps verify the normality of the error terms.



Q-Q plot shows ideal probability distribution of points around the regression line, hence we may conclude that

* The standardized residuals are normally distributed
* The variables are linearly associated