Data Analytics

Yong Zheng

Illinois Institute of Technology
Chicago, IL, 60616, USA

School of Applied Technology
ILLINOIS INSTITUTE OF TECHNOLOGY

Review: Steps in Multiple Linear Regression

Multiple Linear Regression

Important Steps in Multiple Linear Regression

- Data Splits – build a model based on train set, and evaluate it based on the test set; hold-out or N-fold cross validation
- Determine x and y, examine their linear relationships
- Build a multiple linear regression model by parameter estimates → build diff models by using feature selection
- Goodness of fit test
- Residual analysis – the last step to tell your model is qualified
- Interpret the performance of the training process
- Evaluations and predictions – evaluate it based on test set

Schedule

- Build regression models by using R based on Case Study 2
- Feature Selections

Case Study 2: Clerical Data

In any production process in which one or more workers are engaged in a variety of tasks, the total time spent in production varies as a function of the size of the work pool and the level of output of the various activities. For example variables in a large metropolitan department store, the number of hours worked (HOURS) per day by the clerical staff may depend on the following variables:

MAIL: number of pieces of mail processed (open, sort, etc.)
CERT: number of money orders and gift certificates sold
ACC: number of window payments (customer charge accounts) transacted
CHANGE: number of change order transactions processed
CHECK: number of checks cashed
MISC: number of pieces of miscellaneous mail processed on an “as available” basis
TICKETS: number of tickets sold.

The data for 52 working days are stored in the data file clerical.txt, attached to this assignment. The data set contains all the variables listed above and the variable DAY: day of the week (Mon, Tue, Wed, Thu, Fri and Sat) in the following order:
DAY, HOURS, MAIL, CERT, ACC, CHANGE, CHECK, MISC, TICKETS.

Steps of Regression Analysis

1) *Examine the scatterplot of the data.*
   - Does the relationship look linear?
   - Are there points in locations they shouldn't be?
   - Do we need a transformation?

```
mydata=read.table("clerical.txt",header=T)
hours=mydata$hours
mail=mydata$mail
cert=mydata$cert
acc=mydata$acc
change=mydata$change
check=mydata$check
misc=mydata$misc
tickets=mydata$tickets
```

Load data into R
And create variables to store factors

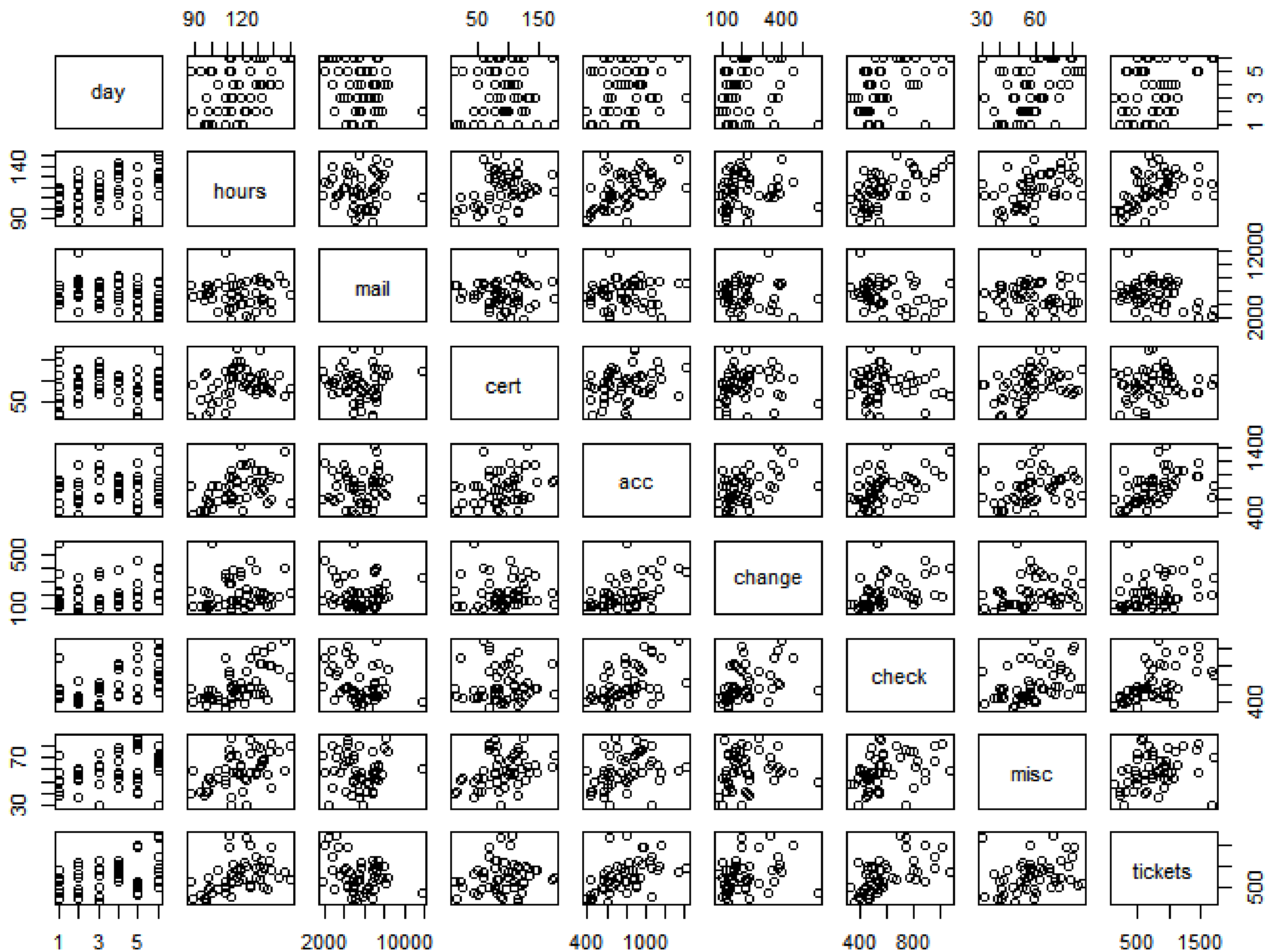
Steps of Regression Analysis

1) *Examine the scatterplot of the data.*
   - Does the relationship look linear?
   - Are there points in locations they shouldn't be?
   - Do we need a transformation?

We choose hours as dependent variable
And others (except DAY) as independent variables
We need to visualize them to see whether there are linear relationships or not

plot(mydata) ➔ this command will help you draw scatterplot of any pairs
plot(x, y) ➔ if you want to draw individual scatterplot

Steps of Regression Analysis



Steps of Regression Analysis

1) *Examine the scatterplot of the data.*
   - Does the relationship look linear?
   - Are there points in locations they shouldn't be?
   - Do we need a transformation?

Collinearity problem detections
If there are two factors with strong correlations, you should remove one of them.

- cor(hours, cert, method="pearson")
- cor(cbind(hours,cert,mail,acc,misc,check,change,tickets))

	hours	cert	mail	acc	misc	check	change	tickets
hours	1.000000000	0.29281923	-0.007650103	0.46151908	0.49901266	0.58731901	0.08479822	0.4495941
cert	0.292819235	1.00000000	0.011282017	0.24521511	0.33892441	-0.01588972	0.03686148	0.1222646
mail	-0.007650103	0.01128202	1.000000000	0.05480359	-0.01594041	-0.27658574	-0.04311752	-0.3117669
acc	0.461519082	0.24521511	0.054803588	1.00000000	0.34892016	0.50899367	0.47780716	0.5087885
misc	0.499012658	0.33892441	-0.015940412	0.34892016	1.00000000	0.38227195	0.16735176	0.2971547
check	0.587319010	-0.01588972	-0.276585736	0.50899367	0.38227195	1.00000000	0.44280516	0.5660733
change	0.084798217	0.03686148	-0.043117518	0.47780716	0.16735176	0.44280516	1.00000000	0.2750750
tickets	0.449594128	0.12226462	-0.311766861	0.50878854	0.29715473	0.56607326	0.27507497	1.0000000

Steps of Regression Analysis

1) *Examine the scatterplot of the data.*
   - Does the relationship look linear?
   - Are there points in locations they shouldn't be?
   - Do we need a transformation?

Try transformation on x variables. Decide to remove mail and use 1/change

```
> mail2=mail*mail
> cor(hours,mail2)
[1] 0.008174511
> logmail=log(mail)
> cor(hours,logmail)
[1] -0.02943603
> mail2=sqrt(mail)
> cor(hours,mail2)
[1] -0.01857279
> mail2=1/mail
> cor(hours,mail2)
[1] 0.04480373
```

```
> change2=change*change
> cor(hours, change2)
[1] 0.0204525
> change2=sqrt(change)
> cor(hours, change2)
[1] 0.1192057
> change2=log(change)
> cor(hours, change2)
[1] 0.1524671
> change2=1/change
> cor(hours, change2)
[1] -0.2069331
```

Steps of Regression Analysis

1) *Examine the scatterplot of the data.*
   – Does the relationship look linear?
   – Are there points in locations they shouldn't be?
   – Do we need a transformation?

Are you going to build the regression model in the next? You should choose an evaluation strategy (hold-out or N-fold cross validation) and split the data first, according to the size of the data.

For domo purpose, we simply use hold-out in this class. N-fold cross validation will be introduced in the future

Steps of Regression Analysis

1) *Split data.*
   – We use hold-out evaluation for example in the class

```
mydata[,"change2"]=change2
mydata=mydata[sample(nrow(mydata)),]
select.data = sample (1:nrow(mydata), 0.8*nrow(mydata))
train.data = mydata[select.data,]
test.data = mydata[-select.data,]
```

Add new variable to the data frame

Do not forget to shuffle the data

We use hold-out evaluation
For example. 80% as training

Next, we will build models on train.data, and evaluate the models on test.data

Steps of Regression Analysis

- Next step: build models
- Notes
  - Linear regression is one technique
  - Given a same technique, we can even build different models
  - One method is to build models by feature selections

Feature Selection Linear Regression

- Feature Selection
  - It refers to the process of selecting useful x variables to build the regression models
- Why we need feature selection?
  - Not all the x variables/features are useful
  - By using different x variables, you can build different models
  - Different model may have different performance

Feature Selection (FS) methods

Feature selection is a general process in data mining. It always has two components. We introduce FS for linear regression only.

1. The criteria for defining the best model
- The p-value in individual parameter test
- Akaike/Bayes Information Criterion (AIC/BIC)
- Optimize coefficient of determination R^2-adj
- Optimize Mallows' Cp Statistics
- Minimize PRESS statistic (a metric similar to errors)

2. The search or rank methods
- For example, forward selection, backward elimination, best subset, stepwise

Criteria for Feature Selection (FS)

The criteria for defining the best model

- The p-value in individual parameter test
- Akaike/Bayes Information Criterion (AIC/BIC)
- Optimize coefficient of determination R^2-adj

Criteria for Feature Selection (FS)

The criteria for defining the best model

- The p-value in individual parameter test

```
> m1=lm(hours~cert+acc+misc+check+change2+tickets, data=train.data)
> summary(m1)

Call:
lm(formula = hours ~ cert + acc + misc + check + change2 + tickets,
    data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-18.3053  -8.6417  -0.9159   7.6568  23.8369

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.718e+01  1.068e+01   5.352 5.99e-06 ***
cert        1.291e-01  6.451e-02   2.002  0.05331 .
acc         2.123e-03  1.195e-02   0.178  0.86006
misc        1.964e-01  1.616e-01   1.216  0.23244
check       4.178e-02  1.526e-02   2.739  0.00975 **
change2     1.450e+03  8.818e+02   1.644  0.10930
tickets     2.424e-03  6.328e-03   0.383  0.70406
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.15 on 34 degrees of freedom
Multiple R-squared:  0.5277,    Adjusted R-squared:  0.4443
F-statistic:  6.33 on 6 and 34 DF,  p-value: 0.0001525
```

Criteria for Feature Selection (FS)

The criteria for defining the best model

- Optimize coefficient of determination R^2-adj

```
> ml=lm(hours~cert+acc+misc+check+change2+tickets, data=train.data)
> summary(ml)

Call:
lm(formula = hours ~ cert + acc + misc + check + change2 + tickets,
    data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-18.3053  -8.6417  -0.9159   7.6568  23.8369

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.718e+01  1.068e+01   5.352 5.99e-06 ***
cert        1.291e-01  6.451e-02   2.002  0.05331 .
acc         2.123e-03  1.195e-02   0.178  0.86006
misc        1.964e-01  1.616e-01   1.216  0.23244
check       4.178e-02  1.526e-02   2.739  0.00975 **
change2     1.450e+03  8.818e+02   1.644  0.10930
tickets     2.424e-03  6.328e-03   0.383  0.70406
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.15 on 34 degrees of freedom
Multiple R-squared:  0.5277,    Adjusted R-squared:  0.4443
F-statistic:  6.33 on 6 and 34 DF,  p-value: 0.0001525
```

Criteria for Feature Selection (FS)

The criteria for defining the best model

- Akaike/Bayes Information Criterion (AIC/BIC)

The *AIC* criterion is defined for a large class of models fit by maximum likelihood:

$$\mathrm{AIC} = -2 \log L + 2 \cdot d$$

where L is the maximized value of the likelihood function for the estimated model.

$$\mathrm{BIC} = \frac{1}{n}\left(\mathrm{RSS} + \log(n) d \hat{\sigma}^2\right).$$

Like C_p, the BIC will tend to take on a small value for a model with a low test error, and so generally we select the model that has the lowest BIC value.

We want to
Minimize AIC/BIC

Criteria for Feature Selection (FS)

The criteria for defining the best model

- The p-value in individual parameter test
  - Indicate x variable is useful or not
- Akaike/Bayes Information Criterion (AIC/BIC)
  - We want to minimize AIC/BIC
- Optimize coefficient of determination R^2-adj
  - We want to maximize Adj-R2, but there may be overfitting problem

Search algorithms – K independent variables

- **Backward elimination**
  - Start with the full model and eliminates one variable at the time until a reasonable candidate regression model is found. It typically uses a criterion based on the goodness-of-fit F-test.

- **Forward selection**
  - Start with the empty model, to add variables one by one, grow the model and select the best model finally

- **Best subset regression:**
  - Computer prints a listing of the best regression equations with 1, 2, 3,…k-1 independent X-variables. It selects the "best" model at each step (for instance the model with highest R^2-adj, or lowest PRESS statistics) It stops when there is no further improvement!

- **Stepwise regression**
  - The combination of backward and forward approaches.

Remarks for model selection

- **There is no unique optimal model** or subset of independent x-variables.
- **Different search algorithms may give different results**. Always run diagnostic methods to check that the model found by the selection method is appropriate and the assumptions are satisfied.

About Backward Elimination

We can use two methods in backward elimination

- **Backward by using p-value as metric**

  In this case, we look at the p-value in the individual parameter tests, and remove x variables one by one. Each time, we remove the x variable with largest p-value ➔ manual process

- **Backward by using AIC/BIC as metric**

  In this case, R will iterate the process and automatically give you the final results by minimizing AIC/BIC metrics ➔ automatic process

Backward Elimination by Using P-value

```
> m1=lm(hours~cert+acc+misc+check+change2+tickets, data=train.data)
> summary(m1)

Call:
lm(formula = hours ~ cert + acc + misc + check + change2 + tickets,
    data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-18.3053  -8.6417  -0.9159   7.6568  23.8369

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.718e+01  1.068e+01   5.352 5.99e-06 ***
cert        1.291e-01  6.451e-02   2.002  0.05331 .
acc         2.123e-03  1.195e-02   0.178  0.86006
misc        1.964e-01  1.616e-01   1.216  0.23244
check       4.178e-02  1.526e-02   2.739  0.00975 **
change2     1.450e+03  8.818e+02   1.644  0.10930
tickets     2.424e-03  6.328e-03   0.383  0.70406
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.15 on 34 degrees of freedom
Multiple R-squared:  0.5277,    Adjusted R-squared:  0.4443
F-statistic:  6.33 on 6 and 34 DF,  p-value: 0.0001525
```

Assume we use 95% as the confidence or significance level

Backward Elimination by Using P-value

```
> m1=lm(hours~cert+misc+check+change2+tickets, data=train.data)
> summary(m1)

Call:
lm(formula = hours ~ cert + misc + check + change2 + tickets,
    data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-18.3575  -8.7982  -0.5318   8.2977  23.5608 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.731e+01  1.051e+01   5.452 4.08e-06 ***
cert        1.310e-01  6.277e-02   2.087  0.04422 *  
misc        2.000e-01  1.581e-01   1.265  0.21435    
check       4.279e-02  1.395e-02   3.067  0.00416 ** 
change2     1.498e+03  8.274e+02   1.811  0.07879 .  
tickets     2.711e-03  6.034e-03   0.449  0.65602    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.98 on 35 degrees of freedom
Multiple R-squared:  0.5272,    Adjusted R-squared:  0.4597 
F-statistic: 7.806 on 5 and 35 DF,  p-value: 5.088e-05
```

Backward Elimination by Using P-value

```
> m1=lm(hours~cert+misc+check+change2, data=train.data)
> summary(m1)

Call:
lm(formula = hours ~ cert + misc + check + change2, data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-17.3299  -8.3910  -0.0577   7.2402  23.3587 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.674e+01  1.032e+01   5.498 3.26e-06 ***
cert        1.352e-01  6.137e-02   2.203  0.03405 *  
misc        2.036e-01  1.561e-01   1.304  0.20052    
check       4.632e-02  1.140e-02   4.064  0.00025 ***
change2     1.493e+03  8.181e+02   1.825  0.07629 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.85 on 36 degrees of freedom
Multiple R-squared:  0.5245,	Adjusted R-squared:  0.4717 
F-statistic: 9.927 on 4 and 36 DF,  p-value: 1.613e-05
```

Backward Elimination by Using P-value

```
> m1=lm(hours~cert+check+change2, data=train.data)
> summary(m1)

Call:
lm(formula = hours ~ cert + check + change2, data = train.data)

Residuals:
    Min      1Q  Median      3Q     Max
-21.078  -9.109  -2.168   8.592  26.516

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.157e+01  9.724e+00   6.332 2.24e-07 ***
cert        1.708e-01  5.548e-02   3.079   0.0039 **
check       5.354e-02  1.006e-02   5.323 5.19e-06 ***
change2     1.399e+03  8.226e+02   1.701   0.0973 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.96 on 37 degrees of freedom
Multiple R-squared:  0.502,     Adjusted R-squared:  0.4617
F-statistic: 12.43 on 3 and 37 DF,  p-value: 8.995e-06
```

Backward Elimination by Using P-value

```
> ml=lm(hours~cert+check, data=train.data)
> summary(ml)

Call:
lm(formula = hours ~ cert + check, data = train.data)

Residuals:
    Min      1Q  Median      3Q     Max
-18.490  -8.450  -2.353   7.601  29.782

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 70.32584    8.45434   8.318 4.38e-10 ***
cert         0.15526    0.05607   2.769  0.00864 **
check        0.05477    0.01028   5.327 4.76e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.25 on 38 degrees of freedom
Multiple R-squared:  0.4631,    Adjusted R-squared:  0.4348
F-statistic: 16.39 on 2 and 38 DF,  p-value: 7.383e-06
```

Backward Elimination by Using P-value

Why we have to remove them one by one?

It is because there may be correlations among x variables. If you remove all of them, probably you also remove some variables which are still useful

Steps of Regression Analysis

Next, you must validate this model is qualified or not by

1), examine F-test

2), perform residual analysis

I ignore these steps, and continue to introduce other feature selection techniques
However, you must validate a model is qualified or not after feature selections

Model selection in R

- Best subset model selection methods (adjR2, Cp) are computed using function `leaps()` in package `leaps`
- Stepwise model selection methods (backward, forward or stepwise) are applied using `step()`
- We will show examples later

Steps of Regression Analysis

2) Build a regression model based on <u>Backward Eliminations by AIC</u>

To realize the backward elimination, you can also use the function step()

step(full, direction="backward", trace=T)

<u>full is the full regression model that adopts all of the x variables</u>

set trace = True or False, can help you track the steps in Backward Elimination

```
> full=lm(hours~cert+acc+check+misc+change2+tickets,data=train.data)
```

Note:

The previous way – we manually drop x variables step by step, is based on the p-value in the individual parameter test.

However, the step function above, will use AIC as the metric to drop x variables.

In this case, you may get a different model by using the step() function. You do not need to drop x variables if the p-value in t-test is larger than alpha, since we no longer use p-value as metrics

```
> full=lm(hours~cert+acc+check+misc+change2+tickets,data=train.data)
> m2=step(full, direction="backward", trace=T)
Start:  AIC=211.11
hours ~ cert + acc + check + misc + change2 + tickets

          Df Sum of Sq    RSS    AIC
- acc      1      4.66 5024.3 209.15
- tickets  1     21.66 5041.3 209.29
- misc     1    218.23 5237.8 210.85
<none>                 5019.6 211.11
- change2  1    399.24 5418.9 212.25
- cert     1    591.75 5611.4 213.68
- check    1   1107.19 6126.8 217.28

Step:  AIC=209.15
hours ~ cert + check + misc + change2 + tickets

          Df Sum of Sq    RSS    AIC
- tickets  1     28.97 5053.2 207.38
- misc     1    229.59 5253.9 208.98
<none>                 5024.3 209.15
- change2  1    470.62 5494.9 210.82
- cert     1    625.33 5649.6 211.96
- check    1   1350.14 6374.4 216.91

Step:  AIC=207.38
hours ~ cert + check + misc + change2

          Df Sum of Sq    RSS    AIC
- misc     1    238.68 5291.9 207.28
<none>                 5053.2 207.38
- change2  1    467.57 5520.8 209.01
- cert     1    681.47 5734.7 210.57
- check    1   2317.81 7371.1 220.86

Step:  AIC=207.27
```

The final model they got.
It is different from the model we got by step-by-step dropping x variables.
It is because they try to minimize the AIC criterion

```
Step:  AIC=207.27
hours ~ cert + check + change2

          Df Sum of Sq    RSS    AIC
<none>                 5291.9 207.28
- change2  1     413.9 5705.8 208.36
- cert     1    1355.7 6647.6 214.63
- check    1    4052.0 9344.0 228.59
```

Steps of Regression Analysis

2) Build a regression model based on Backward Eliminations by AIC

I record this model, m2 = model built by step() using Backward Elimination, adj-R2 = 46.17%

```
> m2=lm(hours~cert+check+change2,data=train.data)
> summary(m2)

Call:
lm(formula = hours ~ cert + check + change2, data = train.data)

Residuals:
    Min      1Q  Median      3Q     Max 
-21.078  -9.109  -2.168   8.592  26.516 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 6.157e+01  9.724e+00   6.332 2.24e-07 ***
cert        1.708e-01  5.548e-02   3.079   0.0039 ** 
check       5.354e-02  1.006e-02   5.323 5.19e-06 ***
change2     1.399e+03  8.226e+02   1.701   0.0973 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.96 on 37 degrees of freedom
Multiple R-squared:  0.502,     Adjusted R-squared:  0.4617 
F-statistic: 12.43 on 3 and 37 DF,  p-value: 8.995e-06
```

Even if p-value is larger than 0.05, But we will NOT Remove it, since we use AIC/BIC as metric this time

Steps of Regression Analysis

3) Build a regression model based on Stepwise Regression

You can also use the function step() to realize stepwise regression

step(Base, scope=list(upper=Full, lower=~1), direction="forward", trace=F)

full is the full regression model that adopts all of the x variables

Base is the model you start from, simply you can build a model with one x variable

```
> base=lm(hours~check,data=train.data)
> step(base, scope=list(upper=full, lower=~1), direction="forward", trace=F)

Call:
lm(formula = hours ~ check + cert + change2, data = train.data)

Coefficients:
(Intercept)        check         cert      change2
  6.157e+01    5.354e-02    1.708e-01    1.399e+03
```

forward model
Set it as "forward"

We can observe that it produces a model as same as the model m2

```
> step(base, scope=list(upper=full, lower=~1), direction="both", trace=F)

Call:
lm(formula = hours ~ check + cert + change2, data = train.data)

Coefficients:
(Intercept)        check         cert      change2
  6.157e+01    5.354e-02    1.708e-01    1.399e+03
```

Stepwise model
Set it as "both"

Steps of Regression Analysis

4) Build a regression model based on Best Subset regression

In this approach, it tries to find the best subset of x variables by selected metric.

The metric you can choose could be Cp, R2 or Adj-R2

leaps(y=train.data[,2],x=train.data[,cbind(4,5,7,8,9,10)],names=names(train.data[,cbind(4,5,7,8,9,10)]),method="adjr2")   Note: you need to install and use the package "leaps"

```
> leaps(y=train.data[,2],x=train.data[,cbind(4,5,7,8,9,10)],names=names(train.data[,cbind(4,5,7,8,9,10)]),method="adjr2")
$which
   cert   acc check  misc tickets change2
1 FALSE FALSE  TRUE FALSE   FALSE   FALSE
1 FALSE  TRUE FALSE FALSE   FALSE   FALSE
1 FALSE FALSE FALSE  TRUE   FALSE   FALSE
1 FALSE FALSE FALSE FALSE    TRUE   FALSE
1  TRUE FALSE FALSE FALSE   FALSE   FALSE
1 FALSE FALSE FALSE FALSE   FALSE    TRUE
2  TRUE FALSE  TRUE FALSE   FALSE   FALSE
2 FALSE FALSE  TRUE  TRUE   FALSE   FALSE
2 FALSE  TRUE  TRUE FALSE   FALSE   FALSE
2 FALSE FALSE  TRUE FALSE   FALSE    TRUE
2 FALSE FALSE  TRUE FALSE    TRUE   FALSE
2 FALSE  TRUE FALSE  TRUE   FALSE   FALSE
```

Steps of Regression Analysis

4) Build a regression model based on Best Subset regression

```
$adjr2
 [1] 0.33819111 0.23086466 0.21292343 0.19054238 0.03807570 0.01329711 0.43482378
[16] 0.25275293 0.46165303 0.43836769 0.43187935 0.42297279 0.41660802 0.39827956
[31] 0.42547704 0.41718146 0.40930131 0.40794547 0.38462284 0.34166708 0.45967422
```

The largest adj-R2 we can observe is 0.4717.

The corresponding model is to use cert, check, misc, change2

```
> m3=lm(hours~cert+check+misc+change2,data=train.data)
> summary(m3)

Call:
lm(formula = hours ~ cert + check + misc + change2, data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-17.3299  -8.3910  -0.0577   7.2402  23.3587

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.674e+01  1.032e+01   5.498 3.26e-06 ***
cert        1.352e-01  6.137e-02   2.203  0.03405 *
check       4.632e-02  1.140e-02   4.064  0.00025 ***
misc        2.036e-01  1.561e-01   1.304  0.20052
change2     1.493e+03  8.181e+02   1.825  0.07629 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.85 on 36 degrees of freedom
Multiple R-squared:  0.5245,    Adjusted R-squared:  0.4717
F-statistic: 9.927 on 4 and 36 DF,  p-value: 1.613e-05
```

Remarks for model selection

- **There is no unique optimal model** or subset of independent x-variables.

- **Different search algorithms may give different results**. Always run diagnostic methods to check that the model found by the selection method is appropriate and the assumptions are satisfied.

Steps of Regression Analysis

- So, currently we get three models

M1 = Backward selection by manually drop x based on p-value
M2 = Backward and Forward selection by step() based on AIC
M3 = Best Subset selection by adjr2

Of course, the next steps, you need to further examine residual analysis to validate they are qualified models or not

Adj-R2:
M1, 43.48%
M2, 46.17%
M3, 47.17%

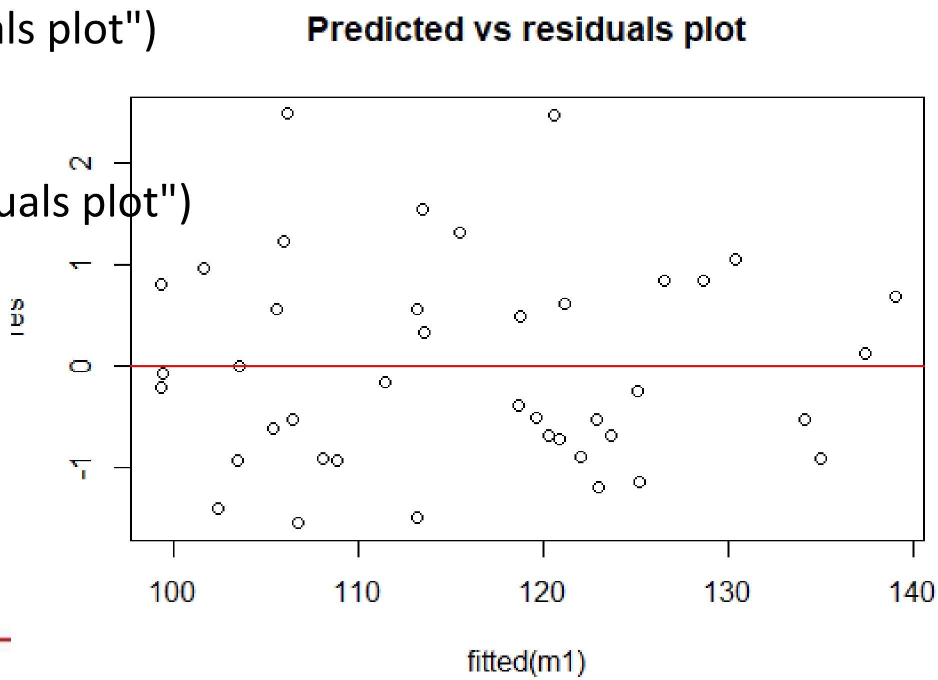
Residual Analysis

Standardized residual vs predicted values

```
res=rstandard(m1)
plot( fitted(m1), res, main="Predicted vs residuals plot")
abline(a=0, b=0, col='red')
```

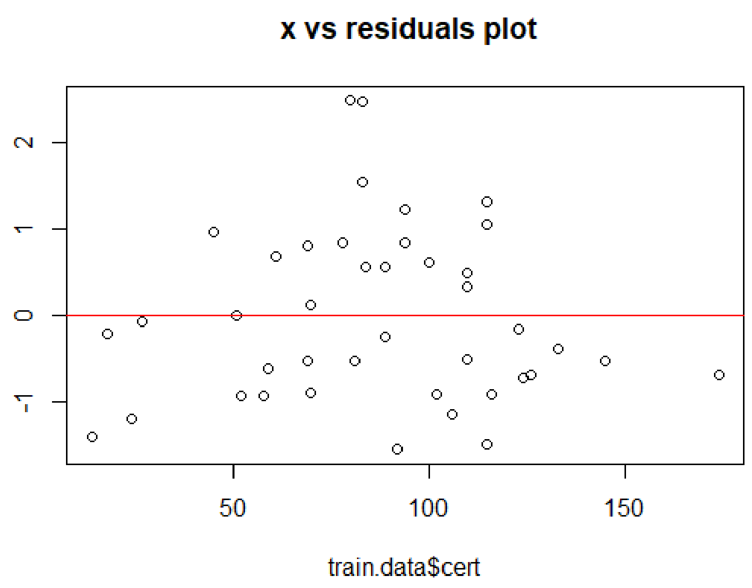
Standardized residual vs x variables

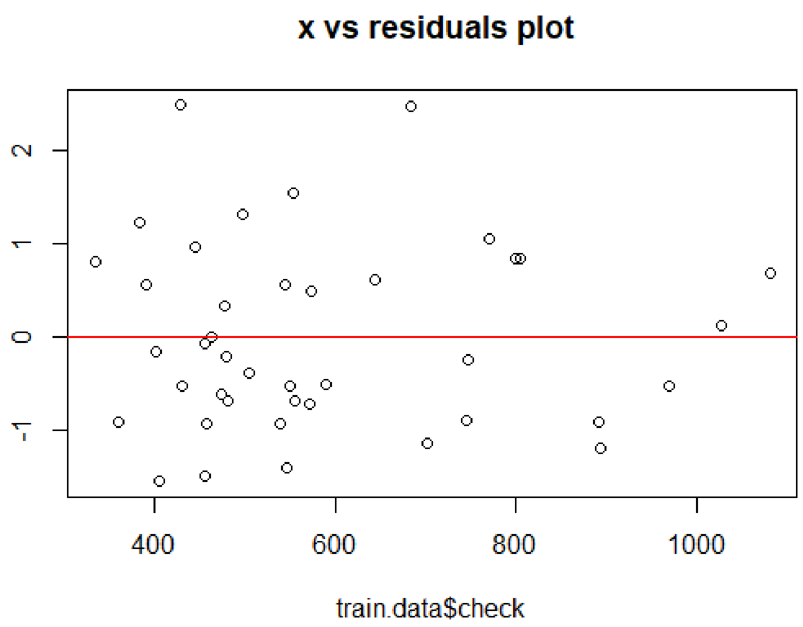
```
plot(train.data$cert, res, main=" x vs residuals plot")
abline(a=0, b=0,col='red')

plot(train.data$check, res, main=" x vs residuals plot")
abline(a=0, b=0,col='red')
```



Residual Analysis

Standardized residual vs x variables



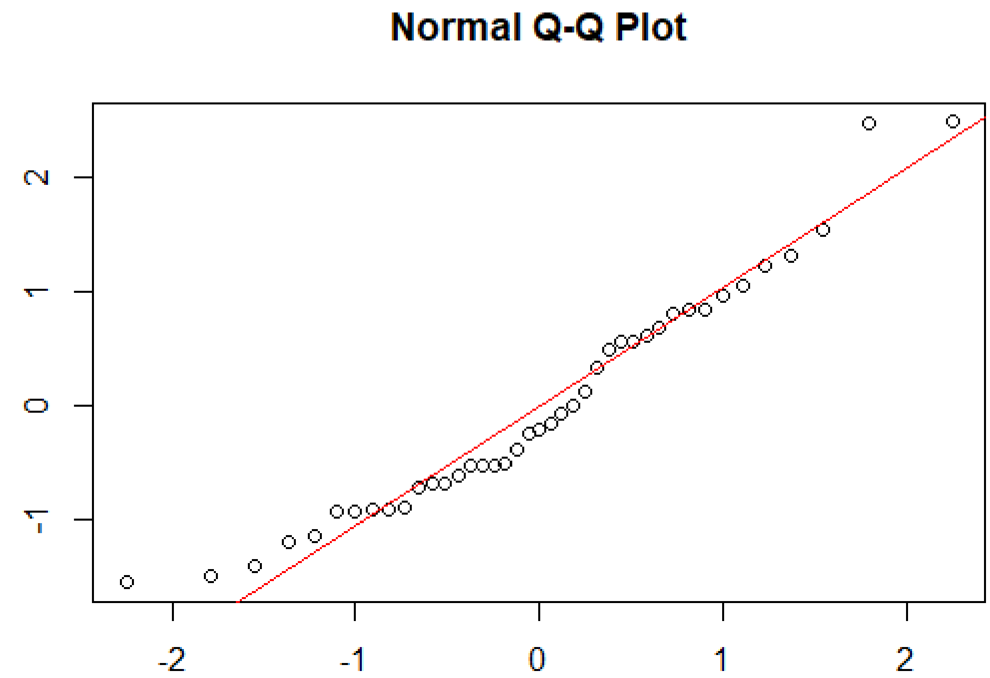


Residual Analysis

Examine residuals are normal or not

```
qqnorm(res)
qqline(res,col=2)

shapiro.test(res)
```

```
> shapiro.test(res)

        Shapiro-Wilk normality test

data:  res
W = 0.95003, p-value = 0.07016
```



Steps of Regression Analysis

- So, currently we get three models

M1 = Backward selection by manually drop x based on p-value
M2 = Backward and Forward selection by step() based on AIC
M3 = Best Subset selection by adjr2

Of course, the next steps, you need to further examine residual analysis to validate they are qualified models or not

Adj-R2:
M1, 43.48%
M2, 46.17%
M3, 47.17%

Measuring predictive performance

Root Mean Square Error :

Best model minimizes RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m} (y_i - \hat{y}_i)^2}{m}}$$

Mean Absolute Error

Best model minimizes MAE

$$MAE = \frac{\sum_{i=1}^{m} | y_i - \hat{y}_i |}{m}$$

Measuring predictive performance

Root Mean Square Error :

Best model minimizes RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{m}}$$

- Using training set to fit model: fit = lm (y~var1+var2, na.action = na.omit, data=train.data);
- Created fitted values using testing data: y_pred=predict.glm(fit, test.data), and y_obs=test.data[,"the name of y-variable"];
- Compute prediction error RMSE: rmse_model1=sqrt((y_obs - y_pred)%*%(y_obs-y_pred))/nrow(test.data)

Steps of Regression Analysis

```
y1=predict.glm(m1,test.data)
y2=predict.glm(m2,test.data)
y3=predict.glm(m3,test.data)
y=test.data[,2]
rmse_1 = sqrt((y-y1)%*%(y-y1))/nrow(test.data)
rmse_2 = sqrt((y-y2)%*%(y-y2))/nrow(test.data)
rmse_3 = sqrt((y-y3)%*%(y-y3))/nrow(test.data)
```

```
> rmse_1
         [,1]
[1,] 3.31041
> rmse_2
          [,1]
[1,] 3.527102
> rmse_3
          [,1]
[1,] 3.070438
```

	Adj-R2	RMSE
M1	43.48%	3.31
M2	46.17%	3.53
M3	47.17%	3.07

Write down and Explain the best model

Y = 56.7 + 0.1352 x1 + 0.04632 x2
+ 0.2036 x3 + 1493 x4

Y = hours

X1 = cert

X2 = check

X3 = misc

X4 = 1/change

	Adj-R2	RMSE
M1	43.48%	3.31
M2	46.17%	3.53
M3	47.17%	3.07

```
> summary(m3)

Call:
lm(formula = hours ~ cert + check + misc + change2, data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-17.3299  -8.3910  -0.0577   7.2402  23.3587

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.674e+01  1.032e+01   5.498 3.26e-06 ***
cert        1.352e-01  6.137e-02   2.203  0.03405 *
check       4.632e-02  1.140e-02   4.064  0.00025 ***
misc        2.036e-01  1.561e-01   1.304  0.20052
change2     1.493e+03  8.181e+02   1.825  0.07629 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.85 on 36 degrees of freedom
Multiple R-squared:  0.5245,    Adjusted R-squared:  0.4717
F-statistic: 9.927 on 4 and 36 DF,  p-value: 1.613e-05
```

Case Study 2: Clerical Data

In any production process in which one or more workers are engaged in a variety of tasks, the total time spent in production varies as a function of the size of the work pool and the level of output of the various activities. For example variables in a large metropolitan department store, the number of hours worked (HOURS) per day by the clerical staff may depend on the following variables:

MAIL: number of pieces of mail processed (open, sort, etc.)
CERT: number of money orders and gift certificates sold
ACC: number of window payments (customer charge accounts) transacted
CHANGE: number of change order transactions processed
CHECK: number of checks cashed
MISC: number of pieces of miscellaneous mail processed on an "as available" basis
TICKETS: number of tickets sold.

The data for 52 working days are stored in the data file clerical.txt, attached to this assignment. The data set contains all the variables listed above and the variable DAY: day of the week (Mon, Tue, Wed, Thu, Fri and Sat) in the following order:
DAY, HOURS, MAIL, CERT, ACC, CHANGE, CHECK, MISC, TICKETS.

Multiple Linear Regression

Important Steps in Multiple Linear Regression

- Data Splits – build a model based on train set, and evaluate it based on the test set
- Determine linear relationship between y and x variables
- Build a multiple linear regression model by parameter estimates ➔ decide feature selection methods at here!!!
- Goodness of fit test
- Residual analysis – the last step to tell your model is qualified
- Interpret the performance of the training process
- Evaluations and predictions – evaluate it based on test set