Lec 05: Variable Selection

MATH 456 - Spring 2016

Navbar: [Home] [Schedule] [Data] [Week 6 Overview] [HW Info] [Google Group] [Google Drive]

Introduction

Variable selection methods are used mainly in exploratory situations where many independent variables have been measured and a final model explaining the dependent variable has not been reached.

In Bioinformatics and other fields that use Machine Learning techniques this technique of selecting variables is also known as *feature selection*.

To do variable selection you need:

1. A general test,
2. Selection criteria, and
3. A selection process.

Consider a model with P variables and you want to test if Q additional variables are useful.
$H_0 : Q$ additional variables are useless, i.e., their β's all $= 0$
$H_A : Q$ additional variables are useful

Ex: Y = FEV1, X1 = ht, X2 = age, X3 = ethnicity, X4 = location.
Test H_0: location does not matter.

A General Test

Likelihood Ratio (Deviance) Test

- Deviance = -2 log likelihood
- Under H_0, the *full model*, the deviance $= D_0, df_0 = N - P - 1$
- Under H_a, the *reduced model*, the deviance $= D_a, df_a = N - P - Q - 1$
- LR (deviance) test statistic is:
- $D_0 - D_a$ is distributed approximately as χ^2 with Q degrees of freedom under H_0 for large N.

If we assume normally distributed residuals, the LR test becomes an exact F=test.

$$F = \frac{(SSR_{red} - SSR_{full})/(df_{full} - df_{red})}{SSR_{full}/df_{full}}$$

Likelihood

Let X be a random variable with pdf f and that depends on the parameter θ. The function $\mathcal{L}(\theta|x) = f_\theta(x)$ then is called the *Likelihood function*. It is the likelihood of θ given the outcome x. Many analyses rely on maximizing this function (Maximum likelihood estimate or MLE), but commonly do so by first taking the log of this function. Hence the *log likelihood*.

Example: Testing adding Q variables to a model

Consider a model to predict depression using age, employment status and whether or not the person was chronically ill in the past year as covariates.

```r
depress <- read.delim("C:/GitHub/MATH456/data/depress_022416.txt")
depress$Employ <- factor(depress$EMPLOY,
                         labels=c("FT", "PT", "Unemp", "Retired", "Houseperson", "Student", "Other"))
full_model <- lm(log(CESD+1) ~ AGE + CHRONILL + Employ, data=depress)
tab <- xtable(summary(full_model), digits=3)
print(tab, type="html")
```

Estimate

Std. Error

t value

Pr(>|t|)

(Intercept)

2.196

0.170

12.919

0.000

AGE

-0.015

0.004

-3.758

0.000

CHRONILL

0.326

0.116

2.814

0.005

EmployPT

0.379

0.167

2.273

0.024

EmployUnemp

0.773

0.265

2.921

0.004

EmployRetired

0.372

0.213

1.743

0.082

EmployHouseperson

0.354

0.203

1.742

0.083

EmployStudent

0.348

0.679

0.513

0.608

EmployOther

0.480

0.484

0.991

0.322

The results of this model show that age and chronic illness are statistically associated with CESD (each p<.006). However employment status is a mixed bag.

Recall that employment is a categorical variable, and all the coefficient estimates shown are the effect of being in that income category has on depression *compared to* being employed full time. For example, the coefficient for PT employment is greater than zero, so they have a higher CESD score compared to someone who is fully employed.

```
exp(.379)
```

```
## [1] 1.460823
```

Specifically while holding all other variables constant, someone who is working part time has 46% higher CESD score as someone who is working full time.

Since only a small constant was added to the CESD score, we can interpret the exponentiated coefficient as the fold change as seen previously with log(Y).

But what about employment status overall? Not all employment categories are significantly different from FT status. To test that employment status affects CESD we need to do a global test that all β's are 0.

$H_0 : \beta_3 = \beta_4 = \beta_5 = \beta_6 = \beta_7 = \beta_8 = 0$
H_A: At least one β_j is not 0.

We fit the reduced model, the one without employment category.

```r
red_model <- lm(log(CESD+1) ~ AGE + CHRONILL, data=depress)
```

and conduct a global F test by running an `anova()`. *Not to be confused with `aov()`*

```r
anova(full_model, red_model)
```

```
## Analysis of Variance Table
##
## Model 1: log(CESD + 1) ~ AGE + CHRONILL + Employ
## Model 2: log(CESD + 1) ~ AGE + CHRONILL
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    285 256.74
## 2    291 270.25 -6   -13.505 2.4986 0.02261 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see that as a whole, employment significantly predicts CESD score.

This only is valid for nested models Meaning all variables in the reduced model are present in the full model.

Selection Criteria

Coefficient of Determination

If the model explains a large amount of variation in the outcome that's good right? So we could consider using R^2 as a selection criteria and trying to find the model that maximizes this value.

The residual sum of squares (RSS in the book or SSE) can be written as $\sum(Y-\hat{Y})^2(1-R^2)$. Therefore minimizing the RSS is equivalent to maximizing the multiple correlation coefficient.

Multiple R^2 Problem: The multiple R^2 *always* increases as predictors are added to the model.

Adjusted R^2 Ok, so let's add an adjustment, or a penalty, to keep this measure in check. $R^2_{adj} = R^2 - \frac{p(1-R^2)}{n-p-1}$

Information Criteria

Mallows Cp

- Compares MSE of a reduced model to the full model.
- Penalized function, as P increases Cp decreases.
- Many investigators recommend selecting those independent variables that minimize the values of Cp.

Akaike Information Criterion (AIC)

- A penalty is applied to the deviance that increases as the number of parameters p increase.
- AIC $= -2LL + 2p$
- Smaller is better

Bayesian Information Criterion (BIC)

- A different penalty function
- BIC $= -2LL + p * ln(n)$
- Compare nested and non-nested models
- BIC identifies the model that is more likely to have generated the observed data.
- Smaller is better

Cross validation (CV)

Estimate the expected level of model fit on a data set that is independent of the data used to train the model on.

1. Randomly split the sample into training sample and validation (testing) sample.
2. Compute regression equation from training sample
    a. Use that equation to compute predicted values in testing sample
    b. Calculate the prediction error on that testing sample.
3. Repeat for different splits of training and testing samples.
4. Average the prediction error across the different subsets of the data to derive a more accurate estimate of model performance.

k-fold cross validation

- Randomly partition the original sample into k equal sized subsamples.
- Compute the regression equation on $k-1$ subsamples (training sample).
- Use this model to calculate the prediction error on the kth held-out subsample (testing sample).
- Repeat this k times (k folds), with each of the k subsamples used exactly once as the validation data.
- Average the k results to produce a single estimation of model predictiveness.

Repeated k-fold CV

- Repeat k-fold CV multiple times, where the data is split differently for each repeat. Results are averaged across repeats.

How to choose k?

- A typical value for k is 10. a.k.a. 10-fold CV
- Afifi recommends $k = 3$ or 4.
- Leave-one-out cross-validation: $k = n$

Methods in R

There are several methods to cross validate a model in R. At this time we are going to use the `cv.lm()` function in the `DAAG` package.

```r
library(DAAG)
```

The `caret` package is also very powerful and flexible cross-validation tool that we are likely to come back to later on in the semester. Here are some resources to start with if you are interested in cross-validation and want to learn more about these tools.

- https://cran.r-project.org/web/packages/caret/vignettes/caret.pdf
- http://topepo.github.io/caret/training.html#builtin

Selection Process

We want to choose a set of independent variables that both will yield a good prediction using as few variables as possible.

Manual

In many situations where regression is used, the investigator has strong justification for including certain variables in the model.

- previous studies
- accepted theory

The investigator may have prior justification for using certain variables but may be open to suggestions for the remaining variables.

The set of independent variables can be broken down into logical subsets

- The usual demographics are entered first (age, gender, ethnicity)
- A set of variables that other studies have shown to affect the dependent variable
- A third set of variables that *could* be associated but the relationship has not yet been examined.

Partially model-driven regression analysis and partially an exploratory analysis.

Automated

Stepwise Regression

- Forward selection:
  - Start with no predictors.
  - Individual X variables added one at a time until optimal model reached
- Backward elimination:
  - Start with all candidate predictors.
  - Individual X variables removed one at a time until optimal model reached
- Stepwise selection: Combines the two
  - Start forward selection
  - At each step check to see if any variables should be removed.

There is a lot of controversy and criticism around these methods so I will not discuss them in great detail.

Best Subset Regression

A "better" method of variable selection considers all possible subsets/combinations of potential variables and finds the model that best fits the data according to a selected criteria.

Manually modified Automated methods

Sometimes you want to have a bit of control over the automated procedures.

- You can enter and remove variables in blocks, e.g., dummy variables representing a nominal X should all be in together, or all excluded together.
- You can force some variables in (e.g. age, gender)

Example: Model Selection

To follow the example in the book, I will use the `Chemical` data set. Refer to the book to learn more about what the variables measure. The raw data and data management file is available on the Data page of the course website.

```
chem <- read.delim("C:/GitHub/MATH456/data/chem_022816.txt", sep="\t")
```

Forward selection, Backward elimination, and Stepwise regression based on model AIC

This uses the `stepAIC` function found in the `MASS` package.

First we define the full, and null models. That null model only contains an intercept, the full model contains all proposed variables.

```
library(MASS)
null <- lm(PE ~ 1, data=chem)
full <- lm(PE ~ ., data=chem)
```

Forward

```
fwd <- stepAIC(null, scope=list(lower=null, upper=full), direction="forward")
```

```
## Start:  AIC=62.71
## PE ~ 1
##
##            Df Sum of Sq    RSS    AIC
## + DE        1    50.889 176.08 57.092
## + NPM1      1    28.012 198.96 60.756
## + PAYOUTR1  1    24.637 202.33 61.261
## + ROR5      1    22.619 204.35 61.559
## <none>                  226.97 62.708
## + EPS5      1     8.140 218.83 63.613
## + SALESGR5  1     3.854 223.11 64.194
##
## Step:  AIC=57.09
## PE ~ DE
##
##            Df Sum of Sq    RSS    AIC
## + PAYOUTR1  1   14.8683 161.21 56.445
## + NPM1      1   14.3281 161.75 56.546
## <none>                  176.08 57.092
## + SALESGR5  1    3.3472 172.73 58.516
## + ROR5      1    2.8155 173.26 58.608
## + EPS5      1    2.2830 173.79 58.700
##
## Step:  AIC=56.45
## PE ~ DE + PAYOUTR1
##
##            Df Sum of Sq    RSS    AIC
## + NPM1      1    42.026 119.18 49.384
## + SALESGR5  1    16.333 144.88 55.241
## + ROR5      1    13.415 147.79 55.839
## <none>                  161.21 56.445
## + EPS5      1     0.602 160.61 58.333
##
## Step:  AIC=49.38
## PE ~ DE + PAYOUTR1 + NPM1
##
##            Df Sum of Sq    RSS    AIC
## + SALESGR5  1   14.2397 104.94 47.567
## <none>                  119.18 49.384
## + ROR5      1    0.3395 118.84 51.299
## + EPS5      1    0.2573 118.93 51.319
##
## Step:  AIC=47.57
## PE ~ DE + PAYOUTR1 + NPM1 + SALESGR5
##
##        Df Sum of Sq    RSS    AIC
## <none>              104.94 47.567
## + EPS5  1   1.61881 103.33 49.101
## + ROR5  1   0.18875 104.75 49.513
```

Backward

```
back <- stepAIC(full, direction="backward")
```

```
## Start:  AIC=51.02
## PE ~ ROR5 + DE + SALESGR5 + EPS5 + NPM1 + PAYOUTR1
##
##            Df Sum of Sq    RSS    AIC
## - ROR5      1     0.266 103.33 49.101
## - EPS5      1     1.696 104.75 49.513
## - DE        1     4.798 107.86 50.388
## <none>                  103.06 51.023
## - SALESGR5  1    14.343 117.40 52.932
## - NPM1      1    28.689 131.75 56.391
## - PAYOUTR1  1    39.076 142.13 58.667
##
## Step:  AIC=49.1
## PE ~ DE + SALESGR5 + EPS5 + NPM1 + PAYOUTR1
##
##            Df Sum of Sq    RSS    AIC
## - EPS5      1     1.619 104.94 47.567
## <none>                  103.33 49.101
## - DE        1    10.502 113.83 50.005
## - SALESGR5  1    15.601 118.93 51.319
## - NPM1      1    41.550 144.88 57.240
## - PAYOUTR1  1    42.288 145.61 57.393
##
## Step:  AIC=47.57
## PE ~ DE + SALESGR5 + NPM1 + PAYOUTR1
##
##            Df Sum of Sq    RSS    AIC
## <none>                  104.94 47.567
## - DE        1    12.738 117.68 49.004
## - SALESGR5  1    14.240 119.18 49.384
## - NPM1      1    39.933 144.88 55.241
## - PAYOUTR1  1    56.008 160.95 58.397
```

Stepwise

```
step <- stepAIC(null, scope=list(upper=full,lower=null), direction="both")
```

```
## Start:  AIC=62.71
## PE ~ 1
##
##            Df Sum of Sq    RSS    AIC
## + DE        1    50.889 176.08 57.092
## + NPM1      1    28.012 198.96 60.756
## + PAYOUTR1  1    24.637 202.33 61.261
## + ROR5      1    22.619 204.35 61.559
## <none>                  226.97 62.708
## + EPS5      1     8.140 218.83 63.613
## + SALESGR5  1     3.854 223.11 64.194
##
## Step:  AIC=57.09
## PE ~ DE
##
##            Df Sum of Sq    RSS    AIC
## + PAYOUTR1  1    14.868 161.21 56.445
## + NPM1      1    14.328 161.75 56.546
## <none>                  176.08 57.092
## + SALESGR5  1     3.347 172.73 58.516
## + ROR5      1     2.816 173.26 58.608
## + EPS5      1     2.283 173.79 58.700
## - DE        1    50.889 226.97 62.708
##
## Step:  AIC=56.45
## PE ~ DE + PAYOUTR1
##
##            Df Sum of Sq    RSS    AIC
## + NPM1      1    42.026 119.18 49.384
## + SALESGR5  1    16.333 144.88 55.241
## + ROR5      1    13.415 147.79 55.839
## <none>                  161.21 56.445
## - PAYOUTR1  1    14.868 176.08 57.092
## + EPS5      1     0.602 160.61 58.333
## - DE        1    41.120 202.33 61.261
```

```
##
## Step:  AIC=49.38
## PE ~ DE + PAYOUTR1 + NPM1
##
##              Df Sum of Sq    RSS    AIC
## + SALESGR5   1    14.240 104.94 47.567
## <none>                    119.18 49.384
## - DE         1    15.222 134.41 50.990
## + ROR5       1     0.339 118.84 51.299
## + EPS5       1     0.257 118.93 51.319
## - NPM1       1    42.026 161.21 56.445
## - PAYOUTR1   1    42.566 161.75 56.546
##
## Step:  AIC=47.57
## PE ~ DE + PAYOUTR1 + NPM1 + SALESGR5
##
##              Df Sum of Sq    RSS    AIC
## <none>                    104.94 47.567
## - DE         1    12.738 117.68 49.004
## + EPS5       1     1.619 103.33 49.101
## - SALESGR5   1    14.240 119.18 49.384
## + ROR5       1     0.189 104.75 49.513
## - NPM1       1    39.933 144.88 55.241
## - PAYOUTR1   1    56.008 160.95 58.397
```

Let's look at the final model chosen by all three methods.

```
library(dplyr)
names(fwd$coefficients)
```

[1] "(Intercept)" "DE" "PAYOUTR1" "NPM1" "SALESGR5"

```
names(back$coefficients)
```

[1] "(Intercept)" "DE" "SALESGR5" "NPM1" "PAYOUTR1"

```
names(step$coefficients)
```

[1] "(Intercept)" "DE" "PAYOUTR1" "NPM1" "SALESGR5"

They all ended at the exact same model. This will not always be the case, but when it is you can be assured that the variables chosen are truly important variable to predict the outcome.

Best Subsets

To perform best subsets we will use the `regsubsets` function found in the `leaps` package. From the help file for `leaps`: *Since the algorithm returns a best model of each size, the results do not depend on a penalty model for model size: it doesn't make any difference whether you want to use AIC, BIC, CIC, DIC, . . .*

```
library(leaps)
regsubsets.out <- regsubsets(PE ~ ROR5 + DE + SALESGR5 + EPS5 + NPM1 + PAYOUTR1,
                     data = chem,
                     nbest = 2,       # 2 best models for each number of predictors
                     nvmax = NULL,    # NULL for no limit on number of variables
                     force.in = NULL, force.out = NULL,
                     method = "exhaustive")
```

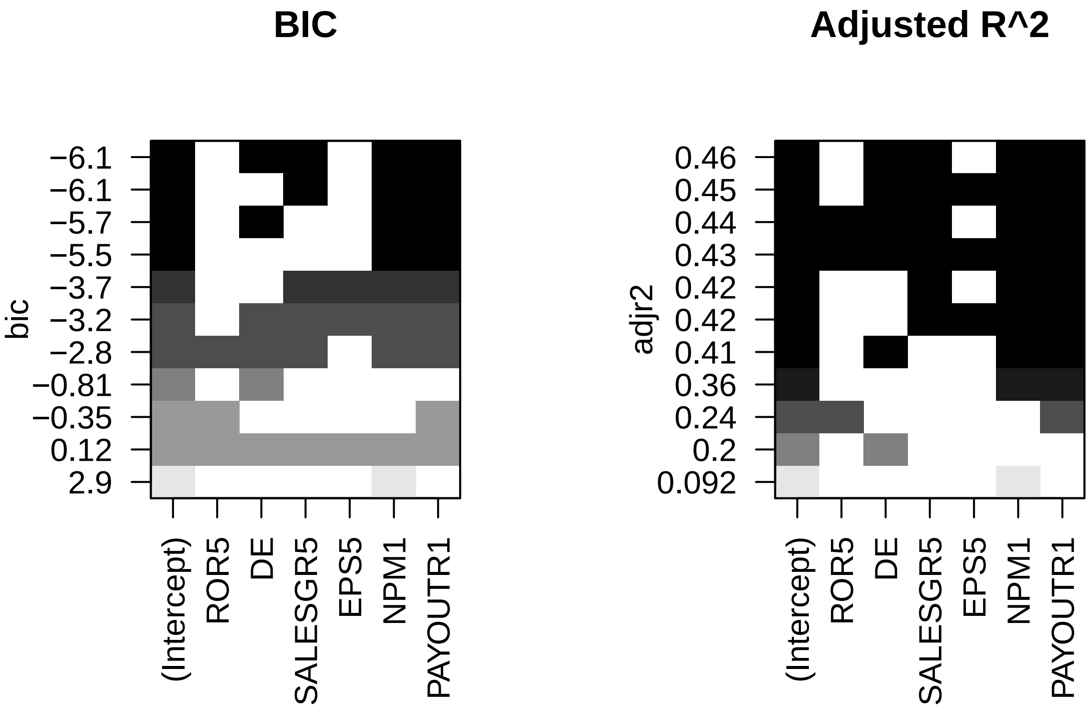
Let’s look at the 2 best models for each size subset.

```
summary(regsubsets.out)
```

```
## Subset selection object
## Call: regsubsets.formula(PE ~ ROR5 + DE + SALESGR5 + EPS5 + NPM1 + 
##     PAYOUTR1, data = chem, nbest = 2, nvmax = NULL, force.in = NULL, 
##     force.out = NULL, method = "exhaustive")
## 6 Variables  (and intercept)
##          Forced in Forced out
## ROR5         FALSE      FALSE
## DE           FALSE      FALSE
## SALESGR5     FALSE      FALSE
## EPS5         FALSE      FALSE
## NPM1         FALSE      FALSE
## PAYOUTR1     FALSE      FALSE
## 2 subsets of each size up to 6
## Selection Algorithm: exhaustive
##          ROR5 DE  SALESGR5 EPS5 NPM1 PAYOUTR1
## 1  ( 1 ) " "  "*" " "      " "  " "  " "     
## 1  ( 2 ) " "  " " " "      " "  "*"  " "     
## 2  ( 1 ) " "  " " " "      " "  "*"  "*"     
## 2  ( 2 ) "*"  " " " "      " "  " "  "*"     
## 3  ( 1 ) " "  " " "*"      " "  "*"  "*"     
## 3  ( 2 ) " "  "*" " "      " "  "*"  "*"     
## 4  ( 1 ) " "  "*" "*"      " "  "*"  "*"     
## 4  ( 2 ) " "  " " "*"      "*"  "*"  "*"     
## 5  ( 1 ) " "  "*" "*"      "*"  "*"  "*"     
## 5  ( 2 ) "*"  "*" "*"      " "  "*"  "*"     
## 6  ( 1 ) "*"  "*" "*"      "*"  "*"  "*"
```

A * in the column means that variable was selected to be included in that model. `Payoutr1` was included in most selections, as was `NPM1`. We can visualize the results based on fitness measures by plotting the `regsubsets.out` output.

```
par(mfrow=c(1,2))
plot(regsubsets.out, scale="bic", main="BIC")
plot(regsubsets.out, scale="adjr2", main="Adjusted R^2")
```



Black indicates that a variable is included in the model, white indicates the variable is not in the model. The y axis (and shading) is oriented such that the top (darker) boxes are better. The top model under both methods (lowest BIC and highest adjusted R^2) is the same: `PE ~ DE + SALESGR5 + NPM1 + PAYOUTR1`. The second best model under BIC is the same as the best model but drops `DE`, but the second best model using adjusted R^2 is vastly different.

Conclusion The final model chosen by best subsets corresponds with the step-wise procedures.

Example: Cross Validation

In the prior example, the top two candidate models were:

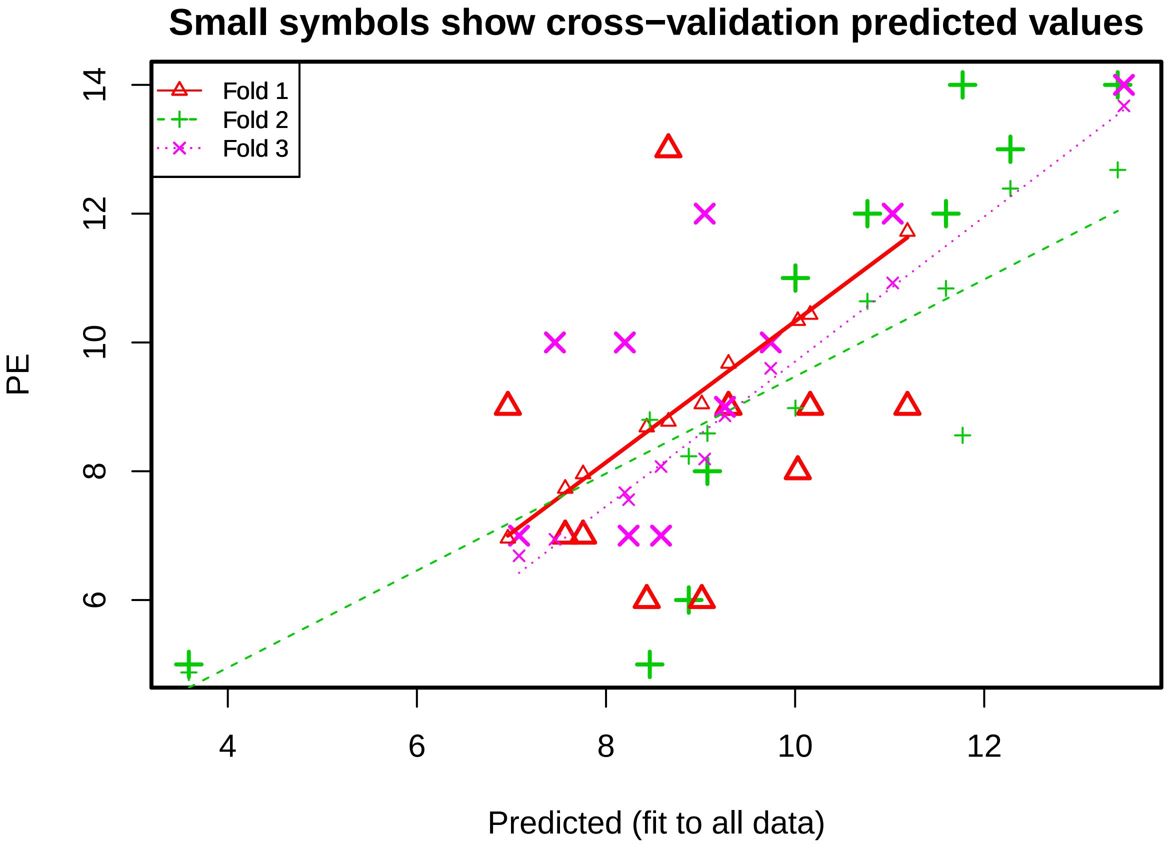
- Model 1: `PE ~ DE + SALESGR5 + NPM1 + PAYOUTR1`
- Model 2: `PE ~ SALESGR5 + NPM1 + PAYOUTR1`

In the interest of parsimony, if the variable `DE` really doesn't contribute a lot of information to the model then perhaps it can be dropped. There are only 30 observations in the chem dataset, so I will conduct only 3-fold cross-validation due to sample size.

```
fit1 <- cv.lm(data=chem, # data set
              form.lm = formula(PE ~ DE + SALESGR5 + NPM1 + PAYOUTR1),  # model
              m=3) # number of partitions
```

```
## Analysis of Variance Table
```

```
##
## Response: PE
##           Df Sum Sq Mean Sq F value Pr(>F)
## DE         1   50.9    50.9   12.12 0.0018 **
## SALESGR5   1    3.3     3.3    0.80 0.3804
## NPM1       1   11.8    11.8    2.81 0.1064
## PAYOUTR1   1   56.0    56.0   13.34 0.0012 **
## Residuals 25  104.9     4.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

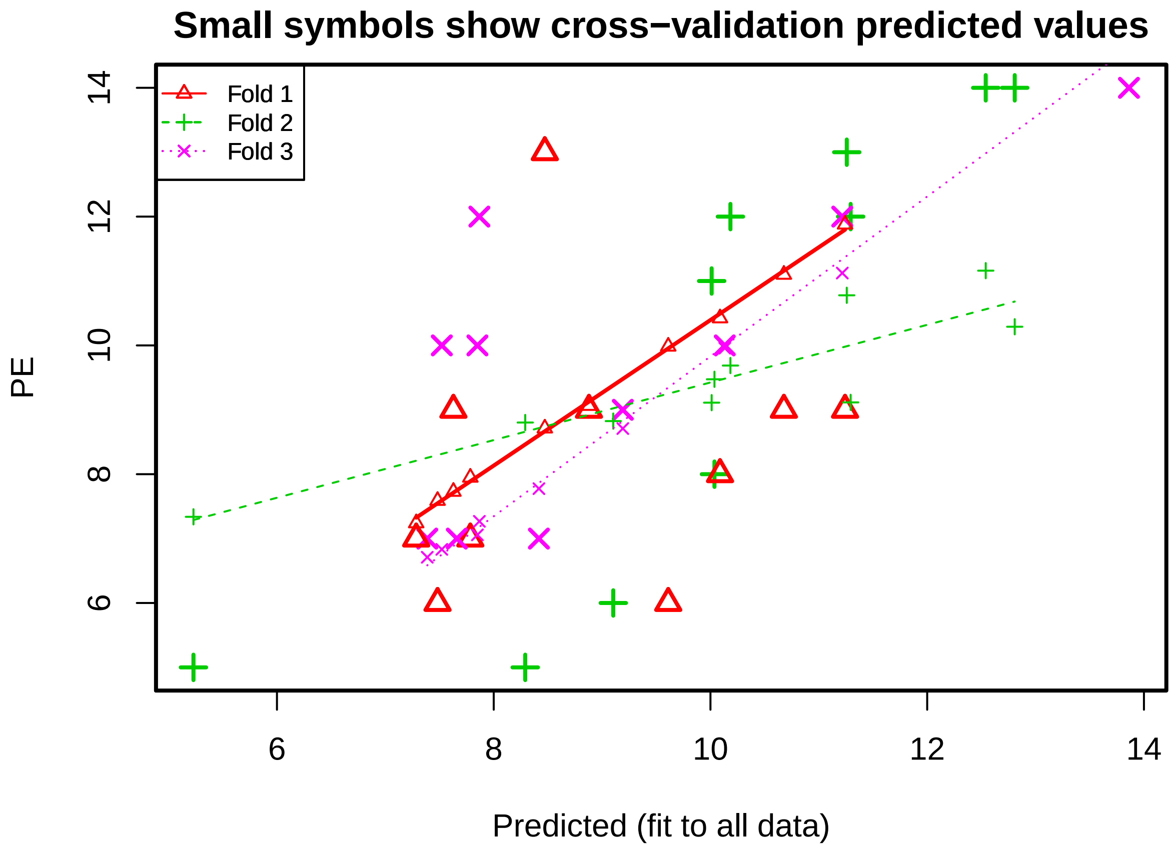


```
##
## fold 1
## Observations in test set: 10
##                  1     3     4      8     13    18    22     25    27   29
## Predicted   10.16 10.03 11.19  9.295  7.569  9.01  8.66  7.758  8.43 6.96
## cvpred      10.43 10.34 11.72  9.672  7.731  9.04  8.77  7.956  8.68 6.96
## PE           9.00  8.00  9.00  9.000  7.000  6.00 13.00  7.000  6.00 9.00
## CV residual -1.43 -2.34 -2.72 -0.672 -0.731 -3.04  4.23 -0.956 -2.68 2.04
##
## Sum of squares = 55.3    Mean square = 5.53    n = 10
##
## fold 2
## Observations in test set: 10
##                   2     5     6     9    15    16    17    20    23    30
## Predicted    9.072  8.46  8.88 10.00 12.28 13.41 3.588 11.60 10.76 11.77
## cvpred       8.588  8.80  8.23  8.98 12.39 12.68 4.876 10.84 10.64  8.56
## PE           8.000  5.00  6.00 11.00 13.00 14.00 5.000 12.00 12.00 14.00
## CV residual -0.588 -3.80 -2.23  2.02  0.61  1.32 0.124  1.16  1.36  5.44
##
## Sum of squares = 58.8    Mean square = 5.88    n = 10
##
## fold 3
## Observations in test set: 10
##                 7    10    11    12    14     19     21    24     26    28
## Predicted    7.46 9.258  8.58 7.080  8.20  9.742 13.478  9.04  8.239 11.03
## cvpred       6.94 8.861  8.07 6.687  7.67  9.598 13.674  8.19  7.559 10.92
## PE          10.00 9.000  7.00 7.000 10.00 10.000 14.000 12.00  7.000 12.00
## CV residual  3.06 0.139 -1.07 0.313  2.33  0.402  0.326  3.81 -0.559  1.08
##
## Sum of squares = 32.3    Mean square = 3.23    n = 10
##
## Overall (Sum over all 10 folds)
##   ms
## 4.88
```

The generated graphic plots the predicted PE (X) against the true PE (Y). The large values are for predictions made on the full data, the small values are for predictions after cross-validating the model. Individual regression lines are shown for each of the folds as well.

Now fit the second model (suppressing the printing of the verbose output).

```
fit2 <- CVlm(data=chem, printit=FALSE,
            form.lm = formula(PE ~ SALESGR5 + NPM1 + PAYOUTR1), m=3)
```



Recall the MSE (mean squared error) is amount of variance in the outcome that was NOT explained by the model, so it is a measure that we want to minimize. We extract the average MS from each model and compare.

```
attributes(fit1)$ms
```

```
## [1] 4.88
```

```
attributes(fit2)$ms
```

```
## [1] 5.84
```

Model 1 is selected as the final model because it has the lowest MSE.

What to watch out for

- Use previous research as a guide
- Variables not included can bias the results
- Significance levels are only a guide
- Perform diagnostics after selection
- ***Use common sense***:
  - A sub-optimal subset may make more sense than optimal one

In addition to the almost dozen entries in the textbook, see the following resources regarding areas of concern.

- http://www.stata.com/support/faqs/statistics/stepwise-regression-problems/
- http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.126.4133&rep=rep1&type=pdf

TLDR; Don't expect a magic bullet and don't use any method blindly.

Penalized methods

I won't go over these methods because we don't have the time to do them justice. However I encourage you to learn more about methods such as Ridge/Lasso regression, and cross-validation methods. Here are a few places to start.

- http://www.stat.ucla.edu/~cocteau/stat120b/lectures/lecture7.pdf
- http://statweb.stanford.edu/~jtaylo/courses/stats203/notes/penalized.pdf
- http://www.stat.ufl.edu/archived/casella/Papers/BL-Final.pdf
- http://www.r-bloggers.com/variable-selection-using-cross-validation-and-other-techniques/

Assigned Reading and additional references

- Afifi Chapter 8
- http://www.statmethods.net/stats/regression.html
- http://www.stat.columbia.edu/~martin/W2024/R10.pdf

- https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/stepAIC.html
- http://www.stat.colostate.edu/~darrenho/AMA/1_regression2.pdf
- https://dynamicecology.wordpress.com/2015/05/21/why-aic-appeals-to-ecologists-lowest-instincts/
- http://www.r-bloggers.com/aic-bic-vs-crossvalidation/
- http://andrewgelman.com/2012/06/27/cross-validation-what-is-it-good-for/
- http://users.stat.umn.edu/~yangx374/papers/ACV_v30.pdf

On Your Own

On Your Own

1. For the lung function data, use an automated selection process to predict FEV1 for the oldest child using age, height, weight and FVC as candidate variables. State and justify the method and criteria you chose.
2. Take the variables you selected in problem 2 and build a linear regression model with `OCFEV1` as the dependent variable, and test whether including the FEV1 of the parents (`MFEV1` and `FFEV1` taken as a pair) in the model significantly improves the regression.
3. Using the Parental HIV data find the best model that predicts the age at which adolescents started drinking alcohol. Since the data were collected retrospectively, only consider variables which might be considered representative of the time before the adolescent started drinking alcohol.