

quiz3

December 1, 2015

1 STAT 510 Quiz 3

1.1 Juan Carlos Apitz

```
In [1]: # import libraries
import pandas as pd
from pandas import Series, DataFrame
import numpy as np

import statsmodels.api as sm
from statsmodels.formula.api import ols

import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

1.2 Question 1

1.3 Part a.

1.3.1 Hald Data

For this data we begin by conducting a model selection analysis based on the R^2 , adjusted R^2 , the AIC, and Mallows's C_p .

To construct all possible regression models, I wrote a script in python that produces the necessary results:

```
In [2]: run construct_models
```

These are the first few rows of the Hald dataset:

	Y	X1	X2	X3	X4	mean
0	78.5	7	26	6	60	95.423077
1	74.3	1	29	15	52	95.423077
2	104.3	11	56	8	20	95.423077
3	87.6	11	31	8	47	95.423077
4	95.9	7	52	6	33	95.423077

The possible models are:

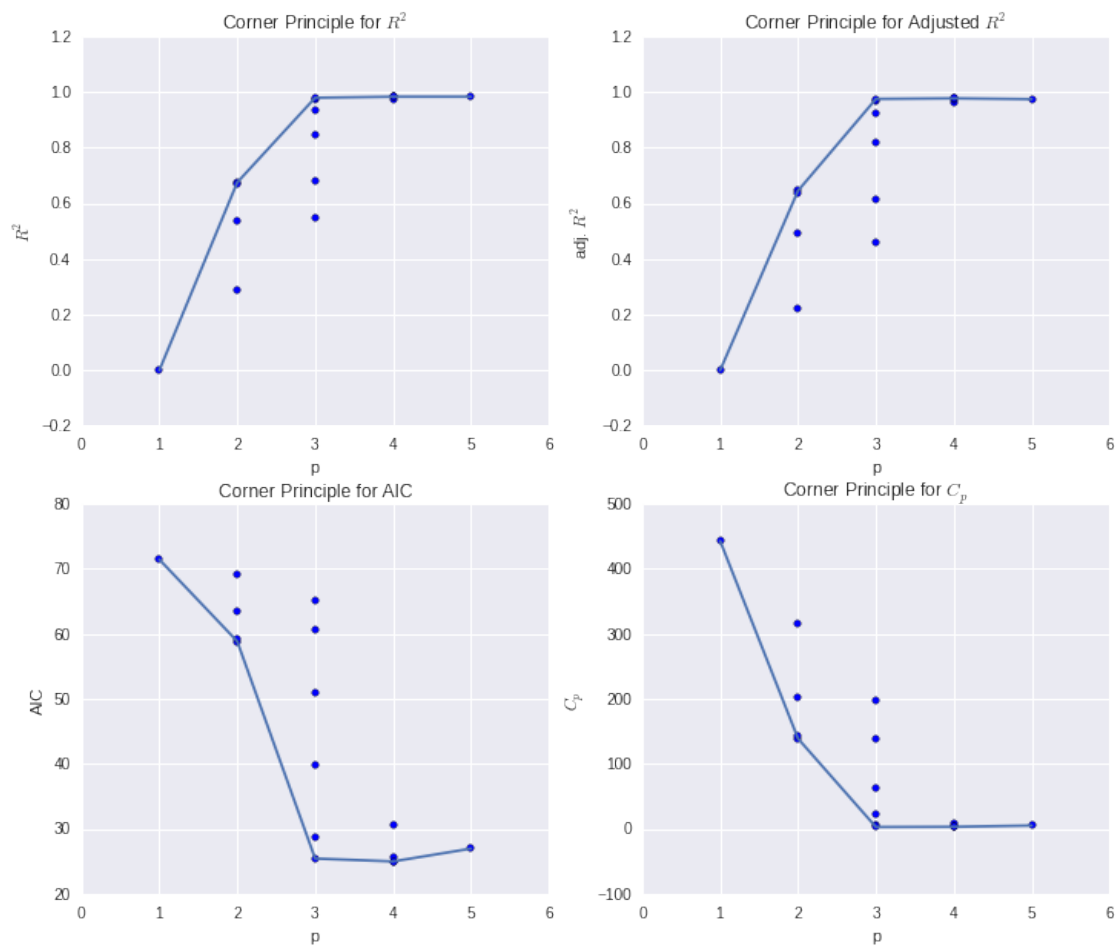
```
Y ~ X1 + X2 + X3 + X4
Y ~ X2 + X3 + X4
Y ~ X1 + X3 + X4
```

```

Y ~ X3 + X4
Y ~ X1 + X2 + X4
Y ~ X2 + X4
Y ~ X1 + X4
Y ~ X4
Y ~ X1 + X2 + X3
Y ~ X2 + X3
Y ~ X1 + X3
Y ~ X3
Y ~ X1 + X2
Y ~ X2
Y ~ X1
Y ~ mean

```

The following plots summarize our analysis:



In [3]: results_tbl

```

Out[3]:
  model  P      R-sq      Adj. R-sq      AIC \
0  model00  5      0.9824      0.9736      24.9443
1  model01  4      0.9728      0.9638      28.5759

```

2	model02	4	0.9813	0.9750	23.7276
3	model03	3	0.9353	0.9223	37.8526
4	model04	4	0.9823	0.9764	22.9739
5	model05	3	0.6801	0.6161	58.6293
6	model06	3	0.9725	0.9670	26.7417
7	model07	2	0.6745	0.6450	56.8516
8	model08	4	0.9823	0.9764	23.0112
9	model09	3	0.8470	0.8164	49.0371
10	model10	3	0.5482	0.4578	63.1167
11	model11	2	0.2859	0.2210	67.0674
12	model12	3	0.9787	0.9744	23.4200
13	model13	2	0.6663	0.6359	57.1780
14	model14	2	0.5339	0.4916	61.5195
15	model15	1	0.0000	0.0000	69.4444

	Cp
0	3.0000
1	5.3375
2	1.4968
3	20.3731
4	1.0182
5	136.2259
6	3.4959
7	136.7308
8	1.0413
9	60.4377
10	196.0947
11	313.1543
12	0.6782
13	140.4864
14	200.5488
15	440.9167

```
In [4]: print 'The models to be considered are model 06: {} and model 12: {}'.format(models[6], models
```

The models to be considered are model 06: $Y \sim X_1 + X_4$ and model 12: $Y \sim X_1 + X_2$.

The above analysis indicates that a model with $p = 3$ might be adequate. The corner principle plots indicate that, models with $P > 3$, do not produce improvements in R^2 , adjusted R^2 , the AIC, or Mallows's C_p . In particular, model 06: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_4 + \epsilon$ and model 12: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$, shows high R^2 , low AIC and C_p that is small and relatively close to $p=3$. The other models with $p=3$, show large values of C_p which is an indication of bias, thus they will not be considered.

1.4 Part b.

Now we implement Forward, Backward, and Stepwise selection and compare with the results in a. above. For this I wrote the module mselector. The python code is included as an appendix.

Import mselector and create new dataset.

```
In [5]: import mselector as s
```

```
dfnew = df.iloc[:, :5]
```

1.4.1 Forward Selection

Run the forward selection algorithm:

```
In [6]: s.foresel(dfnew, 'Y')
```

The optimal model is $Y \sim X4 + X1$

```
/home/jcapitz/anaconda/lib/python2.7/site-packages/scipy/stats/stats.py:1277: UserWarning: kurtosistest
"anyway, n=%i" % int(n))
```

```
Out[6]: <class 'statsmodels.iolib.summary.Summary'>
```

```
"""
                                OLS Regression Results
=====
Dep. Variable:                  Y    R-squared:                0.972
Model:                        OLS    Adj. R-squared:           0.967
Method:                    Least Squares    F-statistic:            176.6
Date:                Tue, 01 Dec 2015    Prob (F-statistic):      1.58e-08
Time:                  13:30:42    Log-Likelihood:         -29.817
No. Observations:                13    AIC:                   65.63
Df Residuals:                    10    BIC:                   67.33
Df Model:                        2
Covariance Type:                nonrobust
=====
                                coef    std err          t      P>|t|      [95.0% Conf. Int.]
-----
Intercept    103.0974      2.124     48.540     0.000     98.365    107.830
X4           -0.6140      0.049    -12.621     0.000     -0.722    -0.506
X1            1.4400      0.138     10.403     0.000      1.132     1.748
=====
Omnibus:                 0.408    Durbin-Watson:           1.788
Prob(Omnibus):           0.816    Jarque-Bera (JB):         0.432
Skew:                   -0.331    Prob(JB):                 0.806
Kurtosis:                2.400    Cond. No.                 97.0
=====

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
"""
```

1.4.2 Backward Selection

Run the backward selection algorithm:

```
In [7]: s.backsel(dfnew, 'Y')
```

The optimal model is $Y \sim X2 + X1$

```
Out[7]: <class 'statsmodels.iolib.summary.Summary'>
```

```
"""
                                OLS Regression Results
=====
Dep. Variable:                  Y    R-squared:                0.979
Model:                        OLS    Adj. R-squared:           0.974
Method:                    Least Squares    F-statistic:            229.5
Date:                Tue, 01 Dec 2015    Prob (F-statistic):      4.41e-09
=====
```

```

Time:                  13:30:42   Log-Likelihood:          -28.156
No. Observations:      13       AIC:                  62.31
Df Residuals:          10       BIC:                  64.01
Df Model:              2
Covariance Type:       nonrobust

```

```

=====
              coef      std err          t      P>|t|      [95.0% Conf. Int.]
-----
Intercept    52.5773      2.286      22.998      0.000      47.483      57.671
X2           0.6623      0.046      14.442      0.000       0.560       0.764
X1           1.4683      0.121      12.105      0.000       1.198       1.739
=====
Omnibus:                1.509   Durbin-Watson:           1.922
Prob(Omnibus):           0.470   Jarque-Bera (JB):       1.104
Skew:                    0.503   Prob(JB):               0.576
Kurtosis:                1.987   Cond. No.               175.
=====

```

Warnings:

```

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
"""

```

1.4.3 Stepwise Selection

Run the stepwise selection algorithm:

```
In [8]: s.stepwsel(dfnew, 'Y')
```

The optimal model is $Y \sim X1 + X2$

```
Out[8]: <class 'statsmodels.iolib.summary.Summary'>
"""

```

```

                                OLS Regression Results
=====
Dep. Variable:                  Y   R-squared:                0.979
Model:                        OLS   Adj. R-squared:           0.974
Method:                    Least Squares   F-statistic:            229.5
Date:                Tue, 01 Dec 2015   Prob (F-statistic):      4.41e-09
Time:                  13:30:42   Log-Likelihood:          -28.156
No. Observations:          13       AIC:                  62.31
Df Residuals:              10       BIC:                  64.01
Df Model:                  2
Covariance Type:       nonrobust
=====
              coef      std err          t      P>|t|      [95.0% Conf. Int.]
-----
Intercept    52.5773      2.286      22.998      0.000      47.483      57.671
X1           1.4683      0.121      12.105      0.000       1.198       1.739
X2           0.6623      0.046      14.442      0.000       0.560       0.764
=====
Omnibus:                1.509   Durbin-Watson:           1.922
Prob(Omnibus):           0.470   Jarque-Bera (JB):       1.104
Skew:                    0.503   Prob(JB):               0.576
Kurtosis:                1.987   Cond. No.               175.
=====

```

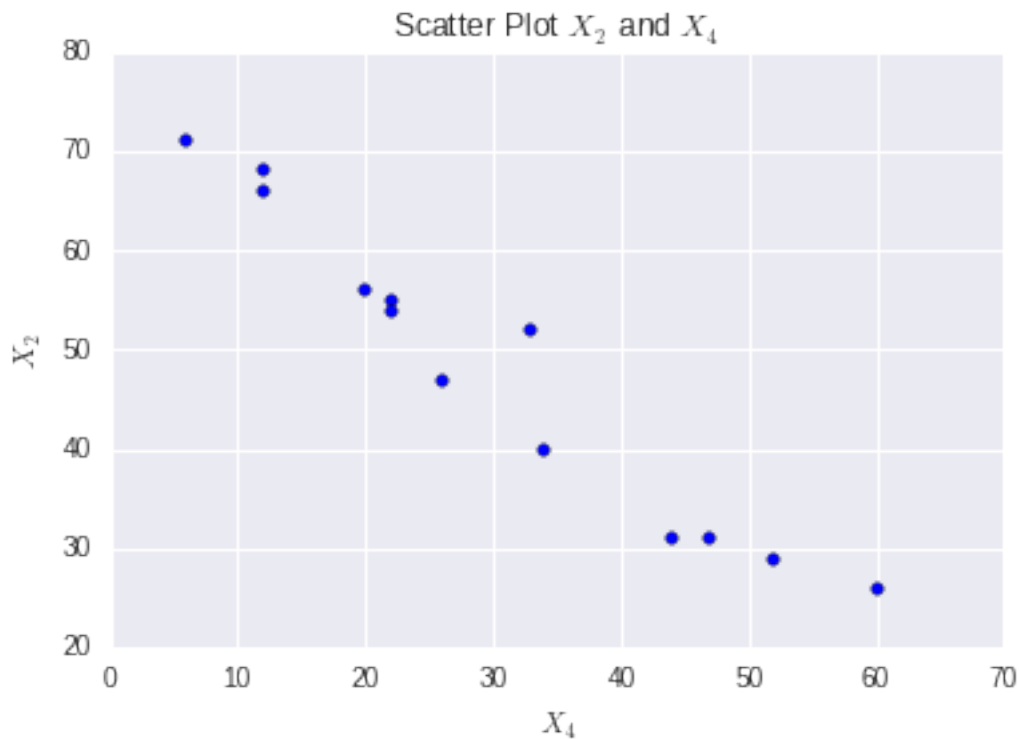
Warnings:

```
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
"""
```

1.5 Conclusion

Based on the results of the analysis in part a. and b. above the model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$ seems most appropriate. In part a. this particular model shows high values of R^2 and adjusted- R^2 as well as low values of AIC and C_p statistics. In addition, both the backward selection and stepwise selection algorithms selected this particular model. In a sense, this result is satisfactory because the backward selection and the stepwise selection algorithm take into account interactions between the explanatory variables, which is the reason the stepwise algorithm “dropped” X_4 in favor of X_2 in the selection process. The reason this variable gets dropped is because of the correlation between X_2 and X_4 , which we can see in the graph below:

```
In [9]: plt.scatter(x=df['X4'], y=df['X2'])
plt.title('Scatter Plot $X_2$ and $X_4$')
plt.ylabel('$X_2$')
plt.xlabel('$X_4$')
plt.show()
```



1.6 Question 2

The DFFITS statistic is composed of two factors:

$$e_i \left[\frac{n - p - 1}{SSE(1 - h_{ii}) - e_i^2} \right]^{1/2}$$

and

$$\left(\frac{h_{ii}}{1 - h_{ii}} \right)^{1/2}$$

The first factor, known as the Studentized Deleted Residual, is effective in capturing extreme Y observations. If an observation y_i is an outlier, this factor will be large in absolute value. Notice that the residual e_i , will also tend to be large.

The second factor is a function of the leverage h_{ii} , with the property that $0 < h_{ii} < 1$. An observation x_i , if it is an outlier, will have high leverage and this second factor will be greater than 1. Therefore, this second factor will be effective in capturing outlying X observations.