BAN404Assignment1

Group8

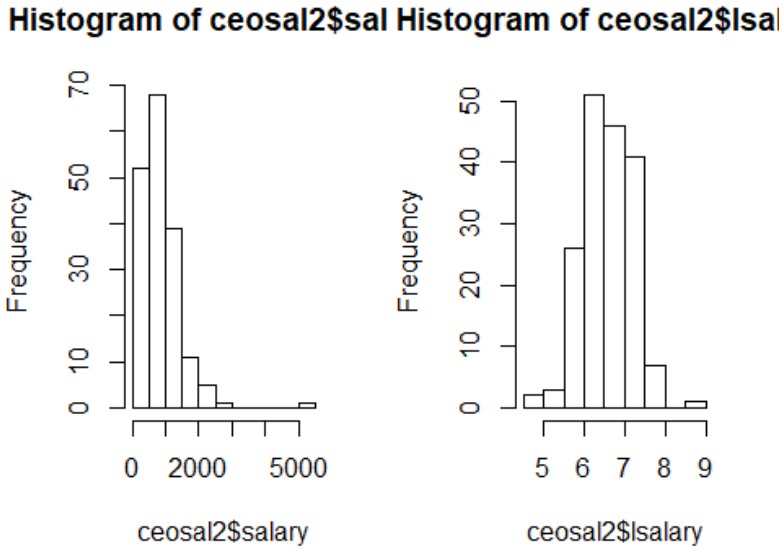
a. Describe relevant features with descriptive statistics.

There are 177 observations in this dataset and 15 variables, including 2 output variables and 13 input variables. We can get a glimpse of the min, mean, median, max and quantile values of each variable with summary () function.

1. Output variables.

The two output variables in the dataset are salary and lsalary. Salary gives us the CEO compensation in thousands of dollars with a huge variation from lowest 100 to highest 5299. The second output variable is lsalary, which is the natural logarithmic of salary.

The lsalary is chosen as the more preferred response variable for the reason that logarithmic transformations can make a highly skewed variable into a more approximately normal distributed variable (Kenneth Benoit, 2011). We plot the histograms of salary and lsalary, from which we see a significant right skew in salary and a distribution more like a normal distribution in lsalary.



Since our task is to predict salary, we can interpret the logarithmically models in this way:

Log Y and Log X: if X increases by 1%, Y would (approximately) increase by β%

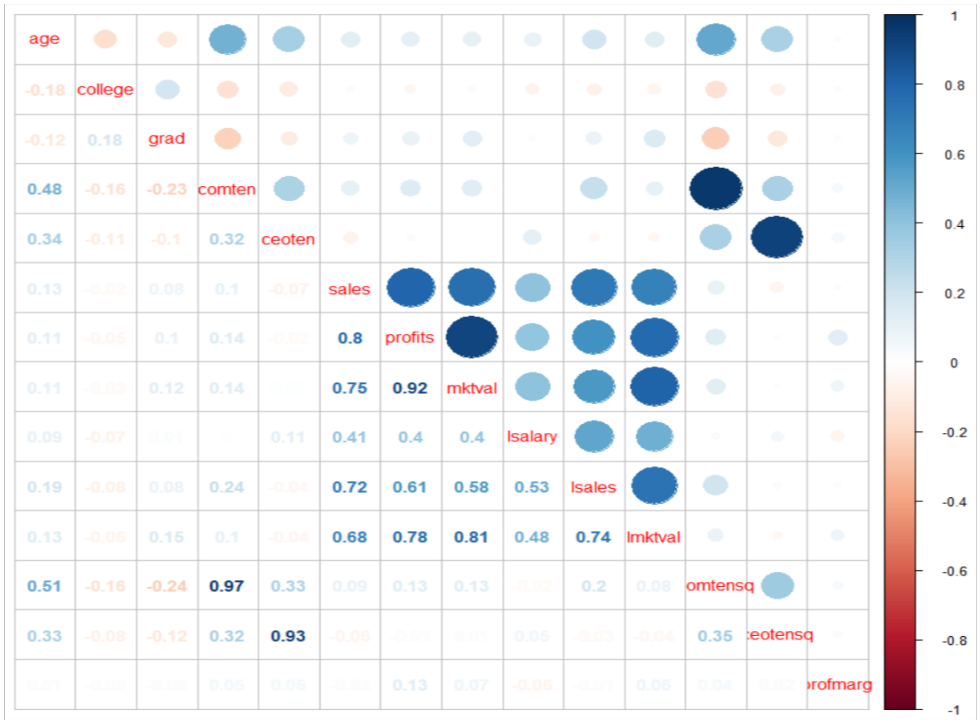
Log Y and X: if X increases by one unit, Y would (approximately) increase by β*100 %

1. Input variables

(1) Qualitative variables. The dataset contains two qualitative predictors, college and grad. College indicates whether the CEOs attended college or not. We find out that 172 out of 177 CEOs have attended college, indicating the variable may not be a key factor affecting the variation of the response. It can probably offer limited explanatory power. The second binary predictor is grad, meaning if the CEOs have finished their graduate study after college. Almost half of the CEOs didn't finish their graduate school. The variable grad has an evident variation, which might help explain the response variable.

(2) Quantitative variables. By conducting the VIF test, we notice the existence of collinearity among 11 quantitative variables.

```
##    Variables       VIF
## 1        age  1.492326
## 2    college  1.082631
## 3       grad  1.148750
## 4     comten 16.074540
## 5     ceoten  8.316840
## 6      sales  4.016435
## 7    profits  8.180653
## 8     mktval  8.148076
## 9    lsalary  1.629541
## 10    lsales  3.744934
## 11   lmktval  4.586639
## 12  comtensq 16.687698
## 13  ceotensq  8.078201
## 14  profmarg  1.093844
```



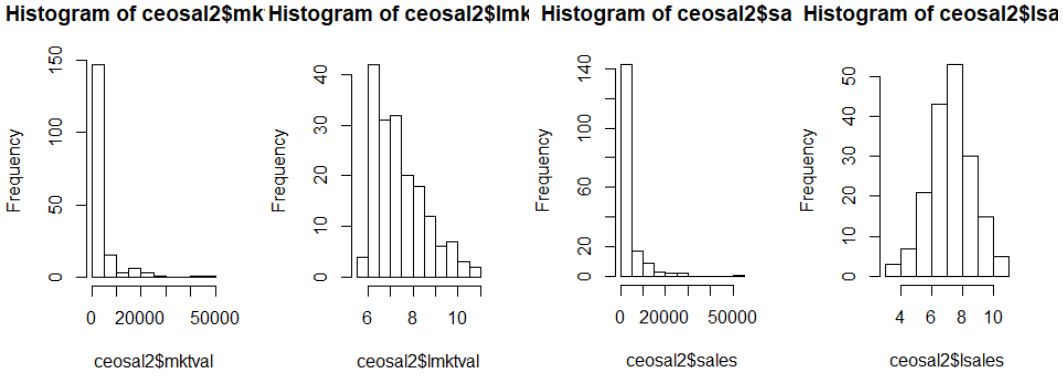
Then we try to identify the pairs of variables which are strongly correlated from the

pairwise correlation plot listed above. Since collinearity reduces the accuracy of the estimates of the regression coefficients, we must drop one of the variables or combine the collinear variables together into a single variable.

a) The correlation between input variable comten and comtensq is 0.97. Comten is the variable shows the number of years that CEO is with the company. Comtensq is the quadratic function of the variable comten. With the presence of comtensq, it causes bias in the estimated coefficient with respect to comten, since we cannot hold comtensq fixed while changing comten. In order to make our linear model easier to interpret, we keep comten in our chosen subset of predictors.

b) The correlation between ceoten and ceotensq is 0.93. Ceoten is the number of years as CEO with company. Ceotensq is the quadratic function of the variable ceoten. Based on the same reasons mentioned above, we keep ceoten in the subset of predictors.

c) The correlation between mktval, the firm's market value in 1990 measured in millions of dollars and lmktval, the natural logarithmic form of the mktval, is 0.81. We can see a significant right skew from the below histogram of mktval while the histogram of lmktval is much more dispersed. Furthermore, lmktval is more correlated to the response variable lsalary than mktval is. Thus, we choose to leave lmktval in the subset of predictors.



d) The correlation between sales, measured in millions of dollars, and lsales, the natural logarithmic form of sales, is 0.72. From the above histograms of sales and lsales, we observe that the dispersion of lsales is much wider than that of the sales. We keep lsales in the subset of predictors due to the same considerations as in the pairs of mktval and lmktval.

e) The correlation between profits, the firm's profits in 1990 in millions of dollars, and lmktval is 0.78. Because lmktval is more correlated to lsalary, we leave it as a chosen predictor instead of the less correlated variable profits.

After removing 5 variables, we get the subset of 8 chosen predictors, namely age, college, grad, comten, ceoten, mktval and profmarg, referring to profits as percentage of sales. We test collinearity again and the VIF values are all below 3, indicating we have relatively controlled collinearity.

```
##   Variables      VIF
## 1       age 1.406955
```

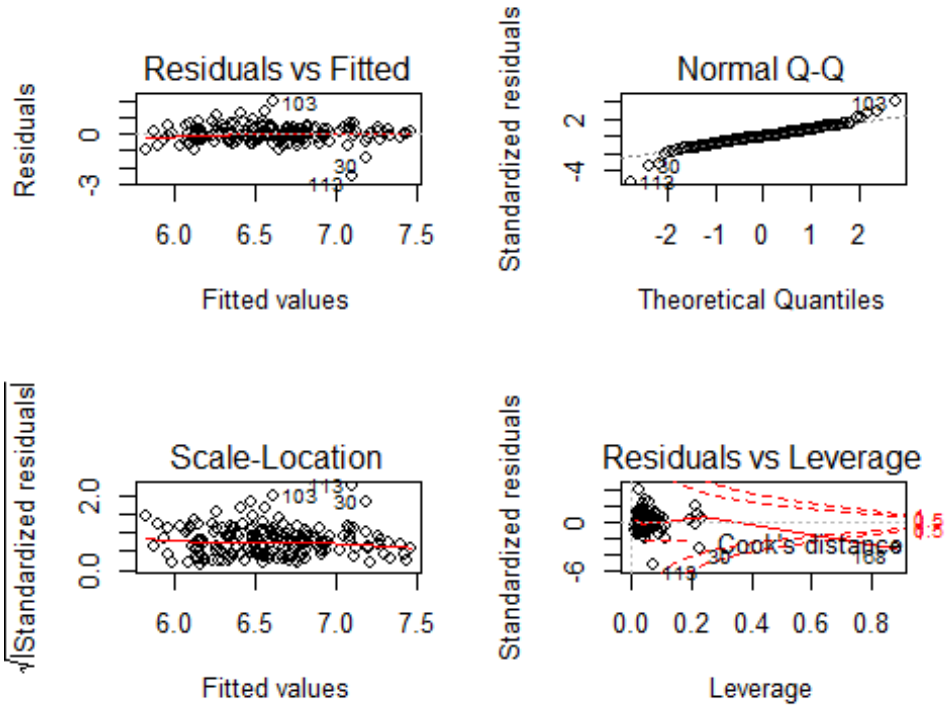
```
## 2   college 1.070333
## 3      grad 1.129973
## 4    comten 1.547079
## 5    ceoten 1.261215
## 6   lsalary 1.560014
## 7    lsales 2.678891
## 8   lmktval 2.330816
## 9  profmarg 1.024191
```

In the following part b and part c, we will use different kinds of linear regressions on both the data frame containing all predictors "df" and the data frame including the 8 chosen predictors "df2" and see if they will get the same model, if not, which model performs better.

Outlier

From the residual plots of standard linear regression on all predictors, we identify several possible outliers including 103, 113, 30 and so on. Combined with the histogram of salary, we are quite sure that observation 103 is an outlier, because its salary is the highest and is almost twice as much as the second highest salary, while its sales and profits aren't that impressive. So, we remove it and refit the model.

After removing the outlier, the RSE drops from 0.49 to 0.47, which means a single outlier caused a strong influence on the interpretation of the fit.



b. Use different methods to predict salary

1. Standard linear regression

Comparing the t-statistics of the regression on all 14 predictors and the t-statistics of the regression on the selected predictors, we can find the presence of collinearity causes a decline in the absolute value of the t-statistics. For instance, the absolute value of t value in variable comten increases from 0.1 to 3.2. In the regression on all 14 predictors, the p value of comten is not significant while in the regression on the subset of predictors, it becomes significant. In order to avoid this situation, it is better to identify and address potential collinearity problems before fitting the linear model.

```
##
## Call:
## lm(formula = lsalary ~ ., data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.44875 -0.23761 -0.01127  0.27549  1.31766
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.708e+00  5.248e-01   8.971 6.96e-16 ***
## age          1.329e-03  5.151e-03   0.258  0.79677
## college     -7.046e-02  2.227e-01  -0.316  0.75216
## grad        -9.586e-02  7.570e-02  -1.266  0.20720
## comten      -1.377e-03  1.164e-02  -0.118  0.90594
## ceoten       4.080e-02  1.397e-02   2.919  0.00401 **
## sales       -7.956e-06  1.167e-05  -0.682  0.49646
## profits      9.100e-05  2.510e-04   0.362  0.71746
## mktval       9.361e-06  1.574e-05   0.595  0.55277
## lsales       1.963e-01  4.553e-02   4.312 2.80e-05 ***
## lmktval      6.005e-02  6.694e-02   0.897  0.37095
## comtensq    -2.337e-04  2.538e-04  -0.921  0.35851
## ceotensq    -9.007e-04  4.709e-04  -1.913  0.05756 .
## profmarg    -2.804e-03  2.068e-03  -1.356  0.17701
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4711 on 162 degrees of freedom
## Multiple R-squared:  0.4072, Adjusted R-squared:  0.3597
## F-statistic: 8.561 on 13 and 162 DF,  p-value: 4.294e-13

##
## Call:
## lm(formula = lsalary ~ ., data = df2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4909 -0.2515 -0.0074  0.2782  1.3805
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  4.7236799  0.4142712  11.402  < 2e-16 ***
## age         -0.0003527  0.0050306  -0.070  0.94419
## college     -0.1062769  0.2225404  -0.478  0.63359
## grad        -0.0769275  0.0756664  -1.017  0.31078
## comten      -0.0114526  0.0035445  -3.231  0.00149 **
## ceoten       0.0165434  0.0054700   3.024  0.00289 **
## lsales       0.1848123  0.0384000   4.813 3.31e-06 ***
## lmktval      0.1102081  0.0474779   2.321  0.02148 *
## profmarg    -0.0021671  0.0020170  -1.074  0.28419
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4738 on 167 degrees of freedom
## Multiple R-squared:  0.3819, Adjusted R-squared:  0.3523
## F-statistic:  12.9 on 8 and 167 DF,  p-value: 2.208e-14
```

2. Subset selection

We will discuss some alternative fitting procedures including subset selection methods, lasso and ridge regression and will briefly explain on how these methods are developed from standard linear regression.

Subset selection procedure identifies a subset of the predictors that are related to the response, then fits a model using least squares on the reduced set of independent variables.

(1) Best subset

We fit a separate least squares regression for each possible combination of the 13 predictors, which means it fitted 2^13=8192 times. As a result, we identify the best model for each subset size of predictors.

Then, we compute the test MSE of the 13 models that contain a given number of predictors and find out the best model contains only one predictor. Finally, we perform best subset selection on the full data set, and get the best model lsalary=4.97+0.22lsales, which means if sales increase by 1%, salary would increase by 0.22% while others remain constant.

```
## [1] 1

## Subset selection object
## Call: regsubsets.formula(lsalary ~ ., data = df, nvmax = nv)
## 13 Variables  (and intercept)
## Selection Algorithm: exhaustive
##          age college grad comten ceoten sales profits mktval lsales
## 1  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 2  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 3  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 4  ( 1 )  " " " "     " "  " "    "*"    " "   " "     "*"    "*"
## 5  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 6  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     " "    "*"
## 7  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     " "    "*"
## 8  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     "*"    "*"
## 9  ( 1 )  " " " "     "*"  " "    "*"    "*"   " "     "*"    "*"
## 10  ( 1 ) " " " "     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 11  ( 1 ) " " "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 12  ( 1 ) "*" "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
```

```
## 13  ( 1 ) "*" "*"     "*"  "*"    "*"    "*"   "*"     "*"    "*"
##           lmktval comtensq ceotensq profmarg
## 1  ( 1 )  " "     " "      " "      " "
## 2  ( 1 )  " "     "*"      " "      " "
## 3  ( 1 )  " "     "*"      " "      " "
## 4  ( 1 )  " "     "*"      " "      " "
## 5  ( 1 )  "*"     "*"      "*"      " "
## 6  ( 1 )  "*"     "*"      "*"      " "
## 7  ( 1 )  "*"     "*"      "*"      "*"
## 8  ( 1 )  "*"     "*"      "*"      "*"
## 9  ( 1 )  "*"     "*"      "*"      "*"
## 10  ( 1 ) "*"     "*"      "*"      "*"
## 11  ( 1 ) "*"     "*"      "*"      "*"
## 12  ( 1 ) "*"     "*"      "*"      "*"
## 13  ( 1 ) "*"     "*"      "*"      "*"

## (Intercept)     lsales
##   4.9712573   0.2214021
```

We use the same fitting process on the subset of predictors and get the best model lsalary= 5.04-0.0082comten+0.24lsales, which means while others remain unchanged, if the observation stays in the company one more year, salary would decrease by 0.8% and if sales increase by 1%, salary would increase by 0.24%. Intuition tells that the correlation between comten and salary is generally positive, so this might not be a good fit.

```
## [1] 2

##  (Intercept)      comten      lsales
##  5.035487485 -0.008193566  0.237899401
```

(2) Forward stepwise

Forward stepwise selection is more computationally efficient compared with best subset selection. It begins with a model with no predictors and adds one to the model at a time until all p predictors are in the model, with 1+p(p+1)/2=92 times of fitting process.

Using forward stepwise methods, we get the same fitted model as using best subset selection method both on all predictors and on the subset of predictors.

```
## [1] 1

## Subset selection object
## Call: regsubsets.formula(lsalary ~ ., data = df, nvmax = nv, method = "forwa
rd")
## 1 subsets of each size up to 13
## Selection Algorithm: forward
##          age college grad comten ceoten sales profits mktval lsales
## 1  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 2  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 3  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 4  ( 1 )  " " " "     " "  " "    "*"    " "   " "     "*"    "*"
## 5  ( 1 )  " " " "     " "  " "    "*"    " "   " "     "*"    "*"
## 6  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     "*"    "*"
## 7  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     "*"    "*"
## 8  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     "*"    "*"
```

```
## 9  ( 1 )  " " " "     "*"  " "    "*"    "*"   " "     "*"    "*"
## 10  ( 1 ) " " " "     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 11  ( 1 ) " " "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 12  ( 1 ) "*" "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 13  ( 1 ) "*" "*"     "*"  "*"    "*"    "*"   "*"     "*"    "*"
##           lmktval comtensq ceotensq profmarg
## 1  ( 1 )  " "     " "      " "      " "
## 2  ( 1 )  " "     "*"      " "      " "
## 3  ( 1 )  " "     "*"      " "      " "
## 4  ( 1 )  " "     "*"      " "      " "
## 5  ( 1 )  " "     "*"      "*"      " "
## 6  ( 1 )  " "     "*"      "*"      " "
## 7  ( 1 )  " "     "*"      "*"      "*"
## 8  ( 1 )  "*"     "*"      "*"      "*"
## 9  ( 1 )  "*"     "*"      "*"      "*"
## 10  ( 1 ) "*"     "*"      "*"      "*"
## 11  ( 1 ) "*"     "*"      "*"      "*"
## 12  ( 1 ) "*"     "*"      "*"      "*"
## 13  ( 1 ) "*"     "*"      "*"      "*"

## (Intercept)     lsales
##   4.9712573   0.2214021

## [1] 2

##  (Intercept)      comten      lsales
##  5.035487485 -0.008193566  0.237899401
```

(3) Backward stepwise

The difference between forward stepwise and backward stepwise is that backward stepwise selection begins with the full least squared model containing all p predictors and remove one at a time.

Using backward stepwise methods, we get the same fitted model as using best subset selection method both on all predictors and on the subset of predictors.

```
## [1] 1

## Subset selection object
## Call: regsubsets.formula(lsalary ~ ., data = df, nvmax = nv, method = "backw
ard")
## 13 Variables  (and intercept)
## Selection Algorithm: backward
##           age college grad comten ceoten sales profits mktval lsales
## 1  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 2  ( 1 )  " " " "     " "  " "    " "    " "   " "     " "    "*"
## 3  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 4  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 5  ( 1 )  " " " "     " "  " "    "*"    " "   " "     " "    "*"
## 6  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     " "    "*"
## 7  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     " "    "*"
## 8  ( 1 )  " " " "     "*"  " "    "*"    " "   " "     "*"    "*"
## 9  ( 1 )  " " " "     "*"  " "    "*"    "*"   " "     "*"    "*"
## 10  ( 1 ) " " " "     "*"  " "    "*"    "*"   "*"     "*"    "*"
```

```
## 11  ( 1 ) " " "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 12  ( 1 ) "*" "*"     "*"  " "    "*"    "*"   "*"     "*"    "*"
## 13  ( 1 ) "*" "*"     "*"  "*"    "*"    "*"   "*"     "*"    "*"
##           lmktval comtensq ceotensq profmarg
## 1  ( 1 )  " "     " "      " "      " "
## 2  ( 1 )  " "     "*"      " "      " "
## 3  ( 1 )  " "     "*"      " "      " "
## 4  ( 1 )  "*"     "*"      " "      " "
## 5  ( 1 )  "*"     "*"      "*"      " "
## 6  ( 1 )  "*"     "*"      "*"      " "
## 7  ( 1 )  "*"     "*"      "*"      "*"
## 8  ( 1 )  "*"     "*"      "*"      "*"
## 9  ( 1 )  "*"     "*"      "*"      "*"
## 10  ( 1 ) "*"     "*"      "*"      "*"
## 11  ( 1 ) "*"     "*"      "*"      "*"
## 12  ( 1 ) "*"     "*"      "*"      "*"
## 13  ( 1 ) "*"     "*"      "*"      "*"

## (Intercept)     lsales
##   4.9712573   0.2214021

## [1] 2

##  (Intercept)      comten      lsales
##  5.035487485 -0.008193566  0.237899401
```

In summary, we compare these three subset selection methods.

The disadvantage of best subset selection is that it is computationally infeasible if the number of predictors is too big.

The disadvantage of stepwise selections is that they are not certain to get the best possible model out of all 2^p models. As in this example, when using best subset selection on all predictors, mktval is selected in the 4-predictor model and is dropped in the 5-predictor model, but is selected in the 8-predictor model again. However, when using forward stepwise selection, mktval is selected in the 4-predictor model and remains selected until the end of the procedure. Likewise, when using backward stepwise selection, mktval is dropped in the 7-predictor model and remains dropped until the end of the procedure.

3. Lasso

Both Lasso and Ridge regression are Shrinkage methods. In standard linear regression, the fitting procedure is to estimate coefficients using the values that minimize RSS. Shrinkage methods is very alike to standard linear regression, except that the shrinkage methods coefficient estimates are the values that minimize RSS plus a shrinkage penalty. Shrinkage approach fits a model involving all predictors and then shrinks the estimated coefficients towards zero relative to the least squares estimates.

What's more, Lasso method can shrink the coefficients to be exactly zero, which means it can also perform variable selection. The lasso methods choose predictors lsales and lmktval. But the coefficients are different using all predictors and the subset of predictors.

Using all predictors, the Lasso model with λ (0.165) chosen by cross-validation is lsalary = 5.71 + 0.09 lsales + 0.03 lmktval.

```
## [1] 0.1651055

## (Intercept)      lsales     lmktval
##  5.70805171  0.08840962  0.03033290
```

Using the subset of predictors, the Lasso model with λ (0.199) chosen by cross-validation is lsalary = 5.93 + 0.07 lsales + 0.01 lmktval.

```
## [1] 0.1988701

## (Intercept)      lsales     lmktval
##  5.93319540  0.07481159  0.01318961
```

4. Ridge regression

Unlike the shrinkage penalty of Lasso, the penalty of ridge regression doesn't set any of the coefficients exactly to zero unless λ is infinite.

We use the built-in cross-validation function, cv.glmnet() to find best lambda. Then fit a ridge regression model on the full data set, using the value of best lambda. After examining the coefficient estimates, as expected none of the coefficients are zero, because ridge regression does not perform variable selection. We compare the coefficients of OLS with those of ridge regression and see the obvious shrinkage in coefficients both in the ridge regression model using all predictors and in that using the subset of predictors.

```
## 14 x 2 sparse Matrix of class "dgCMatrix"
##                       ols        ridge
## (Intercept)  4.708213e+00  6.071323e+00
## age          1.328814e-03  5.417041e-04
## college     -7.045880e-02 -3.866616e-02
## grad        -9.586387e-02  5.163685e-04
## comten      -1.377249e-03 -6.570441e-04
## ceoten       4.079807e-02  1.781824e-03
## sales       -7.955890e-06  4.741948e-06
## profits      9.099592e-05  6.595748e-05
## mktval       9.360869e-06  4.339035e-06
## lsales       1.963258e-01  3.170601e-02
## lmktval      6.005415e-02  3.317968e-02
## comtensq    -2.337184e-04 -2.019322e-05
## ceotensq    -9.006869e-04  2.305842e-05
## profmarg    -2.804013e-03 -4.396270e-04

## 9 x 2 sparse Matrix of class "dgCMatrix"
##                       ols        ridge
## (Intercept)  4.7236799400  5.7518289441
## age         -0.0003527064  0.0006934977
## college     -0.1062768977 -0.0543395727
## grad        -0.0769274857  0.0021664989
## comten      -0.0114526114 -0.0013055292
## ceoten       0.0165434200  0.0028575634
## lsales       0.1848123107  0.0544847616
## lmktval      0.1102081122  0.0606804258
## profmarg    -0.0021670791 -0.0006256393
```

c. Evaluate the predictions

1. The validation set approach

It involves randomly dividing the available set of observations into two parts, a training set and a validation set. The validation estimate of the test error rate can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set, in order to make our codes and reproducible, we set.seed(1) to obtain the same samples elements. Below is the evaluation of the predictions using all predictors.

```
##       ols      best   forward  backward     lasso     ridge
## 1 0.32777 0.2436412 0.2436412 0.2436412 0.2732201 0.2618947
```

Subset

Below is the evaluation of the predictions using the subset of predictors.

```
##         ols      best   forward  backward     lasso     ridge
## 1 0.3106509 0.2388287 0.2388287 0.2388287 0.2875902 0.2747204
```

2. K-fold cross-validation

This approach involves randomly dividing the set of observations into k groups, or folds, of approximately equal size. Moreover, the first fold is treated as a validation set, and the method is fit on the remaining k-1 folds. MSE is then computed on the observations in the held-out fold. Thanks to the use of function, the user can customize the selection of K-fold.

NOTE for function MSE: since we have already conducted the selection of the best predictors in part b using subset selection methods and lasso method, it is not necessary to conduct again a cross validation selection of the models. Hence, we regress lsalary only on the chosen variables.

Below is the evaluation of the predictions using all predictors.

```
##       ols      best   forward  backward     lasso     ridge
## 0.2419397 0.2315033 0.2315033 0.2315033 0.2276783 0.2471654
```

Subset

Below is the evaluation of the predictions using the subset of predictors.

```
##       ols      best   forward  backward     lasso     ridge
## 0.3325171 0.2243590 0.2243590 0.2243590 0.2252655 0.3170986
```

3. Leave-one-out cross-validation

LOOCV has a couple of major advantages over the validation set approach. First, it has far less bias. In LOOCV, we repeatedly fit the statistical learning method using training sets that contain n-1 observations, almost as many as are in the entire data set. Consequently, the LOOCV approach tends not to overestimate the test error rate as much as the validation set approach does. Moreover, performing LOOCV multiple times will always yield the same results: there is no randomness in the training/validation set splits.

However, the LOOCV returns higher MSEs than K-fold cross validation does, due to trade-off between Bias and variance of LOOCV. Indeed, the latest is averaging the outputs of n fitted models, each of which is trained on an almost identical set of observations, thus these outputs are highly (positively) correlated with each other.

"Since the mean of many highly correlated quantities has higher variance than does the mean of many quantities that are not as highly correlated, the test error estimate resulting from LOOCV tends to have higher variance than does the test error estimate resulting from k-fold CV." cit. Pag184 An Introduction to Statistical Learning

```
##       ols      best   forward  backward     lasso     ridge
## 0.3890592 0.2498705 0.2498705 0.2498705 0.2447510 0.3320376
```

Subset

Below is the evaluation of the predictions using the subset of predictors.

```
##       ols      best   forward  backward     lasso     ridge
## 0.3712127 0.2426087 0.2426087 0.2426087 0.2447510 0.3272610
```

Conclusion about MSEs Statistics: since it is a measure of the quality of an estimator-it is always non-negative, and values closer to zero are better. Hence, by comparing the results of the three validation methods on both the data frame "df" containing all predictors and the data frame "df2" including the 8 chosen predictors, subset selection methods and lasso got lower test set MSE than standard linear regression and ridge regression.

As we expected, subset selection methods and lasso can generally yield better prediction accuracy and model interpretability.

- By selecting the predictors or shrinking the estimated coefficients, we can often substantially reduce the variance at the cost of limited increase in bias. This can improve the accuracy in predicting the response for observations.
- Standard linear regression and ridge regression are not likely to generate any coefficient estimates that are exactly zero, making the model difficult to interpret. Instead, subset selection methods and lasso can automatically preform variable selection and make the model much easier to interpret.

d. Choose an additional prediction method

In this section, we conduct three different non-linear methods: polynomial, smooth spline and general additive model (GAM), and we will compare the results with the findings from linear models. We used both df and df2 data frames which are described in part 1 input variables.

According to the book, *an introduction to statistical learning*, it's a common practice to choose power up to four when performing polynomial regression. Hence, we conduct the polynomial regression on the key eight variables: age, ceoten, college, comten, grad, lmktval, lsalary, lsales, and profmarg. Surprisingly, the variable comten, and lsales, which play significant roles in linear models show very little influence in the polynomial regression, which might imply the characteristic of their linearity. From the first step of performing the polynomial regression on each individual variable, we confirm the suitable degree of freedom. Then, a simple validation set approach has been done on each

polynomial model. All the variables obtain relatively reasonable and low mean square errors with adjusted power. Lsales with the power to 4 gained the minimum value of 0.246.

```
## Call:
## lm(formula = lsalary ~ poly(lsales, 4), data = df2[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.20896 -0.36531 -0.05889  0.34640  1.46336
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.57619    0.05343 123.072  < 2e-16 ***
## poly(lsales, 4)1  3.00408    0.50692   5.926 6.43e-08 ***
## poly(lsales, 4)2 -0.32634    0.50692  -0.644    0.521
## poly(lsales, 4)3  0.63673    0.50692   1.256    0.213
## poly(lsales, 4)4 -0.28510    0.50692  -0.562    0.575
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5069 on 85 degrees of freedom
## Multiple R-squared:  0.3057, Adjusted R-squared:  0.273
## F-statistic: 9.357 on 4 and 85 DF,  p-value: 2.578e-06

## [1] 0.2458247
```

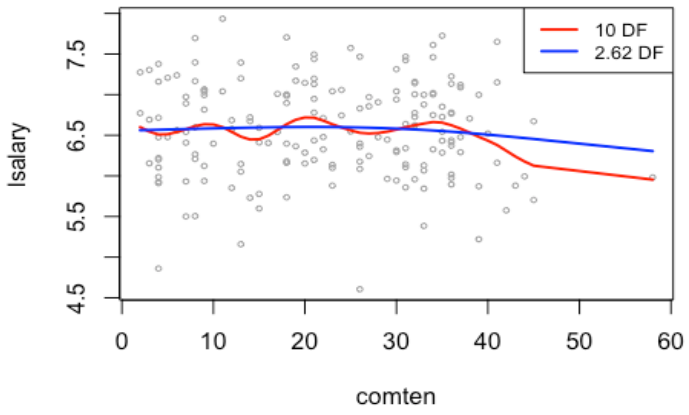
Secondly, we conduct the polynomial regression on combinations of the all key variables with power found in step one and test the models with/without dummy variables. However, after we perform a simple validation set approach, we discover the MSE of model increased dramatically.

```
## Call:
## lm(formula = lsalary ~ poly(age, 3) + poly(ceoten, 3) + poly(lsales,
##     1) + poly(lmktval, 1) + poly(profmarg, 2), data = df2[train,
##     ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.92960 -0.25562  0.00156  0.35530  0.87727
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         6.576185   0.045279 145.237  < 2e-16 ***
## poly(age, 3)1      -0.865903   0.491602  -1.761 0.082041 .
## poly(age, 3)2       0.677668   0.502958   1.347 0.181714
## poly(age, 3)3      -0.549393   0.451163  -1.218 0.226952
## poly(ceoten, 3)1    0.949306   0.507874   1.869 0.065304 .
## poly(ceoten, 3)2   -0.904187   0.455360  -1.986 0.050541 .
## poly(ceoten, 3)3   -0.507552   0.445354  -1.140 0.257871
## poly(lsales, 1)    -0.007818   0.975536  -0.008 0.993626
## poly(lmktval, 1)    3.232259   0.832738   3.881 0.000214 ***
## poly(profmarg, 2)1 -0.653667   0.441876  -1.479 0.143037
## poly(profmarg, 2)2 -2.545662   0.627191  -4.059 0.000115 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.4296 on 79 degrees of freedom
```

After the experiment of polynomial regression, we try the smooth spline model. For the degree of freedom, we first give a manipulated value, then we tested degree of freedom calculated by cross validation. The best model we got was based on the variable comten, and its MSE is 0.34.

```
## Call:
## smooth.spline(x = comten, y = lsalary, df = 10)
##
## Smoothing Parameter  spar= 0.5782737  lambda= 0.0002585595 (14 iterations)
## Equivalent Degrees of Freedom (Df): 10.00116
## Penalized Criterion (RSS): 11.90261
## GCV: 0.3640898

## Call:
## smooth.spline(x = comten, y = lsalary, cv = TRUE)
##
## Smoothing Parameter  spar= 0.9813928  lambda= 0.2115754 (13 iterations)
## Equivalent Degrees of Freedom (Df): 2.621658
## Penalized Criterion (RSS): 14.8547
## PRESS(l.o.o. CV): 0.3509798

## degree of freedom:

## [1] 2.621658

## MSE:

## [1] 0.3405307
```



Lastly, we performed the general additive model (GAM), with the application of natural spline and cubic spline. We find that with the same predictors, the GAM model with natural spline offer smaller MSE value. In addition, the model built by the variables: lsales, comten, lmktval, and profmarg, produces the best result in MSE (0.2292431) among all the other non-linear models.

```
## Call:
## gam(formula = lsalary ~ ns(lsales) + ns(comten) + ns(lmktval) +
##     ns(profmarg), data = df[train, ])
##
## Degrees of Freedom: 89 total; 85 Residual
## Residual Deviance: 20.68472
```

```
## [1] 0.2292431
```

e. Summarize the results

First of all, before we fit linear regressions, it is optimal to have a look at the data in order to identify and overcome possible problems such as collinearity and outlier. In our case, a single outlier causes a strong influence on the interpretation of the fitted model. We can also find the presence of collinearity causes a decline in the absolute value of the t-statistics. What's more, the test MSE of the predictions using the chosen subset of predictors are generally smaller than the test MSE of the predictions using all predictors, which also demonstrate the benefit of overcoming possible problems before fitting a linear model. In practice, there are some other problems such as correlation of error term, non-constant variance of error term, high-leverage points, which are not a problem in our case, so we don't elaborate in our report.

Secondly, regarding the comparison of different prediction methods, the non-linear model yields a more accurate prediction because the true relationship is generally not linear, in which linear models can't represent the real relationship very well. However, the non-linear model is more difficult to interpret. Since our task in this case is to predict the output variable salary instead of explaining how salary will change if the predictors change, we prefer the non-linear model. Otherwise, we would prefer the linear models chosen by subset selection methods or lasso method.

At last, we find that the three validation methods yield slightly different evaluation results due to the Bias-Variance Trade-off. In regards to bias reduction, LOOCV is the best one, followed by k-fold CV and validation set approach is the worst one, due to the different training set portion of the entire data. From the perspective of variance, LOOCV suffers highest variance. What's more, we also find it takes longer time to implement LOOCV than to implement the other two validation methods. Fortunately, there are only less than 200 observations in our data, otherwise it will be computational expensive. Therefore, k-fold CV using proper k can generate accurate estimates of test error rate that suffers neither high bias nor high variance.

Reference

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2015. An Introduction to Statistical Learning. Springer.

Kenneth Benoit. 2011: Linear Regression Models with Logarithmic Transformations

Shea, Justin M. 2018. Wooldridge: 111 Data Sets from “Introductory Econometrics: A Modern Approach, 6e” by Jeffrey M. Wooldridge. https://CRAN.R-project.org/package=wooldridge.