EM algorithm demonstration with air pollution data

Rebekah Chin

2 July 2021

```
# Getting L_int and Psi_int -----------------------------------------------
#import data
airpoldata <- read.delim("G:/My Drive/EM Algorithm and Demo/airpoldata.txt",
                         na.strings="N.A.")
weatherdata <- read.delim("G:/My Drive/EM Algorithm and Demo/weatherdata.txt",
                          na.strings="N.A.")

#write to matrix
CO<-airpoldata$CO
NO2<-airpoldata$NO2
NOX<-airpoldata$NOX
O3<-airpoldata$O3
SO2<-airpoldata$SO2
temp<-weatherdata$Mean..deg..C.
pres<-weatherdata$Mean.Pressure..hPa.
humid<-weatherdata$Mean.Relative.Humidity....
sun<-weatherdata$Total.Bright.Sunshine..hours.
wind<-weatherdata$Mean.Wind.Speed..km.h.
```

The data used in this demonstration is from March 1, 2019 to April 30, 2019 from Mong Kok district in Hong Kong. The variables selected for analysis are carbon monoxide (CO), nitrogen dioxide (NO^2), other nitrogen oxides (NO^x), ozone gas (O^3), sulphur dioxide (SO^2), mean temperature, mean pressure, mean humidity, total bright sunshine in hours, and wind speed.

```
X<-cbind(CO,NO2,NOX,O3,SO2,temp,pres,humid,sun,wind)
X[1:20,]
```

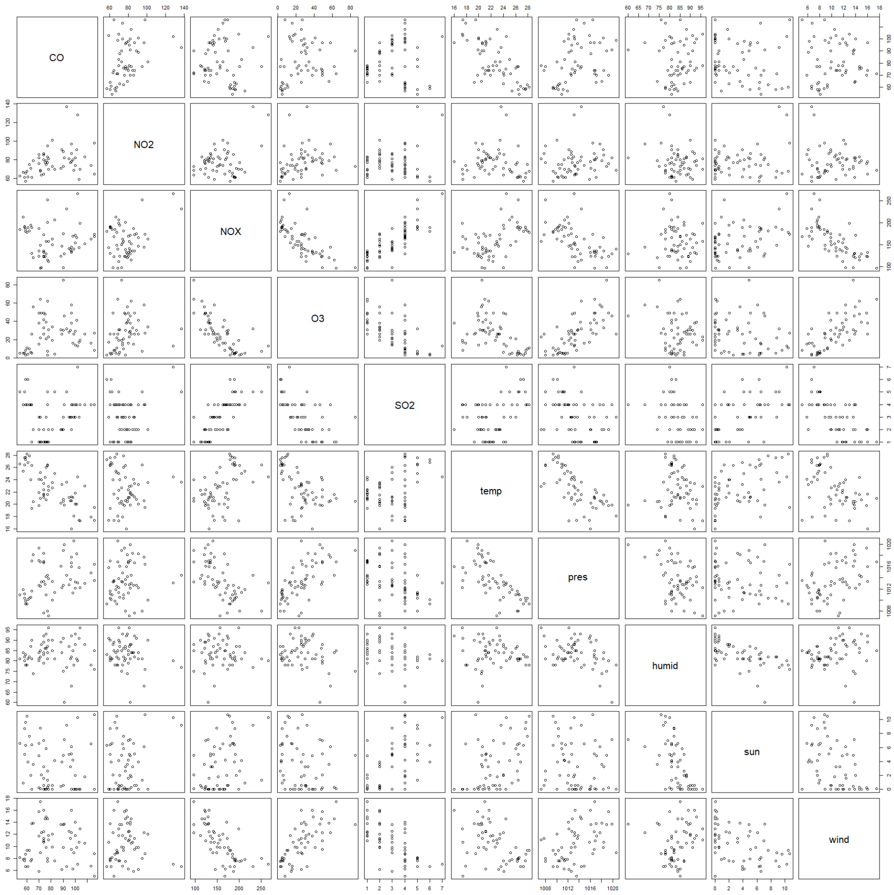
```
##        CO NO2 NOX O3 SO2 temp   pres humid  sun wind
##  [1,]  88  71 124 31   2 20.2 1016.1    89  0.4 13.6
##  [2,]  95  81 141 28   3 21.1 1012.7    87  5.0 12.5
##  [3,]  98  80 169 16   4 21.1 1011.3    85  4.2  6.2
##  [4,]  96  77 139 30   3 20.6 1013.7    81  3.2 11.3
##  [5,] 100  82 172 29   4 21.8 1012.1    88  2.0 14.0
##  [6,] 100  74 154 14   3 20.0 1013.2    93  0.0 11.8
##  [7,] 103  59 143 17   3 17.4 1015.8    91  0.0 10.7
##  [8,]  97  78 132 38   2 16.0 1016.0    92  0.0 16.0
```

```
##  [9,] 104  70 199 19    4 17.3 1012.2    96  0.0 12.9
## [10,] 116  65 168  8    4 17.4 1013.5    85  0.0  5.0
## [11,] 113  74 163 16    4 17.9 1014.9    78  6.5  6.7
## [12,] 116  98 174 27    4 19.5 1016.4    76 10.7  8.8
## [13,]  97  97 144 58    4 20.6 1017.7    68  6.1 12.3
## [14,] 101  91 166 41    4 20.0 1018.3    81  0.0  9.8
## [15,]  99  69 140 26    3 18.1 1020.6    78  0.0  8.9
## [16,]  91  82 129 46    4 19.9 1019.9    60  7.1 13.7
## [17,]  90  73  97 85    3 20.5 1018.9    75  4.9 13.6
## [18,]  97  87 135 49    3 21.2 1016.7    82  5.1  9.5
## [19,] 107  90 188 NA    5 22.9 1014.8    84 10.0  7.8
## [20,] 108  83 152 26    4 23.0 1013.0    88  1.8 10.5
```

```
#drop rows with missing values
X<-na.omit(X)
```

We started off with 61 data points but now have 58 due to missing entries in some data points. However, this would not be an issue as factor analysis is well suited for data sets with incomplete entries.

```
#summary statistics
Z<-scale(X)
mean<-colMeans(X)
Sn<-cov(X)
R<-cov2cor(Sn)
```



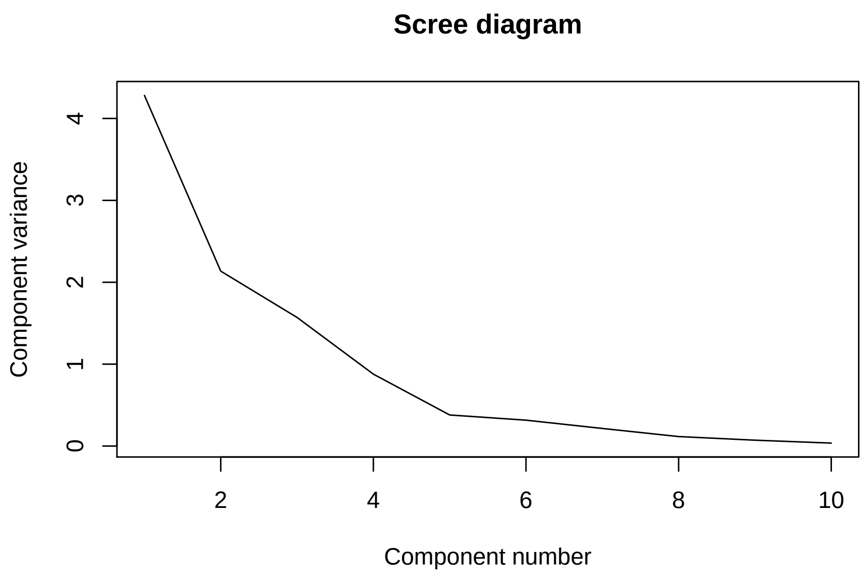
From the graphs, we see that CO and NO^2, NO^x and O^3, NO^x and SO^2, NO^x and wind speed, temperature and pressure, and temperature and number of bright sunlight hours show moderate to high correlation.

```
# principle component analysis to find min dimensions for L ---------------
pc<-princomp(x=X,cor=TRUE)
print(summary(pc,loadings=TRUE))
```

```
## Importance of components:
##                           Comp.1    Comp.2    Comp.3     Comp.4     Comp.5
## Standard deviation     2.069193 1.4614392 1.2531850 0.93704926 0.61602257
## Proportion of Variance 0.428156 0.2135805 0.1570473 0.08780613 0.03794838
## Cumulative Proportion  0.428156 0.6417365 0.7987837 0.88658986 0.92453824
##                            Comp.6     Comp.7     Comp.8      Comp.9     Comp.10
## Standard deviation     0.56215395 0.46368987 0.34030935 0.267521544 0.190299858
## Proportion of Variance 0.03160171 0.02150083 0.01158105 0.007156778 0.003621404
## Cumulative Proportion  0.95613994 0.97764077 0.98922182 0.996378596 1.000000000
##
## Loadings:
##        Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
## CO      0.143  0.378  0.554  0.173  0.287  0.347  0.268  0.151  0.324  0.311
```

```
## NO2           0.440  0.237 -0.736        -0.214                0.127 -0.373
## NOX   -0.419         0.263 -0.142 -0.407 -0.100 -0.269        -0.255  0.639
## O3     0.397  0.204 -0.223 -0.296         0.131  0.411 -0.449 -0.424  0.309
## SO2   -0.389  0.246  0.104  0.247 -0.433  0.387  0.144 -0.456        -0.388
## temp  -0.367 -0.193 -0.360 -0.311                0.302 -0.144  0.637  0.284
## pres   0.344  0.359 -0.142  0.277 -0.188 -0.381 -0.367 -0.371  0.421  0.160
## humid        -0.496  0.462 -0.192  0.238        -0.309 -0.577
## sun   -0.279  0.353 -0.355         0.542  0.347 -0.478        -0.133
## wind   0.398 -0.123 -0.133 -0.221 -0.421  0.615 -0.334  0.250  0.172
```

```
#plot scree diagram
plot(pc$sdev^2, xlab = "Component number",
     ylab = "Component variance", type = "l", main = "Scree diagram")
```




```
n<-3
```

We chose to use the first three components as they account for 79.8 percent of total variability.

```
# calculating L_int using principle component method ----------------------
L<-pc$loadings
L<-L[,1:n]
eigen_R<-eigen(R)
ev_R<-eigen_R$values
for (i in (1:n)){
  L[,i]<-sqrt(ev_R[i])*L[,i]
}
sizeL<-dim(L)

#calculating psi_int using L_int and R
psi<-rep(0,sizeL[1])
```

```r
for (i in (1:sizeL[1])){
  for (j in (1:sizeL[2])){
    psi[i]<-R[i,i]-sum(L[i,j]^2)
  }
}
psi<-diag(psi)
L
```

```
##            Comp.1     Comp.2     Comp.3
## CO     0.29626532  0.5520216  0.6946972
## NO2   -0.03820639  0.6428223  0.2968224
## NOX   -0.86684341  0.1443823  0.3293488
## O3     0.82121206  0.2987268 -0.2791986
## SO2   -0.80461397  0.3600333  0.1306577
## temp  -0.75985370 -0.2816352 -0.4508723
## pres   0.71204197  0.5253549 -0.1776661
## humid  0.15446994 -0.7242446  0.5790635
## sun   -0.57815021  0.5159528 -0.4451638
## wind   0.82256721 -0.1791494 -0.1671362
```

```r
diag(psi)
```

```
##  [1] 0.5173958 0.9118965 0.8915293 0.9220481 0.9829286 0.7967142 0.9684348
##  [8] 0.6646854 0.8018292 0.9720655
```

```r
# Using EM algorithm --------------------------------------------------------
tol=10^-5
maxite<-1000
source("emalg.R")
```

```
## Warning: package 'ropenblas' was built under R version 4.0.5
```

```r
emres<-emalg(tol,maxite,X,L,psi)
```

Here, we used the L and Ψ obtained from the principle component method for factor analysis as the initial values for the EM algorithm. One can also use the L and Ψ obtained from using the maximum likelihood method.

```r
emres$L_new
```

```
##            Comp.1     Comp.2      Comp.3
##  [1,]  0.28978414  0.4630415  0.59968812
##  [2,] -0.08232662  0.3130266  0.09809556
##  [3,] -0.85118189  0.1165780  0.29980062
##  [4,]  0.75645456  0.1624821 -0.34024306
##  [5,] -0.79184265  0.4106652  0.15833353
##  [6,] -0.77753478 -0.3480253 -0.50081571
##  [7,]  0.70587063  0.5526763 -0.12984210
##  [8,]  0.15010601 -0.7689523  0.54953638
##  [9,] -0.53701000  0.4110412 -0.38579909
## [10,]  0.75330340 -0.1929492 -0.15690943
```

```r
diag(emres$psi)
```

```
##  [1] 0.324106167 0.868360171 0.154486756 0.268078452 0.161828279 0.004991027
##  [7] 0.161602432 0.066961849 0.376308907 0.353190433
```

With a tolerance of 10^{-5}, L_{new} and Ψ_{new} converge after 400 iterations. We also notice that Ψ_{new} has values closer to zero, and show great improvement compared to the initial starting value of Ψ. Thus, L_{new} would account for more than 79.8 percent of total variability in the data.