Life Expectancy Model

Junchi Zhang

2018/12/15

There are 19 variables in the data set:

adm: Adult Mortality Rates of both sexes (probability of dying between 15 and 60 years per 1000 population)

ind: Number of Infant Deaths per 1000 population

alcohol: recorded per capita (15+) consumption (in litres of pure alcohol)

exp: Expenditure on health as a percentage of Gross Domestic Product per capita(%)
hb: Hepatitis B (HepB) immunization coverage among 1-year-olds (%)

measles: Hepatitis B (HepB) immunization coverage among 1-year-olds (%)

bmi: Average Body Mass Index of entire population

death5: Number of under-five deaths per 1000 population
polio: Pol3 immunization coverage among 1-year-olds (%)

texp: General government expenditure on health as a percentage of total government expenditure (%)
diph: Diphtheria tetanus toxoid and pertussis (DTP3) immunization coverage among 1-year-olds (%)

hiv: Deaths per 1 000 live births HIV/AIDS (0-4 years)
thinness18: Prevalence of thinness among children and adolescents for Age 10 to 19 (%)
thinness59: Prevalence of thinness among children for Age 5 to 9(%)

income: Human Development Index in terms of income composition of resources (index ranging from 0 to 1)

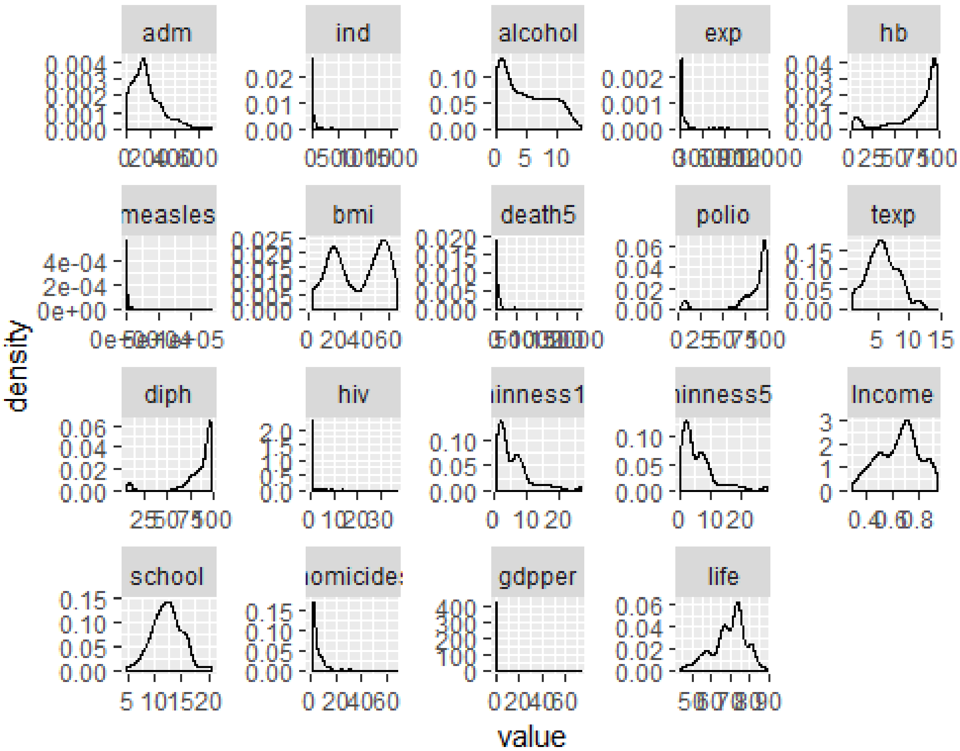
school: Number of years of Schooling(years)

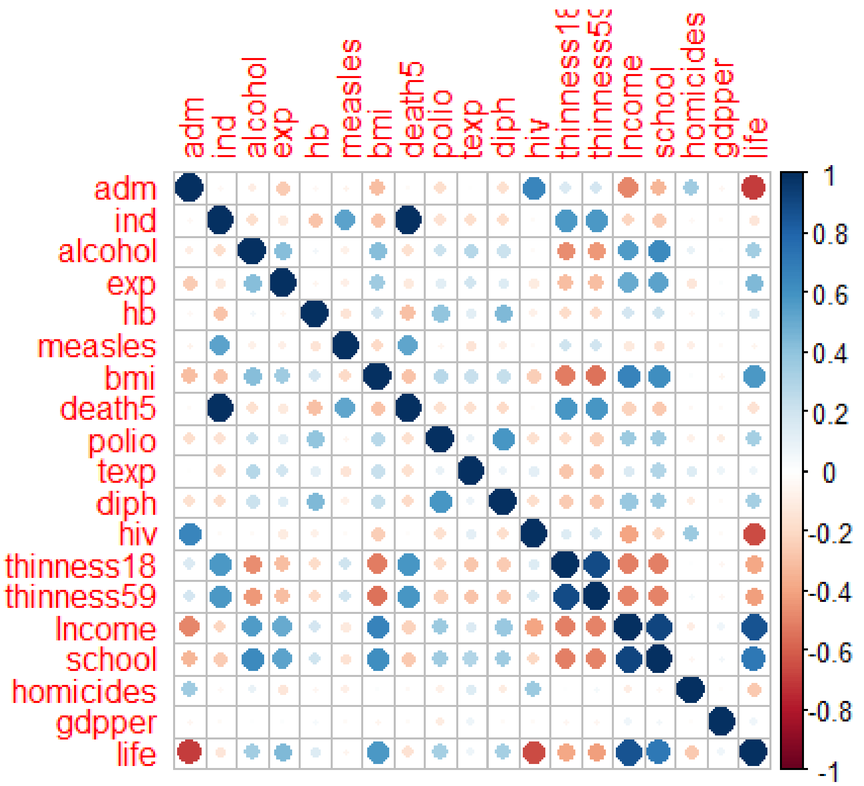
homicides: Intentional homicides (per 100,000 people)
gdpper: Gross Domestic Product per capita (in USD)/Population of the country

life: life expectancy in age

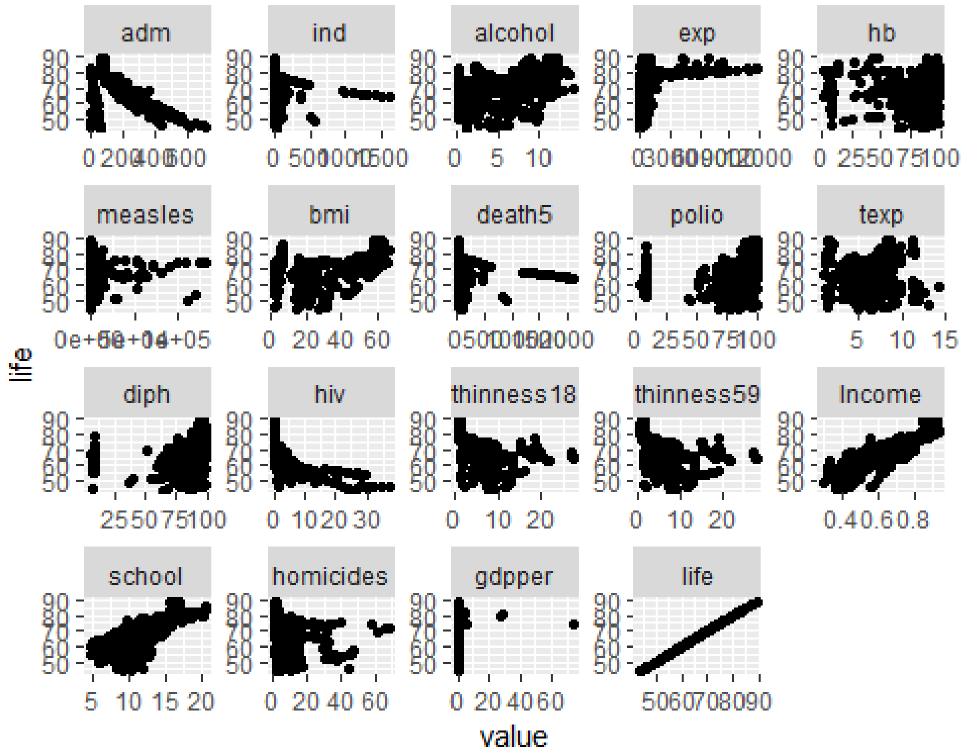
Here, we want to predict the life expectancy, the last variable above, using variables from the 18 variables listed above life expectancy and we have 653 data points in the data set.

Checking the distribution and correlation of each variable. By looking at the correlation plot and correlation martix, we can see that these pairs have high correlation: ind and death5, life and adm, life and income, thinness18 and thinness59, school and income.





By looking at the bivariate correlation plot, it can be seen that, income and school seem to have a linear bivariate relationship with life.



Part A Variable Selection

Looking at stepwise forward and backward selection using AIC. The forward selection generates result: life ~ Income + hiv + adm + school + exp + bmi + alcohol + texp + death5 + ind + gdpper + homicides. Backward selection generates: life ~ adm + ind + alcohol + exp + bmi + death5 + texp + hiv + Income + school + homicides + gdpper. Thus, we have different variables chose by stepwise forward and backward selection. Comparing the AIC value, the second model has a lower AIC value thus variables by backward selection here is more preferred.

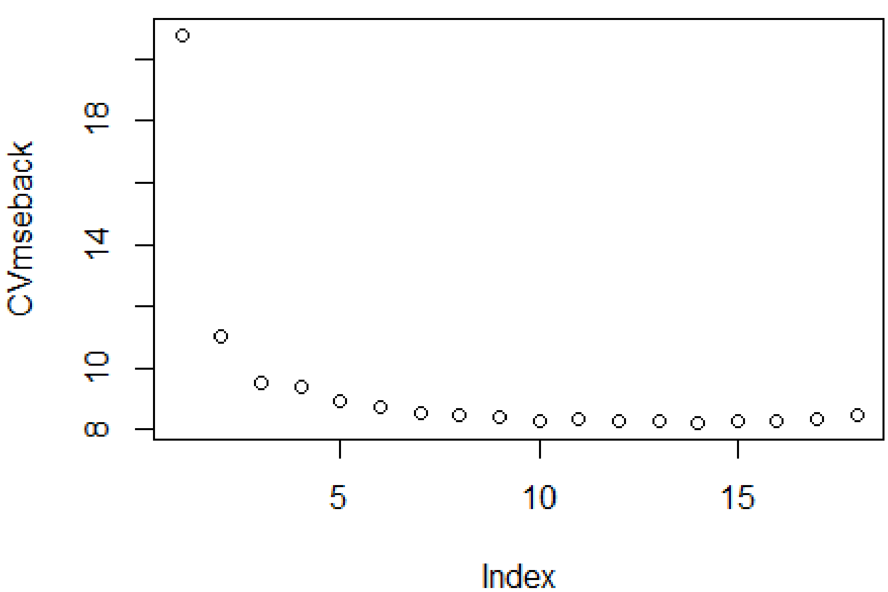
```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## life ~ 1
##
## Final Model:
## life ~ Income + hiv + adm + alcohol + exp + school + texp + homicides +
##     polio
##
##
##          Step Df    Deviance Resid. Df Resid. Dev      AIC
## 1                                   652  53982.310 2884.887
## 2     + Income  1 40480.71726       651  13501.592 1981.928
## 3        + hiv  1  6376.90343       650   7124.689 1566.503
## 4        + adm  1  1089.15253       649   6035.536 1460.169
## 5    + alcohol  1   267.88218       648   5767.654 1432.524
## 6        + exp  1   110.78879       647   5656.865 1421.858
## 7     + school  1    64.12591       646   5592.740 1416.414
## 8       + texp  1    58.60383       645   5534.136 1411.535
## 9  + homicides  1    52.86488       644   5481.271 1407.267
## 10     + polio  1    25.22809       643   5456.043 1406.255
```

```
## Stepwise Model Path 
## Analysis of Deviance Table
## 
## Initial Model:
## life ~ adm + ind + alcohol + exp + hb + measles + bmi + death5 + 
##     polio + texp + diph + hiv + thinness18 + thinness59 + Income + 
##     school + homicides + gdpper
## 
## Final Model:
## life ~ adm + ind + alcohol + exp + death5 + texp + hiv + thinness18 + 
##     thinness59 + Income + school + homicides
## 
## 
##        Step Df   Deviance Resid. Df Resid. Dev      AIC
## 1                               634   5134.149 1384.546
## 2  - gdpper  1  0.1270591       635   5134.276 1382.562
## 3    - diph  1  1.5699791       636   5135.846 1380.762
## 4 - measles  1  2.5175244       637   5138.363 1379.082
## 5     - bmi  1  3.3925462       638   5141.756 1377.513
## 6   - polio  1 11.0289526       639   5152.785 1376.912
## 7      - hb  1  9.4122074       640   5162.197 1376.104
```

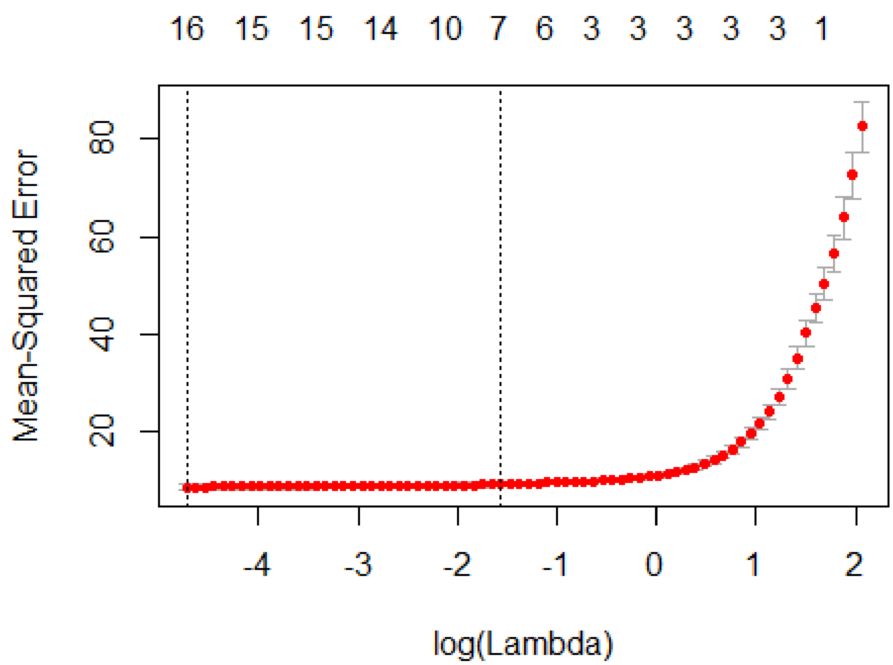
The plot of MSE shows that the 12th model in the backward selection has the lowest MSE. In this case, these variables are chosen: adm, ind, alcohol, exp, death5, texp, hiv, thinness18, thinness59, Income, school, homicides.

```
## Subset selection object
## Call: regsubsets.formula(life ~ ., data, nvmax = 18, method = "backward")
## 18 Variables  (and intercept)
##              Forced in Forced out
## adm              FALSE      FALSE
## ind              FALSE      FALSE
## alcohol          FALSE      FALSE
## exp              FALSE      FALSE
## hb               FALSE      FALSE
## measles          FALSE      FALSE
## bmi              FALSE      FALSE
## death5           FALSE      FALSE
## polio            FALSE      FALSE
## texp             FALSE      FALSE
## diph             FALSE      FALSE
## hiv              FALSE      FALSE
## thinness18       FALSE      FALSE
## thinness59       FALSE      FALSE
## Income           FALSE      FALSE
## school           FALSE      FALSE
## homicides        FALSE      FALSE
## gdpper           FALSE      FALSE
## 1 subsets of each size up to 18
## Selection Algorithm: backward
##           adm ind alcohol exp hb  measles bmi death5 polio texp diph hiv
## 1  ( 1 )  " " " " " "     " " " " " "     " " " "    " "   " "  " "  " "
## 2  ( 1 )  " " " " " "     " " " " " "     " " " "    " "   " "  " "  "*"
## 3  ( 1 )  "*" " " " "     " " " " " "     " " " "    " "   " "  " "  "*"
## 4  ( 1 )  "*" " " " "     " " " " " "     " " "*"    " "   " "  " "  "*"
## 5  ( 1 )  "*" "*" " "     " " " " " "     " " "*"    " "   " "  " "  "*"
```

```
## 6  ( 1 )  "*" "*" "*"     " " " " " "     " " "*"    " "   " "  " "  "*"
## 7  ( 1 )  "*" "*" "*"     "*" " " " "     " " "*"    " "   " "  " "  "*"
## 8  ( 1 )  "*" "*" "*"     "*" " " " "     " " "*"    " "   " "  " "  "*"
## 9  ( 1 )  "*" "*" "*"     "*" " " " "     " " "*"    " "   "*"  " "  "*"
## 10  ( 1 ) "*" "*" "*"     "*" " " " "     " " "*"    " "   "*"  " "  "*"
## 11  ( 1 ) "*" "*" "*"     "*" " " " "     " " "*"    " "   "*"  " "  "*"
## 12  ( 1 ) "*" "*" "*"     "*" " " " "     " " "*"    " "   "*"  " "  "*"
## 13  ( 1 ) "*" "*" "*"     "*" "*" " "     " " "*"    " "   "*"  " "  "*"
## 14  ( 1 ) "*" "*" "*"     "*" "*" " "     " " "*"    "*"   "*"  " "  "*"
## 15  ( 1 ) "*" "*" "*"     "*" "*" " "     "*" "*"    "*"   "*"  " "  "*"
## 16  ( 1 ) "*" "*" "*"     "*" "*" "*"     "*" "*"    "*"   "*"  " "  "*"
## 17  ( 1 ) "*" "*" "*"     "*" "*" "*"     "*" "*"    "*"   "*"  "*"  "*"
## 18  ( 1 ) "*" "*" "*"     "*" "*" "*"     "*" "*"    "*"   "*"  "*"  "*"
##           thinness18 thinness59 Income school homicides gdpper
## 1  ( 1 )  " "        " "        "*"    " "    " "       " "
## 2  ( 1 )  " "        " "        "*"    " "    " "       " "
## 3  ( 1 )  " "        " "        "*"    " "    " "       " "
## 4  ( 1 )  " "        " "        "*"    " "    " "       " "
## 5  ( 1 )  " "        " "        "*"    " "    " "       " "
## 6  ( 1 )  " "        " "        "*"    " "    " "       " "
## 7  ( 1 )  " "        " "        "*"    " "    " "       " "
## 8  ( 1 )  " "        " "        "*"    "*"    " "       " "
## 9  ( 1 )  " "        " "        "*"    "*"    " "       " "
## 10  ( 1 ) " "        " "        "*"    "*"    "*"       " "
## 11  ( 1 ) "*"        " "        "*"    "*"    "*"       " "
## 12  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 13  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 14  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 15  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 16  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 17  ( 1 ) "*"        "*"        "*"    "*"    "*"       " "
## 18  ( 1 ) "*"        "*"        "*"    "*"    "*"       "*"
```



From the plot of Lasso regression, lowest MSE is generated by16 variables. diph and gdpper are excluded. Since the backward selection also dropped these two variables, we no longer consider diph and gdpper.




```
##           adm           ind       alcohol           exp            hb
## -1.127892e-02  1.923228e-02 -1.813417e-01  2.682160e-04 -5.062423e-03
##       measles           bmi        death5         polio          texp
##  1.134366e-05  3.884143e-03 -1.615317e-02  7.786325e-03  1.526233e-01
##          diph           hiv    thinness18    thinness59        Income
##  0.000000e+00 -3.890115e-01  1.261709e-01 -7.275758e-02  4.810183e+01
##        school     homicides        gdpper
## -3.874590e-01 -3.283796e-02  0.000000e+00
```

Part B Modeling

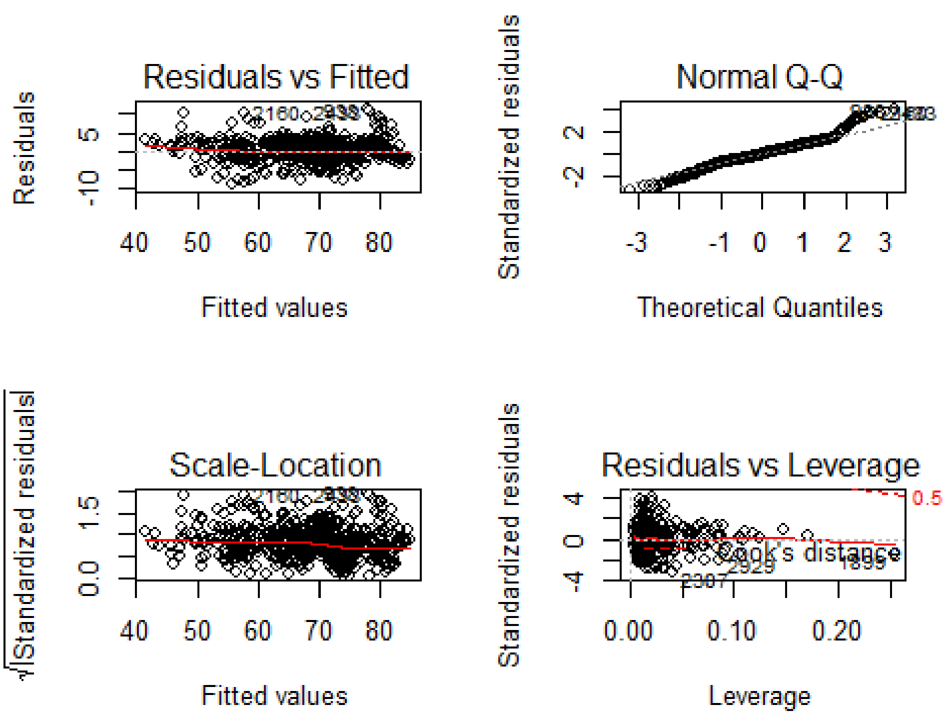
By variable selection, these variables are chosen: adm, ind, alcohol, exp, death5, texp, hiv, thinness18, thinness59, Income, school, homicides. Thus, we would fit different kinds of regression based on these variables from now on.

First, we fit a linear regression. It generates low MSE and by looking at the summary, all variables are significant. The model has MSE 8.19.

```
## Call:
## lm(formula = life ~ adm + ind + alcohol + exp + death5 + texp +
##     hiv + thinness18 + thinness59 + Income + school + homicides,
##     data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7155 -1.5413 -0.0551  1.6762 11.0864
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.528e+01  9.212e-01  49.148  < 2e-16 ***
```

```
## adm         -1.143e-02  1.246e-03  -9.177  < 2e-16 ***
## ind          6.240e-02  1.170e-02   5.333 1.34e-07 ***
## alcohol     -1.513e-01  3.972e-02  -3.810 0.000152 ***
## exp          2.900e-04  7.633e-05   3.799 0.000159 ***
## death5      -4.836e-02  8.868e-03  -5.453 7.08e-08 ***
## texp         1.634e-01  5.260e-02   3.107 0.001977 **
## hiv         -3.756e-01  2.852e-02 -13.170  < 2e-16 ***
## thinness18   1.490e-01  4.935e-02   3.019 0.002639 **
## thinness59  -9.844e-02  4.801e-02  -2.050 0.040742 *
## Income       4.791e+01  2.292e+00  20.907  < 2e-16 ***
## school      -4.072e-01  1.155e-01  -3.525 0.000453 ***
## homicides   -3.696e-02  1.342e-02  -2.754 0.006058 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.84 on 640 degrees of freedom
## Multiple R-squared:  0.9044, Adjusted R-squared:  0.9026
## F-statistic: 504.4 on 12 and 640 DF,  p-value: < 2.2e-16

## [1] 8.19
```

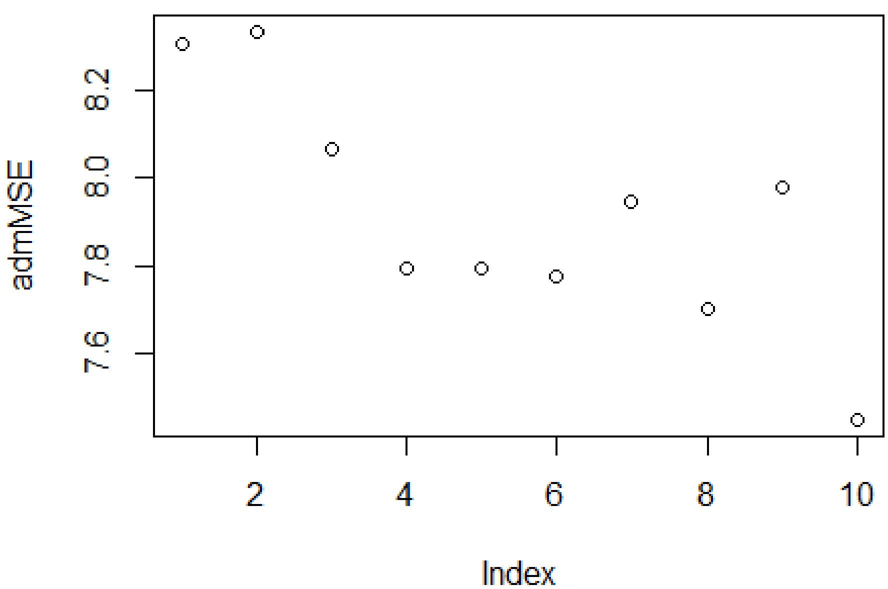
Checking the residual plot vs fitted plot and the square standardized residual vs fitted plot, we can see that the red lines are close to the horizontal line of 0. This means that we do not really need to transform the dependent variable. The two plots also show the residuals are independent. However, the normal qq plot seems to suggest non-normality.

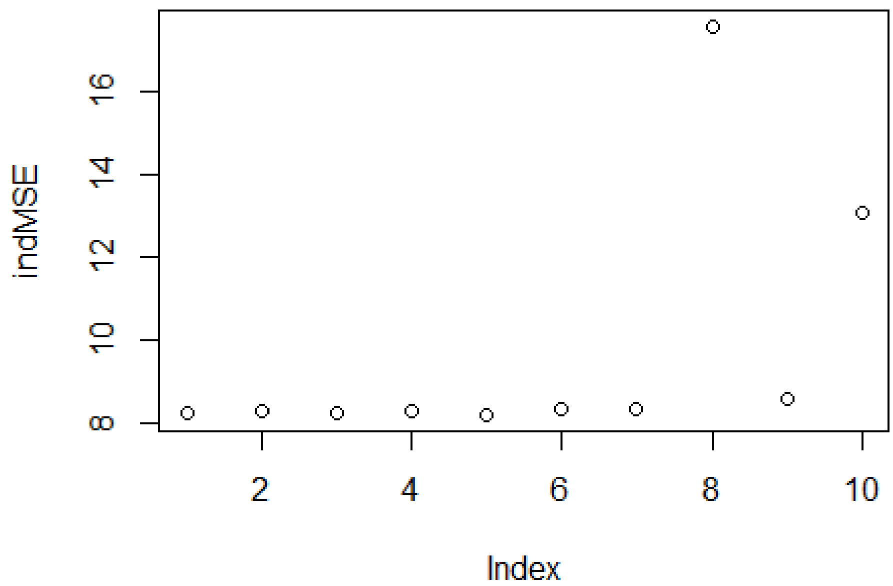


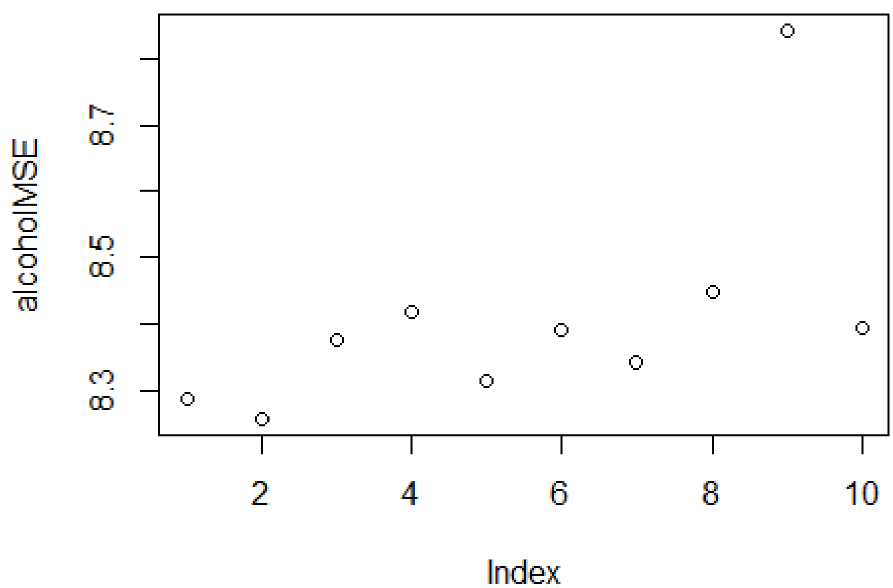
By looking at the table below, we compare the confidence interval of normal approximation to that of bootstrap. It shows that for every variable, the confidence intervals of normal approximation and bootstrap overlap.

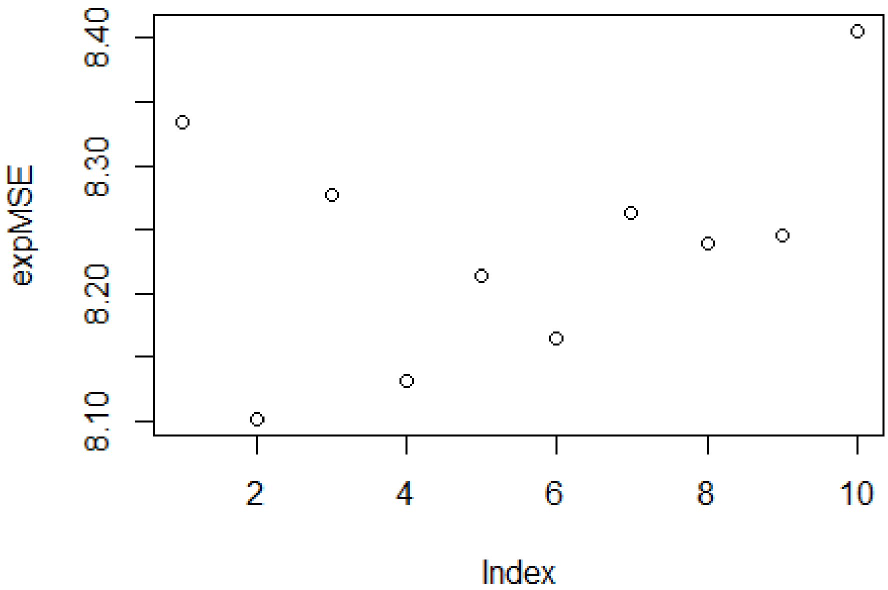
```
               2.5 %   97.5 %     bootstrap 95%
(Intercept) 43.46750 47.08549   (43.3, 47.3 )
adm         -0.01388 -0.00898 (-0.0150, -0.0083 )
ind          0.03942  0.08538 ( 0.0387,  0.0816 )
alcohol     -0.22931 -0.07333 (-0.232, -0.076 )
exp          0.00014  0.00044 ( 0.0002,  0.0005 )
death5      -0.06577 -0.03094 (-0.0633, -0.0304 )
texp         0.06011  0.26669 ( 0.0259,  0.2864 )
hiv         -0.43156 -0.31957 (-0.446, -0.302 )
thinness18   0.05207  0.24588 ( 0.0661,  0.2113 )
thinness59  -0.19272 -0.00416 (-0.1634, -0.0154 )
Income      43.41042 52.41010 (43.3, 52.7 )
school      -0.63407 -0.18041 (-0.630, -0.177 )
homicides   -0.06332 -0.01061 (-0.0755, -0.0100 )
```

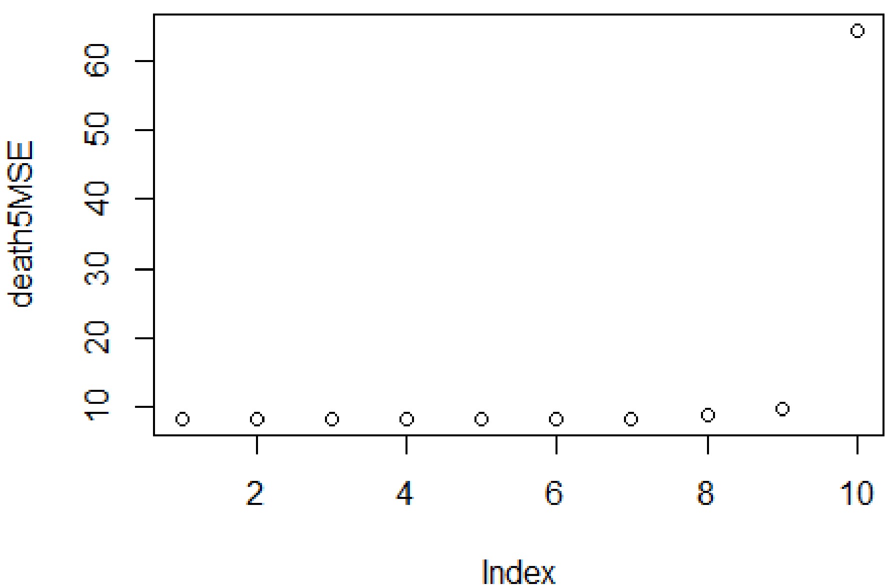
By trying polynomial with degree from 1 to 10 for every variable in the linear model we can see that the degree of polynomial which generates the best MSE result for each variable is (the degree is on the right of each variable): adm 10, ind 3, alcohol 2, exp 6, death5 1, texp 3, hiv 4, thinness18 10, thinness59 10, Income 7, school 8, homicides 5. We can see that many of the degrees are large which seems to be a bit overfitting. Therefore, for those which has small difference between lower degree and higher degree MSE, I will choose to use lower degree. So here are the ones I choose to have lower degree: ind 1, texp 2, texp 1, hiv 1, homicides 1. Thus here is the polynomial model: glm(life~poly(adm,10) + ind+ poly(alcohol,2) + poly(exp,2) + death5+ texp+ hiv+ poly(thinness18,10) + poly(thinness59,10) + poly(Income,7)+ poly(school,8)+homicides,data=data). The polynomial model has MSE=6.61

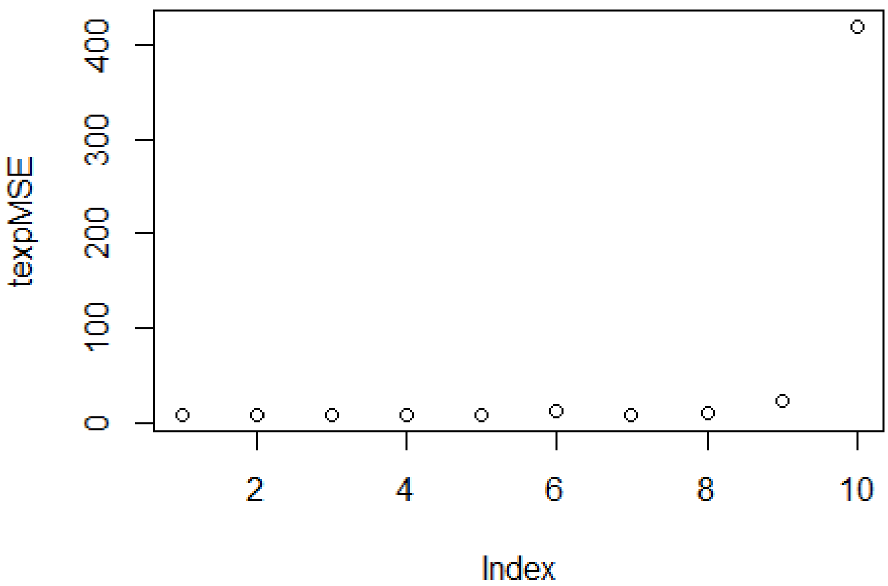


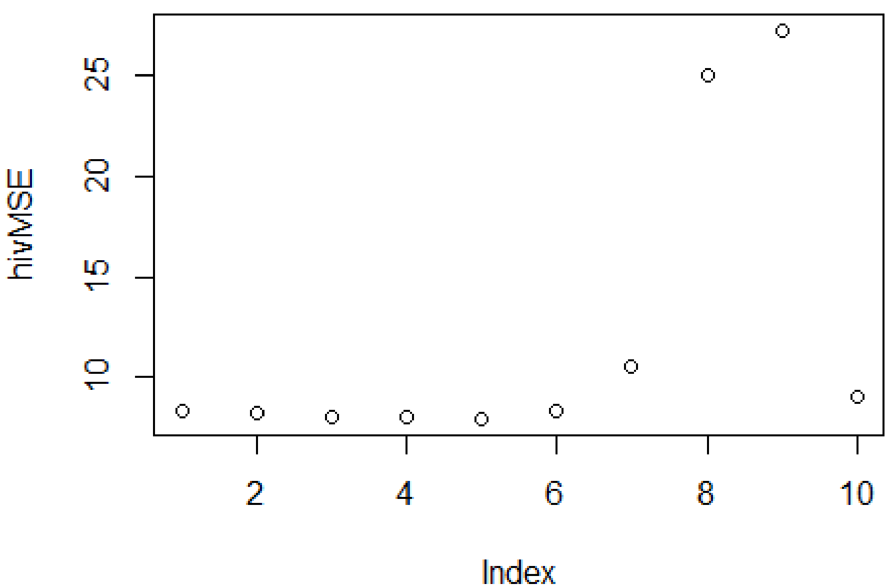

indMSE
8
10
12
14
16
2
4
6
8
10
Index

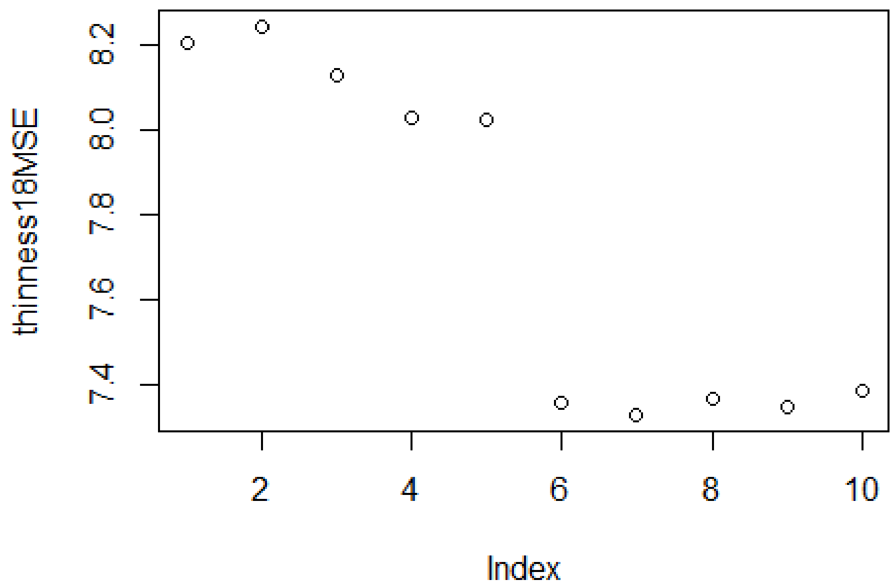

alcoholMSE
8.3
8.5
8.7
2
4
6
8
10
Index

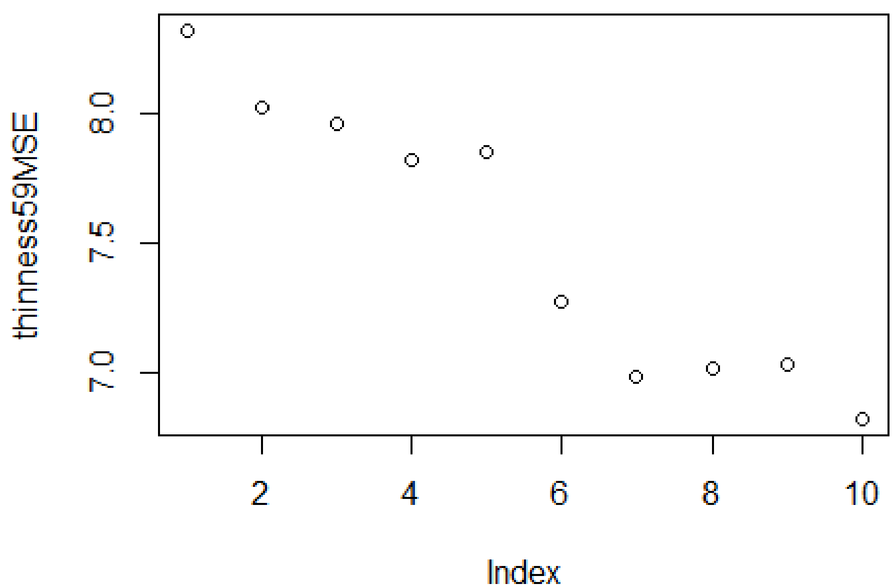

8.40
8.30
8.20
8.10
expMSE
2
4
6
8
10
Index

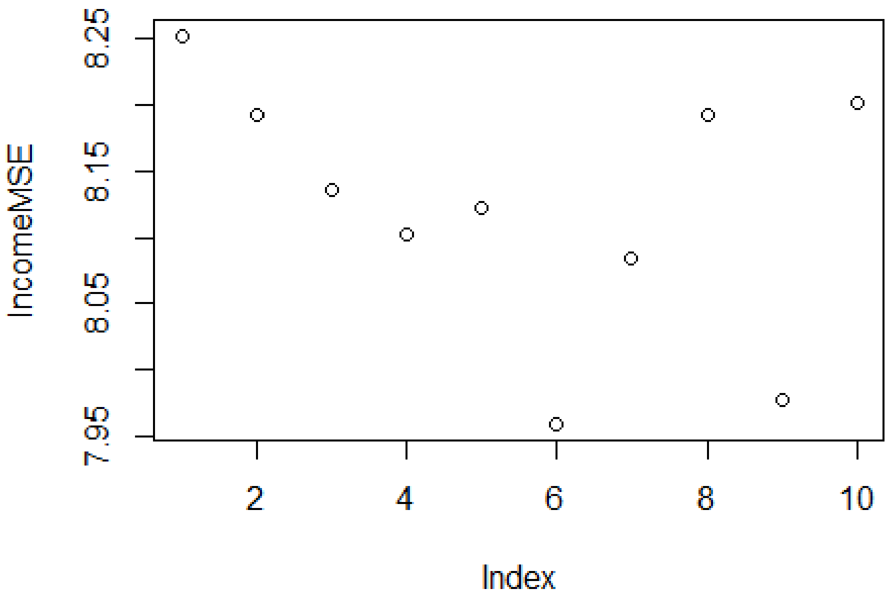

60
50
40
30
20
10
death5MSE
2
4
6
8
10
Index

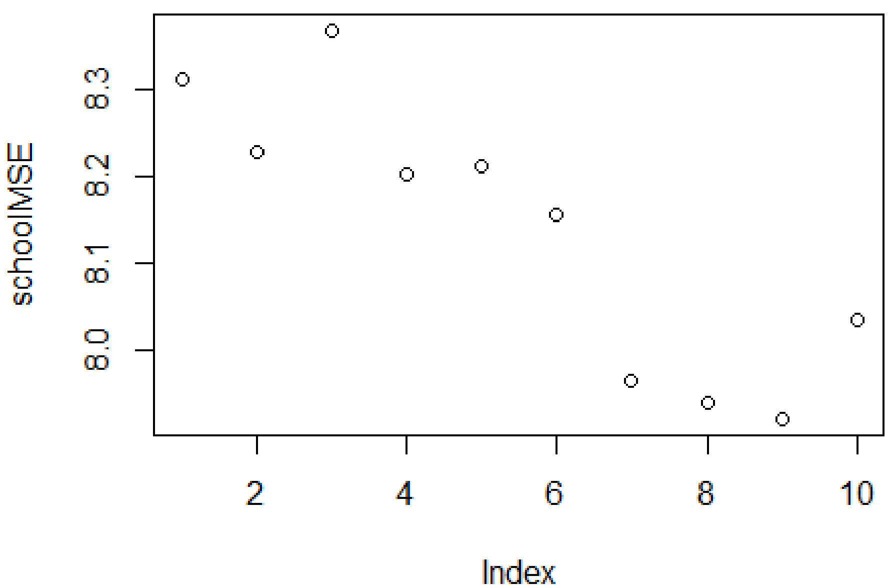

texpMSE
0
100
200
300
400
2
4
6
8
10
Index

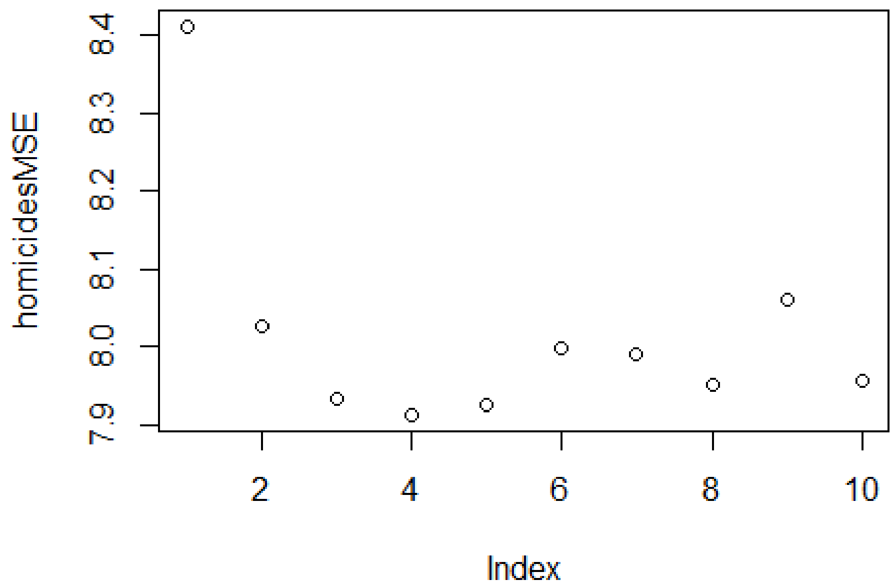

hivMSE
10
15
20
25
2
4
6
8
10
Index


thinness18MSE
8.2
8.0
7.8
7.6
7.4
2
4
6
8
10
Index


thinness59MSE
8.0
7.5
7.0
2
4
6
8
10
Index


IncomeMSE
8.25
8.15
8.05
7.95
2
4
6
8
10
Index

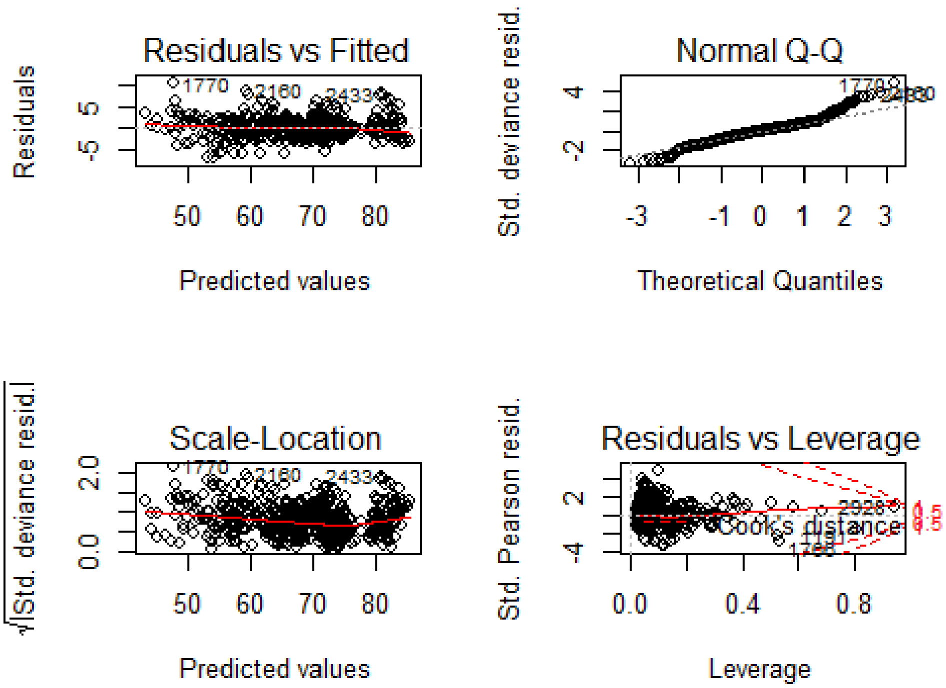

schoolMSE
8.3
8.2
8.1
8.0
2
4
6
8
10
Index




```
## Call:
## lm(formula = life ~ poly(adm, 10) + ind + poly(alcohol, 2) +
##     poly(exp, 2) + death5 + texp + hiv + poly(thinness18, 10) +
##     poly(thinness59, 10) + poly(Income, 7) + poly(school, 8) +
##     homicides, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.964 -1.233  0.016  1.079 10.313
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             70.31051    0.32293  217.72  < 2e-16 ***
## poly(adm, 10)1         -38.34993    4.11112   -9.33  < 2e-16 ***
## poly(adm, 10)2         -12.02157    3.91320   -3.07  0.00222 **
## poly(adm, 10)3          27.03147    2.78009    9.72  < 2e-16 ***
## poly(adm, 10)4          -6.69796    2.97413   -2.25  0.02468 *
## poly(adm, 10)5          -9.68595    2.92296   -3.31  0.00098 ***
## poly(adm, 10)6          13.29256    2.67047    4.98  8.4e-07 ***
## poly(adm, 10)7          -8.57393    2.69437   -3.18  0.00154 **
## poly(adm, 10)8           7.14352    2.71592    2.63  0.00875 **
## poly(adm, 10)9           0.38721    2.77535    0.14  0.88909
## poly(adm, 10)10         -2.49947    2.85593   -0.88  0.38182
## ind                      0.05047    0.01218    4.14  3.9e-05 ***
## poly(alcohol, 2)1       -0.46938    3.70689   -0.13  0.89928
## poly(alcohol, 2)2       -6.04194    2.80983   -2.15  0.03193 *
## poly(exp, 2)1           -5.58171    3.83866   -1.45  0.14645
## poly(exp, 2)2           -3.59075    3.36738   -1.07  0.28670
## death5                  -0.04316    0.00921   -4.69  3.4e-06 ***
## texp                     0.04827    0.04665    1.03  0.30119
## hiv                     -0.43353    0.03821  -11.35  < 2e-16 ***
## poly(thinness18, 10)1  -57.97924  149.42502   -0.39  0.69814
```

```
## poly(thinness18, 10)2  -90.84820  190.17298   -0.48  0.63303
## poly(thinness18, 10)3   -2.23887  133.59367   -0.02  0.98663
## poly(thinness18, 10)4   15.63624   56.56750    0.28  0.78232
## poly(thinness18, 10)5  -84.34331   27.29679   -3.09  0.00210 **
## poly(thinness18, 10)6   22.42033   11.82375    1.90  0.05841 .
## poly(thinness18, 10)7  -15.84216   12.54326   -1.26  0.20708
## poly(thinness18, 10)8   -3.57118   13.77446   -0.26  0.79552
## poly(thinness18, 10)9    8.45143   10.39355    0.81  0.41646
## poly(thinness18, 10)10 -11.36516    6.24402   -1.82  0.06923 .
## poly(thinness59, 10)1   62.52784  152.33790    0.41  0.68162
## poly(thinness59, 10)2  122.89611  191.71516    0.64  0.52175
## poly(thinness59, 10)3  -11.65826  130.33272   -0.09  0.92875
## poly(thinness59, 10)4    8.95192   54.78994    0.16  0.87027
## poly(thinness59, 10)5   70.52536   24.19278    2.92  0.00369 **
## poly(thinness59, 10)6    3.60067   12.39349    0.29  0.77151
## poly(thinness59, 10)7    4.43728   18.97649    0.23  0.81520
## poly(thinness59, 10)8   -2.12172   16.10625   -0.13  0.89524
## poly(thinness59, 10)9   -7.47789    9.97073   -0.75  0.45356
## poly(thinness59, 10)10  -5.26452    5.78671   -0.91  0.36332
## poly(Income, 7)1       154.41778   10.88612   14.18  < 2e-16 ***
## poly(Income, 7)2        -0.63034    6.95110   -0.09  0.92778
## poly(Income, 7)3        -1.71630    4.54448   -0.38  0.70581
## poly(Income, 7)4         9.75248    3.53696    2.76  0.00601 **
## poly(Income, 7)5         6.25585    3.43212    1.82  0.06884 .
## poly(Income, 7)6         1.83258    3.05316    0.60  0.54859
## poly(Income, 7)7         1.28397    2.65471    0.48  0.62881
## poly(school, 8)1       -33.54142    9.13594   -3.67  0.00026 ***
## poly(school, 8)2         6.74669    5.34140    1.26  0.20705
## poly(school, 8)3        -3.44626    3.71905   -0.93  0.35448
## poly(school, 8)4        -6.39135    3.98078   -1.61  0.10890
## poly(school, 8)5         0.61030    3.00308    0.20  0.83903
## poly(school, 8)6         6.53819    2.95921    2.21  0.02752 *
## poly(school, 8)7         8.62879    2.91325    2.96  0.00318 **
## poly(school, 8)8        -1.14664    2.66381   -0.43  0.66702
## homicides               -0.00644    0.01306   -0.49  0.62225
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.28 on 598 degrees of freedom
## Multiple R-squared:  0.943,  Adjusted R-squared:  0.937
## F-statistic:  182 on 54 and 598 DF,  p-value: <2e-16
```

From the plots for the polynomial regression, we can see that for plots of residuals vs fitted and Scale Location, residual does not change with fitted value and the data points are random. Therefore, residuals are independent. For the normal qq plot, we can see that the tail is away from the dot line, and this means that the normal assumption might be violated.



By using spline method on polynomial terms, the regression generates MSE 5.90. Using spline on variables that are not polynomial one at a time, we can see that for variables texp and hiv there are improvements in MSE. After we add these two variables to the spline method, MSE lowered to 5.55 which is the lowest among all models built at this stage. By looking at the plot of the model, there is not much overfitting going on.

```
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + death5 + texp + hiv +
##     s(thinness18) + s(thinness59) + s(Income) + s(school) + homicides
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 70.25261    0.31137  225.62  < 2e-16 ***
## ind          0.04893    0.01126    4.35  1.6e-05 ***
## death5      -0.04079    0.00850   -4.80  2.0e-06 ***
## texp         0.03206    0.04550    0.70     0.48
## hiv         -0.41260    0.03725  -11.08  < 2e-16 ***
## homicides   -0.00464    0.01281   -0.36     0.72
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                edf Ref.df     F p-value
## s(adm)        8.42   8.89 20.01 < 2e-16 ***
## s(alcohol)    2.24   2.79  2.42 0.08603 .
## s(exp)        1.17   1.31  2.88 0.11175
## s(thinness18) 8.35   8.86  3.13 0.00076 ***
## s(thinness59) 9.00   9.00  8.66 2.4e-12 ***
## s(Income)     5.22   6.46 45.55 < 2e-16 ***
```

```
## s(school)     7.81   8.61  4.16 4.9e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.939   Deviance explained = 94.3%
## GCV =  5.494  Scale est. = 5.0884    n = 653
```

```
CVgam(formula(gam.life),data,nfold=10)
```

```
##    GAMscale CV-mse-GAM
##        5.09       5.90
```

```
#ind
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + s(ind) + s(alcohol) + s(exp) + death5 + texp +
##     hiv + s(thinness18) + s(thinness59) + s(Income) + s(school) +
##     homicides
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 72.87437    0.75519   96.50   <2e-16 ***
## death5      -0.03804    0.00905   -4.20    3e-05 ***
## texp         0.04147    0.04564    0.91     0.36
## hiv         -0.40992    0.03720  -11.02   <2e-16 ***
## homicides   -0.00397    0.01272   -0.31     0.75
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                edf Ref.df     F p-value
## s(adm)        8.50   8.91 20.22 < 2e-16 ***
## s(ind)        3.22   3.95  5.39 0.00027 ***
## s(alcohol)    2.48   3.08  2.57 0.05139 .
## s(exp)        1.12   1.24  2.53 0.13273
## s(thinness18) 7.14   7.88  3.07 0.00142 **
## s(thinness59) 7.03   7.80  8.32 9.9e-11 ***
## s(Income)     4.59   5.78 50.69 < 2e-16 ***
## s(school)     8.01   8.72  4.29 2.9e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.938   Deviance explained = 94.3%
## GCV = 5.4928  Scale est. = 5.0967    n = 653
```

```
CVgam(formula(gam.ind),data,nfold=10)
```

```
##    GAMscale CV-mse-GAM
##        5.10       5.93
```

```
#death5
## Family: gaussian
## Link function: identity
##
## Formula:
```

```
## life ~ s(adm) + ind + s(alcohol) + s(exp) + s(death5) + texp +
##     hiv + s(thinness18) + s(thinness59) + s(Income) + s(school) +
##     homicides
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 67.07686    0.84328   79.54  < 2e-16 ***
## ind          0.04763    0.01317    3.62  0.00032 ***
## texp         0.04018    0.04561    0.88  0.37865
## hiv         -0.40817    0.03735  -10.93  < 2e-16 ***
## homicides   -0.00413    0.01270   -0.32  0.74549
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf Ref.df     F p-value
## s(adm)         8.51   8.91 20.22 < 2e-16 ***
## s(alcohol)     2.41   2.99  2.49 0.06291 .
## s(exp)         1.17   1.32  2.51 0.14266
## s(death5)      3.05   3.76  6.33 7.5e-05 ***
## s(thinness18) 7.30   7.98  3.42 0.00058 ***
## s(thinness59) 6.74   7.59  8.32 1.8e-10 ***
## s(Income)      4.78   5.99 48.37 < 2e-16 ***
## s(school)      7.95   8.69  4.15 4.7e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.938   Deviance explained = 94.3%
## GCV = 5.4925  Scale est. = 5.0979    n = 653
```

```
CVgam(formula(gam.death5),data,nfold=10)
```

```
##    GAMscale CV-mse-GAM
##        5.10       5.93
```

```
#texp
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + death5 + s(texp) +
##     hiv + s(thinness18) + s(thinness59) + s(Income) + s(school) +
##     homicides
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 70.44191    0.16440  428.47  < 2e-16 ***
## ind          0.05647    0.01117    5.06  5.7e-07 ***
## death5      -0.04569    0.00842   -5.43  8.4e-08 ***
## hiv         -0.43069    0.03719  -11.58  < 2e-16 ***
## homicides   -0.00867    0.01253   -0.69     0.49
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf Ref.df     F p-value
```

```
## s(adm)         8.47   8.91 20.32 < 2e-16 ***
## s(alcohol)     2.33   2.89  1.96  0.1452
## s(exp)         1.00   1.00  2.76  0.0969 .
## s(texp)        4.27   5.33  6.18 8.0e-06 ***
## s(thinness18) 8.14   8.77  2.39  0.0076 **
## s(thinness59) 9.00   9.00  7.24 4.5e-10 ***
## s(Income)      6.01   7.26 41.71 < 2e-16 ***
## s(school)      8.17   8.79  5.40 4.7e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.942   Deviance explained = 94.6%
## GCV =  5.229  Scale est. = 4.8096    n = 653
```

```
CVgam(formula(gam.texp),data,nfold=10)
```

```
##    GAMscale CV-mse-GAM
##        4.81       5.73
```

```
#hiv
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + death5 + texp + s(hiv) +
##     s(thinness18) + s(thinness59) + s(Income) + s(school) + homicides
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 69.00008    0.30227  228.28  < 2e-16 ***
## ind          0.03918    0.01119    3.50  0.00050 ***
## death5      -0.03288    0.00851   -3.86  0.00012 ***
## texp         0.07843    0.04518    1.74  0.08306 .
## homicides   -0.00114    0.01255   -0.09  0.92781
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                  edf Ref.df     F p-value
## s(adm)         8.31   8.85 19.83 < 2e-16 ***
## s(alcohol)     2.29   2.86  3.13   0.033 *
## s(exp)         1.27   1.48  3.31   0.099 .
## s(hiv)         7.15   8.17 22.93 < 2e-16 ***
## s(thinness18) 8.47   8.90  3.98 4.3e-05 ***
## s(thinness59) 9.00   9.00  9.39 1.6e-13 ***
## s(Income)      5.36   6.55 45.57 < 2e-16 ***
## s(school)      1.00   1.00 20.21 8.3e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.942   Deviance explained = 94.6%
## GCV = 5.1746  Scale est. = 4.7954    n = 653
```

```
CVgam(formula(gam.hiv),data,nfold=10)
```

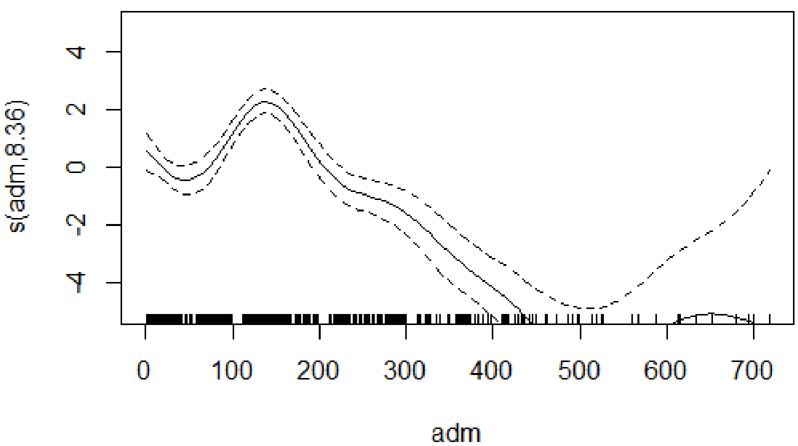
```
##    GAMscale CV-mse-GAM
##        4.80       5.67
```

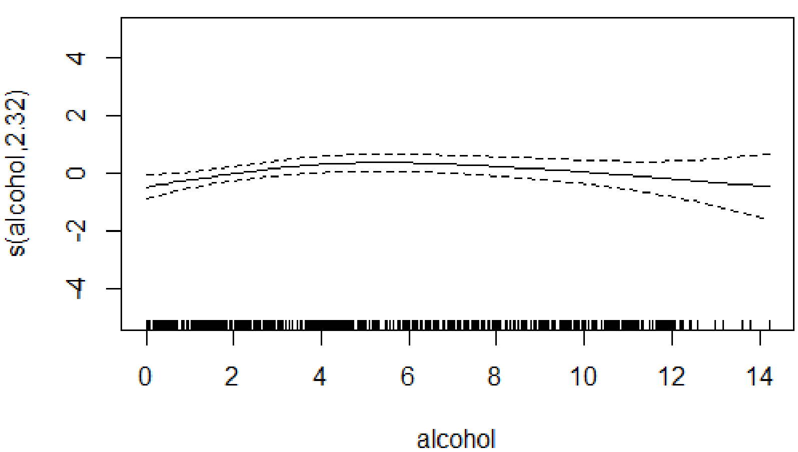
```
#homicides
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + death5 + texp + hiv +
##     s(thinness18) + s(thinness59) + s(Income) + s(school) + s(homicides)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 70.18120    0.31748  221.06  < 2e-16 ***
## ind          0.04300    0.01096    3.93  9.7e-05 ***
## death5      -0.03560    0.00825   -4.31  1.9e-05 ***
## texp         0.03847    0.04519    0.85     0.39
## hiv         -0.43518    0.03941  -11.04  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                edf Ref.df     F p-value
## s(adm)        8.52   8.92 18.27 < 2e-16 ***
## s(alcohol)    2.18   2.71  2.03  0.1286
## s(exp)        1.65   2.04  1.81  0.1783
## s(thinness18) 8.10   8.52  4.61 1.2e-05 ***
## s(thinness59) 6.40   7.46  9.39 8.8e-12 ***
## s(Income)     4.87   6.01 55.83 < 2e-16 ***
## s(school)     1.00   1.00 24.39 1.0e-06 ***
## s(homicides)  8.02   8.72  2.80  0.0038 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.939   Deviance explained = 94.3%
## GCV = 5.4161  Scale est. = 5.0367    n = 653

CVgam(formula(gam.homicides),data,nfold=10)

##    GAMscale CV-mse-GAM
##        5.04       5.88

#final
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + death5 + s(texp) +
##     s(hiv) + s(thinness18) + s(thinness59) + s(Income) + s(school) +
##     homicides
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 69.39766    0.14979  463.31  < 2e-16 ***
## ind          0.04480    0.01130    3.96  8.3e-05 ***
## death5      -0.03620    0.00856   -4.23  2.7e-05 ***
## homicides   -0.00344    0.01239   -0.28     0.78
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

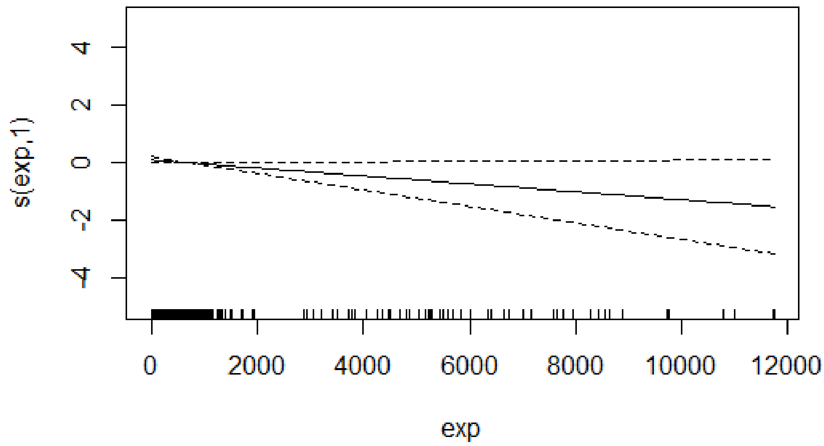
```
##
## Approximate significance of smooth terms:
##                  edf Ref.df     F p-value    
## s(adm)          8.36   8.86 19.31 < 2e-16 ***
## s(alcohol)      2.32   2.88  2.82 0.04619 *  
## s(exp)          1.00   1.00  3.46 0.06354 .  
## s(texp)         4.41   5.49  6.09 8.0e-06 ***
## s(hiv)          7.16   8.18 22.87 < 2e-16 ***
## s(thinness18) 8.35   8.86  3.34 0.00035 ***
## s(thinness59) 9.00   9.00  7.82 5.4e-11 ***
## s(Income)       5.65   6.90 39.21 < 2e-16 ***
## s(school)       7.47   8.40  4.82 8.4e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.946   Deviance explained =   95%
## GCV = 4.9468  Scale est. = 4.5095    n = 653
```

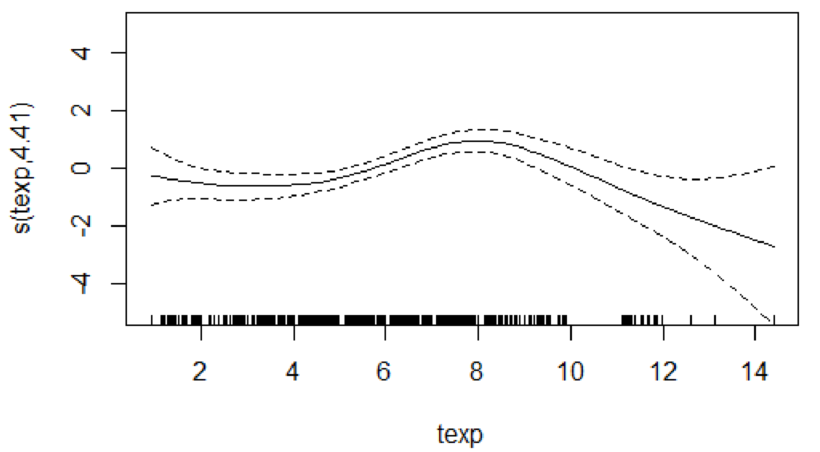
```r
CVgam(formula(plus2),data,nfold=10)
```

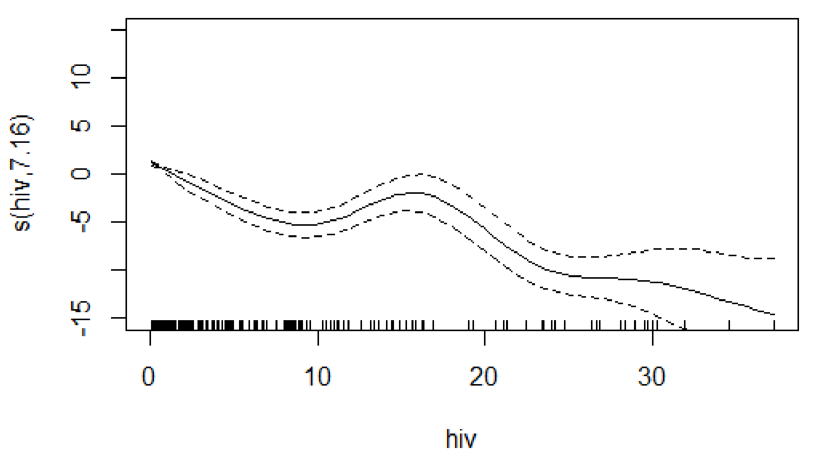
```
##    GAMscale CV-mse-GAM
##        4.51       5.55
```

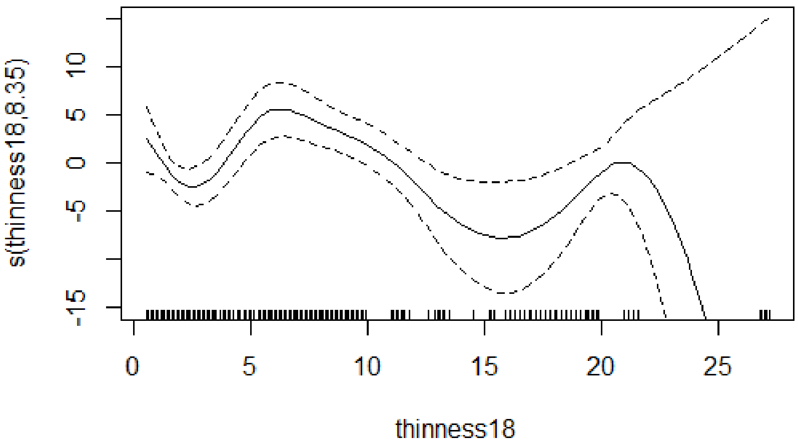


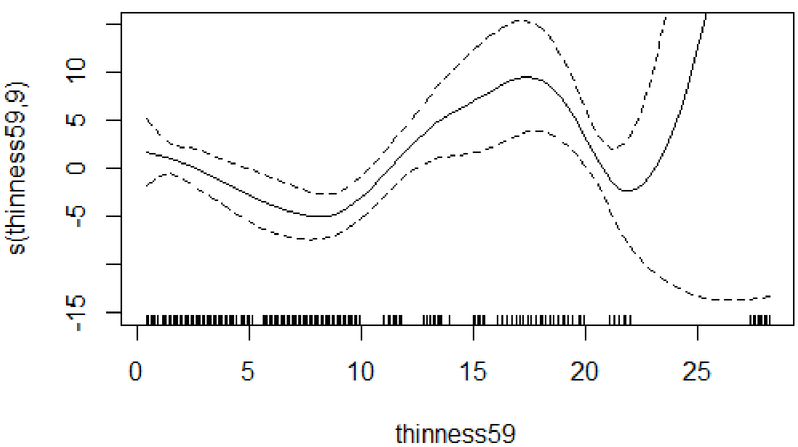


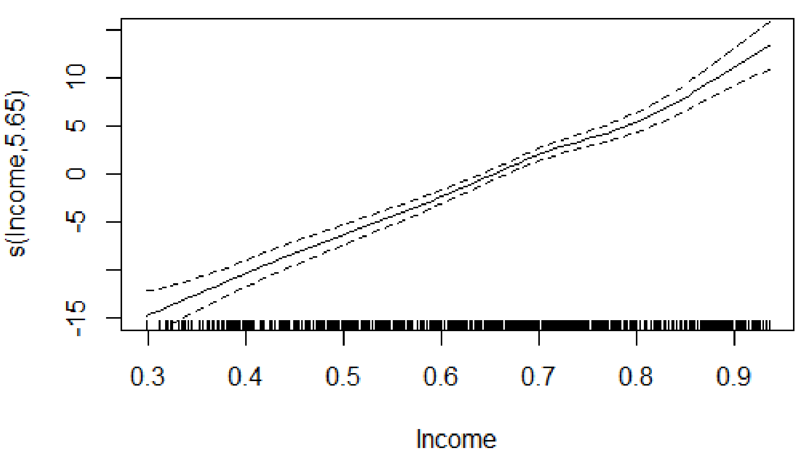

s(exp,1)
4
2
0
-2
-4
0
2000
4000
6000
8000
10000
12000
exp

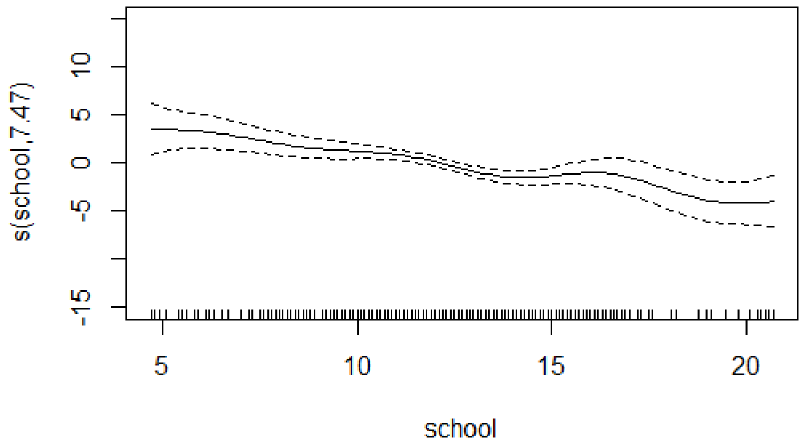

s(texp,4.41)
4
2
0
-2
-4
2
4
6
8
10
12
14
texp


s(hiv,7.16)
10
5
0
-5
-15
0
10
20
30
hiv


s(thinness18,8.35)
-15
-5
0
5
10
0
5
10
15
20
25
thinness18


s(thinness59,9)
-15
-5
0
5
10
0
5
10
15
20
25
thinness59


s(Income,5.65)
-15
-5
0
5
10
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Income



Recall the plot of bivariate correlation, the pairs with high correlations are: ind and death5, life and adm, life and Income, thinness18 and thinness59, school and Income. Among these pairs we chose the ones without dependent variable, life, for the interaction term. Now we check if adding these interaction terms will further improve the model. Looking at the output. the MSE of the model with interaction term (MSE=5.24) is smaller compared to the one without interaction term (MSE =5.55). Thus, model has been improved.

```
## Family: gaussian
## Link function: identity
##
## Formula:
## life ~ s(adm) + ind + s(alcohol) + s(exp) + s(texp) + s(hiv) +
##     homicides + s(ind, death5) + s(thinness18, thinness59) +
##     s(school, Income)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  59.0491     0.9865   59.86   <2e-16 ***
## ind           0.1724     0.0167   10.33   <2e-16 ***
## homicides    -0.0117     0.0124   -0.94     0.35
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                              edf Ref.df     F p-value
## s(adm)                      8.22   8.78 15.63 < 2e-16 ***
## s(alcohol)                  4.16   5.10  3.23  0.0066 **
## s(exp)                      8.70   8.97  2.79  0.0024 **
## s(texp)                     4.39   5.43  6.91 1.4e-06 ***
## s(hiv)                      5.66   6.75 25.43 < 2e-16 ***
## s(ind,death5)              21.10  24.79  7.01 < 2e-16 ***
## s(thinness18,thinness59)  22.67  25.83 11.01 < 2e-16 ***
## s(school,Income)          12.83  16.39 23.77 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Rank: 134/135
## R-sq.(adj) =  0.952   Deviance explained = 95.9%
## GCV = 4.5917  Scale est. = 3.9545    n = 653
```

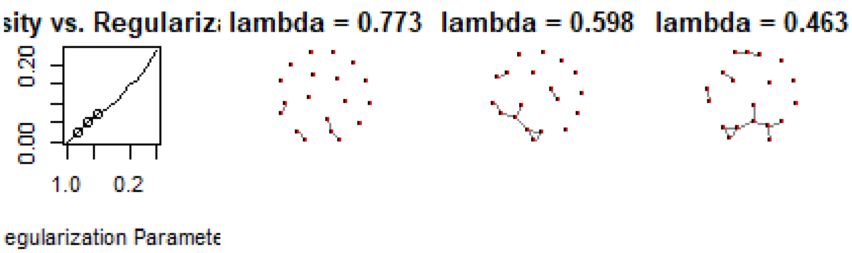
```
##    GAMscale CV-mse-GAM
##        3.95        5.24
```

Trying transformation for life, the result improve MSE value a lot by using square root (MSE=`0.0200`)and log of life(MSE=`0.0013`). In addition, log generates better result thus we proceed with log model.

```
##    GAMscale CV-mse-GAM
##      0.0148      0.0200
```

```
##    GAMscale CV-mse-GAM
##      0.0009      0.0013
```

By looking at the data we can see that sparsity increase with regularization parameter linearly.

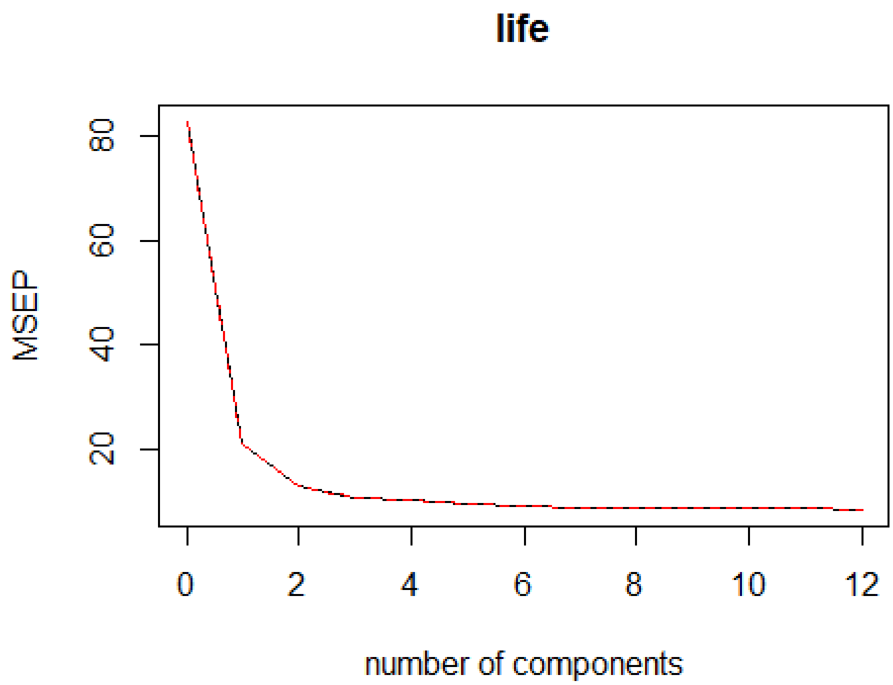


Part C Other Methods

Since there are some highly correlated pairs in the independent variables. Partial least square method is used. It is also useful when there are a large number of independent variables. The lowest cross-validation error occurs when only M=12 partial least squares dimensions are used. By using the partial least square method, MSE turns out to be 7.58 which is higher than all the other models except the linear model.

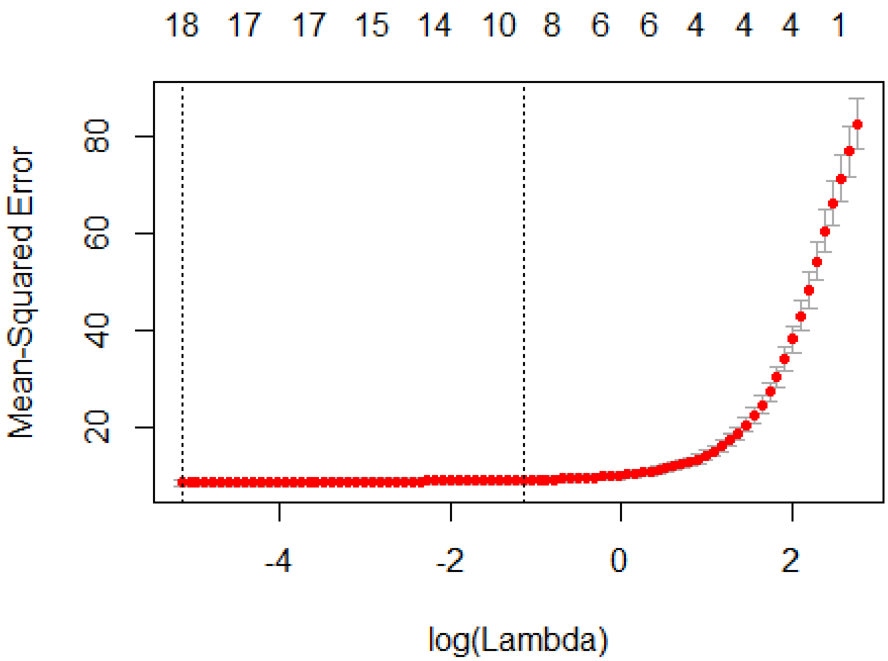
```
## Data:    X dimension: 653 12
##  Y dimension: 653 1
## Fit method: kernelpls
## Number of components considered: 12
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           9.106    4.577    3.598    3.255    3.184    3.095    2.977
## adjCV        9.106    4.573    3.595    3.251    3.181    3.090    2.975
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps
## CV       2.934    2.932    2.931     2.925     2.921     2.881
## adjCV    2.931    2.929    2.928     2.924     2.924     2.877
##
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X       34.89    56.18    62.91    72.88    78.66    81.73    85.37
## life    75.18    84.72    87.68    88.25    88.95    89.75    90.03
##       8 comps  9 comps  10 comps  11 comps  12 comps
```

```
## X        89.17     94.67      97.03      99.87     100.00
## life     90.04     90.05      90.07      90.10      90.44
```




```
## Data:    X dimension: 653 18
##  Y dimension: 653 1
## Fit method: svdpc
## Number of components considered: 12
## TRAINING: % variance explained
##    1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X    30.92    44.88    54.44    62.82    68.61    73.61    78.16    82.61
## y    47.96    77.50    77.99    80.70    80.72    82.12    84.29    84.30
##    9 comps  10 comps  11 comps  12 comps
## X    85.80     88.58     91.23     93.64
## y    84.32     84.66     84.87     85.31
```

By using elnet regression for variable selection, diph and gdpper are dropped, which is the same result that lasso generated.




```
##          adm          ind      alcohol          exp           hb      measles
##    -1.13e-02     3.19e-02    -1.74e-01     2.72e-04    -5.53e-03     9.54e-06
##          bmi       death5        polio         texp         diph          hiv
##     4.43e-03    -2.57e-02     7.74e-03     1.58e-01    -6.05e-04    -3.86e-01
##  thinness18  thinness59       Income       school    homicides       gdpper
##     1.33e-01    -7.74e-02     4.79e+01    -3.92e-01    -3.44e-02     1.18e-03
```

For conclusion, gam(log(life)~s(adm)+ ind+ s(alcohol)+ s(exp) + s(texp)+ s(hiv)+homicides+ s(ind,death5)+s(thinness18,thinness59)+s(school,Income), data=data) generates the best result. This means that the life expectancy is predicted by Adult Mortality Rates, Number of Infant Deaths, alcohol consumption, Expenditure on health, General government expenditure on health, Deaths per 1 000 live births HIV/AIDS, Intentional homicides, Number of Infant Deaths, Number of Infant Deaths, Number of Years of Schooling, and Income.

Appendix

````
```{r}
#data importing
life_data <- read.csv(file="life.csv")
dim(life_data)
data <- na.omit(life_data)
data <- data[apply(data,1,function(z) !any(z==0)),]
dim(data)
#names(data)
#summary(data)
```
```{r}
library(ggplot2)
library(reshape)
# Histogram overlaid with kernel density curve
data2 <- melt(data)
ggplot(data2,aes(x=value))+geom_density()+facet_wrap(~variable,scales="free")
ggplot(data2,aes(x=value))+geom_histogram()+facet_wrap(~variable,scales="free")
library(corrplot)
corrplot(cor(data))
```
```{r}
cor(data)
data3 <- cbind(data2,life=data$life)
ggplot(data3,aes(x=value,y=life))+geom_point()+facet_wrap(~variable,scales="free")
ggplot(data3,aes(x=value,y=life))+stat_bin2d()+facet_wrap(~variable,scales="free")
```
```{r}
# multi linear model foward selection
model1 <- lm(life~.,data=data)
````
````

~~~~
library(MASS)
end <- formula(model1)
start <- lm(life~1,data=data)
step_forward <- stepAIC(start,scope=end,direction="forward",trace = F)
#life ~ Income + hiv + adm + measles + smoke + X5death + thinness18 + ind + texp

# multi linear model backward
end1 <- formula(lm(life~1,data))
step_back <- stepAIC(model1,scope=end1,direction="backward",trace = F)
#life ~ adm + ind + measles + X5death + texp + hiv + thinness18 + Income + smoke


#compare multi linear anova
step_forward$anova

step_back$anova

```
```{r}

library(leaps)

#backward mse
regfit.bcd <- regsubsets(life~.,data,nvmax=18,method="backward")
summary(regfit.bcd)
regfit.summary2 <- summary(regfit.bcd)

library(boot)
CVmseback <- rep(0,18)
for(i in 1:18){
~~~~

~~~~
  tempCols <- which(regfit.summary2$which[i,-1]==TRUE)

  tempCols <- c(tempCols,19)

  tempCols <- as.numeric(tempCols)

  tempGLM <- glm(life~.,data=data[,tempCols])

  tempCV <- cv.glm(tempGLM,data=data[,tempCols],K = 10)

  CVmseback[i] <- tempCV$delta[1]

}

plot(CVmseback)

which.min(CVmseback)

min(CVmseback)

```

```{r}

library(glmnet)

lasso.cv <- cv.glmnet(x=as.matrix(data[,-19]),y=as.matrix(data[,19]),alpha=1,nfolds = 10)

a <- lasso.cv$lambda.min

b <- log(a)

b

plot(lasso.cv)


#look at the selection by lasso

lasso.fit <- glmnet(x=as.matrix(data[,-19]),y=as.matrix(data[,19]),alpha=1,lambda=c(1,exp(b)))

lasso.fit$beta[,2]


```

```{r}

#linear model

lm1 <- lm(life~adm+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+
homicides,data=data)

summary(lm1)

library(DAAG)
~~~~

````
cvlm<-cv.lm(data=data, lm1, m=10, dots =FALSE, seed=29, plotit=TRUE, printit=TRUE)
attr(cvlm, "ms")


```
```{r}
#plot
par(mfrow = c(2, 2))
plot(lm1)
```
```{r}
#bootstrap CI
library(boot)
life <- function(data, i) {
  d <- data[i,]
  fit <- lm(life~adm+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+ homicides,data=d)
  return(coef(fit))
}
bootResults <- boot(data=data,statistic=life,stype="i",R=6000)
boot.ci(bootResults, type="bca",index=1)
boot.ci(bootResults, type="bca",index=2)
boot.ci(bootResults, type="bca",index=3)
boot.ci(bootResults, type="bca",index=4)
boot.ci(bootResults, type="bca",index=5)
boot.ci(bootResults, type="bca",index=6)
boot.ci(bootResults, type="bca",index=7)
boot.ci(bootResults, type="bca",index=8)
boot.ci(bootResults, type="bca",index=9)
boot.ci(bootResults, type="bca",index=10)
boot.ci(bootResults, type="bca",index=11)
````

````
boot.ci(bootResults, type="bca",index=12)
boot.ci(bootResults, type="bca",index=13)

#lm CI (normal approximation)
confint(lm1, level=0.95)

```
```{r}
#adm
admMSE <- rep(0,10)
for(i in 1:10){
  templm <- glm(life~poly(adm,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)
  tempCV <- cv.glm(data,templm,K = 10)
  admMSE[i] <- tempCV$delta[1]
}
plot(admMSE)

#ind
indMSE <- rep(0,10)
for(i in 1:10){
  templm <- glm(life~poly(ind,i)+ adm+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)
  tempCV <- cv.glm(data,templm,K = 10)
  indMSE[i] <- tempCV$delta[1]
}
plot(indMSE)

#alcohol
alcoholMSE <- rep(0,10)
for(i in 1:10){
````

```
  templm <- glm(life~poly(alcohol,i)+ ind+ adm+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  alcoholMSE[i] <- tempCV$delta[1]

}

plot(alcoholMSE)


#exp

expMSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(exp,i)+ ind+ alcohol+ adm+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  expMSE[i] <- tempCV$delta[1]

}

plot(expMSE)


#death5

death5MSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(death5,i)+ ind+ alcohol+ exp+ adm+ texp+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  death5MSE[i] <- tempCV$delta[1]

}

plot(death5MSE)


#texp

texpMSE <- rep(0,10)

for(i in 1:10){
```

```
  templm <- glm(life~poly(texp,i)+ ind+ alcohol+ exp+ death5+ adm+ hiv+ thinness18+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  texpMSE[i] <- tempCV$delta[1]

}

plot(texpMSE)


#hiv

hivMSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(hiv,i)+ ind+ alcohol+ exp+ death5+ texp+ adm+ thinness18+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  hivMSE[i] <- tempCV$delta[1]

}

plot(hivMSE)


#thinness18

thinness18MSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(thinness18,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ adm+ thinness59+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  thinness18MSE[i] <- tempCV$delta[1]

}

plot(thinness18MSE)


#thinness59

thinness59MSE <- rep(0,10)

for(i in 1:10){
```

```
  templm <- glm(life~poly(thinness59,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ adm+ Income+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  thinness59MSE[i] <- tempCV$delta[1]

}

plot(thinness59MSE)




#Income

IncomeMSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(Income,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ adm+ school+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  IncomeMSE[i] <- tempCV$delta[1]

}

plot(IncomeMSE)


#school

schoolMSE <- rep(0,10)

for(i in 1:10){

  templm <- glm(life~poly(school,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ adm+homicides,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  schoolMSE[i] <- tempCV$delta[1]

}

plot(schoolMSE)


#homicides

homicidesMSE <- rep(0,10)

for(i in 1:10){
```

```
  templm <- glm(life~poly(homicides,i)+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+adm,data=data)

  tempCV <- cv.glm(data,templm,K = 10)

  homicidesMSE[i] <- tempCV$delta[1]

}

plot(homicidesMSE)


which.min(admMSE)

which.min(indMSE)

which.min(alcoholMSE)

which.min(expMSE)

which.min(death5MSE)

which.min(texpMSE)

which.min(hivMSE)

which.min(thinness18MSE)

which.min(thinness59MSE)

which.min(IncomeMSE)

which.min(schoolMSE)

which.min(homicidesMSE)


polymodel <- lm(life~poly(adm,10)+ ind+ poly(alcohol,2)+ poly(exp,2)+ death5+ texp+ hiv+ poly(thinness18,10)+ poly(thinness59,10)+ poly(Income,7)+ poly(school,8)+homicides,data=data)


summary(polymodel)


#check MSE

poly <- glm(life~poly(adm,10)+ ind+ poly(alcohol,2)+ poly(exp,2)+ death5+ texp+ hiv+ poly(thinness18,10)+ poly(thinness59,10)+ poly(Income,7)+ poly(school,8)+homicides,data=data)

cv.glm(data,poly,K=10)$delta[1]

```

````
```{r}

par(mfrow = c(2, 2))

plot(poly)

```

```{r}

library(mgcv)

library(gamclass)

gam.life <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ death5+ texp+ hiv+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(gam.life)

CVgam(formula(gam.life),data,nfold=10)


#ind

gam.ind <- gam(life~s(adm)+ s(ind)+ s(alcohol)+ s(exp)+ death5+ texp+ hiv+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(gam.ind)

CVgam(formula(gam.ind),data,nfold=10)


#death5

gam.death5 <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ s(death5)+ texp+ hiv+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(gam.death5)

CVgam(formula(gam.death5),data,nfold=10)


#texp

gam.texp <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ death5+ s(texp)+ hiv+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(gam.texp)

CVgam(formula(gam.texp),data,nfold=10)


#hiv
````
````

gam.hiv <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ death5+ texp+ s(hiv)+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(gam.hiv)

CVgam(formula(gam.hiv),data,nfold=10)

#homicides

gam.homicides <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ death5+ texp+ hiv+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+s(homicides),data=data)

summary(gam.homicides)

CVgam(formula(gam.homicides),data,nfold=10)

#final

plus2 <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp)+ death5+ s(texp)+ s(hiv)+ s(thinness18)+ s(thinness59)+ s(Income)+ s(school)+homicides,data=data)

summary(plus2)

CVgam(formula(plus2),data,nfold=10)

plot(plus2,ylim=c(-15,15))

```

```{r}

interaction <- gam(life~s(adm)+ ind+ s(alcohol)+ s(exp) + s(texp)+ s(hiv)+homicides+ s(ind,death5)+s(thinness18,thinness59)+s(school,Income), data=data)

summary(interaction)

CVgam(formula(interaction),data,nfold = 10)

```

```{r}

interaction1 <- gam(life^0.5~s(adm)+ ind+ s(alcohol)+ s(exp) + s(texp)+ s(hiv)+homicides+ s(ind,death5)+s(thinness18,thinness59)+s(school,Income), data=data)

````
CVgam(formula(interaction1),data,nfold = 10)

interaction2 <- gam(log(life)~s(adm)+ ind+ s(alcohol)+ s(exp) + s(texp)+ s(hiv)+homicides+
s(ind,death5)+s(thinness18,thinness59)+s(school,Income), data=data)

CVgam(formula(interaction2),data,nfold = 10)


```

```{r}

library(huge)

a<-data.matrix(data, rownames.force = NA)


b<-huge(a, lambda = NULL, nlambda = NULL, lambda.min.ratio = NULL, method = "mb",

scr = NULL, scr.num = NULL, cov.output = FALSE, sym = "or", verbose = TRUE)


plot(b, align = FALSE)

```

```{r}

library(pls)

library(dplyr)

set.seed (1000)

set.seed(1)

pls_fit = plsr(life~adm+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+
school+ homicides, data = data, scale = TRUE, validation = "CV")

summary(pls_fit)

validationplot(pls_fit, val.type = "MSEP")


train = data %>%

  sample_frac(0.5)
````

```
test = data %>%
  setdiff(train)

x_train = model.matrix(life~adm+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+ homicides, train)[,-1]
x_test = model.matrix(life~adm+ ind+ alcohol+ exp+ death5+ texp+ hiv+ thinness18+ thinness59+ Income+ school+ homicides, test)[,-1]

y_train = train %>%
  select(life) %>%
  unlist() %>%
  as.numeric()

y_test = test %>%
  select(life) %>%
  unlist() %>%
  as.numeric()

pls_pred = predict(pls_fit, x_test, ncomp = 12)
mean((pls_pred - y_test)^2)

x = model.matrix(life~., data)[,-1]

y = data %>%
  select(life) %>%
  unlist() %>%
  as.numeric()

pcr_fit2 = pcr(y~x, scale = TRUE, ncomp = 12)
summary(pcr_fit2)
```

````
```

```{r}
library(glmnet)
elnet.cv <- cv.glmnet(x=as.matrix(data[,-19]),y=as.matrix(data[,19]),alpha=0.5,nfolds = 10)
a <- elnet.cv$lambda.min
b <- log(a)
b
plot(elnet.cv)

#look at the selection by elnet
elnet.fit <- glmnet(x=as.matrix(data[,-19]),y=as.matrix(data[,19]),alpha=1,lambda=c(1,exp(b)))
elnet.fit$beta[,2]

```
````
````