STAT 108: Lab 6

Tim Lanthier

2/24/2022

Lab 06: Model Selection + Diagnostics

Github Repository: https://github.com/talanthier/lab-06

```
library(tidyverse)
library(knitr)
library(broom)
library(leaps)
library(rms)
library(Sleuth3) #case1201 data
```

In this lab we will be working with SAT data from the 1982 exam. The dataset can be found in the Sleuth3 package (case 1201).

Model Selection

We will start with a full linear model including all possible predictor variables and no interaction terms.

```
sat_scores <- Sleuth3::case1201
full_model <- lm(SAT ~ Takers + Income + Years + Public + Expend + Rank , data = sat_scores)
tidy(full_model) %>% kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-94.659	211.510	-0.448	0.657
Takers	-0.480	0.694	-0.692	0.493
Income	-0.008	0.152	-0.054	0.957
Years	22.610	6.315	3.581	0.001
Public	-0.464	0.579	-0.802	0.427
Expend	2.212	0.846	2.615	0.012
Rank	8.476	2.108	4.021	0.000

Now we will conduct backwards selection on our full model. We will start by using adjusted R^2 as our selection criterion.

```
model_select <- regsubsets(SAT ~ Takers + Income + Years + Public + Expend +
                           Rank , data = sat_scores, method = "backward")
select_summary <- summary(model_select)
coef(model_select,  which.max(select_summary$adjr2)) # choose model which has highest adj R^2
```

```
## (Intercept)       Years      Public      Expend        Rank
## -204.598232   21.890482   -0.663798    2.241640   10.003169
```

So using adjusted R^2 as our criterion, our best model from our backselected models is the model with `Years`, `Public`, `Expend`, and `Rank` as our predictor variables.

```
coef(model_select, which.min(select_summary$bic)) # choose model with smallest BIC
```

```
## (Intercept)       Years      Expend        Rank
## -303.724295   26.095227    1.860866    9.825794
```

Meanwhile with BIC, we find the best model to be the one including just `Years`, `Expend`, and `Rank`. Note that for these cases, `regsubsets` is using the residual sum of squares for its criteria for backwards selection. We are only comparing the backwards selected models using Adj. R^2 and BIC.

Now we will run backwards selection using AIC as the criteria for eliminating predictors.

```
model_select_aic <- step(full_model, direction = "backward")
```

```
## Start:  AIC=333.58
## SAT ~ Takers + Income + Years + Public + Expend + Rank
##
##          Df Sum of Sq   RSS    AIC
## - Income  1       2.0 29844 331.59
## - Takers  1     332.4 30175 332.14
## - Public  1     445.8 30288 332.32
## <none>                29842 333.58
## - Expend  1    4744.9 34587 338.96
## - Years   1    8897.8 38740 344.63
## - Rank    1   11223.0 41065 347.54
##
## Step:  AIC=331.59
## SAT ~ Takers + Years + Public + Expend + Rank
##
##          Df Sum of Sq   RSS    AIC
## - Takers  1     401.3 30246 330.25
## - Public  1     495.5 30340 330.41
## <none>                29844 331.59
## - Expend  1    6904.4 36749 339.99
## - Years   1    9219.7 39064 343.05
## - Rank    1   11645.9 41490 346.06
##
## Step:  AIC=330.25
## SAT ~ Years + Public + Expend + Rank
##
##          Df Sum of Sq    RSS    AIC
## <none>                 30246 330.25
## - Public  1      1462  31708 330.62
## - Expend  1      7343  37589 339.12
## - Years   1      8837  39083 341.07
## - Rank    1    184786 215032 426.33
```

With AIC as our criterion, backward selection has chosen the model with `Public`, `Expend`, `Years`, and `Rank` as the best model. Comparing the 3 different models we found with different criteria, we see that they don't all match up. While AIC and Adj. R^2 agreed on what variables to select, BIC yielded a model with one less variable than the rest. That being said, BIC still selected `Years`, `Expend`, and `Rank` which all appear in the models selected with AIC and Adj. R^2. The fact that our BIC selects a model with fewer variables makes sense since BIC tends to penalize models more complex models more heavily than AIC.

Model Diagnostics

For the remainder of the lab, we will be using our model selected with AIC.

```
model_select_aic_aug <- augment(model_select_aic) %>%
  mutate(obs_num = row_number())

head(model_select_aic_aug,5)
```

```
## # A tibble: 5 x 12
##     SAT Years Public Expend  Rank .fitted .resid   .hat .sigma .cooksd
##   <int> <dbl>  <dbl>  <dbl> <dbl>   <dbl>  <dbl>  <dbl>  <dbl>   <dbl>
## 1  1088  16.8   87.8   25.6  89.7   1059.  28.7  0.100    25.8 0.0304
## 2  1075  16.1   86.2   20.0  90.6   1041.  34.0  0.0788   25.7 0.0320
## 3  1068  16.6   88.3   20.6  89.8   1044.  24.0  0.0894   25.9 0.0185
## 4  1045  16.3   83.9   27.1  86.3   1021.  24.4  0.0585   25.9 0.0117
## 5  1045  17.2   83.6   21.0  88.5   1050.  -4.99 0.113    26.2 0.00106
## # ... with 2 more variables: .std.resid <dbl>, obs_num <int>
```

Now that we have the model predictions and statistics for each of the observations in our dataset, we will examine the leverage for each observation. Since we're conducting multiple linear regression, leverage has the formula

$$H = X(X^T X)^{-1} X^T$$

where X is the original dataframe with the 4 predictor variables in our model. in `model_select_aic_aug`, leverage for each observation is shown as `.hat`. We will say that an observation has high leverage if it is above the threshold

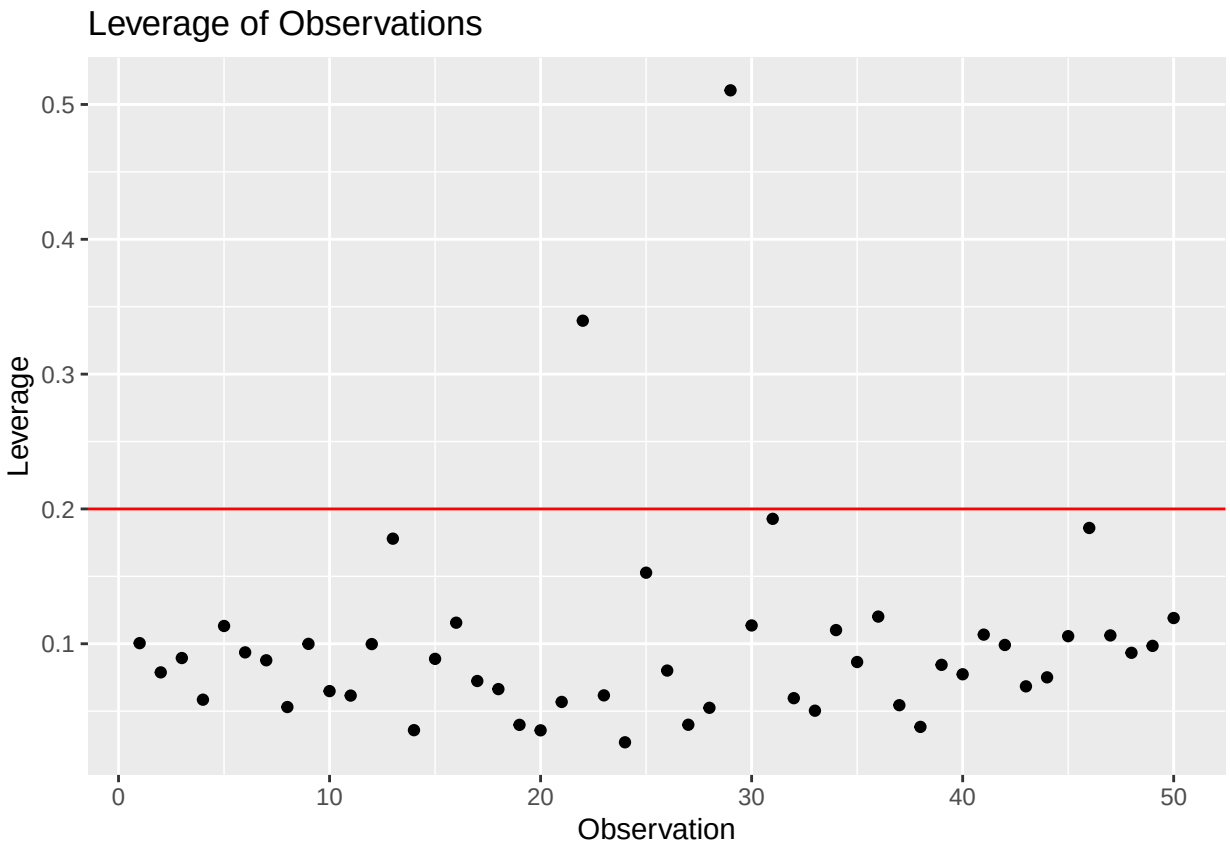
$$\frac{2(p+1)}{n}$$

where n is the number of observations and p the number of predictors. Hence for our data we have a threshold of

$$\frac{2(4+1)}{50} = 0.2$$

Now we will plot the leverage for all of the observations

```
ggplot(model_select_aic_aug, aes(obs_num, .hat)) +
  geom_point() +
  geom_hline(yintercept = 0.2, color = 'red') +
  labs(y = 'Leverage', x = 'Observation', title = 'Leverage of Observations')
```



Looking at our plot of the leverage, we have 2 clear high leverage observations. We also have 3 states which are close to but not above the threshold.

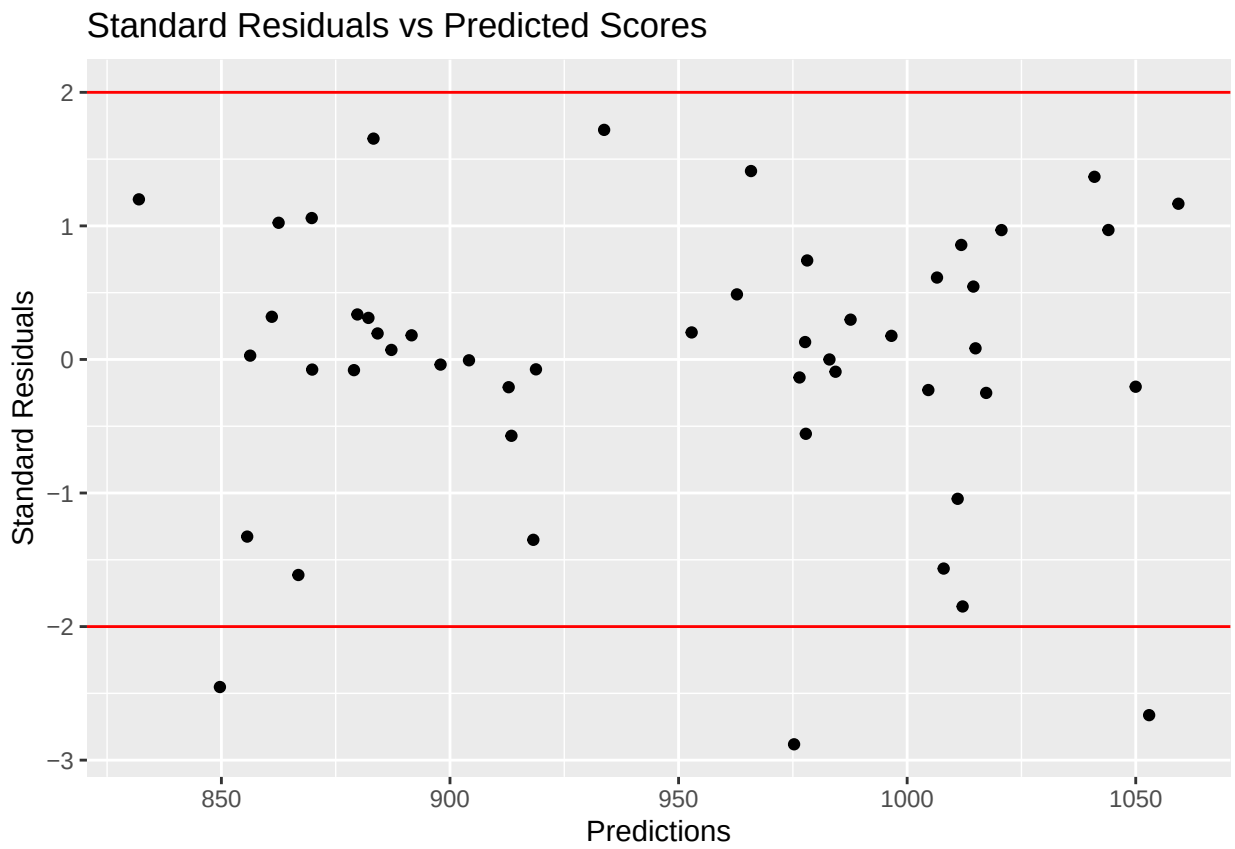
```
sat_scores[which(model_select_aic_aug$.hat > 0.2), ]
```

```
##        State SAT Takers Income Years Public Expend Rank
## 22 Louisiana 975      5    394 16.85   44.8  19.72 82.9
## 29    Alaska 923     31    401 15.32   96.5  50.10 79.6
```

So the 2 states which have high leverage are Louisiana and Alaska.

Now we will examine the standard residuals.

```
ggplot(model_select_aic_aug, aes(.fitted, .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 2, color = 'red') +
  geom_hline(yintercept = -2, color = 'red') +
  labs(x = 'Predictions', y = 'Standard Residuals', title = 'Standard Residuals vs Predicted Scores')
```



According to the standard residuals, we have 3 states lying outside of our threshold. So 3 states have standard residuals with a large magnitude.

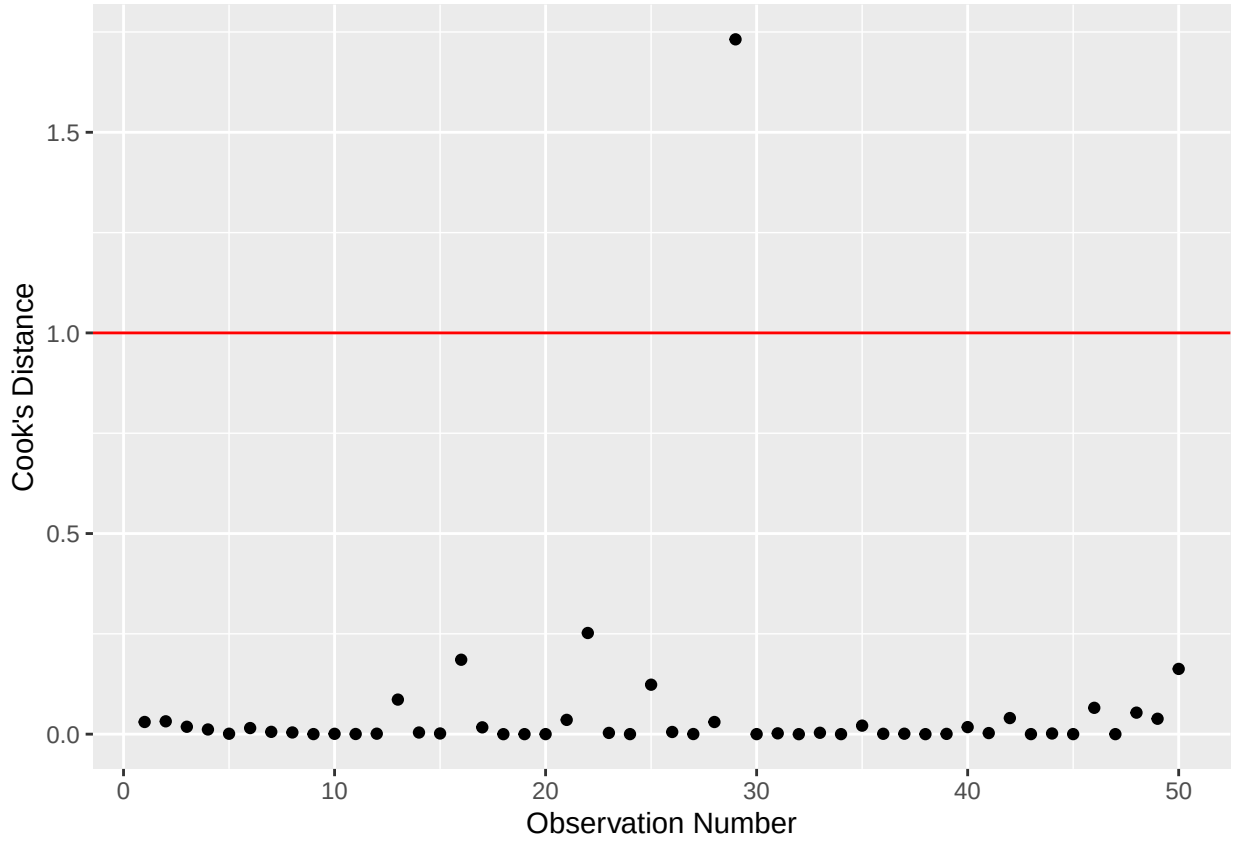
```
sat_scores[which(abs(model_select_aic_aug$.std.resid) > 2), ]
```

```
##            State SAT Takers Income Years Public Expend Rank
## 16   Mississippi 988      3    315 16.76   67.9  15.36 90.1
## 29        Alaska 923     31    401 15.32   96.5  50.10 79.6
## 50 SouthCarolina 790     48    214 15.42   88.1  15.60 74.0
```

As shown above, the 3 states with large standardized residuals are Mississippi, Alaska, and South Carolina.

To check whether these states we identified which have high leverage or high standardized residuals are significantly impacting our model, we will investigate the Cook's distance for our data.

```
ggplot(model_select_aic_aug, aes(obs_num, .cooksd)) +
  geom_point() +
  geom_hline(yintercept = 1, color = 'red') +
  labs(x = 'Observation Number', y = "Cook's Distance")
```



With our threshold of 1 for Cook's distance, we see only one observation lies above our threshold.

```
sat_scores[which(abs(model_select_aic_aug$.cooksd) > 1), ]
```

```
##     State SAT Takers Income Years Public Expend Rank
## 29 Alaska 923     31    401 15.32   96.5   50.1 79.6
```

Here, that state lying above our threshold is Alaska, which we found to have a high magnitude standardized residual as well as a high magnitude leverage. Seeing as we only have a single influential observation, it might be wise to remove that observation from our dataset. While a model based on this dataset will not be useful in predicting SAT scores in Alaska, removing Alaska might result in predictions for the remaining 49 states to be more accurate. If we wanted to use our model to make predictions on Alaska's SAT scores, we would have to include it in our dataset.

Lastly we need to check for collinearity. For this we will use the Variance Inflation Factor (VIF). We will start by building a model with `Expend` as our response variable and the same predictors as our backward selected model based on AIC.

```
expend_model <- lm(Expend ~ Years + Public + Rank, data = sat_scores)
summary(expend_model)
```

```
##
## Call:
## lm(formula = Expend ~ Years + Public + Rank, data = sat_scores)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0866 -3.9495 -0.1809  2.3098 25.1092
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.23862   25.54114  -0.401  0.69037
## Years         2.19154    1.27212   1.723  0.09165 .
## Public        0.25256    0.09047   2.792  0.00761 **
## Rank         -0.28539    0.12423  -2.297  0.02620 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.636 on 46 degrees of freedom
## Multiple R-squared:  0.2102, Adjusted R-squared:  0.1587
## F-statistic: 4.081 on 3 and 46 DF,  p-value: 0.01189
```

Looking at the above model, we get an R^2 of 0.2102. So only approximately 21% of the variance in `Expend` is explained by `Years`, `Public`, and `Rank`. We can calculate the VIF for `Expend` with the following formula

$$\frac{1}{1-R^2} = \frac{1}{1-0.2102} = 1.266$$

So we have a VIF of 1.266. Since we would consider a VIF of greater than 10 as concerning, there doesn't appear to be any concerning collinearity between `Expend` and the other predictors in the model. Now we will calculate the VIF of the remaining predictors.

```
vif(model_select_aic)
```

```
##    Years   Public   Expend     Rank
## 1.301929 1.426831 1.266145 1.129034
```

The VIF for each of the predictors in our model is shown above. As we can see, the VIFs for the remaining predictors are of similar magnitude to that of `Expend`. So once again, the VIF does not indicate any collinearity between any of our variables.