Rahul kumar mahato (Msc data science) CUSB2302222005

project on water pollution of New-york river's data

1976 study exploring the relationship between water quality and land use, Haith (1976) obtained the measurements (shown in Table 1.8) on 20 river basins in New York State. A question of interest here is how the land use around a river basin contributes to the water pollution as measured by the mean nitrogen concentration (mglliter). The data are shown in Table 1.9 and can also be found at the book website & also on my github : https://github.com/rahulkr43/Water-Pollution-in-Newyork-rivers-Data-analysis/tree/main

Analysis :

Import the data excel to R –

```
> library(readxl)
> river_data <- read_excel("river data.xlsx")
> View(river_data)
```

```
> river_data
# A tibble: 20 × 6
   river            y    x1    x2    x3    x4
   <chr>        <dbl> <dbl> <dbl> <dbl> <dbl>
 1 olean          1.1    26    63   1.2 0.29
 2 cassadage      1.01   29    57   0.7 0.009
 3 otka           1.9    54    26   1.8 0.58
 4 neversink      1       2    84   1.9 1.98
 5 hackensack     1.99    3    27  29.4 3.11
 6 wapping        1.42   19    61   3.4 0.56
 7 fishkill       2.04   16    60   5.6 1.11
 8 honeoye        1.65   40    43   1.3 0.24
 9 susquehanna    1.01   28    62   1.1 0.15
10 cherango       1.21   26    60   0.9 0.23
11 tioughnioga    1.33   26    53   0.9 0.18
12 west canada    0.75   15    75   0.7 0.16
13 east canada    0.73    6    84   0.5 0.12
14 saranac        0.8     3    81   0.8 0.35
15 ausable        0.76    2    89   0.7 0.35
16 black          0.87    6    82   0.5 0.15
17 schoharie      0.8    22    70   0.9 0.22
18 raquette       0.87    4    75   0.4 0.18
19 oswegatchic    0.66   21    56   0.5 0.13
20 cohocton       1.25   40    49   1.1 0.13
```

Variables in Study of Water Pollution in New York Rivers.

Variable	Defination
Y	Mean nitrogen concentration (mg/liter) based on samples taken at regular intervals during the spring, summer, and fall months
X_1	Agriculture: percentage of land area currently in agricultural use
X_2	Forest: percentage of forest land
X_3	Residential: percentage of land area in residential use

X_4 use	Commercial/Industrial: percentage of land area in either commercial or industrial

Missing Data Visualization.

```
> sapply(river_data, function(x) sum(is.na(x)))
river     y    x1    x2    x3    x4
    0     0     0     0     0     0
```

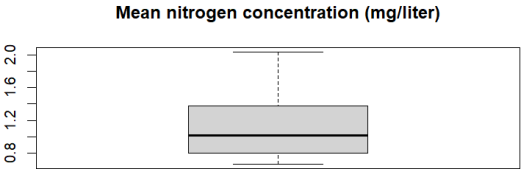
We see that the dataset contains 0 missing records.

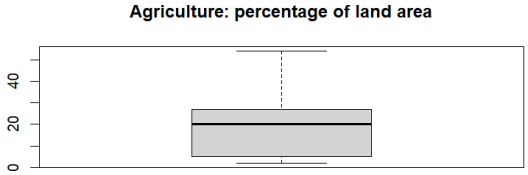
if we have missing values in data the we have to remove these missing values

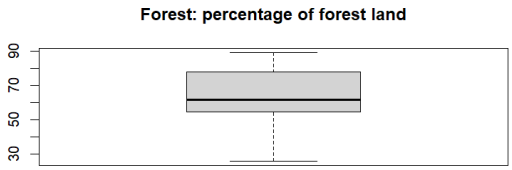
```
> data <- na.omit(river_data)
```

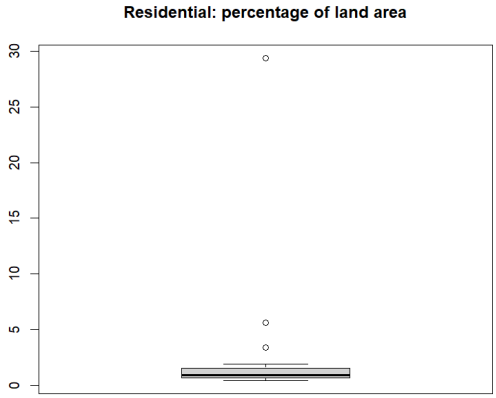
To find the extreme values, we will represent them graphically using the boxplot function.

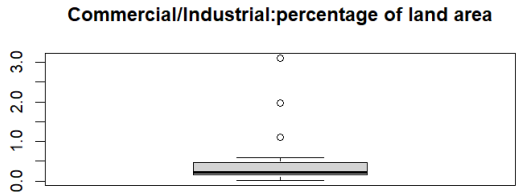
```
> par(mfrow = c(2, 2))
> Y_outliers <- boxplot(data$y , main = 'Mean nitrogen concentration (mg/liter)' )
> x1_outliers <- boxplot(data$x1 , main = 'Agriculture: percentage of land area' )
> x2_outliers <- boxplot(data$x2, main = 'Forest: percentage of forest land' )
> x3_outliers <- boxplot(data$x3, main = 'Residential: percentage of land area' )
> x4_outliers <- boxplot(data$x4, main = 'Commercial/Industrial:percentage of land
area' )
```







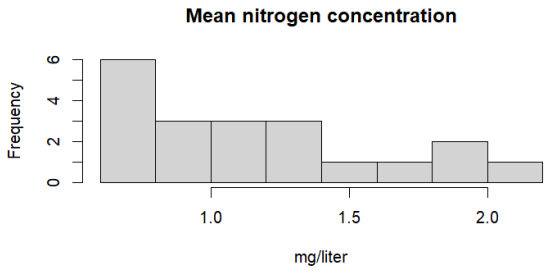


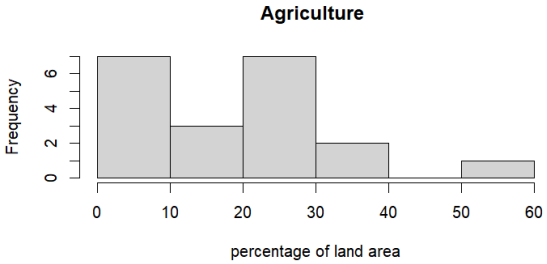


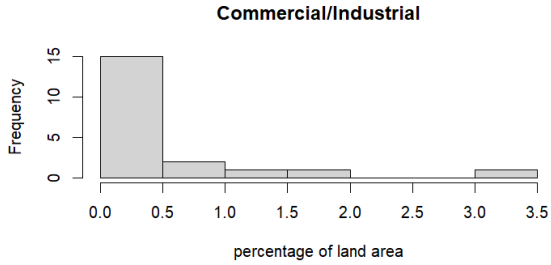
We found several outliers in the dataset. However, since my knowledge in these parameters is null, I cannot say whether these values are correct, erroneous or incompatible with water pollution. In this way, I'm going to choose not to remove the extreme values.

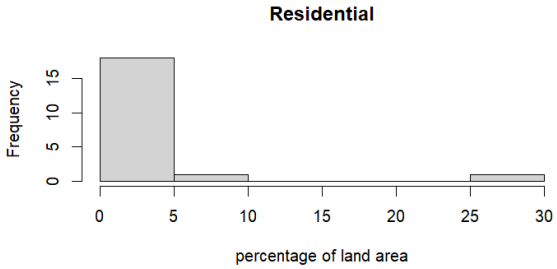
Checking the normality and homogeneity of the variance We will check these conditions for the following numerical variables: • Mean nitrogen concentration (mg/liter) • Agriculture: percentage of land area • Forest: percentage of forest land • Residential: percentage of land area .

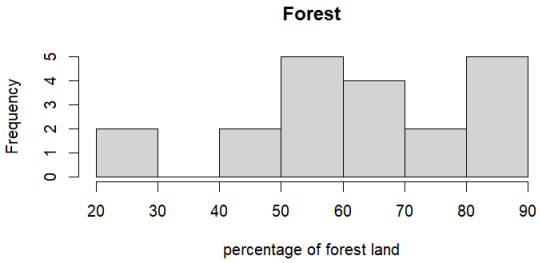
```
> Y_outliers <- hist(data$y ,main ='Mean nitrogen concentration',xlab ='mg/liter')
> x1_outliers <-hist(data$x1 ,main ='Agriculture',xlab ='percentage of land area')
> x2_outliers <- hist(data$x2, main = 'Forest',xlab ='percentage of forest land' )
> x3_outliers <-hist(data$x3,main ='Residential', xlab= 'percentage of land area')
> x4_outliers <- hist(data$x4, main = 'Commercial/Industrial',xlab= 'percentage of
land area' )
```











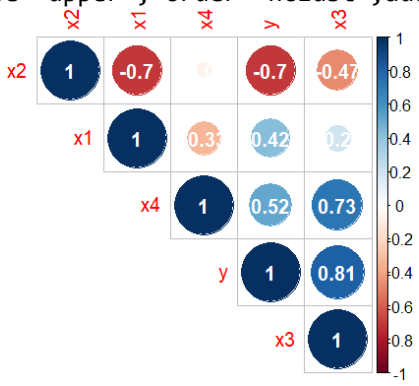
Checking the normality of our Dependent variable

```
> shapiro.test(river_data$y)

        Shapiro-Wilk normality test

data:  river_data$y
W = 0.87505, p-value = 0.01443
```

We observed that Dependent variable Y presents normality, since its p-values are lower than the significance value of 0.05 To check homoscedasticity.

Correlation We will perform a correlation analysis between the different variables to determine which of them exert a greater influence on the Mean nitrogen concentration (mg/liter. Since the data do not follow a normal distribution, we will look at Spearman's correlation coefficient.

```
> numeric_data <- data[, c('y', 'x1', 'x2', 'x3', 'x4')]
> res <- cor(numeric_data, method = 'spearman')
> round(res, 2)
       y    x1    x2    x3    x4
y   1.00  0.42 -0.70  0.81  0.52
x1  0.42  1.00 -0.70  0.20 -0.33
x2 -0.70 -0.70  1.00 -0.47 -0.07
x3  0.81  0.20 -0.47  1.00  0.73
x4  0.52 -0.33 -0.07  0.73  1.00
> library(corrplot)

## corrplot 0.92 loaded

> corrplot(res, type="upper", order="hclust",addCoef.col = "white")
```



Here we can see that how Mean nitrogen concentration (mg/liter) is correlated to other independent variables.

Now, we perform Multiple liner regression-

Multiple Linear Regression Formula

$$Y_i=\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+.....+\beta_p x_{ip}+\varepsilon$$

Where:

- **yi** is the dependent or predicted variable
- **β0** is the y-intercept, i.e., the value of y when both xi and x2 are 0.
- **β1** and **β2** are the regression coefficients representing the change in y relative to a one-unit change in **xi1** and **xi2**, respectively.
- **βp** is the slope coefficient for each independent variable
- **ϵ** is the model's random error (residual) term.

Understanding Multiple Linear Regression

Simple linear regression enables statisticians to predict the value of one variable using the available information about another variable. Linear regression attempts to establish the relationship between the two variables along a straight line.

Multiple regression is a type of regression where the dependent variable shows a **linear** relationship with two or more independent variables. It can also be **non-linear**, where the dependent and independent variables do not follow a straight line.

Both linear and non-linear regression track a particular response using two or more variables graphically. However, non-linear regression is usually difficult to execute since it is created from assumptions derived from trial and error.

Assumptions of Multiple Linear Regression

Multiple linear regression is based on the following assumptions:

1. A linear relationship between the dependent and independent variables

The first assumption of multiple linear regression is that there is a linear relationship between the dependent variable and each of the independent variables. The best way to check the linear relationships is to create scatterplots and then visually inspect the scatterplots for linearity. If the relationship displayed in the scatterplot is not linear, then the analyst will need to run a non-linear regression or transform the data using statistical software, such as SPSS.

2. The independent variables are not highly correlated with each other

The data should not show multicollinearity, which occurs when the independent variables (explanatory variables) are highly correlated. When independent variables show multicollinearity, there will be problems figuring out the specific variable that contributes to the variance in the dependent variable. The best method to test for the assumption is the Variance Inflation Factor method.

3. The variance of the residuals is constant

Multiple linear regression assumes that the amount of error in the residuals is similar at each point of the linear model. This scenario is known as homoscedasticity. When analyzing the data, the analyst should plot the standardized residuals against the predicted values to determine if the points are distributed fairly across all the values of independent variables. To test the assumption, the data can be plotted on a scatterplot or by using statistical software to produce a scatterplot that includes the entire model.

4. Independence of observation

The model assumes that the observations should be independent of one another. Simply put, the model assumes that the values of residuals are independent. To test for this assumption, we use the Durbin Watson statistic.

The test will show values from 0 to 4, where a value of 0 to 2 shows positive autocorrelation, and values from 2 to 4 show negative autocorrelation. The mid-point, i.e., a value of 2, shows that there is no autocorrelation.

5. Multivariate normality

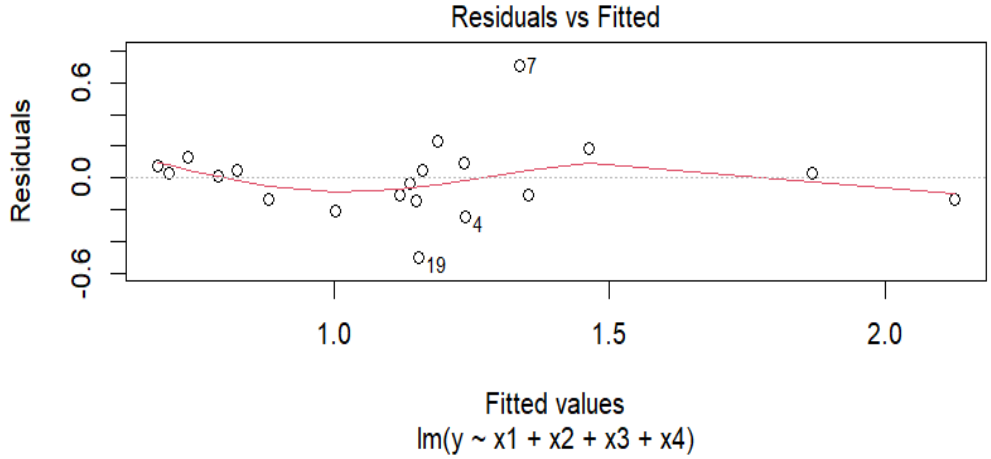
Multivariate normality occurs when residuals are normally distributed. To test this assumption, look at how the values of residuals are distributed. It can also be tested using two main methods, i.e., a histogram with a superimposed normal curve or the Normal Probability Plot method.

```
> mdl <- lm(river_data$y~x1+x2+x3,data = river_data)
> summary(mdl)

Call:
lm(formula = river_data$y ~ x1 + x2 + x3, data = river_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.53216 -0.16155 -0.01698  0.09363  0.79347

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.750332   1.325716   1.320    0.205
x1           0.004603   0.016136   0.285    0.779
x2          -0.011913   0.014955  -0.797    0.437
x3           0.024518   0.031392   0.781    0.446

Residual standard error: 0.2846 on 16 degrees of freedom
Multiple R-squared:  0.6422,     Adjusted R-squared:  0.5751
F-statistic: 9.574 on 3 and 16 DF,  p-value: 0.0007398
> x <- model.matrix(~x1+x2+x3+x4,river_data)  #x=independent variable
> x
   (Intercept) x1 x2   x3    x4
1            1 26 63  1.2 0.290
2            1 29 57  0.7 0.009
3            1 54 26  1.8 0.580
4            1  2 84  1.9 1.980
5            1  3 27 29.4 3.110
6            1 19 61  3.4 0.560
7            1 16 60  5.6 1.110
8            1 40 43  1.3 0.240
9            1 28 62  1.1 0.150
10           1 26 60  0.9 0.230
11           1 26 53  0.9 0.180
12           1 15 75  0.7 0.160
13           1  6 84  0.5 0.120
14           1  3 81  0.8 0.350
15           1  2 89  0.7 0.350
16           1  6 82  0.5 0.150
17           1 22 70  0.9 0.220
18           1  4 75  0.4 0.180
19           1 21 56  0.5 0.130
20           1 40 49  1.1 0.130
attr(,"assign")
[1] 0 1 2 3 4

> y <-river_data$y   # dependent variable
> y
 [1] 1.10 1.01 1.90 1.00 1.99 1.42 2.04 1.65 1.01 1.21 1.33 0.75 0.73 0.80 0.76 0.
87 0.80 0.87 0.66 1.25
```

```
> xtxi <- solve(t(x)%*%x)
> xtxi
            (Intercept)           x1           x2           x3           x4
(Intercept) 21.70017483 -0.257816270 -0.243656717 -0.484972272 -0.035222973
x1          -0.25781627  0.003221030  0.002851830  0.005631034  0.001569166
x2          -0.24365672  0.002851830  0.002765358  0.005595846 -0.001298784
x3          -0.48497227  0.005631034  0.005595846  0.016263067 -0.039344814
x4          -0.03522297  0.001569166 -0.001298784 -0.039344814  0.377809905
> solve(crossprod(x,x),crossprod(x,y))
                  [,1]
(Intercept)  1.721629102
x1           0.005881935
x2          -0.012970870
x3          -0.007543526
x4           0.307872111
> names(mdl)
 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.value
s" "assign"
 [7] "qr"            "df.residual"   "xlevels"       "call"          "terms"
"model"
> mdls<- summary(mdl)
> mdls

Call:
lm(formula = river_data$y ~ x1 + x2 + x3, data = river_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.53216 -0.16155 -0.01698  0.09363  0.79347

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.750332   1.325716   1.320    0.205
x1           0.004603   0.016136   0.285    0.779
x2          -0.011913   0.014955  -0.797    0.437
x3           0.024518   0.031392   0.781    0.446

Residual standard error: 0.2846 on 16 degrees of freedom
Multiple R-squared:  0.6422,     Adjusted R-squared:  0.5751
F-statistic: 9.574 on 3 and 16 DF,  p-value: 0.0007398

> names(mdl)
 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.value
s" "assign"
 [7] "qr"            "df.residual"   "xlevels"       "call"          "terms"
"model"
> mdls<- summary(mdl)
> mdls

Call:
lm(formula = river_data$y ~ x1 + x2 + x3, data = river_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.53216 -0.16155 -0.01698  0.09363  0.79347

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.750332   1.325716   1.320    0.205
```

```
x1           0.004603   0.016136   0.285    0.779
x2          -0.011913   0.014955  -0.797    0.437
x3           0.024518   0.031392   0.781    0.446

Residual standard error: 0.2846 on 16 degrees of freedom
Multiple R-squared:  0.6422,     Adjusted R-squared:  0.5751
F-statistic: 9.574 on 3 and 16 DF,  p-value: 0.0007398

> names(mdls)
 [1] "call"          "terms"         "residuals"     "coefficients"  "aliased"
"sigma"
 [7] "df"            "r.squared"     "adj.r.squared" "fstatistic"    "cov.unscaled
"
> sqrt(deviance(mdl)/df.residual(mdl))
[1] 0.284611
> mdls$sigma
[1] 0.284611
> xtxi<-mdls$cov.unscaled
> sqrt(diag(xtxi))*60975
(Intercept)          x1          x2          x3
 284021.072    3457.079    3203.883    6725.443
> mdls$coef[,2]
(Intercept)          x1          x2          x3
 1.32571565  0.01613649  0.01495466  0.03139213
> 1-deviance(mdl)/sum((y-mean(y))^2)
[1] 0.6422284
> mdls$r.squared
[1] 0.6422284
> new_mdl <-lm(y~x1+x2+x3+x4,data = river_data)
> new_mdl

Call:
lm(formula = y ~ x1 + x2 + x3 + x4, data = river_data)

Coefficients:
(Intercept)           x1           x2           x3           x4
   1.721629     0.005882    -0.012971    -0.007544     0.307872

> summary(new_mdl)
Call:
lm(formula = y ~ x1 + x2 + x3 + x4, data = river_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.49503 -0.13196  0.01955  0.08247  0.70302

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.721629   1.229646   1.400   0.1818
x1           0.005882   0.014981   0.393   0.7001
x2          -0.012971   0.013881  -0.934   0.3649
x3          -0.007544   0.033663  -0.224   0.8257
x4           0.307872   0.162250   1.898   0.0772 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.264 on 15 degrees of freedom
Multiple R-squared:  0.7115,     Adjusted R-squared:  0.6345
F-statistic: 9.248 on 4 and 15 DF,  p-value: 0.0005664
```

```
> plot(new_mdl)
```



Data interpretation:

1.Standard Error of the Regression (Root Mean Squared Error):

➢sqrt(deviance(model)/df.residual(mod1)) and models$sigma both give the same result 0.2709039.

➢This value represents the root mean squared error of the model. It is a measure of the average deviation of the observed values from the fitted values.

2.Standard Errors of Coefficients:

➢trans(tr)=models$cov.unscaled calculates the unscaled covariance matrix.

➢sqrt(diag(trans)*0.2709039) computes the standard errors for each coefficient.

➢The standard errors for the coefficients (x1, x2, x3, x4) and the intercept are displayed:

Intercept: 2.42459329

x1: 0.02953714

x2: 0.02737110

x3: 0.06646571

x4: 0.32184969

3.Coefficients and P-values:

➢models$coefficients[,2] provides the estimated coefficients along with their p-values.

➢For each variable (Intercept, x1, x2, x3, x4), you have the estimated coefficient and its associated p-value:

Intercept: 1.26196270 (p-value < 0.05, likely significant)

x1: 0.01537362 (p-value > 0.05, may not be significant)

x2: 0.01424622 (p-value > 0.05, may not be significant)

x3: 0.03459436 (p-value < 0.05, likely significant)

x4: 0.16751770 (p-value < 0.05, likely significant)

4.R-squared Values:

➢1-deviance(mod1)/sum((dependent_var-mean(dependent_var))^2)and

models$r.squared both give the same result 0.6955827.

➢This value represents the proportion of the variance in the dependent variable that is explained by the independent variables in the model. Approximately 70% of the variability in y is explained by the model.

Conclusion:

p-values for the intercept and some of the coefficients. The R-squared value of approximately 70% suggests that the model explains a substantial amount of the variability in the dependent variable. However In conclusion, the model appears to be statistically significant, as indicated by the significant, further analysis and consideration of the specific context of the data are needed for a more comprehensive interpretation.

The Federal Water Pollution Control Act Amendments of 1972 require that land-use management and control of nonpoint pollution sources be included in areawide water quality planning. At the present time, few tested procedures are available to implement this policy. Statistical techniques, including correlation and regression analyses, offer a promising methodology for the study of land use and nonpoint source impacts on water quality. The methodology has been applied to water quality and land-use data from 20 river basins in New York State. The results indicated that river basin land uses could account for up to 89% of the observed variation in mean river nitrogen concentrations and 63% of the observed variation in mean total suspended solids concentrations. No relationships between phosphorus concentrations and land uses were found in the basins.