Import dataset from the following link:
https://archive.ics.uci.edu/ml/machine-learning-databases/00360/

Perform the below written operations:

a. Read the file in Zip format and get it into R

```
#Read the file in Zip format and get it into R.
#Answer1
forecasturl = paste('https://archive.ics.uci.edu/ml/machine-learning-databases/00360/',
                    'AirQualityUCI.zip', sep='')
# create a temporary directory
td = tempdir()
# create the placeholder file
tf = tempfile(tmpdir=td, fileext=".zip")
# download into the placeholder file
download.file(forecasturl, tf)
# get the name of the first file in the zip archive
fname = unzip(tf, list=TRUE)$Name[1]
fname
# unzip the file to the temporary directory
unzip(tf, files=fname, exdir=td, overwrite=TRUE)
# fpath is the full path to the extracted file
fpath = file.path(td, fname)
fpath
d = read.csv(fpath,sep = ";")
View(d)
```

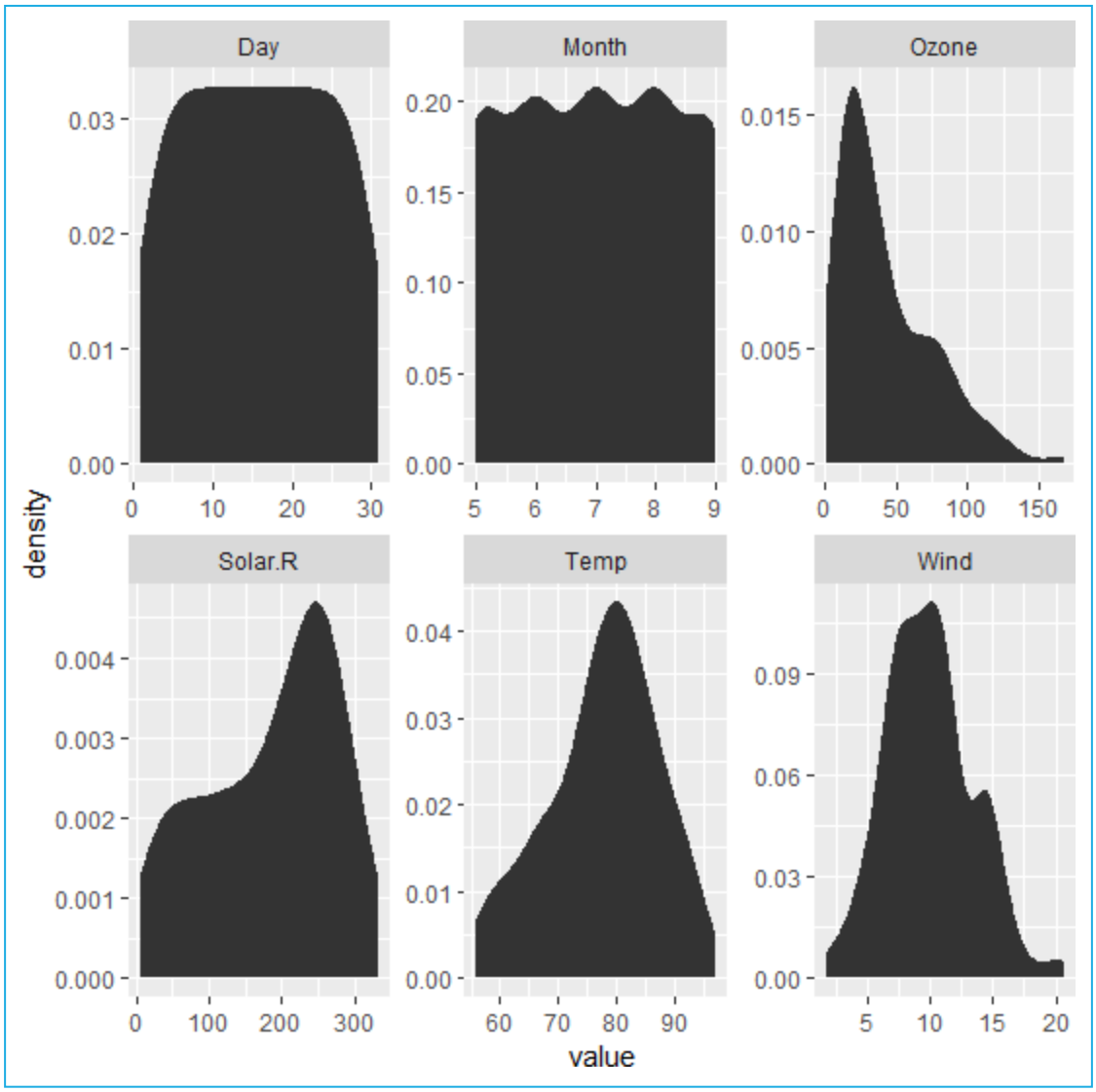
```
> #Read the file in Zip format and get it into R.
> #Answer1
> forecasturl = paste('https://archive.ics.uci.edu/ml/machine-learning-databases/00360/',
+                     'AirQualityUCI.zip', sep='')
> # create a temporary directory
> td = tempdir()
> # create the placeholder file
> tf = tempfile(tmpdir=td, fileext=".zip")
> # download into the placeholder file
> download.file(forecasturl, tf)
trying URL 'https://archive.ics.uci.edu/ml/machine-learning-databases/00360/AirQualityUCI.zip'
Content type 'application/zip' length 1543989 bytes (1.5 MB)
downloaded 1.5 MB

> # get the name of the first file in the zip archive
> fname = unzip(tf, list=TRUE)$Name[1]
> fname
[1] "AirQualityUCI.csv"
> # unzip the file to the temporary directory
> unzip(tf, files=fname, exdir=td, overwrite=TRUE)
> # fpath is the full path to the extracted file
> fpath = file.path(td, fname)
> fpath
[1] "C:\\Users\\VINEET~1\\AppData\\Local\\Temp\\RtmpwVxMFb/AirQualityUCI.csv"
> d = read.csv(fpath,sep = ";")
> View(d)
>
```

	Date	Time	CO.GT.	PT08.S1.CO.	NMHC.GT.	C6H6.GT.	PT08.S2.NMHC.	NOx.GT.	PT08.S3.NOx.	NO2.GT.	PT08.S4.NO2.	PT08.S5.O3.	T	RH	AH	X	X.1
1	10/03/2004	18.00.00	2,6	1360	150	11,9	1046	166	1056	113	1692	1268	13,6	48,9	0,7578	NA	NA
2	10/03/2004	19.00.00	2	1292	112	9,4	955	103	1174	92	1559	972	13,3	47,7	0,7255	NA	NA
3	10/03/2004	20.00.00	2,2	1402	88	9,0	939	131	1140	114	1555	1074	11,9	54,0	0,7502	NA	NA
4	10/03/2004	21.00.00	2,2	1376	80	9,2	948	172	1092	122	1584	1203	11,0	60,0	0,7867	NA	NA
5	10/03/2004	22.00.00	1,6	1272	51	6,5	836	131	1205	116	1490	1110	11,2	59,6	0,7888	NA	NA
6	10/03/2004	23.00.00	1,2	1197	38	4,7	750	89	1337	96	1393	949	11,2	59,2	0,7848	NA	NA
7	11/03/2004	00.00.00	1,2	1185	31	3,6	690	62	1462	77	1333	733	11,3	56,8	0,7603	NA	NA
8	11/03/2004	01.00.00	1	1136	31	3,3	672	62	1453	76	1333	730	10,7	60,0	0,7702	NA	NA
9	11/03/2004	02.00.00	0,9	1094	24	2,3	609	45	1579	60	1276	620	10,7	59,7	0,7648	NA	NA
10	11/03/2004	03.00.00	0,6	1010	19	1,7	561	-200	1705	-200	1235	501	10,3	60,2	0,7517	NA	NA
11	11/03/2004	04.00.00	-200	1011	14	1,3	527	21	1818	34	1197	445	10,1	60,5	0,7465	NA	NA
12	11/03/2004	05.00.00	0,7	1066	8	1,1	512	16	1918	28	1182	422	11,0	56,2	0,7366	NA	NA
13	11/03/2004	06.00.00	0,7	1052	16	1,6	553	34	1738	48	1221	472	10,5	58,1	0,7353	NA	NA
14	11/03/2004	07.00.00	1,1	1144	29	3,2	667	98	1490	82	1339	730	10,2	59,6	0,7417	NA	NA

b. Create Univariate for all the columns.

```
#we can do univariate analysis by the following command too
summary(airquality)
describe(airquality)

#or by visually
library(purrr)
library(tidyr)
library(ggplot2)

airquality%>%
  keep(is.numeric)%>%
  gather()%>%
  ggplot(aes(value)) +
  facet_wrap(~ key,scales = "free") +
  stat_density()
```

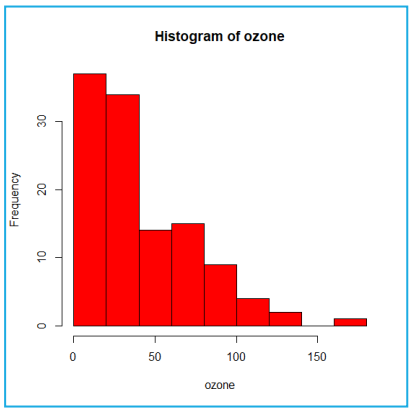
```
> #we can do univariate analysis by the following command too
> summary(airquality)
     Ozone           Solar.R          Wind             Temp           Month            Day
 Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00   Min.   :5.000   Min.   : 1.0
 1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00   1st Qu.:6.000   1st Qu.: 8.0
 Median : 31.50   Median :205.0   Median : 9.700   Median :79.00   Median :7.000   Median :16.0
 Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88   Mean   :6.993   Mean   :15.8
 3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.000   3rd Qu.:23.0
 Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.000   Max.   :31.0
 NA's   :37       NA's   :7
> describe(airquality)
        vars   n   mean    sd median trimmed   mad  min   max range  skew kurtosis   se
Ozone      1 116  42.13 32.99   31.5   37.80 25.95  1.0 168.0   167  1.21     1.11 3.06
Solar.R    2 146 185.93 90.06  205.0  190.34 98.59  7.0 334.0   327 -0.42    -1.00 7.45
Wind       3 153   9.96  3.52    9.7    9.87  3.41  1.7  20.7    19  0.34     0.03 0.28
Temp       4 153  77.88  9.47   79.0   78.28  8.90 56.0  97.0    41 -0.37    -0.46 0.77
Month      5 153   6.99  1.42    7.0    6.99  1.48  5.0   9.0     4  0.00    -1.32 0.11
Day        6 153  15.80  8.86   16.0   15.80 11.86  1.0  31.0    30  0.00    -1.22 0.72
>
```

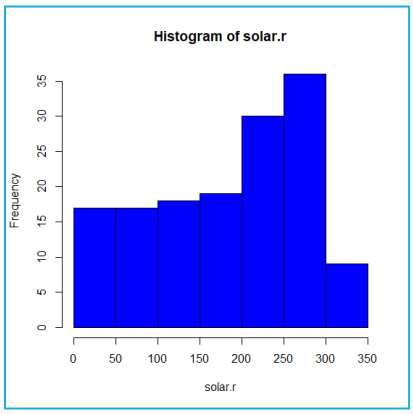


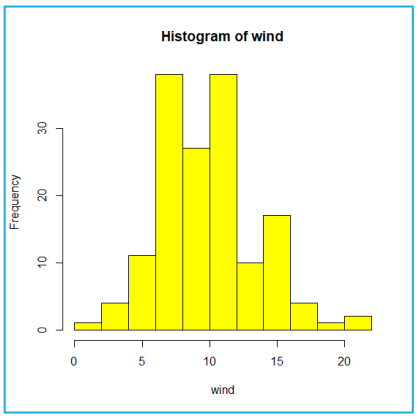
#or we can plot univariate individually for each variable

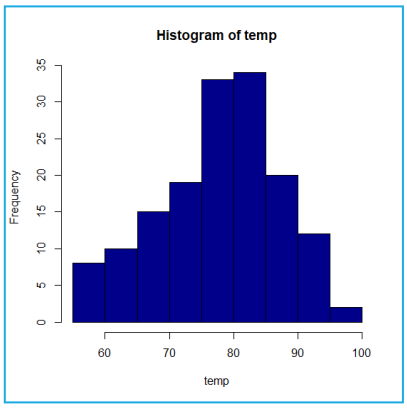
#hence plotting histogram

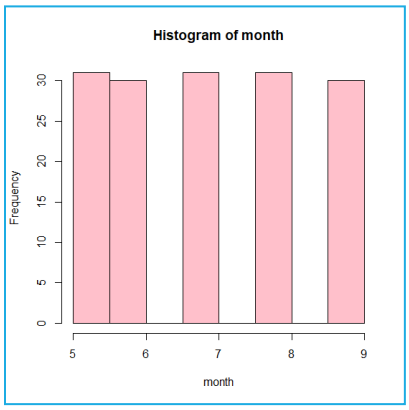
```
hist(airquality$Ozone ,xlab = "ozone", ylab = "Frequency",main="Histogram of ozone",col="red")
hist(airquality$Solar.R ,xlab = "solar.r", ylab = "Frequency",main="Histogram of solar.r",col="blue")
hist(airquality$Wind ,xlab = "wind", ylab = "Frequency",main="Histogram of wind",col="yellow")
hist(airquality$Temp ,xlab = "temp", ylab = "Frequency",main="Histogram of temp",col="darkblue")
hist(airquality$Month ,xlab = "month", ylab = "Frequency",main="Histogram of month",col="pink")
hist(airquality$Day ,xlab = "day", ylab = "Frequency",main="Histogram of day",col="purple")
```

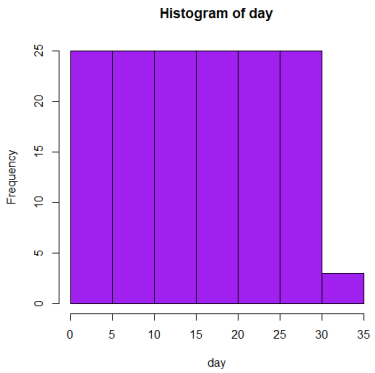












c. Check for missing values in all columns.

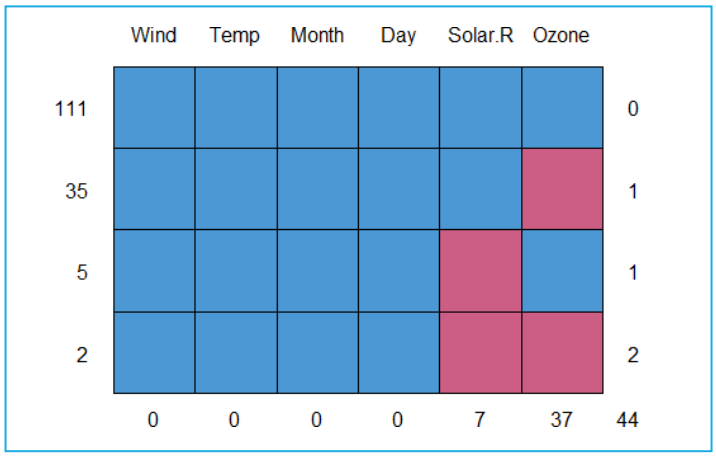
```
     Ozone           Solar.R         Wind             Temp           Month            Day       
 Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00   Min.   :5.000   Min.   : 1.0  
 1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00   1st Qu.:6.000   1st Qu.: 8.0  
 Median : 31.50   Median :205.0   Median : 9.700   Median :79.00   Median :7.000   Median :16.0  
 Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88   Mean   :6.993   Mean   :15.8  
 3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.000   3rd Qu.:23.0  
 Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.000   Max.   :31.0  
 NA's   :37       NA's   :7                                                                      
>
```

#thus ozone and solar.r has missing values

d. Impute the missing values using appropriate methods

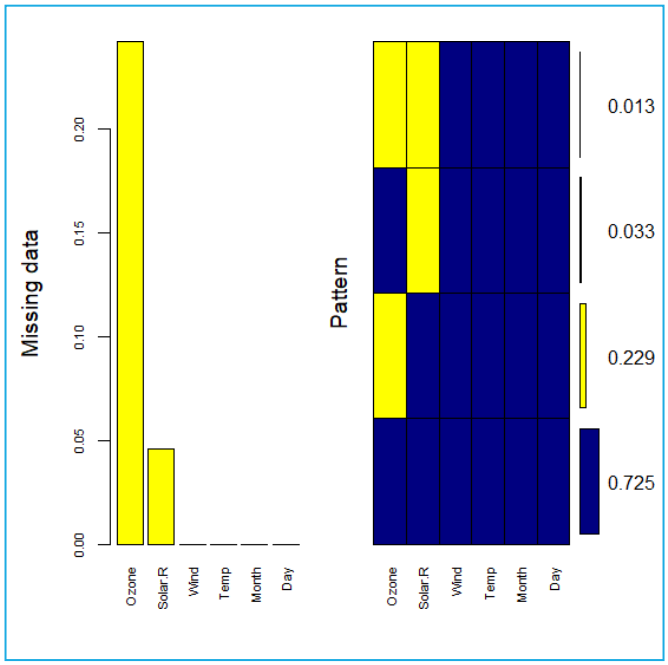
```
#first lets see the structure of airquality
str(airquality)
#Load Mice Library
library(mice)
md.pattern(airquality)
```

```
> md.pattern(airquality)
    Wind Temp Month Day Solar.R Ozone
111    1    1     1   1       1     1  0
35     1    1     1   1       1     0  1
5      1    1     1   1       0     1  1
2      1    1     1   1       0     0  2
       0    0     0   0       7    37 44
>
```



```
#visualizing
library(VIM)
mice_plot <- aggr(airquality, col=c('navyblue','yellow'),
                  numbers=TRUE, sortVars=TRUE,
                  labels=names(airquality), cex.axis=.7,
                  gap=3, ylab=c("Missing data","Pattern"))
```

```
Variables sorted by number of missings:
Variable       Count
   Ozone 0.24183007
 Solar.R 0.04575163
    Wind 0.00000000
    Temp 0.00000000
   Month 0.00000000
     Day 0.00000000
>
```



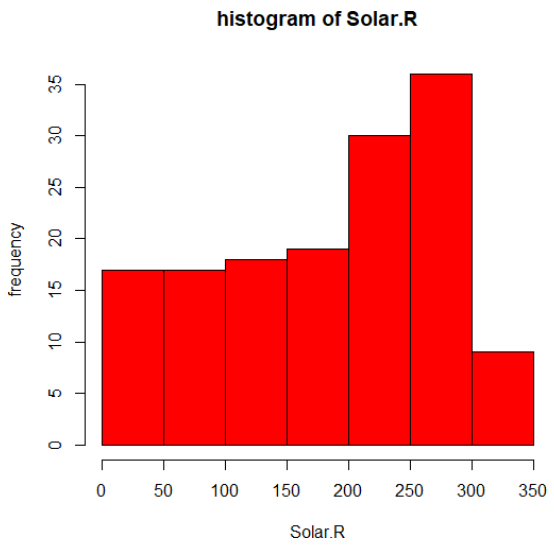
```
# In this case we are using predictive mean matching as imputation method
imputed_Data <- mice(airquality, m=5, maxit = 50, method = 'pmm', seed = 500)
summary(imputed_Data)
completeData <- complete(imputed_Data)
completeData
```

```
> summary(imputed_Data)
Class: mids
Number of multiple imputations:  5
Imputation methods:
  Ozone Solar.R    Wind    Temp   Month     Day
  "pmm"   "pmm"      ""      ""      ""      ""
PredictorMatrix:
        Ozone Solar.R Wind Temp Month Day
Ozone       0       1    1    1     1   1
Solar.R     1       0    1    1     1   1
Wind        1       1    0    1     1   1
Temp        1       1    1    0     1   1
Month       1       1    1    1     0   1
Day         1       1    1    1     1   0
>
```

	Ozone	Solar.R	Wind	Temp	Month	Day
1	41	190	7.4	67	5	1
2	36	118	8.0	72	5	2
3	12	149	12.6	74	5	3
4	18	313	11.5	62	5	4
5	6	115	14.3	56	5	5
6	28	274	14.9	66	5	6
7	23	299	8.6	65	5	7
8	19	99	13.8	59	5	8
9	8	19	20.1	61	5	9
10	12	194	8.6	69	5	10
11	7	275	6.9	74	5	11
12	16	256	9.7	69	5	12
13	11	290	9.2	66	5	13
14	14	274	10.9	68	5	14
15	18	65	13.2	58	5	15

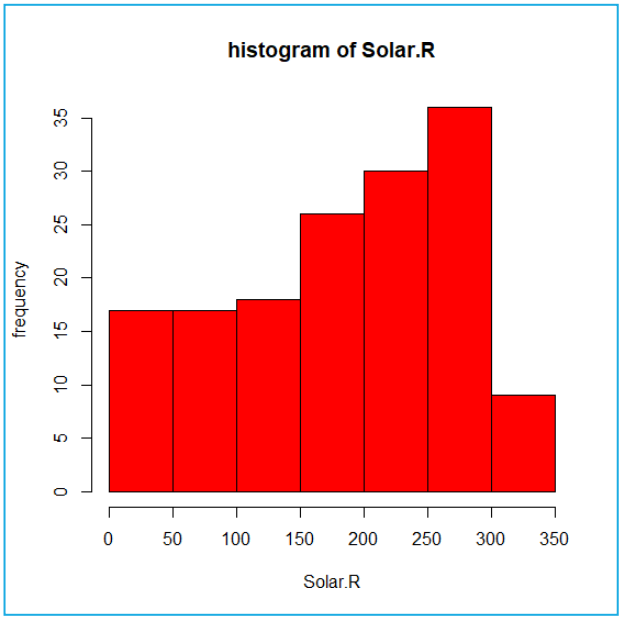
```
#or we an alternate way do it for variable Solar.R in airquality dataset
newair =airquality

dim(newair)
str(newair)
summary(newair)
#before imputing
hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of
Solar.R",col="red")

mean(newair$Solar.R)
mean(newair$Solar.R,na.rm = T)
```

```
> dim(newair)
[1] 153   6
> str(newair)
'data.frame':   153 obs. of  6 variables:
 $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
 $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
 $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
 $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
 $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
 $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
> summary(newair)
     Ozone           Solar.R           Wind             Temp           Month            Day
 Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00   Min.   :5.000   Min.   : 1.0
 1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00   1st Qu.:6.000   1st Qu.: 8.0
 Median : 31.50   Median :205.0   Median : 9.700   Median :79.00   Median :7.000   Median :16.0
 Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88   Mean   :6.993   Mean   :15.8
 3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.000   3rd Qu.:23.0
 Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.000   Max.   :31.0
 NA's   :37       NA's   :7
> #before imputing
> hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of Solar.R",col="red")
> mean(newair$Solar.R)
[1] NA
> mean(newair$Solar.R,na.rm = T)
[1] 185.9315
>
```



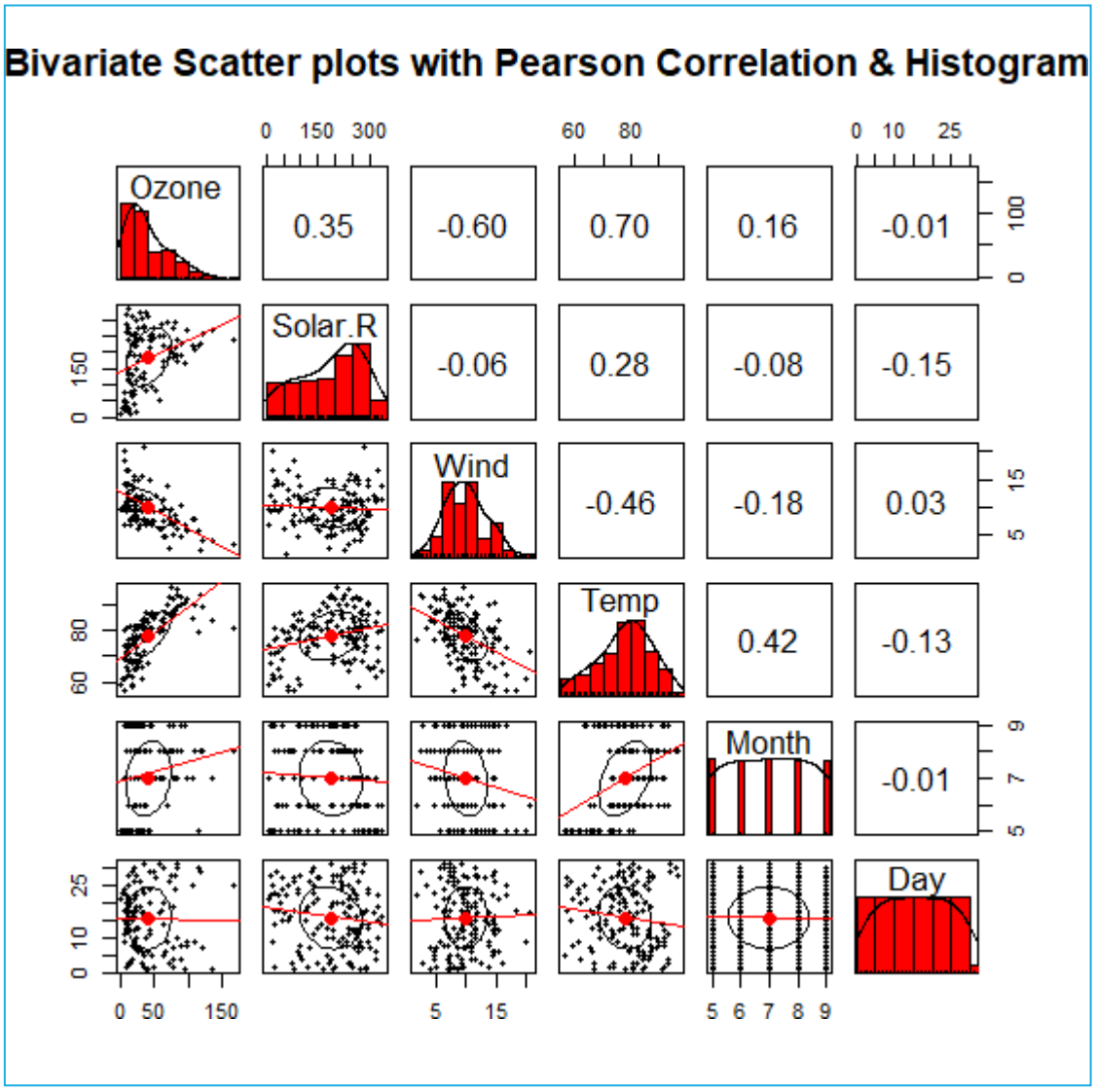
```
#imputed my mean
newair$Solar.R[is.na(newair$Solar.R)]<- mean(newair$Solar.R,na.rm = T)

#check summary after done with imputing
summary(newair)
newair$Solar.R

#visualize after imputing the variable Solar.R with the mean
#lets visualize through histogram
#after imputing
hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of
Solar.R",col="red")
```

```
> #imputed my mean
> newair$Solar.R[is.na(newair$Solar.R)]<- mean(newair$Solar.R,na.rm = T)
> #check summary after done with imputing
> summary(newair)
     Ozone           Solar.R           Wind             Temp           Month            Day
 Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00   Min.   :5.000   Min.   : 1.0
 1st Qu.: 18.00   1st Qu.:120.0   1st Qu.: 7.400   1st Qu.:72.00   1st Qu.:6.000   1st Qu.: 8.0
 Median : 31.50   Median :194.0   Median : 9.700   Median :79.00   Median :7.000   Median :16.0
 Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88   Mean   :6.993   Mean   :15.8
 3rd Qu.: 63.25   3rd Qu.:256.0   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.000   3rd Qu.:23.0
 Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.000   Max.   :31.0
 NA's   :37
> newair$Solar.R
  [1] 190.0000 118.0000 149.0000 313.0000 185.9315 185.9315 299.0000  99.0000  19.0000 194.0000 185.9315 256.0000 290.0000 274.0000  65.0000 334.0000 307.0000
 [18]  78.0000 322.0000  44.0000   8.0000 320.0000  25.0000  92.0000  66.0000 266.0000 185.9315  13.0000 252.0000 223.0000 279.0000 286.0000 287.0000 242.0000
 [35] 186.0000 220.0000 264.0000 127.0000 273.0000 291.0000 323.0000 259.0000 250.0000 148.0000 332.0000 322.0000 191.0000 284.0000  37.0000 120.0000 137.0000
 [52] 150.0000  59.0000  91.0000 250.0000 135.0000 127.0000  47.0000  98.0000  31.0000 138.0000 269.0000 248.0000 236.0000 101.0000 175.0000 314.0000 276.0000
 [69] 267.0000 272.0000 175.0000 139.0000 264.0000 175.0000 291.0000  48.0000 260.0000 274.0000 285.0000 187.0000 220.0000   7.0000 258.0000 295.0000 294.0000
 [86] 223.0000  81.0000  82.0000 213.0000 275.0000 253.0000 254.0000  83.0000  24.0000  77.0000 185.9315 185.9315 185.9315 255.0000 229.0000 207.0000 222.0000
[103] 137.0000 192.0000 273.0000 157.0000  64.0000  71.0000  51.0000 115.0000 244.0000 190.0000 259.0000  36.0000 255.0000 212.0000 238.0000 215.0000 153.0000
[120] 203.0000 225.0000 237.0000 188.0000 167.0000 197.0000 183.0000 189.0000  95.0000  92.0000 252.0000 220.0000 230.0000 259.0000 236.0000 259.0000 238.0000
[137]  24.0000 112.0000 237.0000 224.0000  27.0000 238.0000 201.0000 238.0000  14.0000 139.0000  49.0000  20.0000 193.0000 145.0000 191.0000 131.0000 223.0000
> #after imputing
> hist(newair$Solar.R ,xlab = "Solar.R", ylab = "frequency",main="histogram of Solar.R",col="red")
>
```



e. Create bi-variate analysis for all relationships

```
library(psych)
pairs.panels( airquality[,c(1,2,3,4,5,6)],
        method = "pearson", # correlation method
        hist.col = "red",
        density = TRUE, # show density plots
        ellipses = TRUE, # show correlation ellipses
        lm=TRUE,
        main ="Bivariate Scatter plots with Pearson Correlation & Histogram"
)
```



f. Test relevant hypothesis for valid relations

```
#lets find out the structure
str(airquality)
#we do paired test for continous variables
#some of test are as follows
#define the null hypothesis
#Ho: Mean of first variable - Mean of 2 variable is equal to 0
#Ha: Mean of first variable - Mean of 2 variable is not equal to 0
t.test(x=airquality$Ozone, y=airquality$Solar.R ,alternative = "two.sided",mu=0 ,paired = TRUE)
t.test(x=airquality$Temp, y=airquality$Wind ,alternative = "two.sided",mu=0 ,paired = TRUE)
t.test(x=airquality$Ozone, y=airquality$Temp ,alternative = "two.sided",mu=0 ,paired = TRUE)
t.test(x=airquality$Day, y=airquality$Solar.R ,alternative = "two.sided",mu=0 ,paired = TRUE)

#as p value of this test is <0.05 we reject the null hypo
#and accept the alternative hypothesis which says there
#Mean of 1 variable - Mean of 2 variable is not equal to 0
#thus this are some test that we performed
```

```
        Paired t-test

data:  airquality$Ozone and airquality$Solar.R
t = -17.593, df = 110, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -158.7772 -126.6282
sample estimates:
mean of the differences
              -142.7027

> t.test(x=airquality$Temp, y=airquality$Wind ,alternative = "two.sided",mu=0 ,paired = TRUE)

        Paired t-test

data:  airquality$Temp and airquality$Wind
t = 72.978, df = 152, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 66.08593 69.76374
sample estimates:
mean of the differences
               67.92484
```

```
> t.test(x=airquality$Ozone, y=airquality$Temp ,alternative = "two.sided",mu=0 ,paired = TRUE)

        Paired t-test

data:  airquality$Ozone and airquality$Temp
t = -14.14, df = 115, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -40.74819 -30.73457
sample estimates:
mean of the differences
              -35.74138

> t.test(x=airquality$Day, y=airquality$Solar.R ,alternative = "two.sided",mu=0 ,paired = TRUE)

        Paired t-test

data:  airquality$Day and airquality$Solar.R
t = -22.353, df = 145, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -184.8230 -154.7934
sample estimates:
mean of the differences
              -169.8082
```

g. Create cross tabulations with derived variables

```
attach(airquality)
unique(Wind)
unique(Temp)

#derived variables of wind and temp
x<- cut(Wind,quantile(Wind))
x<- cut(Wind,breaks = seq(1,21,3),labels =
c("wind1","wind2","wind3","wind4","wind5","wind6")) y<- cut(Temp,quantile(Temp))
y<- cut(Temp,breaks = seq(55,100,9),labels = c("temp1","temp2","temp3","temp4","temp5"))
table(x,y)
#or like this using xtabs function
mytable<- xtabs(~x+y,data = airquality)
mytable
#crosstabulate
library(gmodels)
CrossTable(x,y)
```

```
> unique(wind)
 [1]  7.4  8.0 12.6 11.5 14.3 14.9  8.6 13.8 20.1  6.9  9.7  9.2 10.9 13.2 12.0 18.4 16.6  5.7 16.1 20.7 10.3  6.3  1.7  4.6  4.1  5.1  4.0 15.5  3.4  2.3  2.8
> unique(Temp)
 [1] 67 72 74 62 56 66 65 59 61 69 68 58 64 57 73 81 79 76 78 84 85 82 87 90 93 92 80 77 75 83 88 89 91 86 97 94 96 71 63 70
> #derived variables of wind and temp
> x<- cut(wind,quantile(wind))
> x<- cut(wind,breaks = seq(1,21,3),labels = c("wind1","wind2","wind3","wind4","wind5","wind6"))
> y<- cut(Temp,quantile(Temp))
> y<- cut(Temp,breaks = seq(55,100,9),labels = c("temp1","temp2","temp3","temp4","temp5"))
> table(x,y)
       y
x       temp1 temp2 temp3 temp4 temp5
  wind1     0     0     2     1     2
  wind2     0     1    11    10     6
  wind3     4     9    18    14     3
  wind4     4    11    17     8     1
  wind5     4     4    13     3     0
  wind6     3     2     0     0     0
> #or like this using xtabs function
> mytable<- xtabs(~x+y,data = airquality)
> mytable
       y
x       temp1 temp2 temp3 temp4 temp5
  wind1     0     0     2     1     2
  wind2     0     1    11    10     6
  wind3     4     9    18    14     3
  wind4     4    11    17     8     1
  wind5     4     4    13     3     0
  wind6     3     2     0     0     0
>
```

Cell Contents

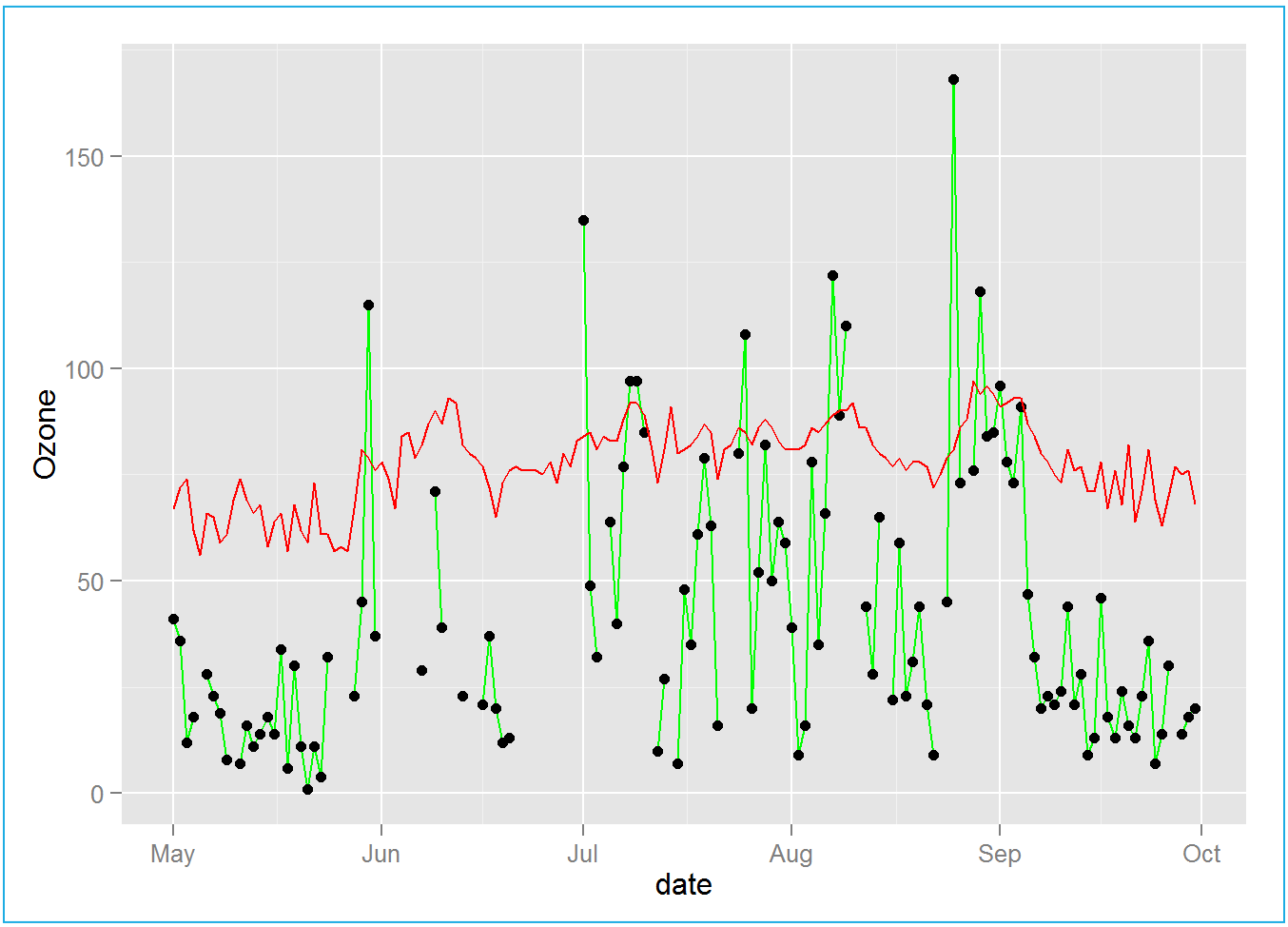
N
Chi-square contribution
N / Row Total
N / Col Total
N / Table Total

Total Observations in Table: 151

x	y temp1	temp2	temp3	temp4	temp5	Row Total
wind1	0	0	2	1	2	5
	0.497	0.894	0.000	0.031	6.464	
	0.000	0.000	0.400	0.200	0.400	0.033
	0.000	0.000	0.033	0.028	0.167	
	0.000	0.000	0.013	0.007	0.013	
wind2	0	1	11	10	6	28
	2.781	3.206	0.009	1.656	6.404	
	0.000	0.036	0.393	0.357	0.214	0.185
	0.000	0.037	0.180	0.278	0.500	
	0.000	0.007	0.073	0.066	0.040	
wind3	4	9	18	14	3	48
	0.124	0.020	0.100	0.571	0.174	
	0.083	0.188	0.375	0.292	0.062	0.318
	0.267	0.333	0.295	0.389	0.250	
	0.026	0.060	0.119	0.093	0.020	
wind4	4	11	17	8	1	41
	0.001	1.836	0.012	0.322	1.565	
	0.098	0.268	0.415	0.195	0.024	0.272
	0.267	0.407	0.279	0.222	0.083	
	0.026	0.073	0.113	0.053	0.007	
wind5	4	4	13	3	0	24
	1.095	0.020	1.126	1.295	1.907	
	0.167	0.167	0.542	0.125	0.000	0.159
	0.267	0.148	0.213	0.083	0.000	
	0.026	0.026	0.086	0.020	0.000	
wind6	3	2	0	0	0	5
	12.617	1.368	2.020	1.192	0.397	
	0.600	0.400	0.000	0.000	0.000	0.033
	0.200	0.074	0.000	0.000	0.000	
	0.020	0.013	0.000	0.000	0.000	
Column Total	15	27	61	36	12	151
	0.099	0.179	0.404	0.238	0.079	

h. check for trends and patterns in time series

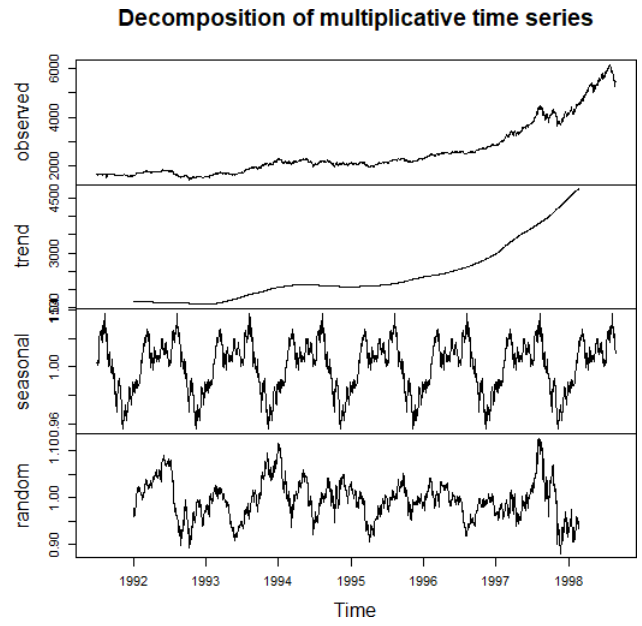
```
ggplot(airquality, aes(x = (Month * 100 + Day), y = Ozone)) + geom_line() +
geom_point()
```



```
ts (AirQualityUCI, frequency = 4, start = c(1959, 2)) # frequency 4 => Quarterly Data
ts (1:10, frequency = 12, start = 1990) # freq 12 => Monthly data.
ts (AirQualityUCI, start=c(2009), end=c(2014), frequency=1) # Yearly Data
ts (1:1000, frequency = 365, start = 1990)# freq 365 => daily data.
tsAirqualityUCI <- EuStockMarkets[, 1] # ts data copied some time series data as below
```

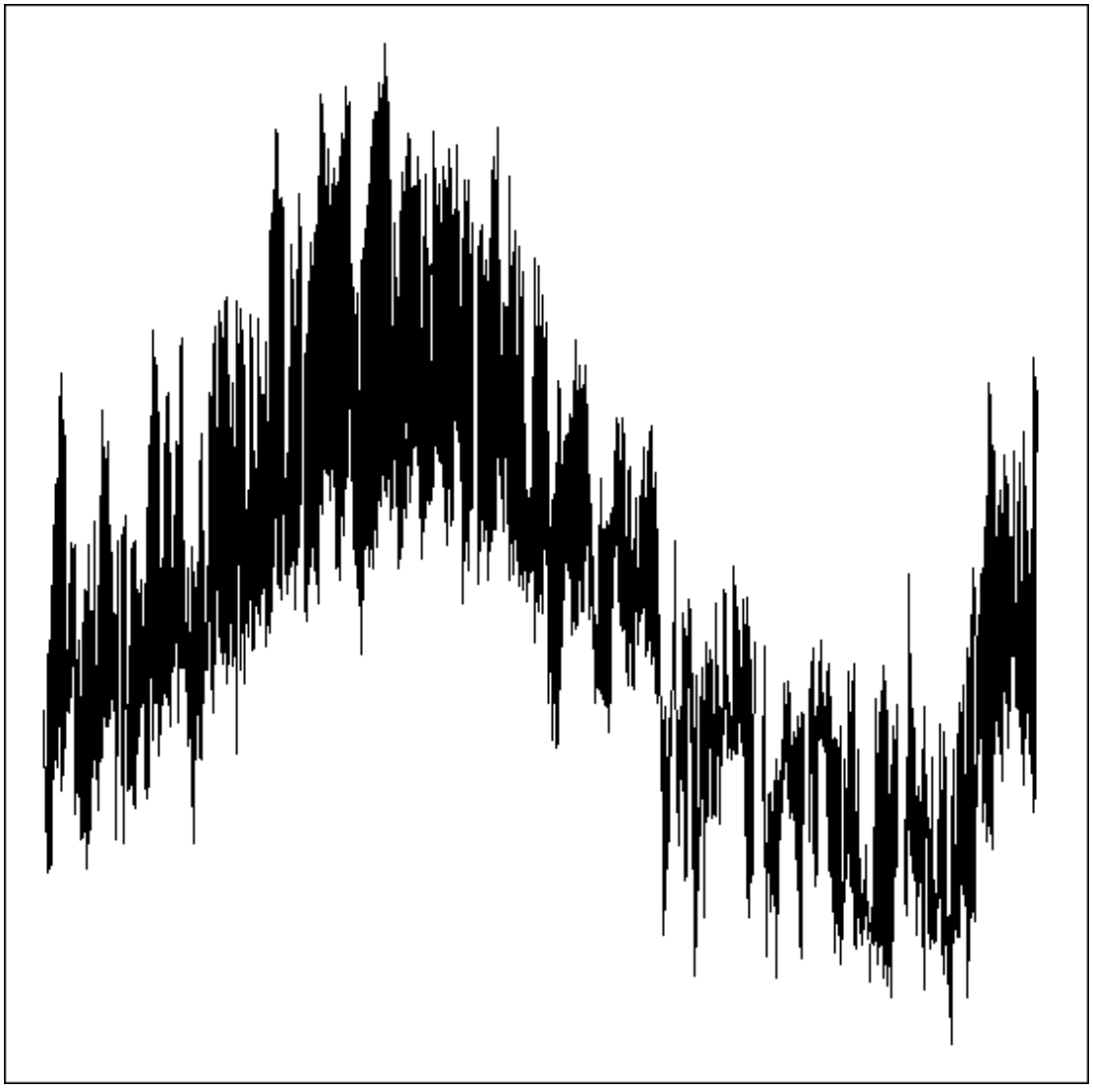
```
> #8. Check for trends and patterns in time series.
> ts (AirQualityUCI, frequency = 4, start = c(1959, 2)) # frequency 4 => Quarterly Data
     Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC) NOx(GT) PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3)    T  RH AH X16 X17
1959 Q2   NA   NA     NA          1360      150       NA           1046     166         1056     113         1692         1268 136 489 NA  NA  NA
1959 Q3   NA   NA      2          1292      112       NA            955     103         1174      92         1559          972 133 477 NA  NA  NA
1959 Q4   NA   NA     NA          1402       88       NA            939     131         1140     114         1555         1074 119 540 NA  NA  NA
1960 Q1   NA   NA     NA          1376       80       NA            948     172         1092     122         1584         1203 110 600 NA  NA  NA
1960 Q2   NA   NA     NA          1272       51       NA            836     131         1205     116         1490         1110 112 596 NA  NA  NA
1960 Q3   NA   NA     NA          1197       38       NA            750      89         1337      96         1393          949 112 592 NA  NA  NA
1960 Q4   NA   NA     NA          1185       31       NA            690      62         1462      77         1333          733 113 568 NA  NA  NA
1961 Q1   NA   NA      1          1136       31       NA            672      62         1453      76         1333          730 107 600 NA  NA  NA
1961 Q2   NA   NA     NA          1094       24       NA            609      45         1579      60         1276          620 107 597 NA  NA  NA
1961 Q3   NA   NA     NA          1010       19       NA            561      NA         1705      NA         1235          501 103 602 NA  NA  NA
1961 Q4   NA   NA     NA          1011       14       NA            527      21         1818      34         1197          445 101 605 NA  NA  NA
1962 Q1   NA   NA     NA          1066        8       NA            512      16         1918      28         1182          422 110 562 NA  NA  NA
1962 Q2   NA   NA     NA          1052       16       NA            553      34         1738      48         1221          472 105 581 NA  NA  NA
1962 Q3   NA   NA     NA          1144       29       NA            667      98         1490      82         1339          730 102 596 NA  NA  NA
1962 Q4   NA   NA      2          1333       64       NA            900     174         1136     112         1517         1102 108 574 NA  NA  NA
1963 Q1   NA   NA     NA          1351       87       NA            960     129         1079     101         1583         1028 105 606 NA  NA  NA
1963 Q2   NA   NA     NA          1233       77       NA            827     112         1218      98         1446          860 108 584 NA  NA  NA
1963 Q3   NA   NA     NA          1179       43       NA            762      95         1328      92         1362          671 105 579 NA  NA  NA
1963 Q4   NA   NA     NA          1236       61       NA            774     104         1301      95         1401          664  95 668 NA  NA  NA
1964 Q1   NA   NA     NA          1286       63       NA            869     146         1162     112         1537          799  83 764 NA  NA  NA
1964 Q2   NA   NA     NA          1371      164       NA           1034     207          983     128         1730         1037  80 811 NA  NA  NA
1964 Q3   NA   NA     NA          1310       79       NA            933     184         1082     126         1647          946  83 798 NA  NA  NA
1964 Q4   NA   NA     NA          1292       95       NA            912     193         1103     131         1591          957  97 712 NA  NA  NA
1965 Q1   NA   NA     NA          1383      150       NA           1020     243         1008     135         1719         1104  98 676 NA  NA  NA
1965 Q2   NA   NA     NA          1581      307       NA           1319     281          799     151         2083         1409 103 642 NA  NA  NA
1965 Q3   NA   NA     NA          1776      461       NA           1488     383          702     172         2333         1704  97 693 NA  NA  NA
1965 Q4   NA   NA     NA          1640      401       NA           1404     351          743     165         2191         1654  96 678 NA  NA  NA
1966 Q1   NA   NA     NA          1313      197       NA           1076     240          957     136         1707         1285  91 640 NA  NA  NA
1966 Q2   NA   NA     NA           965       61       NA            749      94         1325      85         1333          821  82 634 NA  NA  NA
```

```
> ts (1:10, frequency = 12, start = 1990) # freq 12 => Monthly data.
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct
1990   1   2   3   4   5   6   7   8   9  10
> ts (AirQualityUCI, start=c(2009), end=c(2014), frequency=1) # Yearly Data
Time Series:
Start = 2009
End = 2014
Frequency = 1
     Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC) NOx(GT) PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3)    T  RH AH X16 X17
2009   NA   NA     NA          1360      150       NA           1046     166         1056     113         1692         1268 136 489 NA  NA  NA
2010   NA   NA      2          1292      112       NA            955     103         1174      92         1559          972 133 477 NA  NA  NA
2011   NA   NA     NA          1402       88       NA            939     131         1140     114         1555         1074 119 540 NA  NA  NA
2012   NA   NA     NA          1376       80       NA            948     172         1092     122         1584         1203 110 600 NA  NA  NA
2013   NA   NA     NA          1272       51       NA            836     131         1205     116         1490         1110 112 596 NA  NA  NA
2014   NA   NA     NA          1197       38       NA            750      89         1337      96         1393          949 112 592 NA  NA  NA
Warning messages:
1: In data.matrix(data) : NAs introduced by coercion
2: In data.matrix(data) : NAs introduced by coercion
3: In data.matrix(data) : NAs introduced by coercion
4: In data.matrix(data) : NAs introduced by coercion
5: In data.matrix(data) : NAs introduced by coercion
>
```



i. **Find out the most polluted time of the day and the name of the chemical compound.**

```
tsAirqualityUCI <- EuStockMarkets[, 1] # ts data
decomposedRes <- decompose(tsAirqualityUCI, type="mult") # use type = "additive" for
additive components
plot (decomposedRes) # see plot below
stlRes <- stl(tsAirqualityUCI, s.window = "periodic")
plot(AirQualityUCI$T, type = "l")
```



Date	Time	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)
6/8/2004	8:00:00	376	525	125	2746	1708
6/9/2004	8:00:00	357	507	151	2691	2147
10/26/2004	18:00:00	952	325	180	2775	2372
max		1479.0	2682.8	339.7	2775.0	2522.8

Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)
6/8/2004	8:00:00	5.8	1377	-200	36.1	1688
6/9/2004	8:00:00	6.4	1496	-200	36.9	1705
10/26/2004	18:00:00	9.5	1908	-200	52.1	2007
Max		11.9	2039.8	1189.0	63.7	2214.0

Date	Time	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)
6/8/2004	8:00:00	376	525	125	2746	1708
6/9/2004	8:00:00	357	507	151	2691	2147
10/26/2004	18:00:00	952	325	180	2775	2372
max		1479.0	2682.8	339.7	2775.0	2522.8