Department of Computer Science
American International University-Bangladesh
Mid Term Report

Course Name: INTRODUCTION TO DATA SCIENCE

"A report on Data Pre-Processing"

Supervised By:

Dr. Akinul Islam Jony

Associate Professor, Computer Science-AIUB

Submitted By:

Rasel Mahmud

ID: 20-43867-2

Section: B

.

Submission Date: February 27, 2023

Project Title: Applying Data Pre-processing on a Dataset.

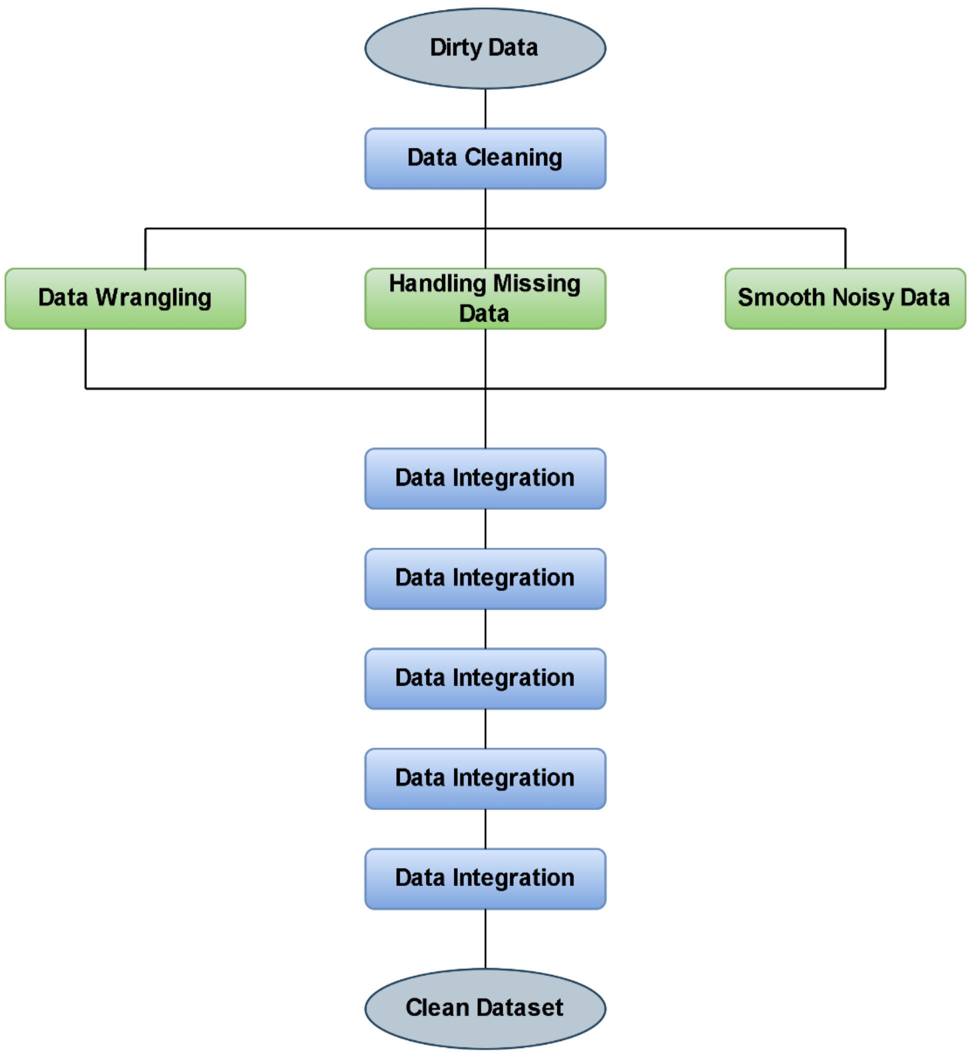
Project overview:

Raw, real-world data such as text, I mages, and videos are messy. Not only can they contain errors and inconsistencies, they are often incomplete and lack a regular and consistent design. Machines like to process information in good order, reading data as 1's and 0's. As such, calculating structured data such as integers and percentages is straightforward. However, unstructured data in the form of text and images must first be cleaned and formatted before analysis. Data preprocessing refers to the steps that transform or encode data so that it can be easily interpreted by a computer. In order for the model to make accurate and accurate predictions, the algorithm must be able to quickly interpret the attributes of the data. Due to their diverse origins, most real-world datasets are particularly vulnerable to missing, inconsistent, and noisy data. Applying data mining algorithms to this noisy data yields poor results because they cannot recognize patterns. Therefore, data preprocessing is important for improving overall data quality. Data preprocessing has four main phases: data cleansing, data integration, data transformation, data reduction, and data discretization. Data cleaning is a step in the data preprocessing process to fill missing values, smooth noisy data, fix discrepancies, and remove outliers. Data integration is the data preparation phase that combines data from multiple sources into one big data store. data warehouse. Data transformation is the technique of transforming high-quality data into different formats by changing the value, structure, or format of the data using techniques such as scaling, normalization, and so on. Data transformation includes data cleaning and data reduction techniques to transform data into an appropriate format. Data transformation includes data cleaning and data reduction techniques to transform data into an appropriate format. Data transformation is an important data preprocessing technique that must be performed on the data before data mining to provide easy-to-understand patterns. This is a systematic process of data preprocessing.

The dataset of the project contains statistics in arrests per 100,000 residents for assault and murder, in each of the 50 US states, in 1973. Also given is the percentage of the population living in urban areas.

Project Solution Design:

In this Project to perform the data Pre-processing the Steps we are going to follow is demonstrate in the below Diagram.



Data Frame:

Code:

```
States=c("Alabama","Alaska","Arizona","Arkansas","California","Colorado","C
onnecticut","Delaware","Florida","Georgia","Hawaii","Idaho","Ilinois","Indi
ana","Iowa","Kansas","Kentucky","Louisiana","Maine","Maryland","Massachuset
ts","Michigan","Minnesota","Mississippi","Missouri","Montana","Nebraska","N
evada","New Hampshire","New Jersey","New Mexico","New York","North Carolina
","North Dakota","Ohio","Oklahoma","Oregon","Pennsylvania","Rhode Island","
South Carolina","South Dakota","Tenessee","Texas","Utah","Vermont","Virgini
a","Washington","West Virginia","Wisconsin","Wyoming")
>   Murder=c(13.2,10,8.1,8.8,9,7.9,3.3,5.9,15.4,17.4,5.3,2.6,10.4,7.2,2.2,6
,9.7,15.4,2.1,11.3,4.4,12.1,2.7,16.1,9,6,4.3,12.2,2.1,7.4,11.4,11.1,13,0.8,
7.3,6.6,4.9,6.3,3.4,14.4,3.8,13.2,12.7,3.2,2.2,8.5,4,5.7,2.6,6.8)
>   Assault=c(236,263,294,190,276,204,110,238,335,NA,46,120,249,113,56,115,
109,249,83,300,149,255,72,259,178,109,102,252,57,159,285,254,337,45,120,151
,159,106,174,879,86,188,201,120,48,156,145,81,53,161)
>   Urban_polpulation=c(58,48,80,50,91,78,77,72,80,60,83,54,83,65,570,66,52
,66,51,67,85,74,66,44,70,53,62,81,56,89,70,6,45,44,75,68,67,72,87,48,45,59,
80,80,32,63,73,39,66,60)
>   Information=data.frame(States,Murder,Assault,Urban_polpulation)
```

Untitled1 × | InputOutput.R × | Practice1.R × | Project.R × | NewInformation × | ProjectCode.R × | Information ×

Filter

	States	Murder	Assault	Urban_polpulation
1	Alabama	13.2	236	58
2	Alaska	10.0	263	48
3	Arizona	8.1	294	80
4	Arkansas	8.8	190	50
5	California	9.0	276	91
6	Colorado	7.9	204	78
7	Connecticut	3.3	110	77
8	Delaware	5.9	238	72
9	Florida	15.4	335	80
10	Georgia	17.4	NA	60
11	Hawaii	5.3	46	83
12	Idaho	2.6	120	54
13	Ilinois	10.4	249	83
14	Indiana	7.2	113	65
15	Iowa	2.2	56	570
16	Kansas	6.0	115	66
17	Kentucky	9.7	109	52

The Software and Language utilized for this project:

The language we are going to use to conduct the project is R and the software we're going to use to process data and shape data is RStudio. R language is designed specifically for statistical computing and analysis. R has powerful tools for data visualization, which helps to analyze and interpret complex data sets more easily. It allows us to create high-quality graphs, charts, and other visual representations of data. And RStudio is a powerful and easy way to interact with R programming. RStudio has built-in data visualization tools that allow users to create high-quality graphs, charts, and other visual representations of data.

Data Pre-processing:

1. **Data Cleaning:**

   I. **Data Munging:** When data is in the unstructured format, we perform data munging. Since in this dataset all the data are in structured format and all the data are per 100,000 residents, there are no data munging steps needed in the data set as the data is in structured format.

   II. **Handling Missing Data:** To deal with missing data, at first we must need to identify the missing values in the dataset. As here Georgia data is missing as we can see. As data is normally distributed here in the Assault column so we can use mean to find the missing value of Georgia state Assault. Mean is affected by the outliers but in this Assault column we don't observe any outliers so we can easily use Mean. But is there are any outliers it that column in that time we need to use the Median. We calculate the mean of the column Assault, except for the empty data, and we add it where the empty data should be.

   When we tried the calculate the mean with the missing value in Assault column we will get

```
> mean(Information$Assault)
[1] NA
```

   Code:

   Mean of the Assult

```
>   meanAssault=mean(Information$Assault,na.rm=TRUE)
>   print(meanAssault)
[1] 182.1837
> Information[is.na(Information$Assault),"Assault"]=meanAssault
> print(Information)
```

Console | Terminal | Background Jobs

R 4.2.2 · ~/

```
           States Murder  Assault Urban Population(%)
1         Alabama   13.2 236.0000                  58
2          Alaska   10.0 263.0000                  48
3         Arizona    8.1 294.0000                  80
4        Arkansas   88.0 190.0000                  50
5      California    9.0 276.0000                  91
6        Colorado    7.9 204.0000                  78
7     Connecticut    3.3 110.0000                  77
8        Delaware    5.9 238.0000                  72
9         Florida   15.4 335.0000                  80
10        Georgia   17.4 182.1837                  60
11         Hawaii    5.3  46.0000                  83
12          Idaho    2.6 120.0000                  54
13        Ilinois   10.4 249.0000                  83
14        Indiana    7.2 113.0000                  65
15           Iowa    2.2  56.0000                 570
16         Kansas    6.0 115.0000                  66
17       Kentucky    9.7 109.0000                  52
18      Louisiana   15.4 249.0000                  66
19          Maine    2.1  83.0000                  51
```

ProjectCode.R | Information

Filter

	States	Murder	Assault	Urban Population(%)
1	Alabama	13.2	236.0000	58
2	Alaska	10.0	263.0000	48
3	Arizona	8.1	294.0000	80
4	Arkansas	88.0	190.0000	50
5	California	9.0	276.0000	91
6	Colorado	7.9	204.0000	78
7	Connecticut	3.3	110.0000	77
8	Delaware	5.9	238.0000	72
9	Florida	15.4	335.0000	80
10	Georgia	17.4	182.1837	60

Showing 1 to 11 of 50 entries, 4 total columns

Here Missing value is Placed.

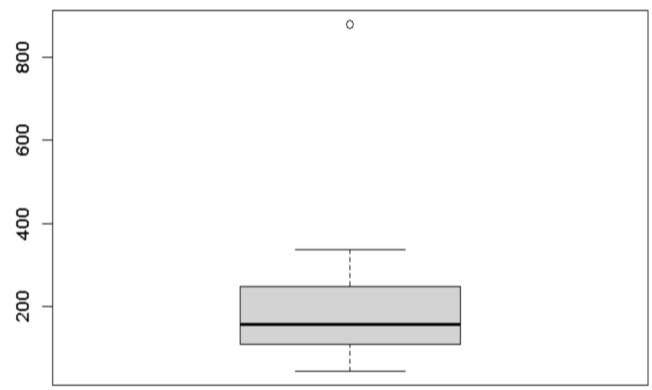
III. __**Smooth Noisy Data:**__ Many Times data is not missing but corrupted which is a big issue and big problem than missing data. When there are outliers in data or dataset contains the random or irrelevant data, we can say that the dataset contains noisy data. But here there are not any random or irrelevant data in the data set ,So we need to identify the outliers in our dataset.

For Assault:

Code:

```
> bp=boxplot(Information$Assault)
> outliers <- bp$out
>  print(outliers)
[1] 879


>  boxplot(Information$Assault)$stats
     [,1]
[1,]   45
[2,]  109
[3,]  159
[4,]  249
[5,]  337
> ResultAssult=Information[(Information$Assault >400 |  Information$Assault < 45),]
> ResultAssult
```



Here we can see a Outlier in the Assault Column from the boxplot. And using the code we found that the Outlier is 879. And we also find out that the minimum value of the Assault column and boxplot is 45 and the maximum value is 337 but we can consider the maximum value as 400.

```
> ResultAssult=Information[(Information$Assault >337 |  Information$Assault < 45),]
> ResultAssult
           States Murder Assault Urban Population(%)
40 South Carolina   14.4     879                  48
> 
```

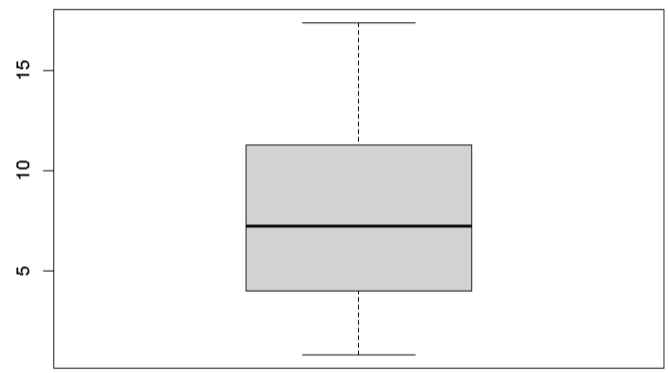
Here we can see the detailed information of the Outlier data information. And we found one Outlier in the Assault Column

For Murder:

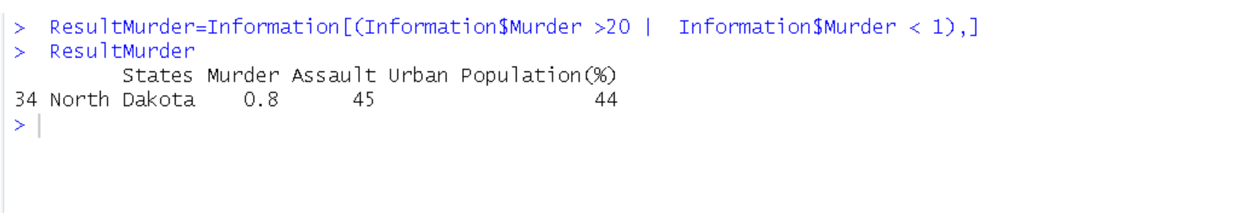
Code:

```
>  bp=boxplot(Information$Murder)
>  outliers <- bp$out
>  print(outliers)
numeric(0)
>  boxplot(Information$Murder)$stats
      [,1]
[1,]  0.80
[2,]  4.00
[3,]  7.25
[4,] 11.30
[5,] 17.40
>  ResultMurder=Information[(Information$Murder >18 |  Information$Murder < 1),]
>  ResultMurder
```

```
Console  Terminal  Background Jobs
R 4.2.2 · ~/
>  bp=boxplot(Information$Murder)
>  outliers <- bp$out
>  print(outliers)
numeric(0)
>  boxplot(Information$Murder)$stats
     [,1]
[1,]  0.80
[2,]  4.00
[3,]  7.25
[4,] 11.30
[5,] 17.40
>
```



Here we can see a Outlier in the Murder Column from the boxplot. We find out that the minimum value of the Murder column and boxplot is 0.80 but we can consider it as this value 0.80 is not very close to the others values in the Murder column of the Dataset, and the maximum value is 17 but we can consider the maximum value as 20.



```
>  ResultMurder=Information[(Information$Murder >20 |  Information$Murder < 1),]
>  ResultMurder
         States Murder Assault Urban Population(%)
34 North Dakota    0.8      45                  44
>
```

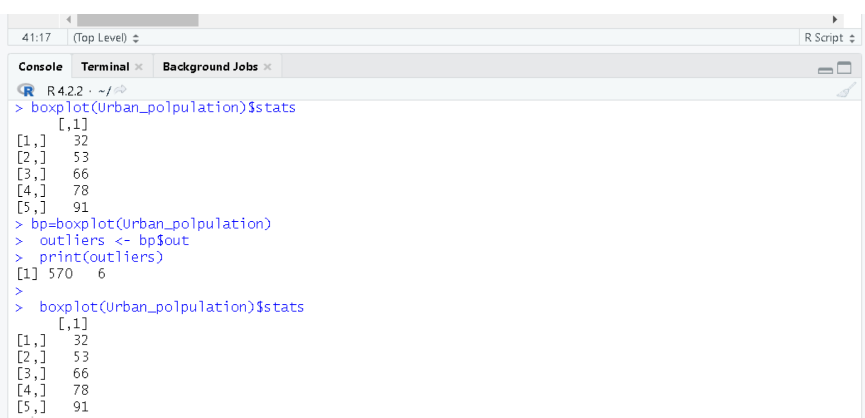
Here we can see the detailed information of the Outlier data information. And we found one Outlier in the Murder Column

For Urban Population (%):

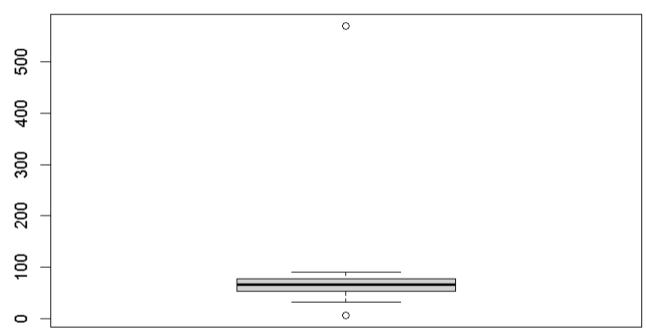
Here we can see two Outlier in the Murder Column from the boxplot. And using the code we found that the Outlier is 570 and 6. And we also find out that the minimum value of the Urban Population(%) column and boxplot is 32 but we can consider it as 30 and the maximum value is 91 but we can consider the maximum value as 100.

Code:

```
> bp=boxplot(Urban_polpulation)
>  outliers <- bp$out
>  print(outliers)
[1] 570   6
> boxplot(Urban_polpulation)$stats
     [,1]
[1,]   32
[2,]   53
[3,]   66
[4,]   78
[5,]   91
>  Result=Information[(Information$"Urban Population(%)" >100 |  Information$"Urban Populati
on(%)" < 30),]
>  Result
     States Murder Assault Urban Population(%)
15     Iowa    2.2      56                 570
32 New York   11.1     254                   6
```



```
41:17   (Top Level) ÷                                    R Script ÷
Console   Terminal ×   Background Jobs ×
R 4.2.2 · ~/
> boxplot(Urban_polpulation)$stats
     [,1]
[1,]   32
[2,]   53
[3,]   66
[4,]   78
[5,]   91
> bp=boxplot(Urban_polpulation)
>  outliers <- bp$out
>  print(outliers)
[1] 570   6
>
>  boxplot(Urban_polpulation)$stats
     [,1]
[1,]   32
[2,]   53
[3,]   66
[4,]   78
[5,]   91
>
```



```
Console   Terminal ×   Background Jobs ×
R 4.2.2 · ~/
> Result=Information[(Information$"Urban Population(%)" >100 |  Information$"Urban Population(%)" < 3
0),]
>  Result
     States Murder Assault Urban Population(%)
15     Iowa    2.2      56                 570
32 New York   11.1     254                   6
> 
```

Here we can see the detailed information of the Outlier data information. And we found one Two in the Urban Population(%) Column.

To deal with the Assault noisy data we can deal with it by the maiden value of that Assault Column. As the mean value is highly influenced by the outliers, it is advised to replace the outliers with the median value. So now we need to calculate the maiden and then insert it in the Assault Column in the South Carolina state row.

```
Code:
>  medianAssault=median(Information$Assault)
>
>  medianAssault
[1] 159
>  Information$Assault[Information$Assault ==879] <-medianAssault
>  Information
```

```
>  medianAssault=median(Information$Assault)
>  medianAssault=median(Information$Assault)
>
>  medianAssault
[1] 159
> 
```

39	Rhode Island	3.4	174.0000	87
40	South Carolina	14.4	159.0000	48
41	South Dakota	3.8	86.0000	45

We can deal with the Murder Outlier in the data transformation as data transformation is also can deal with the smoothing the data.

In the Urban Population(%) column we can replace the two outlier value of Iowa and New York by 57 and 60 which is close to the numbers around them which could be a great solution.

Code:

```
Information$"Urban Population(%)"[Information$"Urban Population(%)" ==570] <-57
Information$"Urban Population(%)"[Information$"Urban Population(%)" ==6] <-60
Information
```

```
13         Ilinois   10.4 249.0000                  83
14         Indiana    7.2 113.0000                  65
15            Iowa    2.2  56.0000                  57
```

```
31      New Mexico   11.4 285.0000                  70
32        New York   11.1 254.0000                  60
33 North Carolina    13.0 337.0000                  45
```

Data Integration:

We can perform the Data Integration step by combining the data from multiple sources in a single file or dataset. Here In this project, we need to add a new column (population level) by Categorized the urban population like [“Small”<50, “Medium”>=50 and “Medium”<60, “Large”>=60 and “Large”<70 and then “Extra-Large”>=70] and add it to out Dataset.

Code:

```
NewInformation=Information %>%
mutate("Population level" = case_when(

Information$"Urban Population(%)"<50   ~ "small",
(Information$"Urban Population(%)"<60 & Information$"Urban Population(%)">=50)  ~ "medium",

(Information$"Urban Population(%)"<70 & Information$"Urban Population(%)">=60)  ~ "large",

(Information$"Urban Population(%)">=70) ~ "extra-large"

))
> NewInformation
```

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Untitled1 × InputOutput.R × Practice1.R × Project.R × ProjectCode.R × New

Filter

	States	Murder	Assault	Urban Population(%)	Population level
1	Alabama	13.2	236.0000	58	medium
2	Alaska	10.0	263.0000	48	small
3	Arizona	8.1	294.0000	80	extra-large
4	Arkansas	8.8	190.0000	50	medium
5	California	9.0	276.0000	91	extra-large
6	Colorado	7.9	204.0000	78	extra-large
7	Connecticut	3.3	110.0000	77	extra-large
8	Delaware	5.9	238.0000	72	extra-large
9	Florida	15.4	335.0000	80	extra-large
10	Georgia	17.4	182.1837	60	large
11	Hawaii	5.3	46.0000	83	extra-large
12	Idaho	2.6	120.0000	54	medium
13	Ilinois	10.4	249.0000	83	extra-large
14	Indiana	7.2	113.0000	65	large
15	Iowa	2.2	56.0000	57	medium
16	Kansas	6.0	115.0000	66	large
17	Kentucky	9.7	109.0000	52	medium
18	Louisiana	15.4	249.0000	66	large
19	Maine	2.1	83.0000	51	medium
20	Maryland	11.3	300.0000	67	large
21	Massachusetts	4.4	149.0000	85	extra-large

Showing 1 to 21 of 50 entries, 5 total columns

Data Transformation:

We Perform the data Transformation so that the data becomes consistent and readable to the system. Here we can round the Murder in each state and perform the data transformation. Here Murders are in decimal, but Murder cannot be like this in decimal format. Data Transformation can be applied to remove the outliers in the data and smooth the data

Code:
```
> NewInformation$Murder=round(NewInformation$Murder)
> View(NewInformation)
```

Untitled1 × | InputOutput.R × | Practice1.R × | Project.R × | ProjectCode

Filter

	States	Murder	Assault	Urban Population(%)	Population level
1	Alabama	13	236.0000	58	medium
2	Alaska	10	263.0000	48	small
3	Arizona	8	294.0000	80	extra-large
4	Arkansas	9	190.0000	50	medium
5	California	9	276.0000	91	extra-large
6	Colorado	8	204.0000	78	extra-large
7	Connecticut	3	110.0000	77	extra-large
8	Delaware	6	238.0000	72	extra-large
9	Florida	15	335.0000	80	extra-large
10	Georgia	17	182.1837	60	large
11	Hawaii	5	46.0000	83	extra-large
12	Idaho	3	120.0000	54	medium
13	Ilinois	10	249.0000	83	extra-large
14	Indiana	7	113.0000	65	large
15	Iowa	2	56.0000	57	medium
16	Kansas	6	115.0000	66	large
17	Kentucky	10	109.0000	52	medium

Again Population level is Numerical so we can transform the Population level variable into a ordered factor variable like [“Small”=1, “Medium”=2, “Large”=3 and then “Extra-Large”=4].

Code:

```
> me=NewInformation$`Population level`
> me=factor(me,levels=c("small","medium","large","extra-large"),labels=c(1,
2,3,4))
> NewInformation$me=me
> colnames(NewInformation)[6]  <- "Ordered Factor Population"
> NewInformation
```

	States	Murder	Assault	Urban Population(%)	Population level	Ordered Factor Population
1	Alabama	13	236	58	medium	2
2	Alaska	10	263	48	small	1
3	Arizona	8	294	80	extra-large	4
4	Arkansas	9	190	50	medium	2
5	California	9	276	91	extra-large	4
6	Colorado	8	204	78	extra-large	4
7	Connecticut	3	110	77	extra-large	4
8	Delaware	6	238	72	extra-large	4
9	Florida	15	335	80	extra-large	4
10	Georgia	17	182	60	large	3
11	Hawaii	5	46	83	extra-large	4
12	Idaho	3	120	54	medium	2
13	Ilinois	10	249	83	extra-large	4
14	Indiana	7	113	65	large	3
15	Iowa	2	56	57	medium	2
16	Kansas	6	115	66	large	3

Data Reduction:

In the Dataset we can see that the Assault is in decimal format which holds a lot of space we can round the Assault column values. So here we will perform the Data Reduction.

Code:

```
> NewInformation$Assault <- round(NewInformation$Assault)
> NewInformation
```

Untitled1 × InputOutput.R × Practice1.R × Project.R × ProjectCode.R × New

Filter

	States	Murder	Assault	Urban Population(%)	Population level	Ordered Factor Population
1	Alabama	13	236	58	medium	2
2	Alaska	10	263	48	small	1
3	Arizona	8	294	80	extra-large	4
4	Arkansas	9	190	50	medium	2
5	California	9	276	91	extra-large	4
6	Colorado	8	204	78	extra-large	4
7	Connecticut	3	110	77	extra-large	4
8	Delaware	6	238	72	extra-large	4
9	Florida	15	335	80	extra-large	4
10	Georgia	17	182	60	large	3
11	Hawaii	5	46	83	extra-large	4
12	Idaho	3	120	54	medium	2
13	Ilinois	10	249	83	extra-large	4
14	Indiana	7	113	65	large	3
15	Iowa	2	56	57	medium	2
16	Kansas	6	115	66	large	3

Data Discretization:

In data Discretization we need to convert the continuous value to a more manageable part. But here we can see we don't need to do the data Discretization as all the data in the data set is already in the manageable parts. It is important for numerical data. Previously we add a new column in the dataset "Order Factor Variable" which we can consider as data Discretization.

Code:

```
> NewInformation=NewInformation %>%
+   mutate("Ordered Factor Population" = case_when(
+     NewInformation$"Population level"=="small"    ~ 1,
+     (NewInformation$"Population level"=="medium") ~ 2,
+     (NewInformation$"Population level"=="large")  ~3,
+     (NewInformation$"Population level"=="extra-large") ~ 4
+   ))
> NewInformation
View(NewInformation)
```

Untitled1 × InputOutput.R × Practice1.R × Project.R × ProjectCode.R × NewInformation ×

Filter

	States	Murder	Assault	Urban Population(%)	Population level	Ordered Factor Population
19	Maine	2	83.0000	51	medium	2
20	Maryland	11	300.0000	67	large	3
21	Massachusetts	4	149.0000	85	extra-large	4
22	Michigan	12	255.0000	74	extra-large	4
23	Minnesota	3	72.0000	66	large	3
24	Mississippi	16	259.0000	44	small	1
25	Missouri	9	178.0000	70	extra-large	4
26	Montana	6	109.0000	53	medium	2
27	Nebraska	4	102.0000	62	large	3
28	Nevada	12	252.0000	81	extra-large	4
29	New Hampshire	2	57.0000	56	medium	2
30	New Jersey	7	159.0000	89	extra-large	4
31	New Mexico	11	285.0000	70	extra-large	4
32	New York	11	254.0000	60	large	3
33	North Carolina	13	337.0000	45	small	1
34	North Dakota	1	45.0000	44	small	1
35	Ohio	7	120.0000	75	extra-large	4

Discussion & Conclusion:

For data processing, we will gradually improve the data and use the R language constructs and techniques. After all data preprocessing techniques were successfully applied, the dataset was cleaner and nicer. Nevertheless, I didn't have to use every step of the technique for this project. I learned about current data and data preprocessing in the industry. Expand your toolbox with more experience. Improve the accuracy of your dataset by preprocessing the data. Any values that are inaccurate or missing due to human error or problems are removed. Improved consistency. And More importantly I have faced many problems while dealing with the data and in one moment I made a incorrect column and add it to the data frame and got the wrong answers. So we need to be careful while doing the data preprocessing a dataset and adding new data to the dataset.