Session 10 Assignment 1

Import dataset from the following link: AirQuality Data Set

Perform the following written operations:

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

2. Create Univariate for all the columns.

3. Check for missing values in all columns.

4. Impute the missing values using appropriate methods.

5. Create bi-variate analysis for all relationships.

6. Test relevant hypothesis for valid relations.

7. Create cross tabulations with derived variables.

8. Check for trends and patterns in time series. 9. Find out the most polluted time of the day and the name of the chemical compound.

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

Ans:-

mydata<-read\_csv("AirqualityUCI.zip")

library(readr)

AirQualityUCI <- read\_delim("AirQualityUCI.zip",

";", escape\_double = FALSE, trim\_ws = TRUE)

View(AirQualityUCI)

Multiple files in zip: reading 'AirQualityUCI.csv'

Parsed with column specification:

cols(`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;;` = col\_character()

)

number of columns of result is not a multiple of vector length (arg 1)9357 parsing failures.

row # A tibble: 5 x 5 col row col expected actual file expected *<int>* *<chr>* *<chr>* *<chr>* *<chr>* actual 1 1 NA 1 columns 6 columns 'AirqualityUCI.zip' file 2 2 NA 1 columns 5 columns 'AirqualityUCI.zip' row 3 3 NA 1 columns 6 columns 'AirqualityUCI.zip' col 4 4 NA 1 columns 6 columns 'AirqualityUCI.zip' expected 5 5 NA 1 columns 6 columns 'AirqualityUCI.zip'

... ................................. ... ..................................................... ........ .................................................................................................................................................................................. ...... ............................................................................... .... ............................................................................... ... ............................................................................... ... ............................................................................... ........ ...............................................................................

See problems(...) for more details.

Multiple files in zip: reading 'AirQualityUCI.csv'

Missing column names filled in: 'X16' [16], 'X17' [17]Parsed with column specification:

cols(

Date = col\_character(),

Time = col\_character(),

`CO(GT)` = col\_character(),

`PT08.S1(CO)` = col\_integer(),

`NMHC(GT)` = col\_integer(),

`C6H6(GT)` = col\_character(),

`PT08.S2(NMHC)` = col\_integer(),

`NOx(GT)` = col\_integer(),

`PT08.S3(NOx)` = col\_integer(),

`NO2(GT)` = col\_integer(),

`PT08.S4(NO2)` = col\_integer(),

`PT08.S5(O3)` = col\_integer(),

T = col\_number(),

RH = col\_number(),

AH = col\_character(),

X16 = col\_character(),

X17 = col\_character()

)

Other method

## a quicker way that doesnt require that you know which files - just does all  
## \ allows you to use the . in .zip, the . is a special character  
## $ is tells the pattern to search is the end? not sure about this one  
for (i in dir(pattern="\.zip$"))  
unzip(i)

*2. Create Univariate for all the columns*.

AirQualityUCI[AirQualityUCI==-200.0]<-NA

for(i in 1:ncol(AirQualityUCI)){AirQualityUCI[is.na(AirQualityUCI[,i]),i] <- mean(AirQualityUCI[,i], na.rm = TRUE)}

summary(AirQualityUCI)

AirQualityUCI[7:14,]

hist(AirQualityUCI$`NOx(GT)`,col="red")

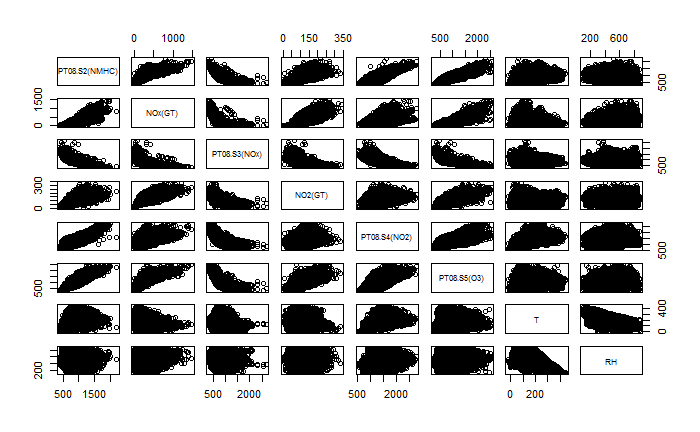
dotchart(AirQualityUCI$`PT08.S2(NMHC)`,labels = row.names(AirQualityUCI$`PT08.S1(CO)`),cex=0.5, color = "blue")

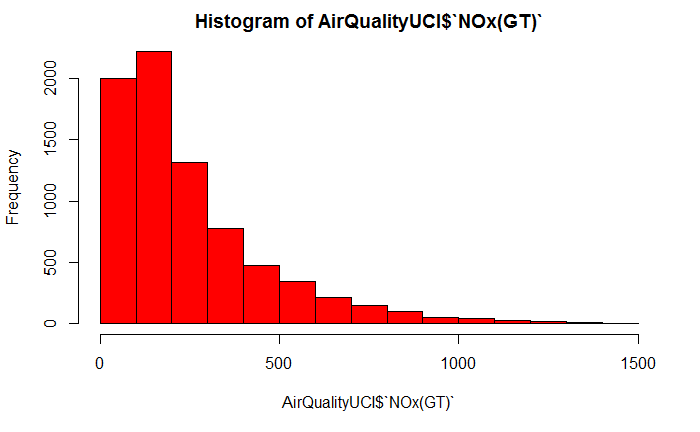
pairs(AirQualityUCI[7:14])

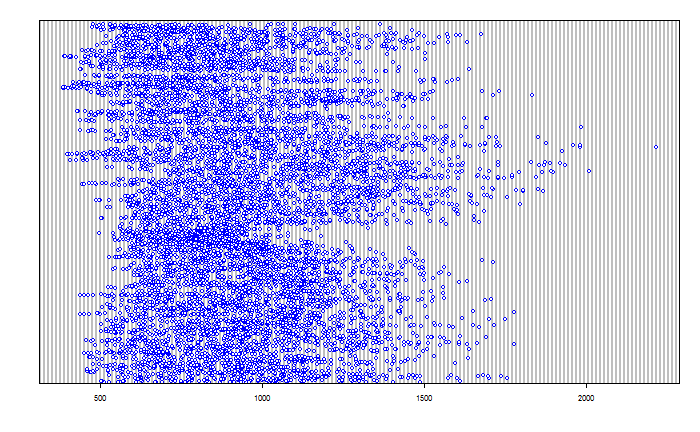
| **Date**  <chr> | **Time**  <chr> | **CO(GT)**  <chr> | **PT08.S1(CO)**  <dbl> | **NMHC(GT)**  <dbl> | **C6H6(GT)**  <chr> | **PT08.S2(NMHC)**  <dbl> |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11/03/2004 | 00.00.00 | 1,2 | 1185 | 31 | 3,6 | 690 |  |
| 11/03/2004 | 01.00.00 | 1 | 1136 | 31 | 3,3 | 672 |  |
| 11/03/2004 | 02.00.00 | 0,9 | 1094 | 24 | 2,3 | 609 |  |
| 11/03/2004 | 03.00.00 | 0,6 | 1010 | 19 | 1,7 | 561 |  |
| 11/03/2004 | 04.00.00 | *NA* | 1011 | 14 | 1,3 | 527 |  |
| 11/03/2004 | 05.00.00 | 0,7 | 1066 | 8 | 1,1 | 512 |  |
| 11/03/2004 | 06.00.00 | 0,7 | 1052 | 16 | 1,6 | 553 |  |
| 11/03/2004 | 07.00.00 | 1,1 | 1144 | 29 | 3,2 | 667 |  |

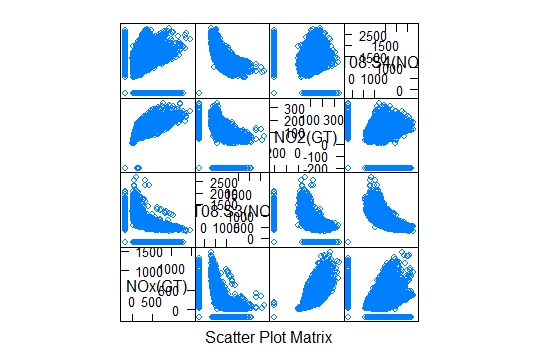
8 rows | 1-7 of 17 columns

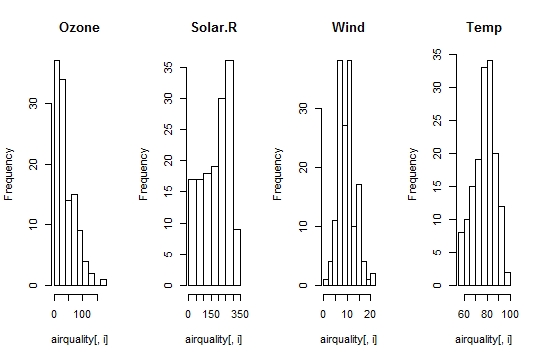
univariateTable(~Date +Time + CO(GT) + PT08.S1(CO)+ NMHC(GT)+ C6H6(GT)+ PT08.S2(NMHC)+ NOx(GT)+ PT08.S3(NOx) ,data=AirqualityUCI)











*3. Check for missing values in all columns.*

> colSums(is.na(AirQualityUCI)) # Number of missing per column/variable

Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT)

114 114 114 114 114 114

PT08.S2(NMHC) NOx(GT) PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3)

114 114 114 114 114 114

T RH AH X16 X17

114 114 114 9471 9471

# Pattern of missing values

library(mice)

md.pattern(AirQualityUCI) # pattern or missing values in data.

Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC) NOx(GT)

9357 1 1 1 1 1 1 1 1

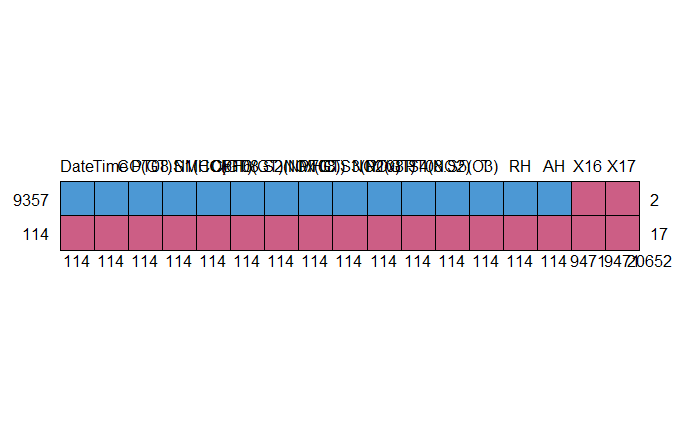
114 0 0 0 0 0 0 0 0

114 114 114 114 114 114 114 114

PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3) T RH AH X16 X17

9357 1 1 1 1 1 1 1 0 0 2

114 0 0 0 0 0 0 0 0 0 17

114 114 114 114 114 114 114 9471 9471 20652 

|  |
| --- |
| > str(AirQualityUCI)  Classes ‘tbl\_df’, ‘tbl’ and 'data.frame': 9471 obs. of 17 variables:  $ Date : chr "10/03/2004" "10/03/2004" "10/03/2004" "10/03/2004" ...  $ Time : chr "18.00.00" "19.00.00" "20.00.00" "21.00.00" ...  $ CO(GT) : chr "2,6" "2" "2,2" "2,2" ...  $ PT08.S1(CO) : int 1360 1292 1402 1376 1272 1197 1185 1136 1094 1010 ...  $ NMHC(GT) : int 150 112 88 80 51 38 31 31 24 19 ...  $ C6H6(GT) : chr "11,9" "9,4" "9,0" "9,2" ...  $ PT08.S2(NMHC): int 1046 955 939 948 836 750 690 672 609 561 ...  $ NOx(GT) : int 166 103 131 172 131 89 62 62 45 -200 ...  $ PT08.S3(NOx) : int 1056 1174 1140 1092 1205 1337 1462 1453 1579 1705 ...  $ NO2(GT) : int 113 92 114 122 116 96 77 76 60 -200 ...  $ PT08.S4(NO2) : int 1692 1559 1555 1584 1490 1393 1333 1333 1276 1235 ...  $ PT08.S5(O3) : int 1268 972 1074 1203 1110 949 733 730 620 501 ...  $ T : num 136 133 119 110 112 112 113 107 107 103 ...  $ RH : num 489 477 540 600 596 592 568 600 597 602 ...  $ AH : chr "0,7578" "0,7255" "0,7502" "0,7867" ...  $ X16 : chr NA NA NA NA ...  $ X17 : chr NA NA NA NA ...  - attr(\*, "spec")=List of 2  ..$ cols :List of 17  .. ..$ Date : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ Time : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ CO(GT) : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ PT08.S1(CO) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ NMHC(GT) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ C6H6(GT) : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ PT08.S2(NMHC): list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ NOx(GT) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ PT08.S3(NOx) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ NO2(GT) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ PT08.S4(NO2) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ PT08.S5(O3) : list()  .. .. ..- attr(\*, "class")= chr "collector\_integer" "collector"  .. ..$ T : list()  .. .. ..- attr(\*, "class")= chr "collector\_number" "collector"  .. ..$ RH : list()  .. .. ..- attr(\*, "class")= chr "collector\_number" "collector"  .. ..$ AH : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ X16 : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  .. ..$ X17 : list()  .. .. ..- attr(\*, "class")= chr "collector\_character" "collector"  ..$ default: list()  .. ..- attr(\*, "class")= chr "collector\_guess" "collector"  ..- attr(\*, "class")= chr "col\_spec"  > summary(AirQualityUCI)  Date Time CO(GT) PT08.S1(CO) NMHC(GT)  Length:9471 Length:9471 Length:9471 Min. :-200 Min. :-200.0  Class :character Class :character Class :character 1st Qu.: 921 1st Qu.:-200.0  Mode :character Mode :character Mode :character Median :1053 Median :-200.0  Mean :1049 Mean :-159.1  3rd Qu.:1221 3rd Qu.:-200.0  Max. :2040 Max. :1189.0  NA's :114 NA's :114  C6H6(GT) PT08.S2(NMHC) NOx(GT) PT08.S3(NOx) NO2(GT)  Length:9471 Min. :-200.0 Min. :-200.0 Min. :-200 Min. :-200.00  Class :character 1st Qu.: 711.0 1st Qu.: 50.0 1st Qu.: 637 1st Qu.: 53.00  Mode :character Median : 895.0 Median : 141.0 Median : 794 Median : 96.00  Mean : 894.6 Mean : 168.6 Mean : 795 Mean : 58.15  3rd Qu.:1105.0 3rd Qu.: 284.0 3rd Qu.: 960 3rd Qu.: 133.00  Max. :2214.0 Max. :1479.0 Max. :2683 Max. : 340.00  NA's :114 NA's :114 NA's :114 NA's :114  PT08.S4(NO2) PT08.S5(O3) T RH AH  Min. :-200 Min. :-200.0 Min. :-200.0 Min. :-200.0 Length:9471  1st Qu.:1185 1st Qu.: 700.0 1st Qu.: 109.0 1st Qu.: 341.0 Class :character  Median :1446 Median : 942.0 Median : 172.0 Median : 486.0 Mode :character  Mean :1391 Mean : 975.1 Mean : 168.2 Mean : 465.3  3rd Qu.:1662 3rd Qu.:1255.0 3rd Qu.: 241.0 3rd Qu.: 619.0  Max. :2775 Max. :2523.0 Max. : 446.0 Max. : 887.0  NA's :114 NA's :114 NA's :114 NA's :114  X16 X17  Length:9471 Length:9471  Class :character Class :character  Mode :character Mode :character          > is.na(AirQualityUCI)  Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC) NOx(GT) PT08.S3(NOx)  [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [2,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [3,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [4,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [5,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [6,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [7,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [8,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [9,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [10,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [11,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [12,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [13,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [14,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [15,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [16,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [17,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [18,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [19,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [20,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [21,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [22,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [23,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [24,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [25,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [26,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [27,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [28,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [29,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [30,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [31,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [32,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [33,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [34,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [35,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [36,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [37,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [38,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [39,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [40,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [41,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [42,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [43,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [44,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [45,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [46,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [47,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [48,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [49,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [50,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [51,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [52,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [53,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [54,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [55,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [56,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [57,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  [58,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  NO2(GT) PT08.S4(NO2) PT08.S5(O3) T RH AH X16 X17  [1,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [2,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [3,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [4,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [5,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [6,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [7,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [8,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [9,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [10,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [11,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [12,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [13,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [14,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [15,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [16,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [17,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [18,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [19,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [20,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [21,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [22,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [23,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [24,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [25,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [26,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [27,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [28,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [29,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [30,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [31,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [32,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [33,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [34,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [35,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [36,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [37,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [38,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [39,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [40,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [41,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [42,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [43,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [44,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [45,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [46,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [47,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [48,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [49,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [50,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [51,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [52,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [53,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [54,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [55,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [56,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [57,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [58,] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE  [ reached getOption("max.print") -- omitted 9413 rows ]  > library(Amelia)  > library(mlbench)  > # create a missing map  > missmap(AirQualityUCI, col=c("black", "grey"), legend=FALSE)  Warning messages:  1: In if (class(obj) == "amelia") { :  the condition has length > 1 and only the first element will be used  2: Unknown or uninitialised column: 'arguments'.  3: Unknown or uninitialised column: 'arguments'.  4: Unknown or uninitialised column: 'imputations'. |
|  |
| |  | | --- | | > | |



colSums(is.na(AirQualityUCI)) # Number of missing per column/variable

> colSums(is.na(AirQualityUCI)) # Number of missing per column/variable

Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT)

114 114 114 114 114 114

PT08.S2(NMHC) NOx(GT) PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3)

114 114 114 114 114 114

T RH AH X16 X17

114 114 114 9471 9471

4. Impute the missing values using appropriate methods.

Ans:-

colSums(is.na(AirQualityUCI)) # Number of missing per column/variable

#filling the missing values by NA

library(plyr)

AirQualityUCI[AirQualityUCI==-200.0]<-NA

#Replacing the NA by mean of each columns

for(i in 1:ncol(AirQualityUCI)){

AirQualityUCI[is.na(AirQualityUCI[,i]),i] <- mean(AirQualityUCI[,i], na.rm = TRUE)}

summary(AirQualityUCI)

Mode :character Mode :character Mode :character Median :1063

Mean :1100

3rd Qu.:1231

Max. :2040

NA's :480

NMHC(GT) C6H6(GT) PT08.S2(NMHC) NOx(GT)

Min. : 7.0 Length:9471 Min. : 383.0 Min. : 2.0

1st Qu.: 67.0 Class :character 1st Qu.: 734.5 1st Qu.: 98.0

Median : 150.0 Mode :character Median : 909.0 Median : 180.0

Mean : 218.8 Mean : 939.2 Mean : 246.9

3rd Qu.: 297.0 3rd Qu.:1116.0 3rd Qu.: 326.0

Max. :1189.0 Max. :2214.0 Max. :1479.0

NA's :8557 NA's :480 NA's :1753

PT08.S3(NOx) NO2(GT) PT08.S4(NO2) PT08.S5(O3) T

Min. : 322.0 Min. : 2.0 Min. : 551 Min. : 221.0 Min. :-19.0

1st Qu.: 658.0 1st Qu.: 78.0 1st Qu.:1227 1st Qu.: 731.5 1st Qu.:118.0

Median : 806.0 Median :109.0 Median :1463 Median : 963.0 Median :178.0

Mean : 835.5 Mean :113.1 Mean :1456 Mean :1022.9 Mean :183.2

3rd Qu.: 969.5 3rd Qu.:142.0 3rd Qu.:1674 3rd Qu.:1273.5 3rd Qu.:244.0

Max. :2683.0 Max. :340.0 Max. :2775 Max. :2523.0 Max. :446.0

NA's :480 NA's :1756 NA's :480 NA's :480 NA's :480

RH AH X16 X17

Min. : 92.0 Length:9471 Length:9471 Length:9471

1st Qu.:358.0 Class :character Class :character Class :character

Median :496.0 Mode :character Mode :character Mode :character

Mean :492.3

3rd Qu.:625.0

Max. :887.0

NA's :480

*5. Create bi-variate analysis for all relationships*.

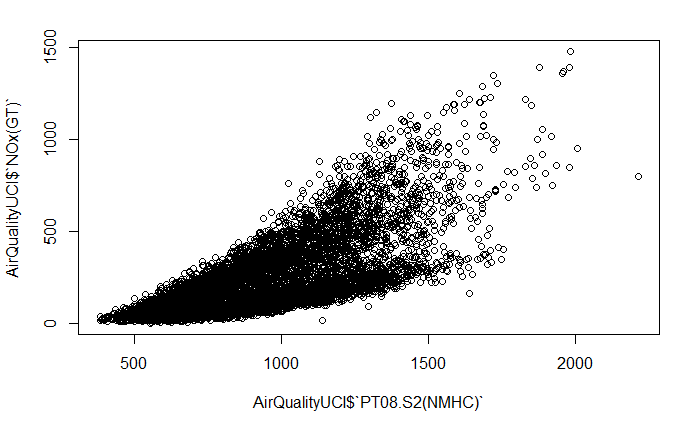
summary(AirQualityUCI)

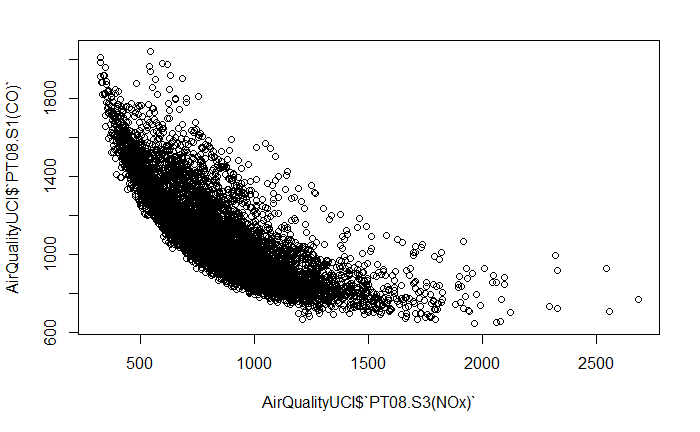
plot(AirQualityUCI$`NOx(GT)`~AirQualityUCI$`PT08.S2(NMHC)`)

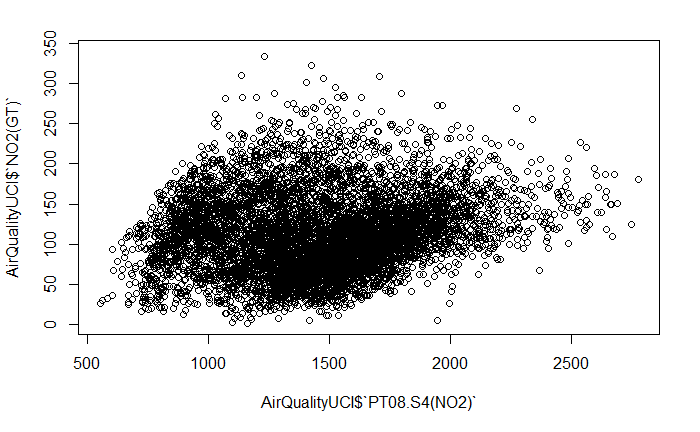
plot(AirQualityUCI$`PT08.S1(CO)`~AirQualityUCI$`PT08.S3(NOx)`)

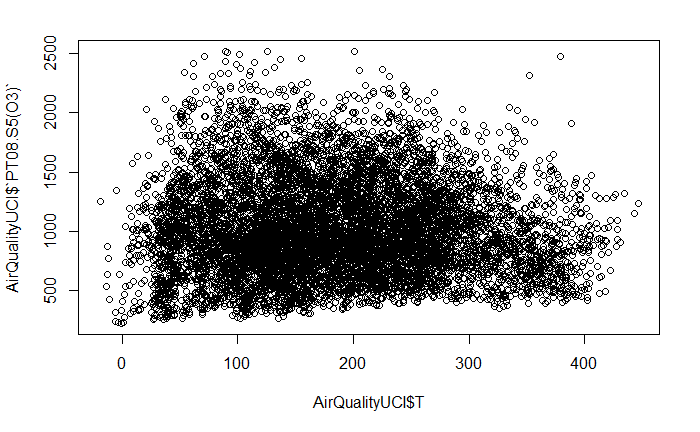
plot(AirQualityUCI$`NO2(GT)`~AirQualityUCI$`PT08.S4(NO2)`)

plot(AirQualityUCI$`PT08.S5(O3)`~AirQualityUCI$T)









***6. Test relevant hypothesis for valid relations.***

plot(AirQualityUCI$`PT08.S1(CO)`,AirQualityUCI$T)

lm(formula=AirQualityUCI$`PT08.S3(NOx)`~AirQualityUCI$`NOx(GT)`)

lm(formula = AirQualityUCI$`PT08.S1(CO)`~AirQualityUCI$T)

lm(formula = AirQualityUCI$`NMHC(GT)`~AirQualityUCI$`PT08.S2(NMHC)`)

plot(AirQualityUCI$`PT08.S5(O3)`,AirQualityUCI$`NOx(GT)`)

lm(formula =AirQualityUCI$`PT08.S5(O3)`~AirQualityUCI$`NOx(GT)` )

pnorm(1.49)

pnorm(1.097)

qnorm(0.9318879)

qnorm(0.8636793)

Call:

lm(formula = AirQualityUCI$`PT08.S3(NOx)` ~ AirQualityUCI$`NOx(GT)`)

Coefficients:

(Intercept) AirQualityUCI$`NOx(GT)`

1022.2737 -0.8165

Call:

lm(formula = AirQualityUCI$`PT08.S1(CO)` ~ AirQualityUCI$T)

Coefficients:

(Intercept) AirQualityUCI$T

1077.9402 0.1195

Call:

lm(formula = AirQualityUCI$`NMHC(GT)` ~ AirQualityUCI$`PT08.S2(NMHC)`)

Coefficients:

(Intercept) AirQualityUCI$`PT08.S2(NMHC)`

-410.0522 0.6663

Call:

lm(formula = AirQualityUCI$`PT08.S5(O3)` ~ AirQualityUCI$`NOx(GT)`)

Coefficients:

(Intercept) AirQualityUCI$`NOx(GT)`

670.796 1.548

library(car)

mod=lm(AirQualityUCI$`PT08.S5(O3)` ~ AirQualityUCI$`NOx(GT)`)

summary(mod)

predict(mod)

Call:

lm(formula = AirQualityUCI$`PT08.S5(O3)` ~ AirQualityUCI$`NOx(GT)`)

Residuals:

Min 1Q Median 3Q Max

-978.34 -172.18 -16.95 143.35 1324.95

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 670.79645 4.48936 149.4 <2e-16 \*\*\*

AirQualityUCI$`NOx(GT)` 1.54807 0.01411 109.7 <2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 250.4 on 7394 degrees of freedom

(2075 observations deleted due to missingness)

Multiple R-squared: 0.6194, Adjusted R-squared: 0.6194

F-statistic: 1.204e+04 on 1 and 7394 DF, p-value: < 2.2e-16

pnorm(1.49)

pnorm(1.097)

qnorm(0.9318879)

qnorm(0.8636793)

[1] 0.9318879

[1] 0.8636793

[1] 1.49

[1] 1.097

1 2 3 4 5 6 7

927.7768 830.2481 873.5942 937.0653 873.5942 808.5751 766.7771 766.7771

9 11 12 13 14 15 16 17

740.4598 703.3060 695.5656 723.4310 822.5077 940.1614 870.4980 844.1808

18 19 20 21 22 23 24 25

817.8635 831.7962 896.8153 991.2479 955.6421 969.5748 1046.9785 1105.8054

26 27 28 29 30 31 32 33

1263.7090 1214.1706 1042.3343 816.3154 743.5559 859.6615 876.6903 797.7386

35 36 37 38 39 41 42 43

703.3060 717.2387 757.4886 839.5366 1146.0553 960.2864 1005.1805 892.1711

44 45 46 47 48 49 50 51

918.4884 923.1326 964.9306 946.3537 903.0076 989.6998 983.5075 1197.1418

52 53 54 55 56 57 59 60

1094.9688 1062.4593 1135.2188 969.5748 885.9788 799.2866 839.5366 766.7771

61 62 63 64 65 66 67 68

752.8444 885.9788 1067.1035 1127.4784 1057.8151 1129.0265 1040.7862 907.6518

69 70 71 72 73 74 75 76

853.4692 855.0173 884.4307 899.9115 1022.2093 1099.6131 1102.7092 1108.9015

77 78 79 80 81 83 84 85

1002.0844 937.0653 964.9306 940.1614 868.9500 779.1617 752.8444 738.9117

86 87 88 89 90 91 92 93

785.3540 827.1520 853.4692 893.7192 943.2575 920.0364 845.7289 830.2481

94 95 96 97 98 99 100 101

844.1808 933.9691 949.4498 918.4884 1074.8439 1173.9206 1006.7286 896.8153

102 103 104 105 107 108 109 110

906.1038 831.7962 834.8923 837.9885 772.9694 807.0270 884.4307 1023.7574

111 112 113 114 115 116 117 118

1228.1032 1410.7760 1280.7378 1164.6322 981.9594 935.5172 916.9403 907.6518

119 120 121 122 123 124 125 126

892.1711 912.2961 1156.8918 1296.2185 1166.1803 1067.1035 969.5748 808.5751

127 128 129 131 132 133 134 135

867.4019 793.0943 737.3636 762.1328 729.6233 797.7386 824.0558 1077.9400

136 137 138 139 140 141 142 143

1025.3055 1283.8339 1156.8918 1006.7286 1060.9112 1077.9400 949.4498 955.6421

144 145 146 147 148 149 150 151

964.9306 955.6421 950.9979 927.7768 1161.5360 955.6421 872.0461 875.1423

152 153 155 156 157 158 159 160

817.8635 779.1617 754.3925 714.1425 742.0079 918.4884 1166.1803 1254.4205

161 162 163 164 165 166 167 168

1104.2573 1012.9209 995.8921 986.6036 920.0364 879.7865 958.7383 898.3634

169 170 171 172 173 174 175 176

1133.6707 1307.0550 1207.9783 1190.9495 983.5075 946.3537 909.1999 813.2193

177 179 180 181 182 183 184 185

765.2290 763.6809 728.0752 776.0655 885.9788 1195.5937 1322.5358 1211.0744

186 187 188 189 190 191 192 193

1017.5651 935.5172 901.4595 882.8826 901.4595 937.0653 927.7768 1002.0844

194 195 196 197 198 199 200 201

1104.2573 1098.0650 946.3537 870.4980 817.8635 865.8538 830.2481 745.1040

203 204 205 206 207 208 209 210

701.7579 698.6618 757.4886 848.8250 1166.1803 1223.4590 1062.4593 1008.2767

211 212 213 214 215 216 217 218

968.0267 943.2575 977.3152 998.9882 1056.2670 1088.7765 1028.4016 1059.3631

219 220 221 222 223 224 225 227

995.8921 1056.2670 892.1711 867.4019 831.7962 803.9308 785.3540 738.9117

228 229 230 231 232 233 234 235

735.8156 768.3251 848.8250 933.9691 898.3634 927.7768 972.6710 952.5460

236 237 238 239 240 241 242 243

930.8730 864.3058 817.8635 793.0943 855.0173 890.6230 1003.6325 1025.3055

244 245 246 247 248 249 251 252

989.6998 898.3634 859.6615 941.7095 876.6903 808.5751 802.3828 755.9405

253 254 255 256 257 258 259 260

707.9502 782.2578 859.6615 842.6327 861.2096 834.8923 830.2481 828.7000

261 262 263 264 265 266 267 268

745.1040 768.3251 822.5077 875.1423 938.6133 957.1902 1082.5842 961.8344

269 270 271 272 273 275 276 277

859.6615 844.1808 814.7674 785.3540 706.4022 697.1137 694.0176 734.2675

278 279 280 281 282 283 284 285

803.9308 1020.6613 1002.0844 848.8250 862.7577 859.6615 836.4404 876.6903

286 287 288 289 290 291 292 293

872.0461 903.0076 884.4307 1000.5363 1029.9497 1039.2382 949.4498 868.9500

294 295 296 297 299 300 301 302

771.4213 752.8444 743.5559 749.7482 701.7579 690.9214 721.8829 830.2481

303 304 305 306 307 308 309 310

1017.5651 1048.5266 1037.6901 1011.3728 991.2479 992.7959 918.4884 873.5942

311 312 313 314 315 316 317 318

868.9500 943.2575 836.4404 873.5942 927.7768 808.5751 796.1905 791.5463

319 320 321 323 324 325 326 327

779.1617 788.4501 737.3636 738.9117 723.4310 737.3636 793.0943 940.1614

328 329 330 331 332 333 334 335

1034.5939 1008.2767 881.3346 851.9212 855.0173 865.8538 825.6039 975.7671

336 337 338 339 340 341 342 343

972.6710 912.2961 1053.1708 964.9306 932.4210 853.4692 796.1905 794.6424

344 345 347 348 349 350 351 352

735.8156 729.6233 698.6618 689.3733 737.3636 800.8347 955.6421 983.5075

353 354 355 356 357 358 359 360

876.6903 872.0461 834.8923 875.1423 834.8923 864.3058 884.4307 856.5654

361 362 363 364 365 366 367 368

915.3922 1062.4593 1028.4016 903.0076 824.0558 786.9020 814.7674 793.0943

369 371 372 373 374 375 376 377

774.5174 740.4598 782.2578 830.2481 875.1423 1040.7862 1096.5169 1029.9497

378 379 380 381 382 383 384 385

949.4498 844.1808 850.3731 830.2481 808.5751 844.1808 867.4019 864.3058

386 387 388 389 390 391 392 393

856.5654 856.5654 824.0558 793.0943 772.9694 816.3154 779.1617 786.9020

395 396 397 398 399 400 401 402

759.0367 720.3348 754.3925 776.0655 822.5077 868.9500 870.4980 872.0461

403 404 405 406 407 408 409 410

842.6327 834.8923 882.8826 845.7289 859.6615 882.8826 926.2287 974.2190

411 412 413 414 415 416 417 419

940.1614 887.5269 828.7000 872.0461 858.1135 842.6327 805.4789 789.9982

420 421 422 423 424 425 426 427

731.1713 789.9982 811.6712 833.3443 876.6903 868.9500 865.8538 813.2193

428 429 430 431 432 433 434 435

772.9694 748.2002 779.1617 783.8059 789.9982 803.9308 814.7674 833.3443

436 437 438 439 440 441 443 444

774.5174 768.3251 768.3251 742.0079 703.3060 704.8541 731.1713 755.9405

445 446 447 448 449 450 451 452

788.4501 950.9979 1071.7477 853.4692 855.0173 861.2096 813.2193 811.6712

453 454 455 456 457 458 459 460

830.2481 828.7000 816.3154 822.5077 865.8538 837.9885 817.8635 791.5463

461 462 463 464 465 467 468 469

765.2290 755.9405 759.0367 734.2675 742.0079 734.2675 751.2963 842.6327

470 471 472 473 474 475 476 477

966.4787 1048.5266 1108.9015 1166.1803 1056.2670 1009.8247 980.4113 884.4307

478 479 480 481 482 483 484 485

853.4692 824.0558 844.1808 851.9212 870.4980 875.1423 783.8059 776.0655

486 487 488 489 491 492 493 494

794.6424 776.0655 738.9117 743.5559 694.0176 738.9117 802.3828 991.2479

495 496 497 498 499 500 501 502

1026.8536 950.9979 893.7192 887.5269 986.6036 901.4595 935.5172 972.6710

503 504 505 506 507 508 509 510

906.1038 868.9500 858.1135 890.6230 873.5942 796.1905 763.6809 759.0367

511 512 513 515 516 517 518 519

819.4116 802.3828 748.2002 759.0367 769.8732 907.6518 1200.2379 1255.9686

520 521 522 523 524 528 529 530

1141.4111 968.0267 872.0461 834.8923 828.7000 923.1326 997.4402 1084.1323

531 532 533 534 535 536 537 539

1087.2285 909.1999 867.4019 859.6615 865.8538 800.8347 760.5848 728.0752

540 541 542 543 544 545 546 547

757.4886 811.6712 916.9403 964.9306 1056.2670 974.2190 949.4498 929.3249

548 549 550 551 552 553 554 555

916.9403 906.1038 901.4595 892.1711 957.1902 1016.0170 1003.6325 1090.3246

556 557 558 559 560 561 563 564

949.4498 848.8250 901.4595 822.5077 783.8059 777.6136 763.6809 757.4886

565 566 567 568 617 618 619 620

783.8059 916.9403 985.0556 997.4402 1036.1420 1023.7574 989.6998 961.8344

621 622 623 624 625 626 627 628

915.3922 862.7577 822.5077 918.4884 971.1229 940.1614 913.8441 839.5366

629 630 631 632 633 635 636 637

802.3828 810.1231 765.2290 726.5271 709.4983 707.9502 783.8059 924.6807

638 639 640 641 642 643 644 645

1180.1129 1138.3149 1067.1035 952.5460 833.3443 881.3346 929.3249 927.7768

646 647 648 649 650 651 652 653

848.8250 839.5366 892.1711 916.9403 950.9979 946.3537 811.6712 788.4501

654 655 656 657 659 660 661 662

772.9694 757.4886 724.9791 703.3060 689.3733 704.8541 755.9405 906.1038

663 664 665 666 667 668 669 670

907.6518 906.1038 844.1808 855.0173 859.6615 848.8250 800.8347 896.8153

671 672 673 674 675 676 677 678

771.4213 856.5654 957.1902 913.8441 940.1614 766.7771 796.1905 853.4692

679 680 681 683 684 685 686 687

817.8635 800.8347 757.4886 737.3636 777.6136 785.3540 920.0364 1147.6034

688 689 690 691 692 693 694 695

1135.2188 963.3825 924.6807 859.6615 927.7768 950.9979 867.4019 834.8923

696 697 698 699 700 701 726 727

968.0267 1043.8824 1175.4687 1029.9497 813.2193 817.8635 913.8441 799.2866

728 729 731 732 733 734 735 736

853.4692 827.1520 742.0079 796.1905 844.1808 903.0076 884.4307 862.7577

737 738 739 740 741 742 743 744

862.7577 848.8250 827.1520 810.1231 810.1231 808.5751 813.2193 830.2481

745 746 747 748 749 750 751 752

859.6615 916.9403 937.0653 864.3058 960.2864 940.1614 786.9020 745.1040

753 755 756 757 758 759 760 761

765.2290 729.6233 729.6233 734.2675 749.7482 765.2290 743.5559 805.4789

762 763 764 765 766 767 768 769

836.4404 920.0364 765.2290 711.0464 735.8156 755.9405 771.4213 786.9020

770 771 772 773 774 775 776 777

814.7674 768.3251 763.6809 768.3251 772.9694 734.2675 735.8156 726.5271

779 780 781 782 783 784 785 786

695.5656 695.5656 721.8829 731.1713 740.4598 768.3251 780.7097 831.7962

787 788 789 790 791 792 793 794

802.3828 752.8444 760.5848 808.5751 819.4116 807.0270 825.6039 867.4019

795 796 797 798 799 800 801 803

807.0270 772.9694 765.2290 754.3925 742.0079 709.4983 698.6618 703.3060

804 805 806 807 808 809 810 811

788.4501 901.4595 1178.5649 1107.3534 989.6998 876.6903 882.8826 850.3731

812 813 814 815 816 817 818 819

844.1808 868.9500 895.2672 841.0846 955.6421 1046.9785 1071.7477 988.1517

820 821 822 823 824 825 827 828

893.7192 906.1038 837.9885 769.8732 783.8059 743.5559 740.4598 793.0943

829 830 831 832 833 834 835 856

845.7289 1077.9400 1096.5169 1077.9400 954.0941 898.3634 875.1423 1149.1514

857 858 859 860 861 862 863 864

1177.0168 1133.6707 1068.6516 972.6710 1028.4016 933.9691 968.0267 1059.3631

865 866 867 868 869 870 871 872

1118.1900 1214.1706 1164.6322 1079.4881 893.7192 855.0173 856.5654 819.4116

873 875 876 877 878 879 880 881

789.9982 768.3251 820.9597 949.4498 1215.7187 1102.7092 958.7383 932.4210

882 883 884 885 886 887 888 889

937.0653 867.4019 1026.8536 1104.2573 1045.4305 1057.8151 1149.1514 1088.7765

890 891 892 893 894 895 896 897

1170.8245 1146.0553 1017.5651 927.7768 935.5172 932.4210 856.5654 830.2481

1044 1045 1046 1047 1048 1049 1050 1051

762.1328 782.2578 1048.5266 1214.1706 1225.0071 1050.0747 1042.3343 947.9018

1052 1053 1054 1055 1056 1057 1058 1059

918.4884 940.1614 1073.2958 1105.8054 1110.4496 1057.8151 1105.8054 1026.8536

1060 1061 1062 1063 1064 1065 1067 1068

822.5077 858.1135 841.0846 920.0364 899.9115 839.5366 740.4598 780.7097

1069 1070 1071 1072 1073 1074 1075 1076

786.9020 834.8923 927.7768 961.8344 920.0364 885.9788 878.2384 800.8347

1077 1078 1079 1080 1081 1082 1083 1084

822.5077 811.6712 824.0558 810.1231 898.3634 944.8056 921.5845 807.0270

1085 1086 1087 1088 1089 1091 1092 1093

752.8444 776.0655 765.2290 780.7097 768.3251 706.4022 692.4695 723.4310

1094 1095 1096 1097 1098 1099 1100 1101

735.8156 732.7194 745.1040 765.2290 760.5848 759.0367 754.3925 734.2675

1102 1103 1104 1105 1106 1107 1108 1109

759.0367 799.2866 796.1905 822.5077 862.7577 876.6903 825.6039 850.3731

1110 1111 1112 1113 1115 1116 1117 1118

828.7000 755.9405 721.8829 697.1137 686.2772 724.9791 779.1617 850.3731

1119 1120 1121 1122 1123 1124 1125 1126

878.2384 845.7289 841.0846 822.5077 819.4116 817.8635 803.9308 803.9308

1127 1128 1129 1130 1131 1132 1133 1134

811.6712 855.0173 898.3634 879.7865 862.7577 808.5751 793.0943 768.3251

1135 1136 1137 1139 1140 1141 1142 1143

731.1713 709.4983 700.2099 723.4310 760.5848 844.1808 1156.8918 1042.3343

1144 1145 1146 1147 1148 1149 1150 1151

868.9500 889.0749 853.4692 875.1423 903.0076 858.1135 901.4595 856.5654

1152 1153 1154 1155 1156 1157 1158 1159

910.7480 937.0653 944.8056 950.9979 879.7865 824.0558 842.6327 782.2578

1160 1161 1163 1164 1165 1166 1167 1168

754.3925 734.2675 707.9502 715.6906 788.4501 1167.7283 1094.9688 1169.2764

1169 1170 1171 1172 1173 1174 1175 1176

1065.5554 974.2190 881.3346 855.0173 932.4210 975.7671 887.5269 946.3537

1177 1178 1179 1180 1181 1182 1183 1184

1016.0170 1014.4690 1081.0362 904.5557 793.0943 805.4789 807.0270 749.7482

1185 1187 1188 1189 1190 1191 1192 1193

723.4310 718.7868 788.4501 885.9788 1124.3823 1240.4878 1057.8151 895.2672

1194 1195 1196 1197 1198 1199 1200 1201

841.0846 831.7962 988.1517 981.9594 1082.5842 1051.6228 1053.1708 1090.3246

1202 1203 1204 1205 1206 1207 1208 1209

1184.7572 1050.0747 949.4498 912.2961 937.0653 882.8826 811.6712 794.6424

1211 1212 1213 1214 1215 1216 1217 1218

742.0079 776.0655 921.5845 1000.5363 920.0364 994.3440 966.4787 1003.6325

1219 1220 1221 1222 1223 1224 1225 1226

998.9882 983.5075 907.6518 957.1902 995.8921 1003.6325 1003.6325 1105.8054

1227 1228 1229 1230 1231 1232 1233 1235

1064.0074 963.3825 940.1614 929.3249 957.1902 870.4980 802.3828 724.9791

1236 1237 1238 1239 1240 1241 1242 1243

707.9502 717.2387 786.9020 793.0943 831.7962 853.4692 833.3443 827.1520

1244 1245 1246 1247 1248 1249 1250 1251

771.4213 762.1328 794.6424 822.5077 808.5751 847.2769 972.6710 912.2961

1252 1253 1254 1255 1256 1257 1259 1260

839.5366 955.6421 964.9306 833.3443 788.4501 729.6233 711.0464 721.8829

1261 1262 1263 1264 1265 1266 1267 1268

759.0367 745.1040 805.4789 816.3154 808.5751 791.5463 782.2578 769.8732

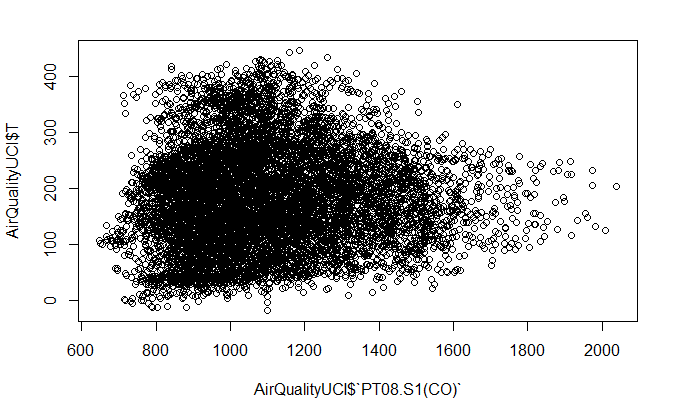
1269 1270 1271 1272 1273 1274 1275 1276

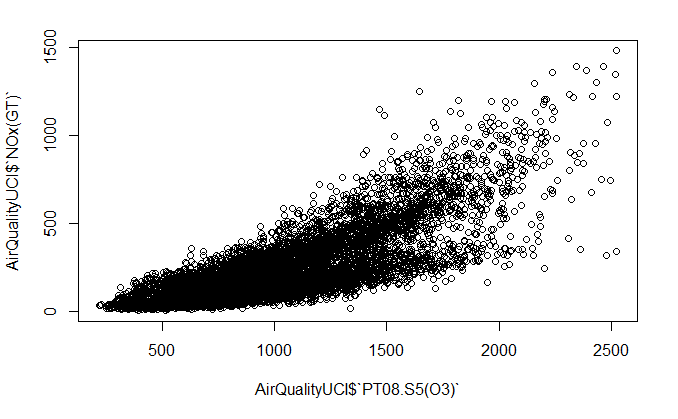
779.1617 813.2193 777.6136 873.5942 893.7192 994.3440 933.9691 906.1038

1277 1278 1279 1280 1281 1283 1284 1285

935.5172 907.6518 820.9597 728.0752 711.0464 709.4983 765.2290 906.1038

[ reached getOption("max.print") -- omitted 6396 entries ]





*7. Create cross tabulations with derived variables.*

mydata<-AirQualityUCI

View(mydata)

# 2-Way Frequency Table

attach(mydata)

mytable <- table(A,B) # A will be rows, B will be columns

mytable # print table

margin.table(mytable, 1) # A frequencies (summed over B)

margin.table(mytable, 2) # B frequencies (summed over A)

prop.table(mytable) # cell percentages

prop.table(mytable, 1) # row percentages

prop.table(mytable, 2) # column percentages

Chi-squared approximation may be incorrect

Pearson's Chi-squared test

data: mytable

X-squared = 2450, df = 2401, p-value = 0.2382

8. Check for trends and patterns in time series.

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

copie[326] 326 327 328 329 330 331 332 333 334 335 336 337 338

NAs introduced by coercionNAs introduced by coercionNAs introduced by coercionNAs introduced by coercionNAs introduced by coercion Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC)

1959 Q2 NA NA NA 1360 150 NA 1046

1959 Q3 NA NA 2 1292 112 NA 955

1959 Q4 NA NA NA 1402 88 NA 939

1960 Q1 NA NA NA 1376 80 NA 948

1960 Q2 NA NA NA 1272 51 NA 836

1960 Q3 NA NA NA 1197 38 NA 750

1960 Q4 NA NA NA 1185 31 NA 690

1961 Q1 NA NA 1 1136 31 NA 672

1961 Q2 NA NA NA 1094 24 NA 609

1961 Q3 NA NA NA 1010 19 NA 561

1961 Q4 NA NA NA 1011 14 NA 527

1962 Q1 NA NA NA 1066 8 NA 512

1962 Q2 NA NA NA 1052 16 NA 553

1962 Q3 NA NA NA 1144 29 NA 667

1962 Q4 NA NA 2 1333 64 NA 900

1963 Q1 NA NA NA 1351 87 NA 960

1963 Q2 NA NA NA 1233 77 NA 827

1963 Q3 NA NA NA 1179 43 NA 762

1963 Q4 NA NA NA 1236 61 NA 774

1964 Q1 NA NA NA 1286 63 NA 869

1964 Q2 NA NA NA 1371 164 NA 1034

1964 Q3 NA NA NA 1310 79 NA 933

1964 Q4 NA NA NA 1292 95 NA 912

1965 Q1 NA NA NA 1383 150 NA 1020

1965 Q2 NA NA NA 1581 307 NA 1319

1965 Q3 NA NA NA 1776 461 NA 1488

1965 Q4 NA NA NA 1640 401 NA 1404

1966 Q1 NA NA NA 1313 197 NA 1076

1966 Q2 NA NA NA 965 61 NA 749

1966 Q3 NA NA 1 913 26 NA 629

1966 Q4 NA NA NA 1080 55 NA 805

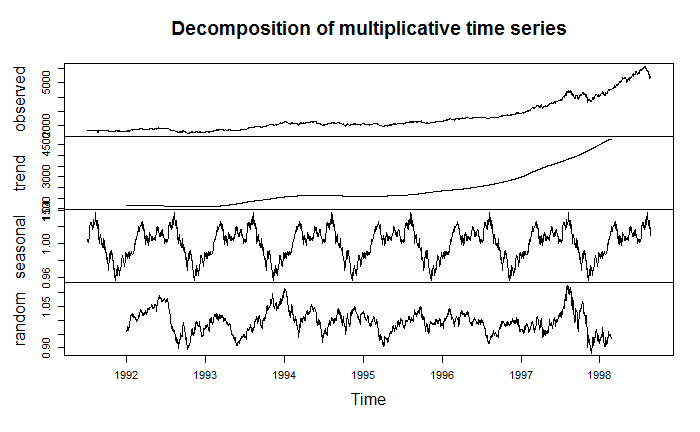
1967 Q1 NA

#plot time series

tsAirqualityUCI <- EuStockMarkets[, 1] # ts data

decomposedRes <- decompose(tsAirqualityUCI, type="mult") # use type = "additive" for additive components

plot (decomposedRes) # see plot below



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

#plot time series

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")

118 119 120 121 122 123 124 125 126 127 128 129 130

[131

PT08.S4(NO2) is the highest pollution at 18.00 hrs

PTO\*s4

132 133 134 135 136 137 138 139 140 141 142 143

[144] 144 145 146 147 148 149 150 151 152 153 154 155 156

[157] 157 158 159 160 161 162 163 164 165 166 167 168 169

[1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 |

70] 170 171 172 173 174 175 176 177 178 179 180 181 182

[183] 183 184 185 186 187 188 189 190 191 192 193 194 195

[196] 196 197 198 199 200 201 202 203 204 205 206 207 208

[209] 209 210 211 212 213 214 215 216 217 218 219 220 221

[222] 222 223 224 225 226 227 228 229 230 231 232 233 234

[235] 235 236 237 238 239 240 241 242 243 244 245 246 247

[248] 248 249 250 251 252 253 254 255 256 257 258 259 260

[261] 261 262 263 264 265 266 267 268 269 270 271 272 273

[274] 274 275 276 277 278 279 280 281 282 283 284 285 286

[287] 287 288 289 290 291 292 293 294 295 296 297 298 299

[300] 300 301 302 303 304 305 306 307 308 309 310 311 312

[313] 313 314 315 316 317 318 319 320 321 322 323 324 325

[326] 326 327 328 329 330 331 332 333 334 335 336 337 338

NAs introduced by coercionNAs introduced by coercionNAs introduced by coercionNAs introduced by coercionNAs introduced by coercion Date Time CO(GT) PT08.S1(CO) NMHC(GT) C6H6(GT) PT08.S2(NMHC)

1959 Q2 NA NA NA 1360 150 NA 1046

1959 Q3 NA NA 2 1292 112 NA 955

1959 Q4 NA NA NA 1402 88 NA 939

1960 Q1 NA NA NA 1376 80 NA 948

1960 Q2 NA NA NA 1272 51 NA 836

1960 Q3 NA NA NA 1197 38 NA 750

1960 Q4 NA NA NA 1185 31 NA 690

1961 Q1 NA NA 1 1136 31 NA 672

1961 Q2 NA NA NA 1094 24 NA 609

1961 Q3 NA NA NA 1010 19 NA 561

1961 Q4 NA NA NA 1011 14 NA 527

1962 Q1 NA NA NA 1066 8 NA 512

1962 Q2 NA NA NA 1052 16 NA 553

1962 Q3 NA NA NA 1144 29 NA 667

1962 Q4 NA NA 2 1333 64 NA 900

1963 Q1 NA NA NA 1351 87 NA 960

1963 Q2 NA NA NA 1233 77 NA 827

1963 Q3 NA NA NA 1179 43 NA 762

1963 Q4 NA NA NA 1236 61 NA 774

1964 Q1 NA NA NA 1286 63 NA 869

1964 Q2 NA NA NA 1371 164 NA 1034

1964 Q3 NA NA NA 1310 79 NA 933

1964 Q4 NA NA NA 1292 95 NA 912

1965 Q1 NA NA NA 1383 150 NA 1020

1965 Q2 NA NA NA 1581 307 NA 1319

1965 Q3 NA NA NA 1776 461 NA 1488

1965 Q4 NA NA NA 1640 401 NA 1404

1966 Q1 NA NA NA 1313 197 NA 1076

1966 Q2 NA NA NA 965 61 NA 749

1966 Q3 NA NA 1 913 26 NA 629

1966 Q4 NA NA NA 1080 55 NA 805

1967 Q1 NA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 |

[989] 989 990 991 992 993 994 995 996 997 998 999 1000