

Advanced Regression Methods With R

- Robust Regression
- Weighted Regression
- Ridge Regression

Robust Regression

```
gala <- read.csv("M:/DataMining/Data/galapagos.csv",header=TRUE)
attach(gala)
ols_reg <- lm(Species ~ Area + Elevation + Nearest + Scrub + Adjacent, gala)
summary(ols_reg)
attach(ols_reg)
qqnorm(residuals)

# load package nortest
lillie.test(residuals)
cvm.test(residuals)
```

Call:

```
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,  
    data = gala)
```

Residuals:

```
   Min    1Q  Median    3Q   Max  
-111.679 -34.898  -7.862  33.460 182.584
```

Coefficients:

```
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 7.068221 19.154198  0.369 0.715351  
Area        -0.023938  0.022422 -1.068 0.296318  
Elevation   0.319465  0.053663  5.953 3.82e-06 ***  
Nearest     0.009144  1.054136  0.009 0.993151  
Scruz       -0.240524  0.215402 -1.117 0.275208  
Adjacent    -0.074805  0.017700 -4.226 0.000297 ***
```

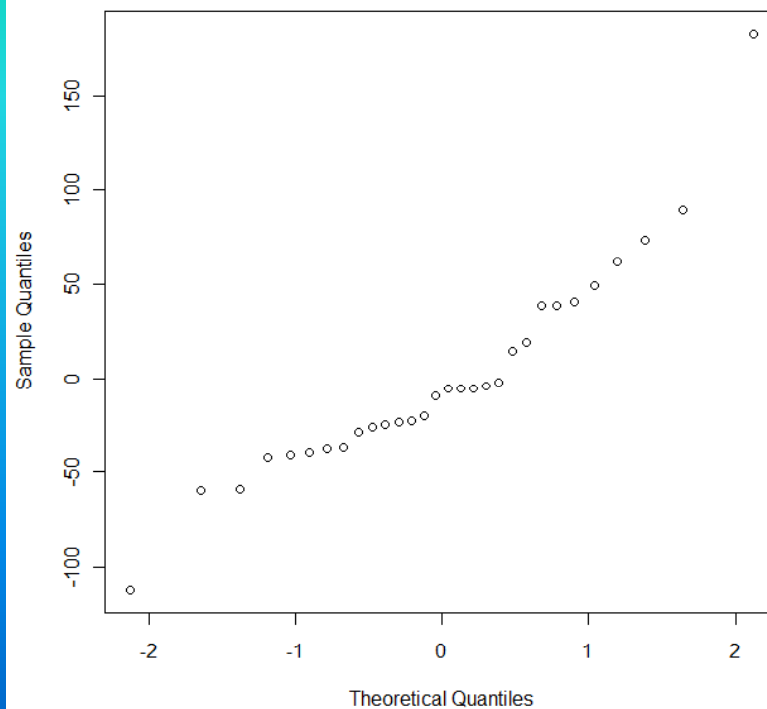
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 60.98 on 24 degrees of freedom

Multiple R-squared: 0.7658, Adjusted R-squared: 0.7171

F-statistic: 15.7 on 5 and 24 DF, p-value: 6.838e-07

Normal Q-Q Plot



```
lillie.test(residuals)
```

Lilliefors (Kolmogorov-Smirnov) normality test

```
data: residuals  
D = 0.1813, p-value = 0.01310
```

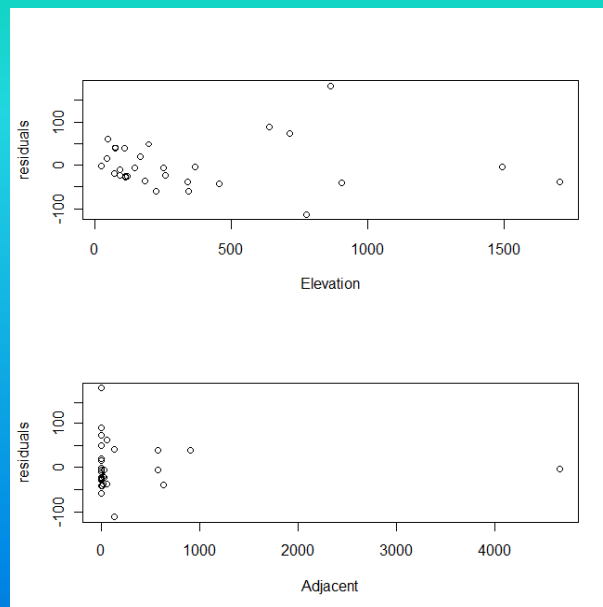
```
cvm.test(residuals)
```

Cramer-von Mises normality test

```
data: residuals  
W = 0.1509, p-value = 0.02149
```

Clearly, residuals are not normally distributed.

```
layout(1:2)  
plot(residuals~Elevation)  
plot(residuals~Adjacent)
```



Robust Regression

- Ordinary Least Squares (OLS) works well when residuals are normal, but performs poorly when residuals are not normal or error variance is not constant.
- Robust Regression Methods –
 - Least Absolute Deviation (LAD) – minimize $\sum |e_i|$
 - Huber's Method – (hybrid of OLS and LAD)
minimize $\sum (e_i)^2$, $|e_i| \leq c$
 $\sum |e_i| - c^2/2$, $|e_i| > c$

To run ROBUST REGRESSION in R, load the package MASS, and read data galapagos.csv.

```
rob_reg <- rlm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent)
summary(rob_reg)
```

```
Call: rlm(formula = Species ~ Area + Elevation + Nearest + Scruz +
  Adjacent)
```

```
Residuals:
```

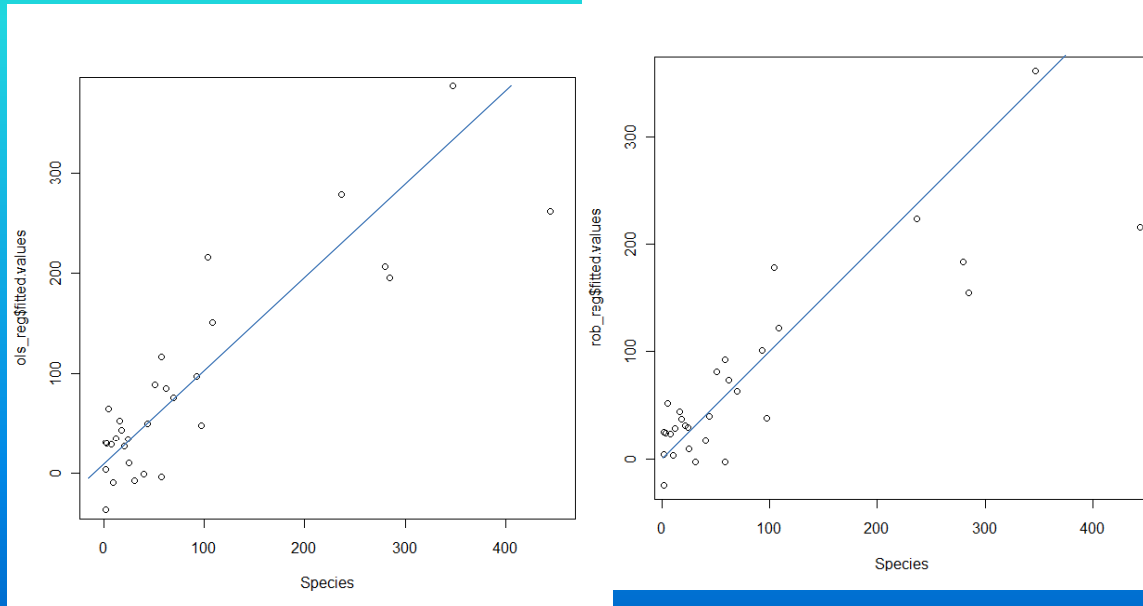
```
  Min   1Q  Median   3Q   Max
-74.389 -18.353 -6.364  21.187 229.082
```

```
Coefficients:
```

```
              Value Std. Error t value
(Intercept)  6.3611  12.3897   0.5134
Area         -0.0061  0.0145  -0.4214
Elevation    0.2476  0.0347   7.1320*
Nearest      0.3592  0.6819   0.5267
Scruz        -0.1952  0.1393  -1.4013
Adjacent     -0.0546  0.0114  -4.7648*
```

```
Residual standard error: 29.76 on 24 degrees of freedom ( * significant)
```

```
# eqsplot (package MASS) plots scatterplot on equal scales
eqsplot(Species, ols_reg$fitted.values)
eqsplot(Species, rob_reg$fitted.values)
```



Weighted Regression

The data file
Bid_Cost.csv shows
the values of X = size
of a bid in million \$,
and Y = cost to the
firm preparing the bid,
for 12 recent bids.

BidSize	BidCost
2.13	15.5
1.21	11.1
11	62.6
6	35.4
5.6	24.9
6.91	28.1
2.97	15
3.35	23.2
10.39	42
1.1	10
4.36	20
8	47.5

WEIGHTED REGRESSION EXAMPLE continued (data file = bid_cost.csv)

```
bid <- read.csv("K:/DataMining/Data/Bid_Cost.csv",header=TRUE)
attach(bid)
```

```
out1 <- lm(BidCost ~ BidSize)
```

Call:
lm(formula = BidCost ~ BidSize)

Residuals:

Min	1Q	Median	3Q	Max
-9.143	-4.090	1.106	3.904	8.703

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.2289	3.2517	1.301	0.223
BidSize	4.5153	0.5285	8.544	6.59e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

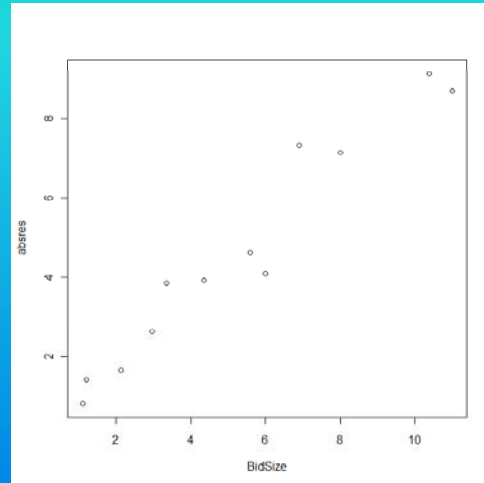
Residual standard error: 5.87 on 10 degrees of freedom
Multiple R-squared: 0.8795, Adjusted R-squared: 0.8675
F-statistic: 73 on 1 and 10 DF, p-value: 6.588e-06

WEIGHTED REGRESSION EXAMPLE continued (data file = bid_cost.csv)

```
absres <- abs(out1$residuals)
plot(absres~BidSize)
```

Absolute residual seems to be a linear function of BidSize. We will fit a 2nd degree equation -

```
BidSize2 <- BidSize^2
res_reg <- lm(absres~BidSize+BidSize2)
summary(res_reg)
```



Call:

```
lm(formula = absres ~ BidSize + BidSize2)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.29732	-0.34033	-0.08335	0.33580	1.21061

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.09063	0.65975	-0.137	0.8938
BidSize	0.99428	0.26562	3.743	0.0046 **
BidSize2	-0.01384	0.02179	-0.635	0.5410

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7118 on 9 degrees of freedom
Multiple R-squared: 0.9492, Adjusted R-squared: 0.938
F-statistic: 84.16 on 2 and 9 DF, p-value: 1.495e-06

Above output shows that:

Abs(residual) is a LINEAR function of BidSize (and not quadratic, since BidSize² term is not significant).

```
out2 <- lm(BidCost~BidSize, weights = 1/BidSize)
summary(out2)
```

OUTPUT FROM OLS

```
lm(formula = BidCost ~ BidSize)
```

Residuals:

Min	1Q	Median	3Q	Max
-9.143	-4.090	1.106	3.904	8.703

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.2289	3.2517	1.301	0.223
BidSize	4.5153	0.5285	8.544	6.59e-06 ***

OUTPUT FROM WEIGHTED LS

```
lm(formula = BidCost ~ BidSize, weights = 1/BidSize)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.2573	1.7352	3.03	0.0127 *
BidSize	4.3195	0.4301	10.04	1.53e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared: 0.9098, Adjusted R-squared: 0.9008

RIDGE REGRESSION

Data = $\{X_{1i}, \dots, X_{pi}, Y_i\}, i = 1, 2, \dots, n$

Matrix $X_{n \times (p+1)}$ = data matrix with 1st column of all 1 (for the intercept term)

OLS estimate of vector $\beta = (\beta_0, \beta_1, \dots, \beta_p)$ is

$$\hat{\beta} = (X'X)^{-1} X'Y$$

Ridge Regression estimate of vector $\beta = (\beta_0, \beta_1, \dots, \beta_p)$ is

$$\hat{\beta}_R = (X'X + kI)^{-1} X'Y$$

where $k > 0$ is a suitably chosen constant.

NOTE: (1) When some of the predictors are correlated, the matrix $(X'X)$ is ill-conditioned, and adding a small positive k reduces ill-conditioning.

Another way to look at the ridge regression estimates is that it minimizes the error sum of squares while keeping all β 's from getting too large.

(2) There are several methods for selecting the ridge constant k . One way is to vary k from 0 to some upper value U (found by trial and error), plot the estimated coefficients of predictors as a function of k , and select smallest k for which the coefficients become stable.

Example (Ridge Regression)

```
navy <- read.csv("K:/DataMining/Data/Navy.csv",header=TRUE)
> attach(navy)
```

```
> navy.out1 <- lm(Y~X1+X2+X3+X4+X5+X6+X7)
> summary(navy.out)
```

Call:

```
lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	148.2206	221.6269	0.669	0.512613
X1	-1.2874	0.8057	-1.598	0.128510
X2	1.8096	0.5152	3.512	0.002673 **
X3	0.5904	1.8001	0.328	0.746931
X4	-21.4817	10.2226	-2.101	0.050823 .
X5	5.6194	14.7562	0.381	0.708056
X6	-14.5147	4.2262	-3.434	0.003163 **
X7	29.3603	6.3704	4.609	0.000250 ***

STATS24x7.com © 2010 ADI-NV, INC

19

```
# load the package HH
library(HH)
```

```
vif(navy.out1)
  X1      X2      X3      X4      X5      X6      X7
2.165539 4.500146 1.405882 2.352975 3.653326 37.184830 63.712775
```

Let us look at correlations among a X5, X6, X7.

```
# combine X5, X6, X7 by columns to create a matrix v so correlation
# can be calculated
```

```
v <- cbind(X5,X6,X7)
cor(v)
  X5      X6      X7
X5 1.0000000 0.6763194 0.7589389
X6 0.6763194 1.0000000 0.9781898
X7 0.7589389 0.9781898 1.0000000
```

STATS24x7.com © 2010 ADI-NV, INC

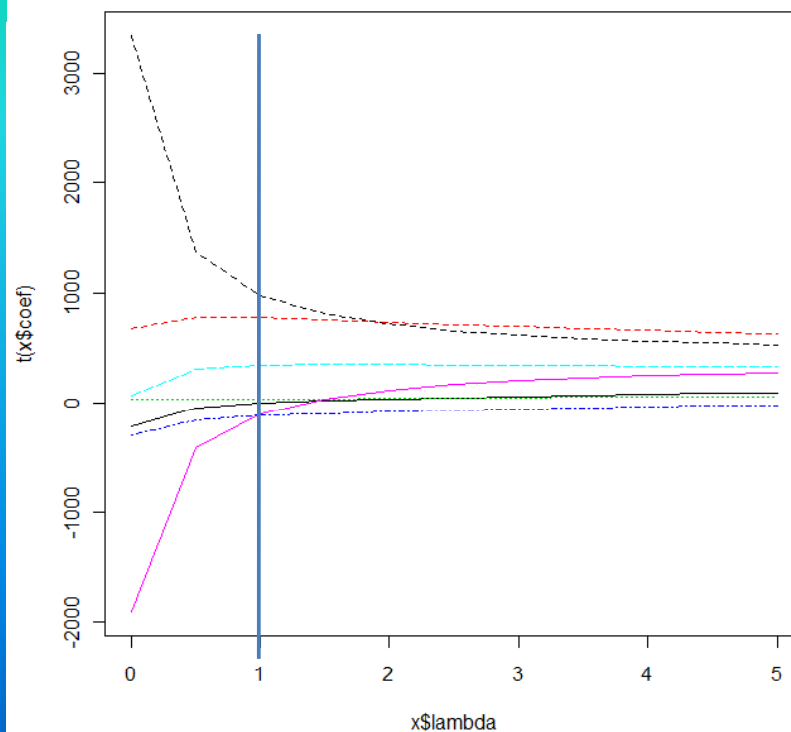
20

Earlier, we had taken care of multicollinearity by removing one of the variables from a correlated subset of predictors.

We will now use ridge regression using the R-package MASS.

```
# load(MASS)
# run lm.ridge and plot the coefficients as function of k(lambda)
plot(lm.ridge(Yc~X1+X2+X3+X4+X5+X6+X7, lambda=seq(0, 5, 0.5)))
```

RIDGE TRACE
PLOT for navy
data



```
out2 <- lm.ridge(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7,  
lambda=1)
```

```
out2$coef
```

X1	X2	X3	X4
-1.169476	779.494163	36.992115	- 117.856248
X5	X6	X7	
346.294850	-95.989837	979.232177	