R Notebook

Climate Change

There have been many studies documenting that the average global temperature has been increasing over the last century. The consequences of a continued rise in global temperature will be dire. Rising sea levels and an increased frequency of extreme weather events will affect billions of people.

In this problem, we will attempt to study the relationship between average global temperature and several other factors.

The file **climate_change.csv** contains climate data from May 1983 to December 2008. The available variables include:

- **Year**: the observation year.
- **Month**: the observation month.
- **Temp**: the difference in degrees Celsius between the average global temperature in that period and a reference value. This data comes from the Climatic Research Unit at the University of East Anglia.
- **CO2, N2O, CH4, CFC.11, CFC.12**: atmospheric concentrations of carbon dioxide (CO_2), nitrous oxide (N_2O), methane (CH_4), trichlorofluoromethane (CCl_3F; commonly referred to as CFC-11) and dichlorodifluoromethane (CCl_2F_2; commonly referred to as CFC-12), respectively. This data comes from the ESRL/NOAA Global Monitoring Division.
  - CO2, N2O and CH4 are expressed in ppmv (parts per million by volume – i.e., 397 ppmv of CO2 means that CO2 constitutes 397 millionths of the total volume of the atmosphere)
  - CFC.11 and CFC.12 are expressed in ppbv (parts per billion by volume).
- **Aerosols**: the mean stratospheric aerosol optical depth at 550 nm. This variable is linked to volcanoes, as volcanic eruptions result in new particles being added to the atmosphere, which affect how much of the sun's energy is reflected back into space. This data is from the Godard Institute for Space Studies at NASA.
- **TSI**: the total solar irradiance (TSI) in W/m2 (the rate at which the sun's energy is deposited per unit area). Due to sunspots and other solar phenomena, the amount of energy that is given off by the sun varies substantially with time. This data is from the SOLARIS-HEPPA project website.
- **MEI**: multivariate El Nino Southern Oscillation index (MEI), a measure of the strength of the El Nino/La Nina-Southern Oscillation (a weather effect in the Pacific Ocean that affects global temperatures). This data comes from the ESRL/NOAA Physical Sciences Division.

1.1. Creating Our First Model

Read in the data file

```r
climate_change = read.csv('climate_change.csv')
head(climate_change)
```

```
##   Year Month   MEI    CO2     CH4     N2O  CFC.11  CFC.12      TSI
## 1 1983     5 2.556 345.96 1638.59 303.677 191.324 350.113 1366.102
## 2 1983     6 2.167 345.52 1633.71 303.746 192.057 351.848 1366.121
## 3 1983     7 1.741 344.15 1633.22 303.795 192.818 353.725 1366.285
## 4 1983     8 1.130 342.25 1631.35 303.839 193.602 355.633 1366.420
## 5 1983     9 0.428 340.17 1648.40 303.901 194.392 357.465 1366.234
## 6 1983    10 0.002 340.30 1663.79 303.970 195.171 359.174 1366.059
##   Aerosols  Temp
## 1   0.0863 0.109
## 2   0.0794 0.118
## 3   0.0731 0.137
```

```
## 4   0.0673 0.176
## 5   0.0619 0.149
## 6   0.0569 0.093
```

Split the data into a training set (for obs up to and including 2006) and a test set (remaining years)

```r
train = subset(climate_change, Year <= 2006)
test = subset(climate_change, Year > 2006)
```

Build a linreg model to predict the dependent var Temp, using MEI, CO2, CH4, N20, CFC.11, CFC.12, TSI, and Aerosols as independent variables (Year and Month should NOT be iused in the model. Use the training set to build the model

```r
mod = lm(Temp ~ MEI + CO2 + CH4 + N2O + CFC.11 + CFC.12 + TSI + Aerosols, data = train)
summary(mod)
```

```
##
## Call:
## lm(formula = Temp ~ MEI + CO2 + CH4 + N2O + CFC.11 + CFC.12 +
##     TSI + Aerosols, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25888 -0.05913 -0.00082  0.05649  0.32433
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.246e+02  1.989e+01  -6.265 1.43e-09 ***
## MEI          6.421e-02  6.470e-03   9.923  < 2e-16 ***
## CO2          6.457e-03  2.285e-03   2.826  0.00505 **
## CH4          1.240e-04  5.158e-04   0.240  0.81015
## N2O         -1.653e-02  8.565e-03  -1.930  0.05467 .
## CFC.11      -6.631e-03  1.626e-03  -4.078 5.96e-05 ***
## CFC.12       3.808e-03  1.014e-03   3.757  0.00021 ***
## TSI          9.314e-02  1.475e-02   6.313 1.10e-09 ***
## Aerosols    -1.538e+00  2.133e-01  -7.210 5.41e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09171 on 275 degrees of freedom
## Multiple R-squared:  0.7509, Adjusted R-squared:  0.7436
## F-statistic: 103.6 on 8 and 275 DF,  p-value: < 2.2e-16
```

2.1 Understanding the Model

2.2 Understanding the Model

Compute the correlations between all the variables in the training set. Which of the following independent variables is N2O highly correlated with (absolute correlation greater than 0.7)? And which of the following independent variables is CFC.11 highly correlated with?

N2O : CO2, CH4, and CFC.12 CFC.11: CH4, CFC.12

```r
cor(train)
```

```
##                 Year         Month           MEI         CO2         CH4
## Year      1.00000000 -0.0279419602 -0.0369876842  0.98274939  0.91565945
## Month    -0.02794196  1.0000000000  0.0008846905 -0.10673246  0.01856866
## MEI      -0.03698768  0.0008846905  1.0000000000 -0.04114717 -0.03341930
## CO2       0.98274939 -0.1067324607 -0.0411471651  1.00000000  0.87727963
## CH4       0.91565945  0.0185686624 -0.0334193014  0.87727963  1.00000000
## N2O       0.99384523  0.0136315303 -0.0508197755  0.97671982  0.89983864
## CFC.11    0.56910643 -0.0131112236  0.0690004387  0.51405975  0.77990402
## CFC.12    0.89701166  0.0006751102  0.0082855443  0.85268963  0.96361625
## TSI       0.17030201 -0.0346061935 -0.1544919227  0.17742893  0.24552844
## Aerosols -0.34524670  0.0148895406  0.3402377871 -0.35615480 -0.26780919
## Temp      0.78679714 -0.0998567411  0.1724707512  0.78852921  0.70325502
##                  N2O      CFC.11        CFC.12         TSI    Aerosols
## Year      0.99384523  0.56910643  0.8970116635  0.17030201 -0.34524670
## Month     0.01363153 -0.01311122  0.0006751102 -0.03460619  0.01488954
## MEI      -0.05081978  0.06900044  0.0082855443 -0.15449192  0.34023779
## CO2       0.97671982  0.51405975  0.8526896272  0.17742893 -0.35615480
## CH4       0.89983864  0.77990402  0.9636162478  0.24552844 -0.26780919
## N2O       1.00000000  0.52247732  0.8679307757  0.19975668 -0.33705457
## CFC.11    0.52247732  1.00000000  0.8689851828  0.27204596 -0.04392120
## CFC.12    0.86793078  0.86898518  1.0000000000  0.25530281 -0.22513124
## TSI       0.19975668  0.27204596  0.2553028138  1.00000000  0.05211651
## Aerosols -0.33705457 -0.04392120 -0.2251312440  0.05211651  1.00000000
## Temp      0.77863893  0.40771029  0.6875575483  0.24338269 -0.38491375
##                 Temp
## Year      0.78679714
## Month    -0.09985674
## MEI       0.17247075
## CO2       0.78852921
## CH4       0.70325502
## N2O       0.77863893
## CFC.11    0.40771029
## CFC.12    0.68755755
## TSI       0.24338269
## Aerosols -0.38491375
## Temp      1.00000000
```

3. Simplifying the Model

Given that the correlations are so high, let us focus on the N2O variable and build a model with only MEI, TSI, Aerosols and N2O as independent variables. Remember to use the training set to build the model.

```
mod_reduced = lm(Temp ~ MEI + TSI + Aerosols + N2O, data = train)
summary(mod_reduced)
```

```
##
## Call:
## lm(formula = Temp ~ MEI + TSI + Aerosols + N2O, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.27916 -0.05975 -0.00595  0.05672  0.34195
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.162e+02  2.022e+01  -5.747 2.37e-08 ***
## MEI          6.419e-02  6.652e-03   9.649  < 2e-16 ***
## TSI          7.949e-02  1.487e-02   5.344 1.89e-07 ***
## Aerosols    -1.702e+00  2.180e-01  -7.806 1.19e-13 ***
## N2O          2.532e-02  1.311e-03  19.307  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09547 on 279 degrees of freedom
## Multiple R-squared:  0.7261, Adjusted R-squared:  0.7222
## F-statistic: 184.9 on 4 and 279 DF,  p-value: < 2.2e-16
```

4. Automatically Building the Model

We have many variables in this problem, and as we have seen above, dropping some from the model does not decrease model quality. R provides a function, step, that will automate the procedure of trying different combinations of variables to find a good compromise of model simplicity and R2. This trade-off is formalized by the Akaike information criterion (AIC) - it can be informally thought of as **the quality of the model with a penalty for the number of variables in the model**.

The step function has one argument - the name of the initial model. It returns a simplified model. Use the step function in R to derive a new model, with the full model as the initial model (HINT: If your initial full model was called "climateLM", you could create a new model with the step function by typing step(climateLM). Be sure to save your new model to a variable name so that you can look at the summary. For more information about the step function, type ?step in your R console.)

It is interesting to note that **the step function does not address the collinearity of the variables**, except that adding highly correlated variables will not improve the R2 significantly. The consequence of this is that the step function will not necessarily produce a very interpretable model - just a model that has balanced quality and simplicity for a particular weighting of quality and simplicity (AIC).

```r
step(mod)
```

```
## Start:  AIC=-1348.16
## Temp ~ MEI + CO2 + CH4 + N2O + CFC.11 + CFC.12 + TSI + Aerosols
##
##            Df Sum of Sq    RSS     AIC
## - CH4       1   0.00049 2.3135 -1350.1
## <none>                  2.3130 -1348.2
## - N2O       1   0.03132 2.3443 -1346.3
## - CO2       1   0.06719 2.3802 -1342.0
## - CFC.12    1   0.11874 2.4318 -1335.9
## - CFC.11    1   0.13986 2.4529 -1333.5
## - TSI       1   0.33516 2.6482 -1311.7
## - Aerosols  1   0.43727 2.7503 -1301.0
## - MEI       1   0.82823 3.1412 -1263.2
##
## Step:  AIC=-1350.1
## Temp ~ MEI + CO2 + N2O + CFC.11 + CFC.12 + TSI + Aerosols
##
##            Df Sum of Sq    RSS     AIC
## <none>                  2.3135 -1350.1
## - N2O       1   0.03133 2.3448 -1348.3
```

```
## - CO2       1   0.06672 2.3802 -1344.0
## - CFC.12    1   0.13023 2.4437 -1336.5
## - CFC.11    1   0.13938 2.4529 -1335.5
## - TSI       1   0.33500 2.6485 -1313.7
## - Aerosols  1   0.43987 2.7534 -1302.7
## - MEI       1   0.83118 3.1447 -1264.9
```

```
##
## Call:
## lm(formula = Temp ~ MEI + CO2 + N2O + CFC.11 + CFC.12 + TSI +
##     Aerosols, data = train)
##
## Coefficients:
## (Intercept)          MEI          CO2          N2O       CFC.11
##  -1.245e+02    6.407e-02    6.401e-03   -1.602e-02   -6.609e-03
##      CFC.12          TSI     Aerosols
##   3.868e-03    9.312e-02   -1.540e+00
```

```r
mod_reduced_best = lm(Temp ~ MEI + CO2 + N2O + CFC.11 + CFC.12 + TSI + Aerosols, data = train)

summary(mod_reduced_best)
```

```
##
## Call:
## lm(formula = Temp ~ MEI + CO2 + N2O + CFC.11 + CFC.12 + TSI +
##     Aerosols, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25770 -0.05994 -0.00104  0.05588  0.32203
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.245e+02  1.985e+01  -6.273 1.37e-09 ***
## MEI          6.407e-02  6.434e-03   9.958  < 2e-16 ***
## CO2          6.402e-03  2.269e-03   2.821 0.005129 **
## N2O         -1.602e-02  8.287e-03  -1.933 0.054234 .
## CFC.11      -6.609e-03  1.621e-03  -4.078 5.95e-05 ***
## CFC.12       3.868e-03  9.812e-04   3.942 0.000103 ***
## TSI          9.312e-02  1.473e-02   6.322 1.04e-09 ***
## Aerosols    -1.540e+00  2.126e-01  -7.244 4.36e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09155 on 276 degrees of freedom
## Multiple R-squared:  0.7508, Adjusted R-squared:  0.7445
## F-statistic: 118.8 on 7 and 276 DF,  p-value: < 2.2e-16
```

5. Testing on Unseen Data

We have developed an understanding of how well we can fit a linear regression to the training data, but does the model quality hold when applied to unseen data? Using the model produced from the step function, calculate temperature predictions for the testing data set, using the predict function.

```r
temp_pred = predict(mod_reduced_best, newdata = test)
```

```r
SSE = sum((test$Temp - temp_pred) ^ 2)
SST = sum((test$Temp - mean(train$Temp)) ^ 2)

R_squared = 1 - (SSE/SST)

R_squared
```

```
## [1] 0.6286051
```