Midterm Exam

Andus Kong

February 7, 2016

Download the data from CCLE and load it into your workspace.

```r
# write your commands here
download.file(url = "https://ccle.ucla.edu/mod/resource/view.php?id=1021920", destfile = "PCRData.csv")
```

```
## Warning in download.file(url = "https://ccle.ucla.edu/mod/resource/
## view.php?id=1021920", : downloaded length 2806 != reported length 200
```

```r
midtermdata <- read.csv("C:/Users/wkong_000/Desktop/Statistics/Stats 102B/Data Sets/PCRData.csv")
```

Task 1 *Fit a linear model to predict the standing height of a female based on all of the x predictors. [Use function lm()] Print a summary of the linear model. How many coeffecients associated with the variables are significant? Do any of the signs of the coefficients surprise you?*

```r
# write your commands here
m1 <- lm(Y ~ X1 + X2 + X3 +X4 + X5 + X6 + X7 + X8 + X9, data = midtermdata)
summary(m1)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9,
##     data = midtermdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9974 -1.3111  0.0082  1.2150  3.6043
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -4.7038   220.2047  -0.021    0.983
## X1            0.7921     0.1543   5.133 3.36e-05 ***
## X2            1.9907     4.3534   0.457    0.652
## X3           -2.2089     5.0067  -0.441    0.663
## X4            0.7921     0.5474   1.447    0.161
## X5           -0.9127     4.4214  -0.206    0.838
## X6            2.2905     4.6926   0.488    0.630
## X7            0.9582     0.6970   1.375    0.182
## X8            0.8447     1.6465   0.513    0.613
## X9           -0.5118     1.9435  -0.263    0.795
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.881 on 23 degrees of freedom
## Multiple R-squared:  0.8937, Adjusted R-squared:  0.8521
## F-statistic: 21.49 on 9 and 23 DF,  p-value: 3.657e-09
```

```
#####################
# Answer to comments
#####################
# only 1 variable, upper arm length, is significant
# the negative coefficients suprise me because bigger body parts usually mean taller people
######################
```

Some of the coefficients were negative. Apart from the ratio based measurements, it does not seem to make sense that larger measurements of a body part would correspond to a smaller prediction for height. Part of the problem is multicolinearity. Many of the X-variables are correlated with one another. This often means that an increase of one variable cannot be distinguished from an increase of another variable. When fitting linear models, the model gets 'confused' and has a hard time correctly assigning the correct amount of dependence upon the variables.

One way to improve our model is by using backward stepwise selection of variables. This can be achieved using the `step()` function in R. Provide `step` with the full model and tell it that the direction of variable selction is backwards. R will then eliminate variables in the model, one at a time, removing the variable that will influence the AIC criteria the least.

Task 2 *Perform backwards variable selction with the step() function. Fit the resulting model and print its summary.*

```
# write your commands here
step(m1)
```

```
## Start:  AIC=49.8
## Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9
##
##        Df Sum of Sq     RSS    AIC
## - X5    1     0.151  81.569 47.863
## - X9    1     0.245  81.664 47.901
## - X3    1     0.689  82.107 48.080
## - X2    1     0.740  82.158 48.101
## - X6    1     0.843  82.261 48.142
## - X8    1     0.932  82.350 48.177
## <none>               81.418 49.802
## - X7    1     6.691  88.109 50.408
## - X4    1     7.413  88.831 50.678
## - X1    1    93.261 174.679 72.993
##
## Step:  AIC=47.86
## Y ~ X1 + X2 + X3 + X4 + X6 + X7 + X8 + X9
##
##        Df Sum of Sq     RSS    AIC
## - X3    1     0.589  82.158 46.100
## - X2    1     0.650  82.218 46.125
## - X8    1     0.835  82.404 46.199
## - X9    1     2.580  84.149 46.891
## <none>               81.569 47.863
## - X7    1     6.588  88.157 48.426
## - X4    1     7.673  89.242 48.830
## - X6    1    70.270 151.839 66.368
## - X1    1   109.767 191.336 73.998
```

```
##
## Step:  AIC=46.1
## Y ~ X1 + X2 + X4 + X6 + X7 + X8 + X9
##
##        Df Sum of Sq     RSS    AIC
## - X2    1     0.153  82.311 44.162
## - X9    1     2.564  84.722 45.114
## - X8    1     3.057  85.215 45.306
## <none>               82.158 46.100
## - X7    1     6.345  88.502 46.555
## - X4    1     7.675  89.832 47.047
## - X6    1    71.835 153.993 64.833
## - X1    1   111.962 194.120 72.475
##
## Step:  AIC=44.16
## Y ~ X1 + X4 + X6 + X7 + X8 + X9
##
##        Df Sum of Sq     RSS    AIC
## - X9    1     2.512  84.823 43.154
## - X8    1     4.751  87.062 44.014
## <none>               82.311 44.162
## - X7    1     6.456  88.768 44.654
## - X4    1    10.044  92.355 45.961
## - X1    1   113.972 196.284 70.841
## - X6    1   128.365 210.676 73.176
##
## Step:  AIC=43.15
## Y ~ X1 + X4 + X6 + X7 + X8
##
##        Df Sum of Sq     RSS    AIC
## - X8    1     2.727  87.549 42.198
## - X7    1     4.223  89.045 42.757
## <none>               84.823 43.154
## - X4    1    18.107 102.930 47.539
## - X1    1   134.364 219.187 72.483
## - X6    1   151.855 236.677 75.016
##
## Step:  AIC=42.2
## Y ~ X1 + X4 + X6 + X7
##
##        Df Sum of Sq     RSS    AIC
## <none>               87.549 42.198
## - X7    1     7.191  94.740 42.803
## - X4    1    15.511 103.060 45.581
## - X1    1   135.302 222.851 71.030
## - X6    1   157.088 244.637 74.108

##
## Call:
## lm(formula = Y ~ X1 + X4 + X6 + X7, data = midtermdata)
##
## Coefficients:
## (Intercept)           X1           X4           X6           X7
##     17.7654       0.8481       0.9049       1.2886       0.8464
```

```r
m2 <- lm(Y ~ X1 + X4 + X6 + X7, data = midtermdata)
summary(m2)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X4 + X6 + X7, data = midtermdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0196 -1.2973 -0.1849  1.1132  4.4430
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.7654    11.5676   1.536   0.1358
## X1            0.8481     0.1289   6.578 3.92e-07 ***
## X4            0.9049     0.4063   2.227   0.0341 *
## X6            1.2886     0.1818   7.088 1.04e-07 ***
## X7            0.8464     0.5581   1.516   0.1406
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.768 on 28 degrees of freedom
## Multiple R-squared:  0.8857, Adjusted R-squared:  0.8694
## F-statistic: 54.24 on 4 and 28 DF,  p-value: 8.702e-13
```

If we go back to the full model with all of the x-variables used as predictors, we can see that part of the problem is that many of the variables are correlated with one another.

Task 3 *Print out the correlation matrix between x-variables. Round the results to two decimal places. Which pairs of variables appear to be highly correlated with one another?*

```r
# write your commands here
cor(midtermdata[2:10])
```

```
##            X1         X2        X3          X4          X5          X6
## X1 1.00000000  0.1440924 0.2791324  0.14830208  0.18633213 0.226357258
## X2 0.14409236  1.0000000 0.4707681  0.64522081  0.71597688 0.661645525
## X3 0.27913242  0.4707681 1.0000000  0.50504720  0.36582628 0.728430798
## X4 0.14830208  0.6452208 0.5050472  1.00000000  0.60074246 0.549978622
## X5 0.18633213  0.7159769 0.3658263  0.60074246  1.00000000 0.714977891
## X6 0.22635726  0.6616455 0.7284308  0.54997862  0.71497789 1.000000000
## X7 0.36800330  0.1467500 0.4276572  0.34707842 -0.02977045 0.282080993
## X8 0.11456027 -0.5816926 0.4425459 -0.19007952 -0.38708026 0.002849043
## X9 0.02296179 -0.1007389 0.4396086 -0.09938554 -0.41186673 0.341416268
##             X7           X8          X9
## X1  0.36800330  0.114560275  0.02296179
## X2  0.14675004 -0.581692567 -0.10073892
## X3  0.42765715  0.442545869  0.43960859
## X4  0.34707842 -0.190079517 -0.09938554
## X5 -0.02977045 -0.387080265 -0.41186673
## X6  0.28208099  0.002849043  0.34141627
## X7  1.00000000  0.243357197  0.39776808
```

```
## X8  0.24335720  1.000000000  0.50849160
## X9  0.39776808  0.508491601  1.00000000
```

```
##################
# Answers to questions
##################
# Upper Arm Length is positively correlated with Hand Length, Upper Leg Length, and Lower Leg Length
# Upper Leg Length is positively correlated with Lower Leg Length and Hand Length
# There are a few others but the ones mentioned about are the most correlated
# Lower Leg length is postively correlated with Forearm Length
##################
```

One method to address this issue of multicolinearity is through Principal Components Regression. The big idea is this: perform principal components analysis on the matrix of predictor variables. The resulting principal components will be uncorrelated with each other. Regress the Y variable against the principal components.

Task 4 *Perform principal components analysis using the correlation matrix of the matrix of x-variables. Print out the resulting PCA loadings (eigenvector matrix). Reexpress the x-variable data in its principal components. Don't forget to center the X matrix before doing your analysis.*

```
# write your commands here
x <- midtermdata[2:10]
means <- colMeans(x)
xc <- (apply(x, 1, FUN = function(x) x - means))
xc <- (t(xc))
r <- cor(xc)
e <- eigen(r)
Qr <- e$vectors
xcs <- scale(x)
pc <- xcs %*% Qr
Qr
```

```
##              [,1]        [,2]        [,3]       [,4]       [,5]
##  [1,] -0.18536481 -0.15297190  0.80199329 -0.2796327  0.3690077
##  [2,] -0.44137447  0.23470984 -0.09736505  0.2322078  0.2542986
##  [3,] -0.39326301 -0.33377183 -0.16609566 -0.2330887 -0.1218365
##  [4,] -0.41830380  0.08072687  0.02751068  0.2041069 -0.5771173
##  [5,] -0.41268584  0.29967465 -0.01293216 -0.3506906 -0.0544079
##  [6,] -0.46445423 -0.10120241 -0.25221653 -0.1634478  0.2718401
##  [7,] -0.21398124 -0.35780291  0.37983818  0.5839423 -0.2176867
##  [8,]  0.08510891 -0.54614454 -0.05312137 -0.4555959 -0.3660767
##  [9,] -0.04621437 -0.52654232 -0.32878999  0.2713665  0.4393179
##              [,6]        [,7]          [,8]         [,9]
##  [1,]  0.23296940 -0.17413203 -0.0003038997  0.009399926
##  [2,]  0.31883601  0.39764116  0.5775622175 -0.169347845
##  [3,]  0.31642689  0.49617008 -0.5132376010  0.165694754
##  [4,]  0.37180134 -0.55206581  0.0000343842  0.003377492
##  [5,] -0.46627870 -0.02723322  0.1785987972  0.603091308
##  [6,] -0.37946995 -0.27926740 -0.1830834850 -0.595237501
##  [7,] -0.48185955  0.24766543  0.0017569453 -0.002455874
##  [8,] -0.03655144  0.04167094  0.5659445511 -0.163448008
##  [9,]  0.10377255 -0.34466154  0.1315125155  0.446115119
```

Now that we have expressed our data in its principal components, we will attempt to perform regression again.

Task 5 *Fit a linear regression model to predict the standing heights (Y) based on the Principal Components of X. Print a summary of the resulting linear model.*

```
# write your commands here
Y <- data.frame(midtermdata[,1])
colnames(Y) <- "Y"
dat <- data.frame(Y, pc)
m3 <- lm(Y ~ X1 + X2 + X3 +X4 + X5 + X6 + X7 + X8 + X9, data = dat)
summary(m3)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9,
##     data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9974 -1.3111  0.0082  1.2150  3.6043
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 164.5636     0.3275 502.452  < 2e-16 ***
## X1           -2.1436     0.1746 -12.279 1.39e-11 ***
## X2           -0.8031     0.2128  -3.774 0.000984 ***
## X3            1.3435     0.3305   4.065 0.000478 ***
## X4           -0.8796     0.3799  -2.316 0.029849 *
## X5            0.6313     0.4255   1.484 0.151452
## X6           -0.5816     0.6048  -0.962 0.346174
## X7           -1.2065     0.6897  -1.749 0.093562 .
## X8            3.9995    11.7267   0.341 0.736153
## X9           -6.5854    16.1070  -0.409 0.686429
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.881 on 23 degrees of freedom
## Multiple R-squared:  0.8937, Adjusted R-squared:  0.8521
## F-statistic: 21.49 on 9 and 23 DF,  p-value: 3.657e-09
```

We may find that many of the principal components do not contribute significantly towards predicting Y. We can eliminate the latter principal components, as they contribute the least towards capturing the variation in X.

Task 6 *Improve the linear regression model by removing some of the latter principal components. It may be safer to eliminate one or two principal components at a time and check the model fit. Print a summary of the final linear model.*

```
# write your commands here
pc1 <- xcs %*% Qr[,1:8]
dat <- data.frame(Y, pc1)
```

```r
m4 <- lm(Y ~ X1 + X2 + X3 +X4 + X5 + X6 + X7 + X8, data = dat)
pc2 <- xcs %*% Qr[,1:3]
dat <-data.frame(Y, pc2)
m5 <- lm(Y ~ X1 + X2 + X3, data = dat)
summary(m5)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3, data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3840 -1.3302 -0.1698  1.8581  3.9423
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 164.5636     0.3589 458.461  < 2e-16 ***
## X1           -2.1436     0.1913 -11.204 4.71e-12 ***
## X2           -0.8031     0.2332  -3.444 0.001767 **
## X3            1.3435     0.3622   3.709 0.000875 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.062 on 29 degrees of freedom
## Multiple R-squared:  0.839,  Adjusted R-squared:  0.8224
## F-statistic: 50.38 on 3 and 29 DF,  p-value: 1.278e-11
```

Task 7

Compare the summary information of the model resulting from Backwards elimination (Task 2) and from Principal components regression (Task 6). Which model do you think does a better job?

ANSWER:

I believe the variable selection method does a better job in this case. The R-squared is higher for the Backwards Elimination regression than the R-squared for the principal component regression, which indicates that the stepwise regression model captures more of the variance. Also the Backwards Elimination model makes more sense in that the coefficients of the model are positive which we would expect the relationship to be. The Principal Component Regression, however, has negative coefficients which does not make much sense.

Variable selection methods and Principal Components Regression are different approaches to dealing with issues of multicolinearity and dimension reduction. Each method has its own strengths and may be more appropriate to use in different situations.