IC272 – Data Science 3 *Aryan Garg*

REPORT *B19153*

Lab Assignment – 2 *+91-8219383122*

---

1. Bar graph of missing values in each field attribute.

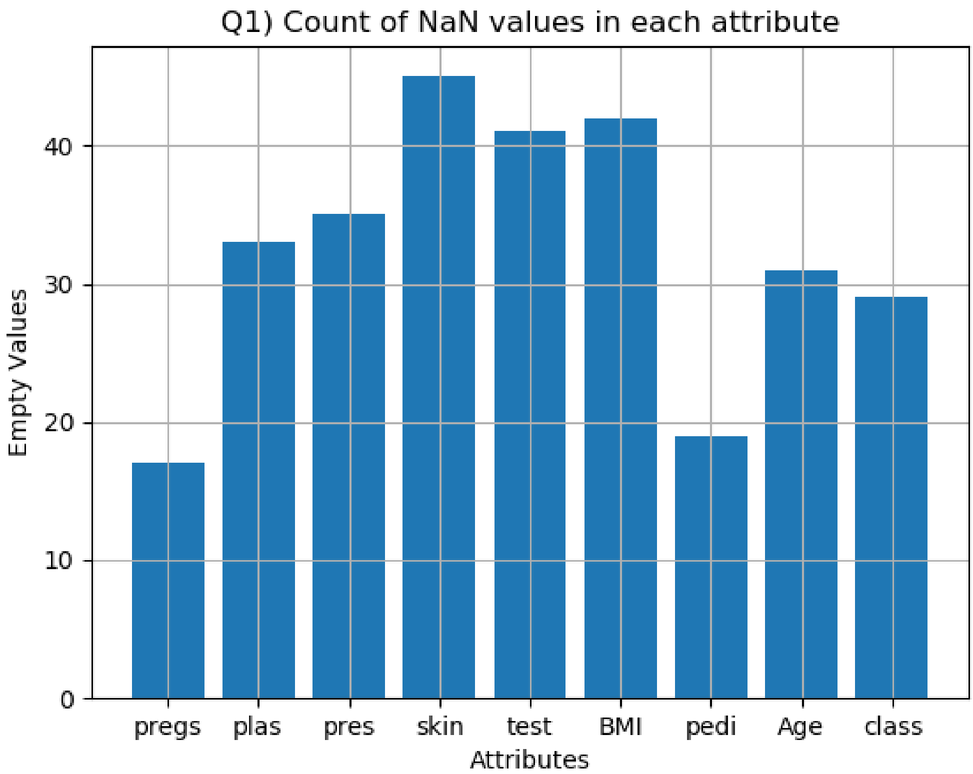

Figure 1

Observation(s):

i. The missing values in the class attribute will not allow our model to train as this is the result of the case. It hampers training.

ii. Looks as if the data was filled manually due to the high number of missing values.

2. 39 records were dropped and their indices are printed by the code. Here's the output for the (a) part.

```
Q2(a)
Number of records dropped with more than 2 attribute values missing: 39

The following records(index) were dropped with more than 2 missing attributes:
[1, 39, 40, 53, 54, 83, 89, 103, 125, 136, 145, 210, 211, 212, 213, 249, 250, 254, 280, 281, 284, 314, 321, 335, 429,
430, 449, 450, 451, 471, 472, 473, 474, 718, 719, 720, 721, 753, 766]
```

Figure 2

As was our observation in Q1, we dropped the records where values in the class record were missing. Here's the code output:

```
Q2 (b)
Number of records dropped with missing values in class attribute: 21

The following records(index) were dropped with missing class values:
[8, 13, 28, 29, 35, 62, 92, 95, 107, 110, 130, 131, 132, 133, 149, 182, 188, 218, 308, 746, 748]
```

Figure 3

3. Output for Q3.

```
Q3 Counting missing values in each attribute and the whole file...

Missing values in each attribute:
pregs : 0
plas : 12
pres : 9
skin : 8
test : 8
BMI : 12
pedi : 2
Age : 18
class : 0

Total missing values in the file: 69
```

Figure 4

Observation(s):

i. We have considerably removed total missing values but at the same time reduced our training data set. But this trade-off is logical and beneficial because we have dropped records which don't have an outcome or have more than equal to 1/3rd values missing. These records were anyhow polluting our data.

4. Missing values have been imputed with the Nan Mean, i.e. mean of the non-missing values of each attribute. And then the means, medians and modes are computed for comparison with the original file which is a pre-processed file/file without missing values

```
(a)-(i)
Comparing mean, median and mode of the two files...
  --------------------------------------------
  Attribute      |    Mean    |   Original File Mean
  --------------------------------------------
  pregs          |     3.886 |     3.845
  plas           |   120.667 |   120.895
  pres           |    69.001 |    69.105
  skin           |    20.349 |    20.536
  test           |    77.814 |    79.799
  BMI            |    32.009 |    31.993
  pedi           |     0.476 |     0.472
  Age            |    33.094 |    33.241
  class          |     0.343 |     0.349
```

Figure 5

```
  ---------------------------------------------
  Attribute      |  Median   |Original File Median
  ---------------------------------------------
  pregs          |     3.000 |     3.000
  plas           |   118.000 |   117.000
  pres           |    72.000 |    72.000
  skin           |    23.000 |    23.000
  test           |    36.000 |    30.500
  BMI            |    32.009 |    32.000
  pedi           |     0.382 |     0.372
  Age            |    29.000 |    29.000
  class          |     0.000 |     0.000
```

Figure 6

```
----------------------------------------------
Attribute     |    Mode    |   Original File Mode
----------------------------------------------
pregs         |       1.0 |          1
plas          |      99.0 |         99
pres          |      70.0 |         70
skin          |       0.0 |          0
test          |       0.0 |          0
BMI           |      32.0 |       32.0
pedi          |     0.254 |      0.254
Age           |      22.0 |         22
class         |       0.0 |          0
```

Figure 7

Moving on to compute the error we are generating using the RMSE-loss function by imputing mean:

```
(a)-(ii)
RMSE values (attribute-wise):
pregs     ->    0.000
plas      ->   31.237
pres      ->   11.391
skin      ->   11.919
test      ->   74.321
BMI       ->   12.228
pedi      ->    0.157
Age       ->    8.517
class     ->    0.000
```

Figure 8

Bar graph of the same:

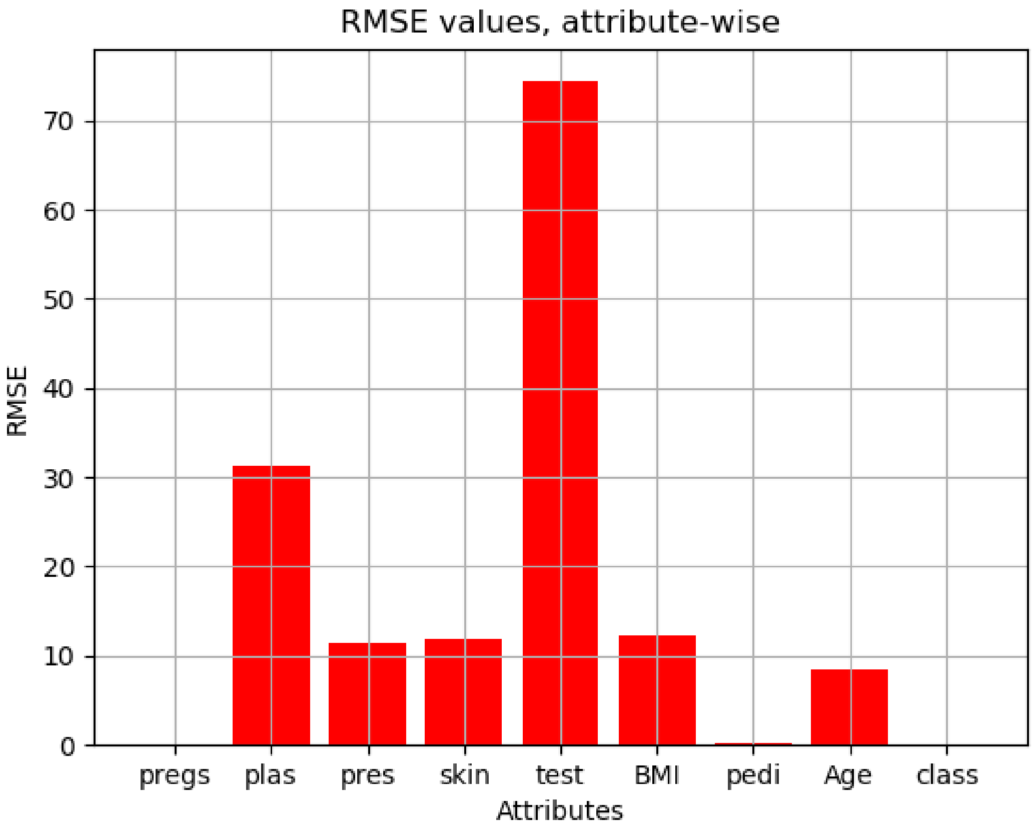

Figure 9

(b) Now we will linearly interpolate the missing values. We expect a lower loss with this method of imputation as it preserves relation with all elements of the data.

First, we compare it with the original file:

```
Q4(b)
Replacing values by linear interpolation(column-wise)...
[+] Done.

(b)-(i)
File1 -> Linearly interpolated imputation
File2 -> Original
Comparing mean, median and mode of the two files...
  ---------------------------------------------
  Attribute    |   Mean    |   Original File Mean
  ---------------------------------------------
  pregs        |     3.886 |     3.845
  plas         |   120.350 |   120.895
  pres         |    69.109 |    69.105
  skin         |    20.393 |    20.536
  test         |    77.355 |    79.799
  BMI          |    32.046 |    31.993
  pedi         |     0.477 |     0.472
  Age          |    33.216 |    33.241
  class        |     0.343 |     0.349
```

Figure 10

Attribute	Median	Original File Median
pregs	3.000	3.000
plas	117.000	117.000
pres	72.000	72.000
skin	23.000	23.000
test	27.000	30.500
BMI	32.250	32.000
pedi	0.382	0.372
Age	29.000	29.000
class	0.000	0.000

Figure 12

Attribute	Mode	Original File Mode
pregs	1.0	1
plas	99.0	99
pres	70.0	70
skin	0.0	0
test	0.0	0
BMI	32.0	32.0
pedi	0.254	0.254
Age	22.0	22
class	0.0	0

Figure 11

Observation(s):

i. It is clear that linear interpolation not only preserves relationship with other data entries, but also gives a better fit than mean imputation as is clear from the descriptive analysis comparison of the two files.

Proceeding to analyse loss and the bar graph of the same, here are the results:

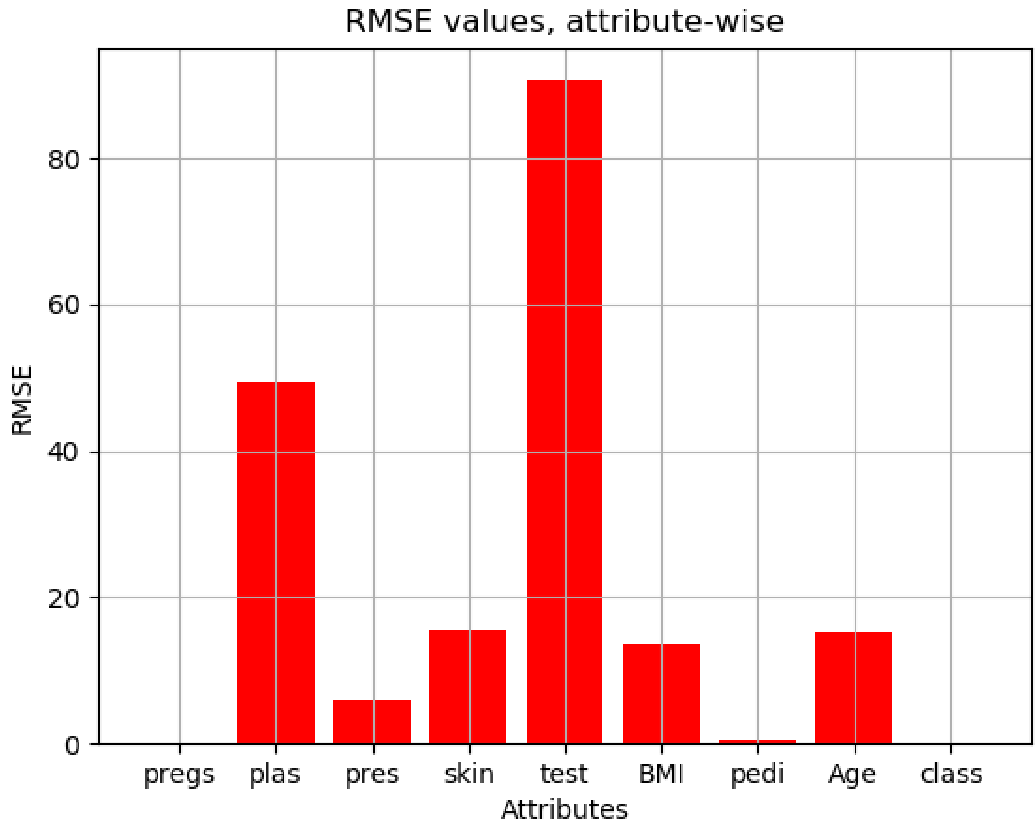

Figure 13

Observation(s):

- **i.** Loss in most of the attributes have gone down.
- **ii.** Loss in ‘test’ and ‘plas’ attribute have gone up.

Possible reason(s):

- **i.** For this increase in loss, a possible reason could be that the data could be highly scattered which ultimately gives a poor linear interpolation.
- **ii.** The decrease in loss was expected due to the preservation of data relationship in most of the cases.

5. **Outlier detection and manipulation.**
   Here is the boxplot of 'Age' and 'BMI' attribute.

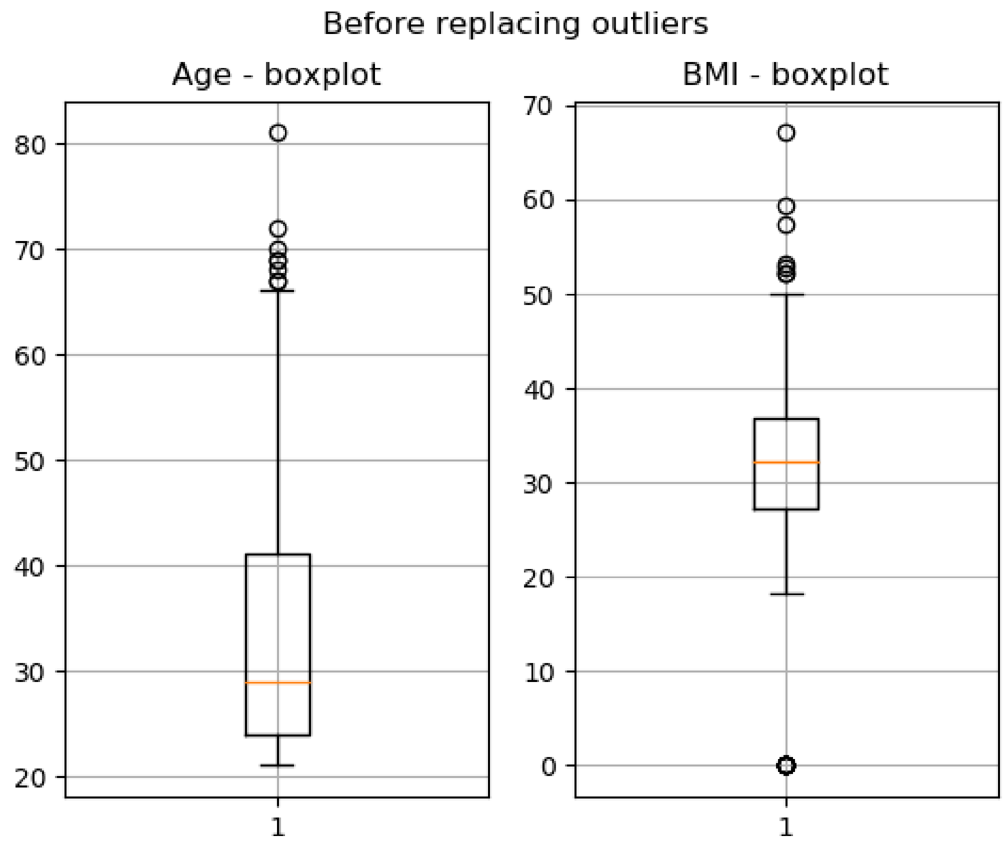

Figure 14

Many outliers are present in both fields and they are <u>mostly above the upper-bounds</u>.

Now we proceed to find these outlier values and replace them with the median which is found using the data series with the current outliers as found in the above boxplot.

Here's the boxplot after replacing the outliers with attribute medians:

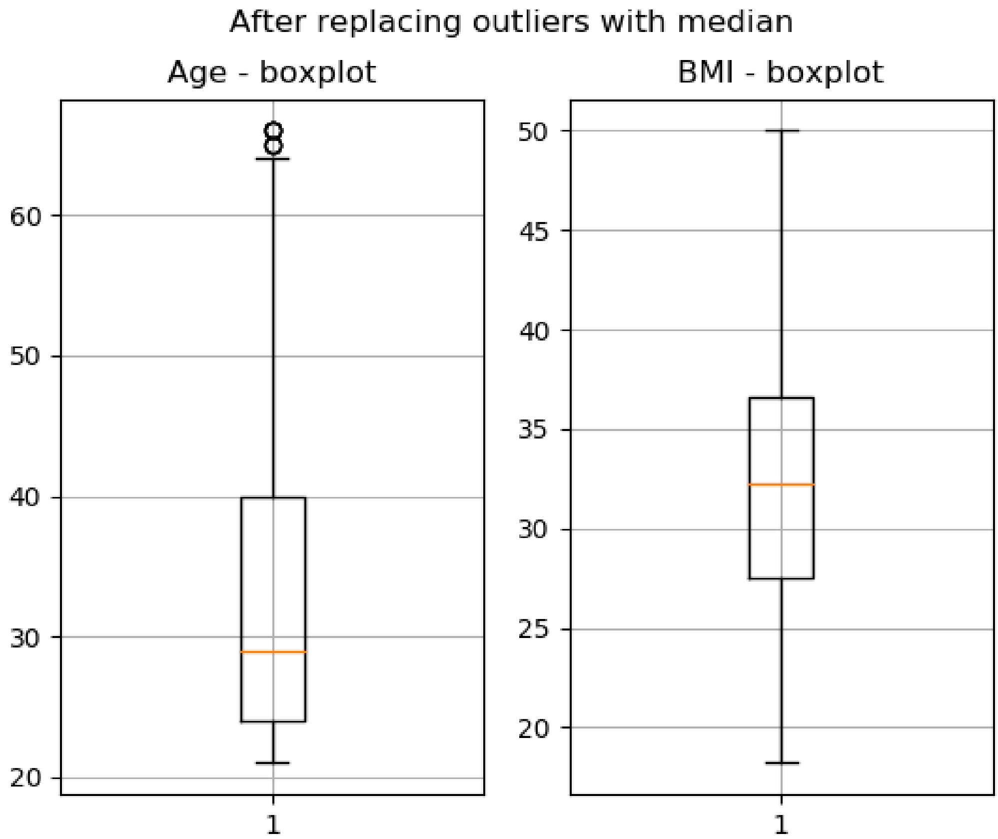

Figure 15

Observation(s):

- **i.** Median remains the same after replacement
- **ii.** There are still outliers in the 'Age' attribute

Reason(s):

- **i.** Median (Q2) is the central quantity of the data whose position is fixed(index-wise). Hence, the median is bound to remain the same as we are only replacing values towards the front and the back of the data.
- **ii.** The third quartile (q3) of both attributes has decreased (because there were majorly upper outliers which were replaced by lower values) which ultimately decreases the outlier upper-bound: q3 + (1.5 * IQR), thus exposing some earlier values which used to lie within the former upper bound

Here’s proof that the program successfully executed itself and finished with 0 warnings and errors:

```
Q5 Outlier detection and manipulation under progress...
q1: 24.0 | q3: 41.0
q1: 27.3 | q3: 36.8
q1: 24.0 | q3: 40.0
q1: 27.5 | q3: 36.6
[+] Program finished successfully.
---------------------------- xxx ---------------------------
>>>
```