

3.2.1 - R: Variable Selection

Stat 5100: Dr. Bean

Example: (Textbook tables 9.1 & 9.5) A hospital surgical unit was interested in predicting survival time for patients who undergo a particular liver operation. Data are reported for 108 patients on the following variables: blood-clotting score, prognostic index, enzyme function test score, liver function test score, age (in years), gender (0=male, 1=female), indicators of alcohol use (none, moderate, heavy), and survival time (in days). Which (if any) of these predictors should be used in a linear model?

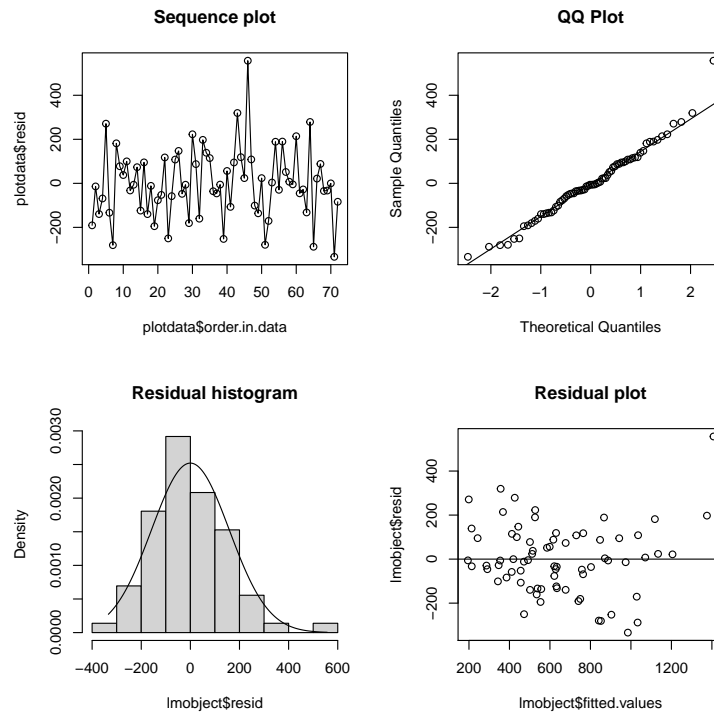
Create train/test data and check assumptions

```
# Set a random seed so that results are reproducible
set.seed(2341)

# Load the data
library(stat5100)
data(surgical)

# We commonly will make our test set be 1/3 of the data, and have our training
# set be the other 2/3 of the data. There are a variety of ways to randomly
# split up the data this way, here is one efficient way to do it:
n <- nrow(surgical)
train_index <- sample(1:n, size = (2/3)*n)
train_surgical <- surgical[train_index, ]
test_surgical <- surgical[-train_index, ]

# Check initial assumptions using the training data
surgical_train_lm <- lm(Time ~ bloodclot + prognostic + enzyme + liver,
                        data = train_surgical)
stat5100::visual_assumptions(surgical_train_lm)
```



```
stat5100::brown_forsythe_lm(surgical_train_lm)

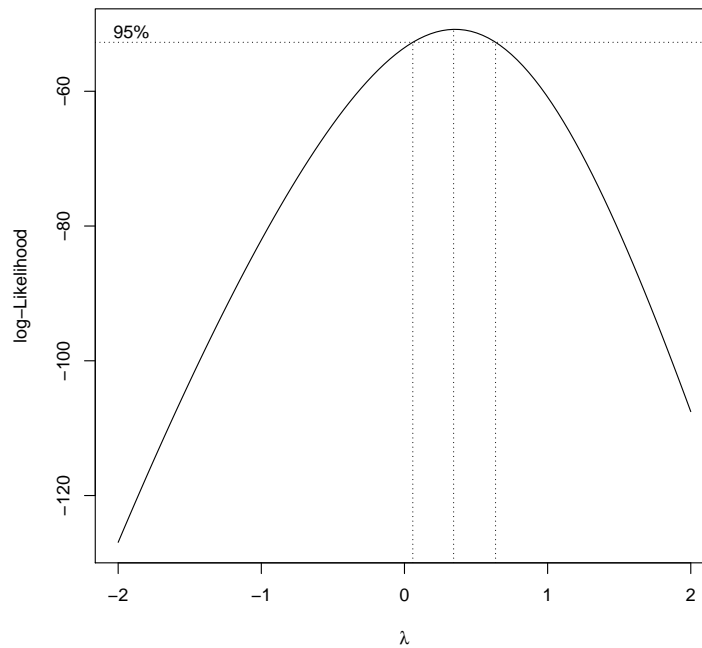
## [1] "Brown-forsythe test for constant variance in the residuals:"
## [1] "T-statistic: -0.8723, p-value: 0.386"

stat5100::cor_normality_lm(surgical_train_lm)

## Correlation test of normality:
##           resid expected_norm
## resid       1.0000000    0.9868459
## expected_norm 0.9868459    1.0000000
##
## Total observations: 72
## Make sure to consult with table B.6 for your final result.
```

Consider a possible transformation on the response

```
MASS::boxcox(surgical_train_lm)
```



```
# Make a log-transform (Make sure to transform on both training and testing data)
train_surgical <- cbind(train_surgical, logTime = log(train_surgical$Time))
test_surgical <- cbind(test_surgical, logTime = log(test_surgical$Time))

# Fit a new log model
surgical_logtrain_lm <- lm(logTime ~ bloodclot + prognostic + enzyme + liver,
                           data = train_surgical)
```

Perform variable selection with R^2 , adjusted R^2 , or Mallow's C_p :

Note that the output below shows results for *all* possible combinations of variables in the model.

```
olsrr::ols_step_all_possible(surgical_logtrain_lm)
```

##	Index	N	Predictors	R-Square	Adj. R-Square
## 3	1	1	enzyme	0.40896416	0.400520790
## 4	2	1	liver	0.36064018	0.351506464
## 2	3	1	prognostic	0.28135339	0.271087005
## 1	4	1	bloodclot	0.00904029	-0.005116277
## 8	5	2	prognostic enzyme	0.69553296	0.686707831
## 10	6	2	enzyme liver	0.51490091	0.500840063
## 9	7	2	prognostic liver	0.47040155	0.455050871
## 6	8	2	bloodclot enzyme	0.45254253	0.436674193
## 7	9	2	bloodclot liver	0.36400730	0.345572731
## 5	10	2	bloodclot prognostic	0.28847685	0.267852989
## 11	11	3	bloodclot prognostic enzyme	0.73495360	0.723260372
## 14	12	3	prognostic enzyme liver	0.70644450	0.693493521
## 13	13	3	bloodclot enzyme liver	0.52249652	0.501430188
## 12	14	3	bloodclot prognostic liver	0.47137051	0.448048616
## 15	15	4	bloodclot prognostic enzyme liver	0.73540593	0.719609269
##	Mallow's Cp				

```
## 3      81.660955
## 4      93.897461
## 2     113.974309
## 1     182.928906
## 8      11.096556
## 10     56.835857
## 9      68.103897
## 6      72.626125
## 7      95.044844
## 5     114.170520
## 11      3.114539
## 14     10.333558
## 13     56.912510
## 12     69.858541
## 15      5.000000
```

Perform variable selection with

Note that the output below shows results for only a few different model choices. This function from the “olsrr” package will show more information criteria (including SBC, AIC, and more that we don’t talk about in our class) but it will not show every single possible variable combination like the last section. On the second table, each of the result rows refers to a specific model number, which you can reference with the first table in the output.

```
olsrr::ols_step_best_subset(surgical_logtrain_lm)
```

```
##           Best Subsets Regression
## -----
## Model Index   Predictors
## -----
##      1         enzyme
##      2      prognostic enzyme
##      3    bloodclot prognostic enzyme
##      4    bloodclot prognostic enzyme liver
## -----
```

```
##
##                               Subsets Regression Summary
## -----
```

## Model	R-Square	Adj. R-Square	Pred R-Square	C(p)	AIC	SBIC	SBC	MSEP
## 1	0.4090	0.4005	0.3619	81.6610	67.3344	-139.6069	74.1644	10.1639
## 2	0.6955	0.6867	0.6658	11.0966	21.5758	-183.1567	30.6825	5.3128
## 3	0.7350	0.7233	0.6948	3.1145	13.5925	-190.1630	24.9758	4.6940
## 4	0.7354	0.7196	0.6851	5.0000	15.4695	-188.1226	29.1295	4.7570

```
## -----
```

```
## AIC: Akaike Information Criteria
## SBIC: Sawa's Bayesian Information Criteria
## SBC: Schwarz Bayesian Criteria
## MSEP: Estimated error of prediction, assuming multivariate normality
## FPE: Final Prediction Error
## HSP: Hocking's Sp
## APC: Amemiya Prediction Criteria
```

Perform backward variable selection

In the function below, use the option “prem” to specify the p-value that must be met for a variable to be taken out of the model.

```
olsrr::ols_step_backward_p(surgical_logtrain_lm, prem = 0.10)

##
##
## Elimination Summary
## -----
##      Variable      Adj.      C(p)      AIC      RMSE
## Step  Removed  R-Square  R-Square
## -----
##    1   liver      0.735    0.7233    3.1145    13.5925    0.2553
## -----
```

In the output above, because no variables are removed, this tells us that we will want to include all variables in the model.

Perform forward variable selection

In the function below, use the option “penter” to specify the p-value that must be met for a variable to be included in the output.

```
olsrr::ols_step_forward_p(surgical_logtrain_lm, penter = 0.10)

##
##
## Selection Summary
## -----
##      Variable      Adj.      C(p)      AIC      RMSE
## Step  Entered  R-Square  R-Square
## -----
##    1   enzyme      0.4090    0.4005    81.6610    67.3344    0.3757
##    2  prognostic    0.6955    0.6867    11.0966    21.5758    0.2716
##    3  bloodclot     0.7350    0.7233     3.1145    13.5925    0.2553
## -----
```

In the output above, notice that all four variables enter the model (because there are four steps), which tells us that the optimal model should include all four predictor variables.

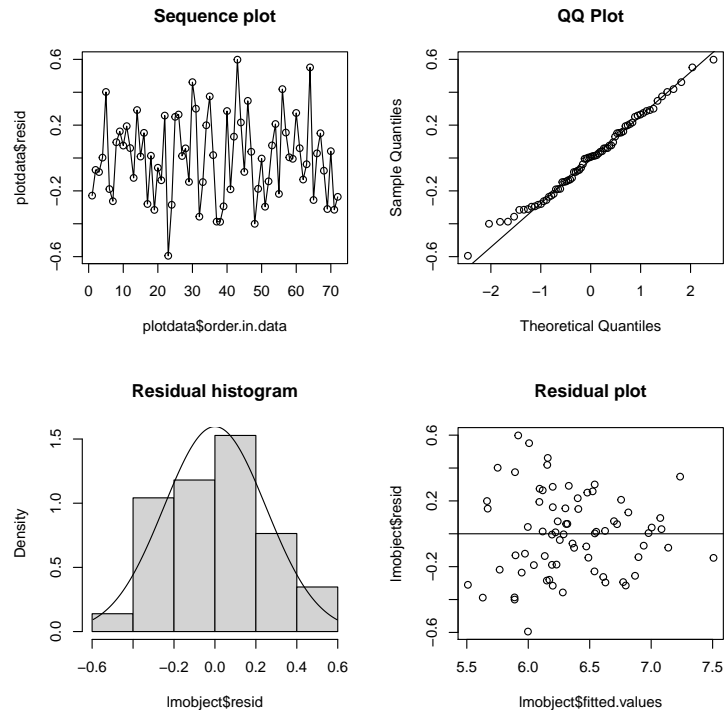
Perform hybrid forward/backward selection:

```
olsrr::ols_step_both_p(surgical_logtrain_lm, penter = 0.10, prem = 0.10)

##
##
## Stepwise Selection Summary
## -----
##      Variable      Added/      Adj.      C(p)      AIC      RMSE
## Step  Variable  Removed  R-Square  R-Square
## -----
##    1   enzyme    addition    0.409    0.401    81.6610    67.3344    0.3757
##    2  prognostic  addition    0.696    0.687    11.0970    21.5758    0.2716
##    3  bloodclot   addition    0.735    0.723     3.1150    13.5925    0.2553
## -----
```

Check validity of tentative model

```
surgical_final_lm <- lm(logTime ~ bloodclot + prognostic + enzyme,  
                        data = train_surgical)  
stat5100::visual_assumptions(surgical_final_lm)
```



```
stat5100::brown_forsythe_lm(surgical_final_lm)  
  
## [1] "Brown-forsythe test for constant variance in the residuals:"  
## [1] "T-statistic: 3.4794, p-value: 9e-04"  
  
stat5100::cor_normality_lm(surgical_final_lm)  
  
## Correlation test of normality:  
##           resid expected_norm  
## resid           1.0000000      0.9954407  
## expected_norm 0.9954407      1.0000000  
##  
## Total observations: 72  
## Make sure to consult with table B.6 for your final result.
```

Test the trained model on the testing dataset

```
test_predicted <- predict(surgical_final_lm, newdata = test_surgical)  
  
# Get mean-squared predicted error  
mspr <- mean((test_predicted - test_surgical$logTime)^2)  
mspr  
  
## [1] 0.08472427
```