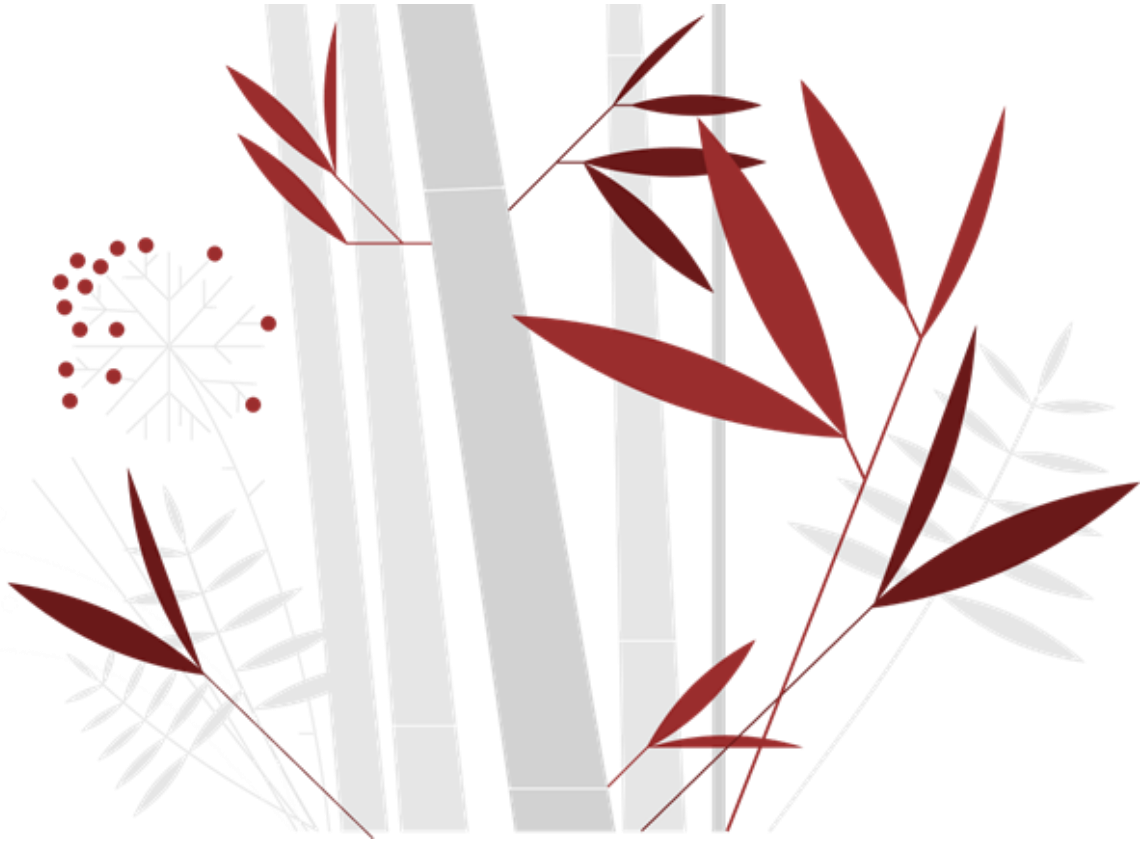
PREDICTIVE MODELING (PM) PROJECT-CODED

BY

Harsh Patel

9th June 2024



Problem-1: CPU Usage

The comp-activ database comprises activity measures of computer systems. Data was gathered from a Sun Sparcstation 20/712 with 128 Mbytes of memory, operating in a multi-user university department. Users engaged in diverse tasks, such as internet access, file editing, and CPU-intensive programs.

Being an aspiring data scientist, you aim to establish a linear equation for predicting 'usr' (the percentage of time CPUs operate in user mode). Your goal is to analyse various system attributes to understand their influence on the system's 'usr' mode.

Data Variables:

- lread - Reads (transfers per second) between system memory and user memory
- lwrite - writes (transfers per second) between system memory and user memory
- scall - Number of systems calls of all types per second
- sread - Number of systems read calls per second.
- swrite - Number of systems write calls per second.
- fork - Number of system fork calls per second.
- exec - Number of system exec calls per second.
- rchar - Number of characters transferred per second by system read calls
- wchar - Number of characters transfreed per second by system write calls
- pgout - Number of pages out requests per second
- ppgout - Number of pages, paged out per second
- pgfree - Number of pages per second placed on the free list.
- pgscan - Number of pages checked if they can be freed per second
- atch - Number of page attaches (satisfying a page fault by reclaiming a page in memory) per second
- pgin - Number of page-in requests per second
- ppgin - Number of pages paged in per second
- pflt - Number of page faults caused by protection errors (copy-on-writes).
- vflt - Number of page faults caused by address translation.
- runqsz - Process run queue size (The number of kernel threads in memory that are waiting for a CPU to run. Typically, this value should be less than 2. Consistently higher values mean that the system might be CPU-bound.)
- freemem - Number of memory pages available to user processes
- freeswap - Number of disk blocks available for page swapping.
- usr - Portion of time (%) that cpus run in user mode (This is our dependent variable)

1.1 Data Pre-Processing and EDA:

First, we will look at first and last five rows using function head and tail respectively, of the dataset from excel file called Clustering clean ads data that we loaded using read excel function. In fig-1 and fig-2 shows below shows the dataset.

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	...	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	freeswap	usr
0	1	0	2147	79	68	0.2	0.2	40671.0	53995.0	0.0	...	0.0	0.0	1.6	2.6	16.00	26.40	CPU_Bound	4670	1730946	95
1	0	0	170	18	21	0.2	0.2	448.0	8385.0	0.0	...	0.0	0.0	0.0	0.0	15.63	16.83	Not_CPU_Bound	7278	1869002	97
2	15	3	2162	159	119	2.0	2.4	NaN	31950.0	0.0	...	0.0	1.2	6.0	9.4	150.20	220.20	Not_CPU_Bound	702	1021237	87
3	0	0	160	12	16	0.2	0.2	NaN	8670.0	0.0	...	0.0	0.0	0.2	0.2	15.60	16.80	Not_CPU_Bound	7248	1863704	98
4	5	1	330	39	38	0.4	0.4	NaN	12185.0	0.0	...	0.0	0.0	1.0	1.2	37.80	47.60	Not_CPU_Bound	633	1760253	90

5 rows × 22 columns

Table-1 first five rows

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	...	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	freeswap	usr
8187	16	12	3009	360	244	1.6	5.81	405250.0	85282.0	8.02	...	55.11	0.6	35.87	47.90	139.28	270.74	CPU_Bound	387	986647	80
8188	4	0	1596	170	146	2.4	1.80	89489.0	41764.0	3.80	...	0.20	0.8	3.80	4.40	122.40	212.60	Not_CPU_Bound	263	1055742	90
8189	16	5	3116	289	190	0.6	0.60	325948.0	52640.0	0.40	...	0.00	0.4	28.40	45.20	60.20	219.80	Not_CPU_Bound	400	969106	87
8190	32	45	5180	254	179	1.2	1.20	62571.0	29505.0	1.40	...	18.04	0.4	23.05	24.25	93.19	202.81	CPU_Bound	141	1022458	83
8191	2	0	985	55	46	1.6	4.80	111111.0	22256.0	0.00	...	0.00	0.2	3.40	6.20	91.80	110.00	CPU_Bound	659	1756514	94

Table-2 last five rows

Now, we use shape function the dataset and we get that there are 8192 row and 22 columns. Then, we use info function and found out the data type of each column and used value counts functions on the categorical variables as shown in the below table. we will check for the duplicated rows are present or not using duplicate function and found out that in the dataset there are 80 duplicated rows present and we removed those rows using drop duplicates function on the whole dataset.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8192 entries, 0 to 8191
Data columns (total 22 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   lread     8192 non-null   int64
 1   lwrite    8192 non-null   int64
 2   scall     8192 non-null   int64
 3   sread     8192 non-null   int64
 4   swrite    8192 non-null   int64
 5   fork      8192 non-null   float64
 6   exec      8192 non-null   float64
 7   rchar     8088 non-null   float64
 8   wchar     8177 non-null   float64
 9   pgout     8192 non-null   float64
 10  ppgout    8192 non-null   float64
 11  pgfree    8192 non-null   float64
 12  pgscan    8192 non-null   float64
 13  atch      8192 non-null   float64
 14  pgin      8192 non-null   float64
 15  ppgin     8192 non-null   float64
 16  pflt      8192 non-null   float64
 17  vflt      8192 non-null   float64
 18  runqsz    8192 non-null   object
 19  freemem   8192 non-null   int64
 20  freeswap  8192 non-null   int64
 21  usr       8192 non-null   int64
dtypes: float64(13), int64(8), object(1)
```

Table-3 Information table

From the above table we can say that there are missing values in “rchar” and “wchar” variables by imputing the median of their respective variables into the empty or null values. Other, thing we can say from the above table is that there are 13 float, 6 integer and 1 oblject data type variables present in our dataset.

lread	0
lwrite	0
scall	0
sread	0
swrite	0
fork	0
exec	0
rchar	104
wchar	15
pgout	0
ppgout	0
pgfree	0
pgscan	0
atch	0
pgin	0
ppgin	0
pflt	0
vflt	0
runqsz	0
freemem	0
freeswap	0
usr	0

Table-4 Missing values present before Treatment

lread	0
lwrite	0
scall	0
sread	0
swrite	0
fork	0
exec	0
rchar	0
wchar	0
pgout	0
ppgout	0
pgfree	0
pgscan	0
atch	0
pgin	0
ppgin	0
pflt	0
vflt	0
runqsz	0
freemem	0
freeswap	0
usr	0

Tabel-5 No missing values after treatment

Now, we will use describe function on this dataset and obtain the min, max, std, count, mean, q1, q2, q3 values for all the numerical columns as shown in the below table. We can observe that for some the numerical variables the min, max values are too larger i.e. they are far apart from each other indicating presence of outliers in them as well as there is variation between variables so we will need to scale them.

	count	mean	std	min	25%	50%	75%	max
lread	8192.0	1.955969e+01	53.353799	0.0	2.00	7.0	20.000	1845.00
lwrite	8192.0	1.310620e+01	29.891726	0.0	0.00	1.0	10.000	575.00
scall	8192.0	2.306318e+03	1633.617322	109.0	1012.00	2051.5	3317.250	12493.00
sread	8192.0	2.104800e+02	198.980146	6.0	86.00	166.0	279.000	5318.00
swrite	8192.0	1.500582e+02	160.478980	7.0	63.00	117.0	185.000	5456.00
fork	8192.0	1.884554e+00	2.479493	0.0	0.40	0.8	2.200	20.12
exec	8192.0	2.791998e+00	5.212456	0.0	0.20	1.2	2.800	59.56
rchar	8192.0	1.964728e+05	238446.012054	278.0	34860.50	125473.5	265394.750	2526649.00
wchar	8192.0	9.581275e+04	140728.464118	1498.0	22977.75	46619.0	106037.000	1801623.00
pgout	8192.0	2.285317e+00	5.307038	0.0	0.00	0.0	2.400	81.44
ppgout	8192.0	5.977229e+00	15.214590	0.0	0.00	0.0	4.200	184.20
pgfree	8192.0	1.191971e+01	32.363520	0.0	0.00	0.0	5.000	523.00
pgscan	8192.0	2.152685e+01	71.141340	0.0	0.00	0.0	0.000	1237.00
atch	8192.0	1.127505e+00	5.708347	0.0	0.00	0.0	0.600	211.58
pgin	8192.0	8.277960e+00	13.874978	0.0	0.60	2.8	9.765	141.20
ppgin	8192.0	1.238859e+01	22.281318	0.0	0.60	3.8	13.800	292.61
pflt	8192.0	1.097938e+02	114.419221	0.0	25.00	63.8	159.600	899.80
vflt	8192.0	1.853158e+02	191.000603	0.2	45.40	120.4	251.800	1365.00
freemem	8192.0	1.763456e+03	2482.104511	55.0	231.00	579.0	2002.250	12027.00
freeswap	8192.0	1.328126e+06	422019.426957	2.0	1042623.50	1289289.5	1730379.500	2243187.00
usr	8192.0	8.396887e+01	18.401905	0.0	81.00	89.0	94.000	99.00

Table-6 Description of Numerical variables

We also checked the value counts for the categorical variables and can be seen in the below table.

```
runqsz
Not_CPU_Bound    4331
CPU_Bound        3861
Name: count, dtype: int64
```

Table-7 Value counts of Categorical variable

Now, will divide the dataset into dataset for numerical variables and categorical variables, and concatenate them after we have performed EDA on numerical and categorical variables as per requirement.

First of all, we will perform univariate analysis on the numerical and categorical dataset and drawn observation from them. We will also perform outlier treatment on the numerical dataset as per requirements.

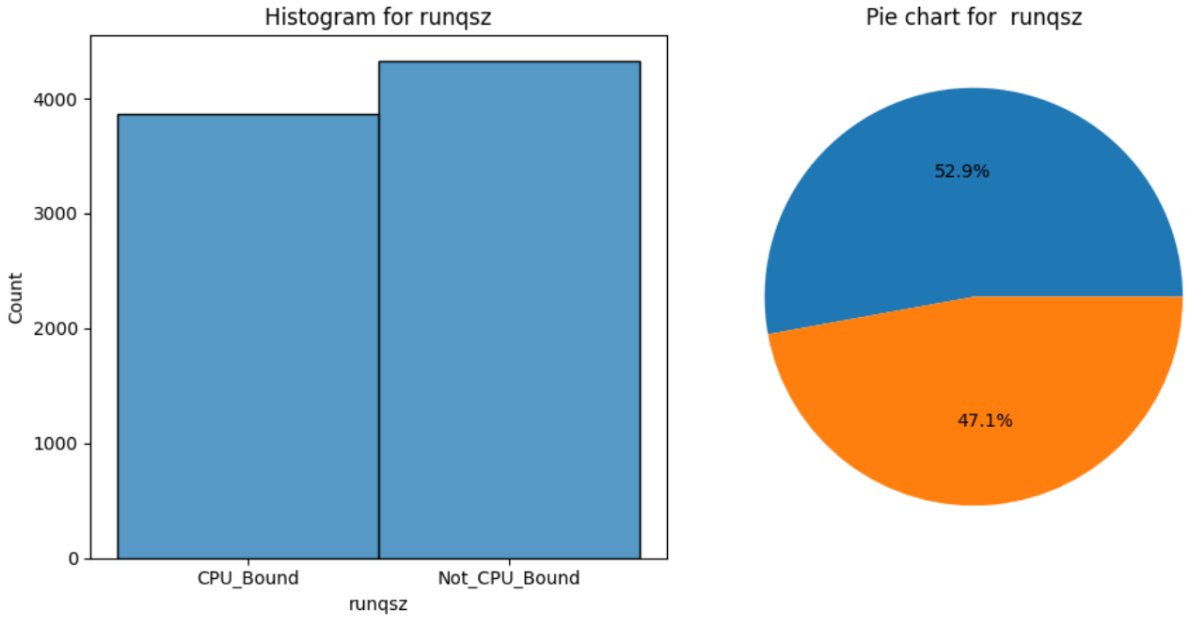
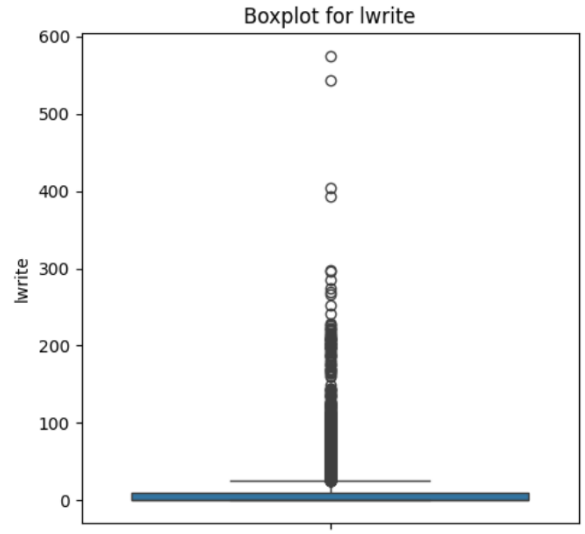


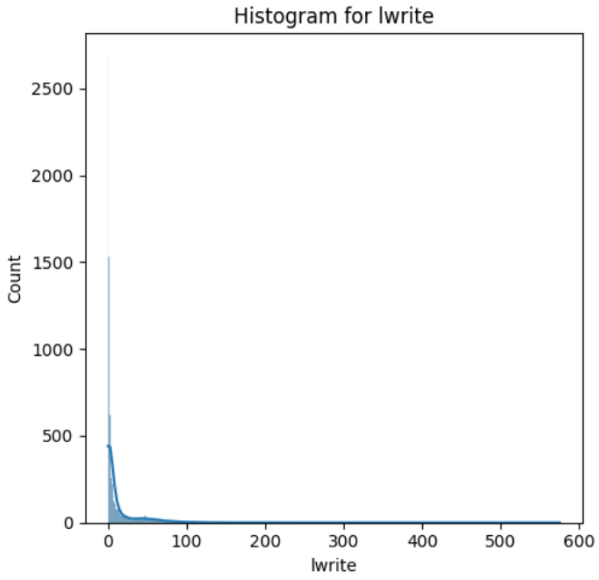
Fig-1 Count plot and Pie-chart for Run sequence variable

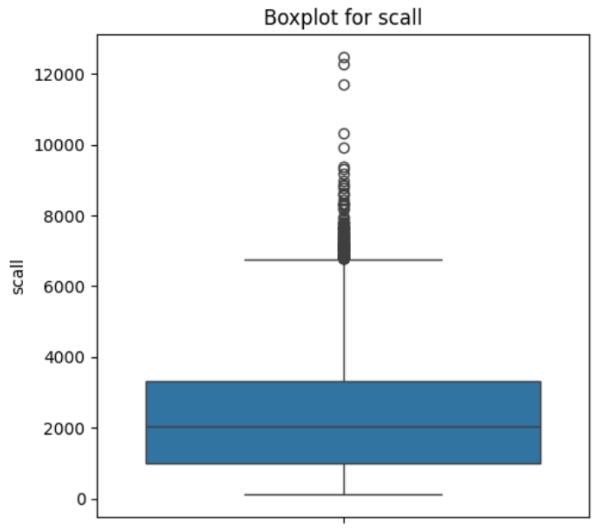
Insights:

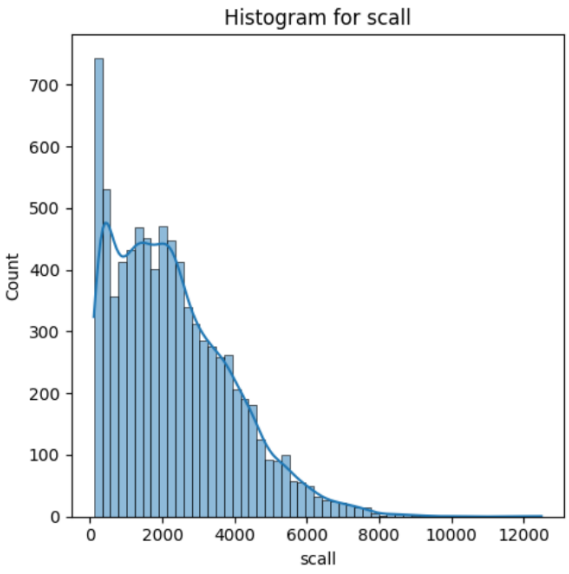
- A process run queue size below 2 signifies that the system is not constrained by CPU usage.
- Instances where the system is not limited by CPU resources outnumber those where it is CPU-bound.

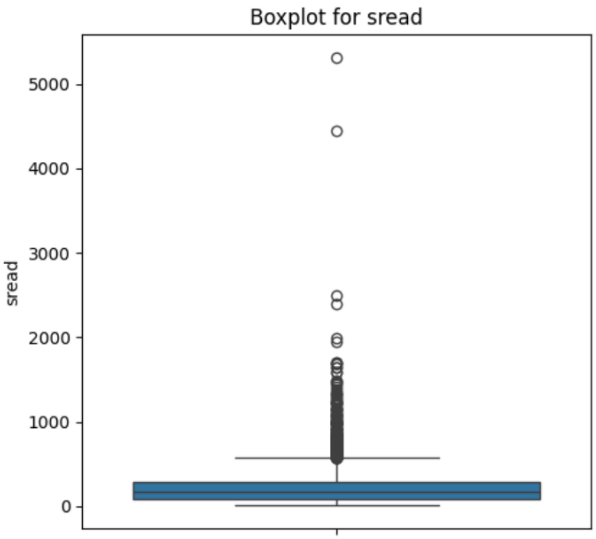
Now, as shown below figure we can see the that we have performed Univariate analysis on the numerical dataset that has presence of outliers.

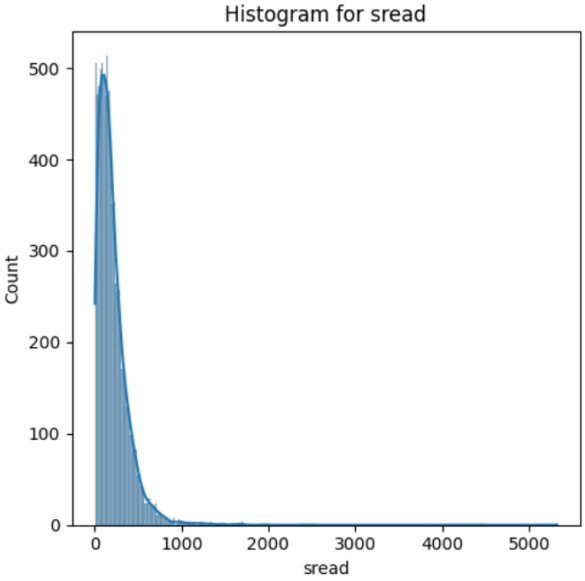

Boxplot for lwrite
lwrite
0
100
200
300
400
500
600

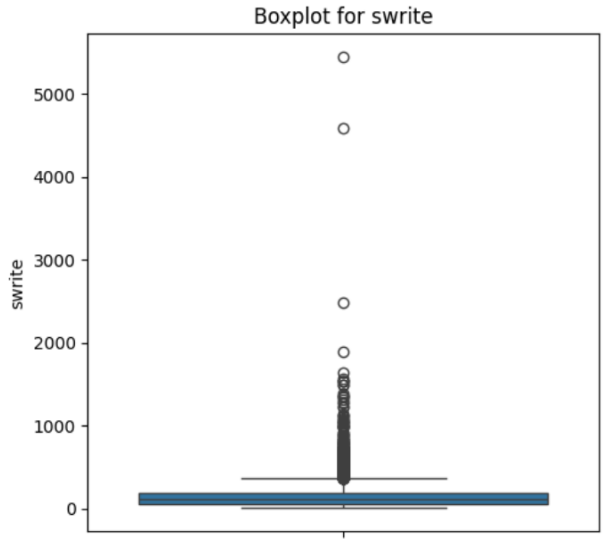

Histogram for lwrite
Count
lwrite
0
500
1000
1500
2000
2500
0
100
200
300
400
500
600

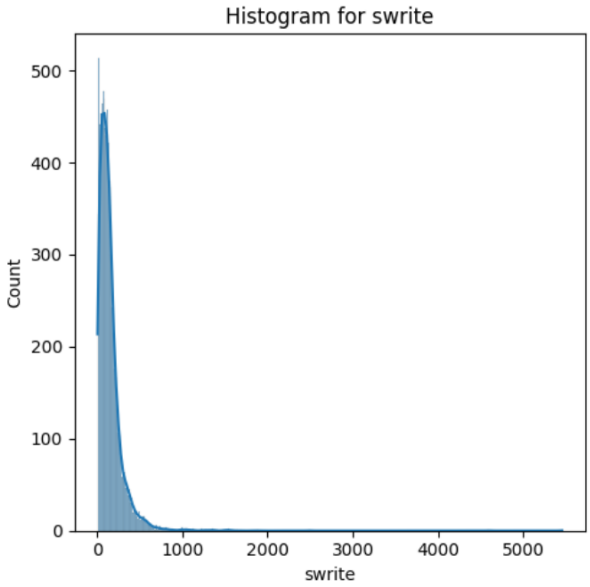

Boxplot for scall
scall
0
2000
4000
6000
8000
10000
12000

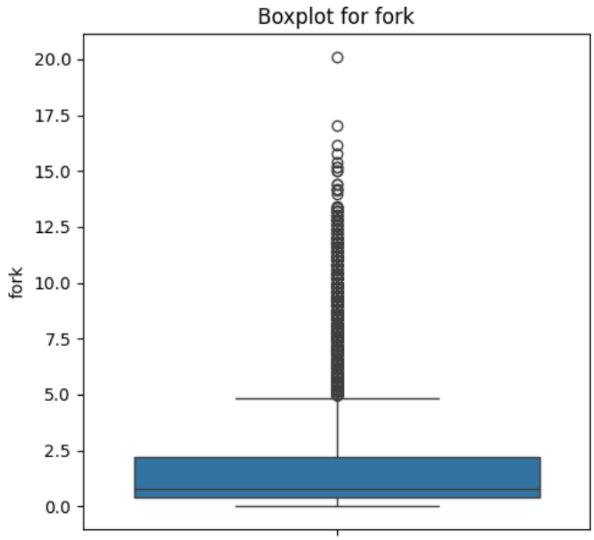

Histogram for scall
Count
scall
0
100
200
300
400
500
600
700
0
2000
4000
6000
8000
10000
12000

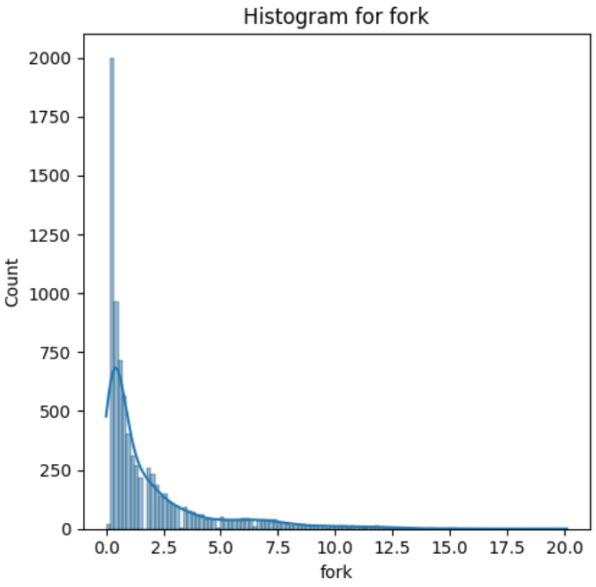

Boxplot for sread
sread
0
1000
2000
3000
4000
5000

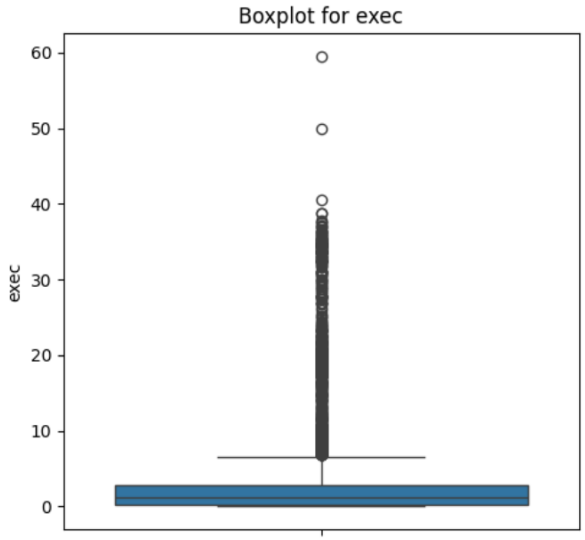

Histogram for sread
Count
sread
0
100
200
300
400
500
0
1000
2000
3000
4000
5000

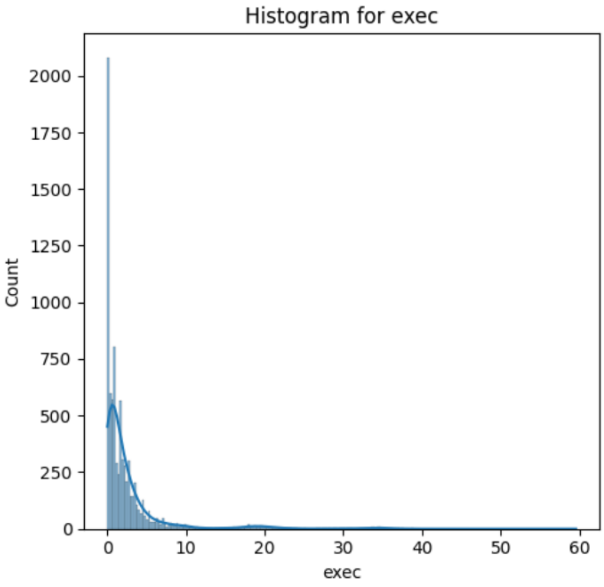

Boxplot for swrite
swrite
0
1000
2000
3000
4000
5000

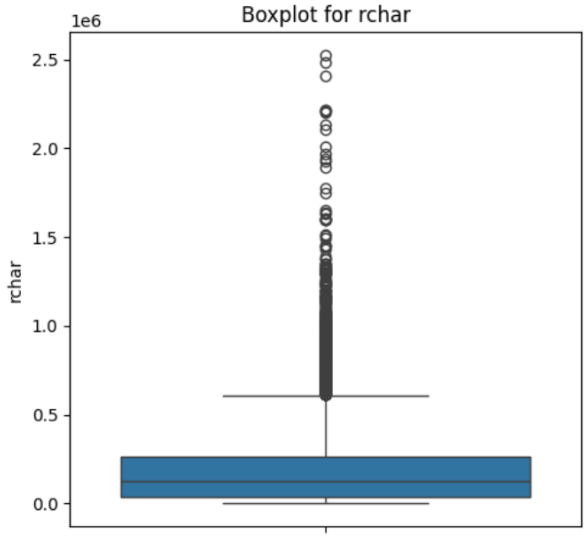

Histogram for swrite
Count
swrite
0
100
200
300
400
500
0
1000
2000
3000
4000
5000

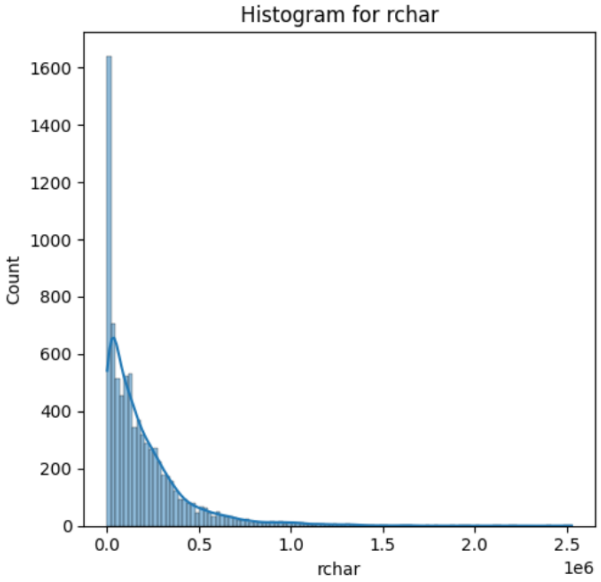

Boxplot for fork
fork
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

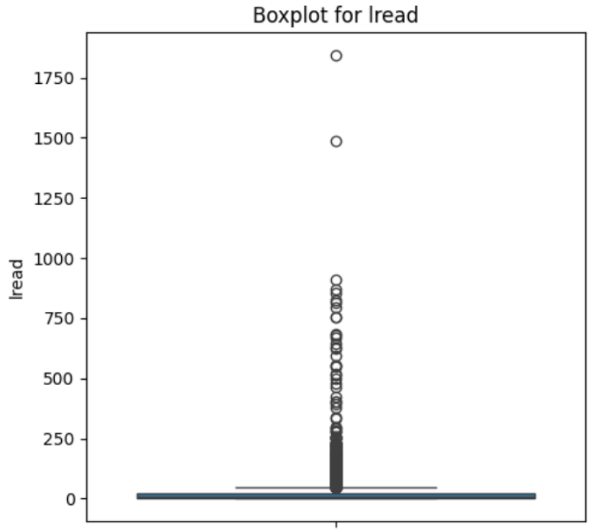

Histogram for fork
Count
fork
0
250
500
750
1000
1250
1500
1750
2000
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

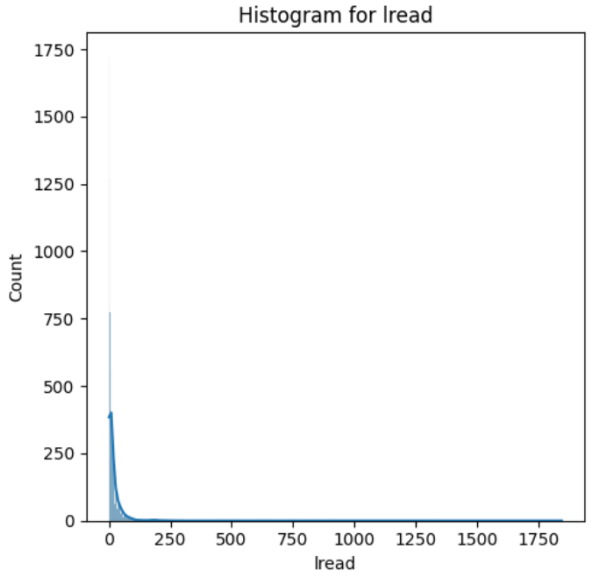

Boxplot for exec
exec
0
10
20
30
40
50
60

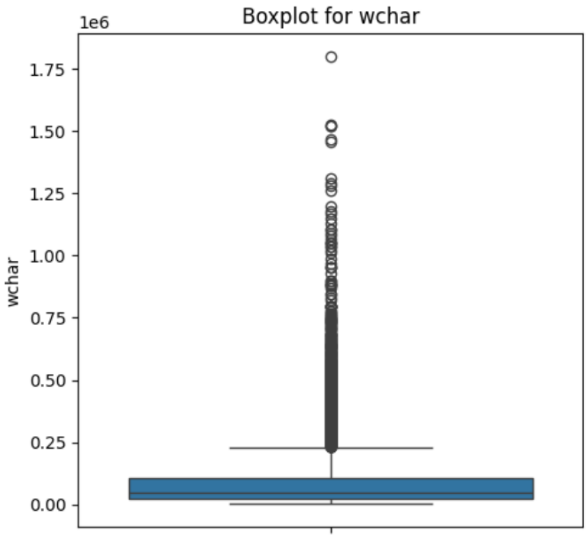

Histogram for exec
Count
exec
0
250
500
750
1000
1250
1500
1750
2000
0
10
20
30
40
50
60

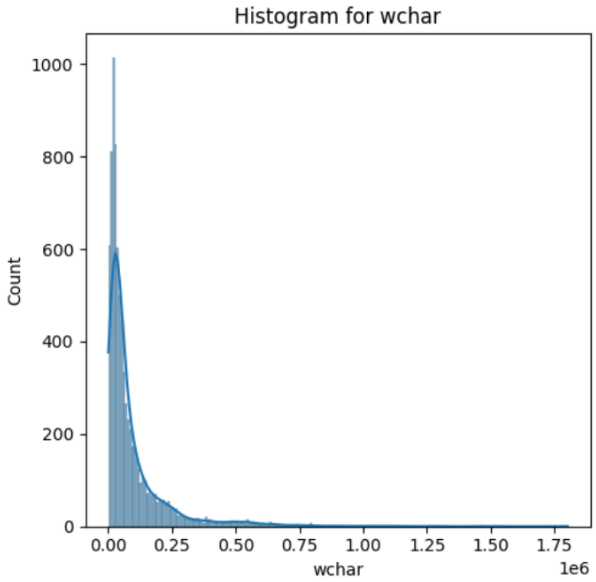

Boxplot for rchar
1e6
rchar
2.5
2.0
1.5
1.0
0.5
0.0

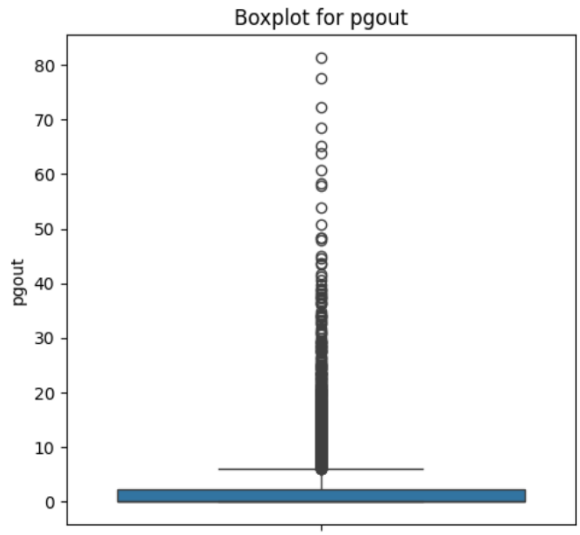

Histogram for rchar
Count
rchar
1e6
0.0
0.5
1.0
1.5
2.0
2.5
0
200
400
600
800
1000
1200
1400
1600

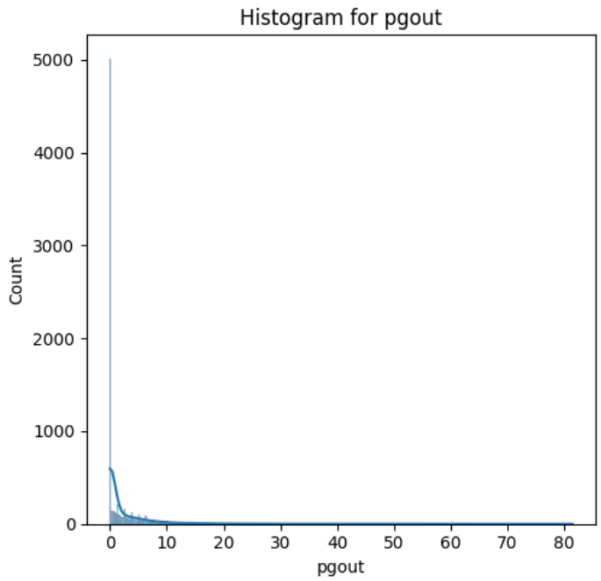

Boxplot for lread
lread
1750
1500
1250
1000
750
500
250
0

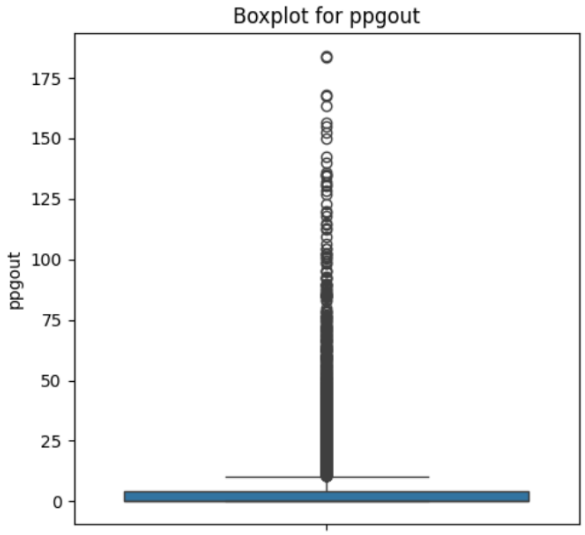

Histogram for lread
Count
lread
0
250
500
750
1000
1250
1500
1750

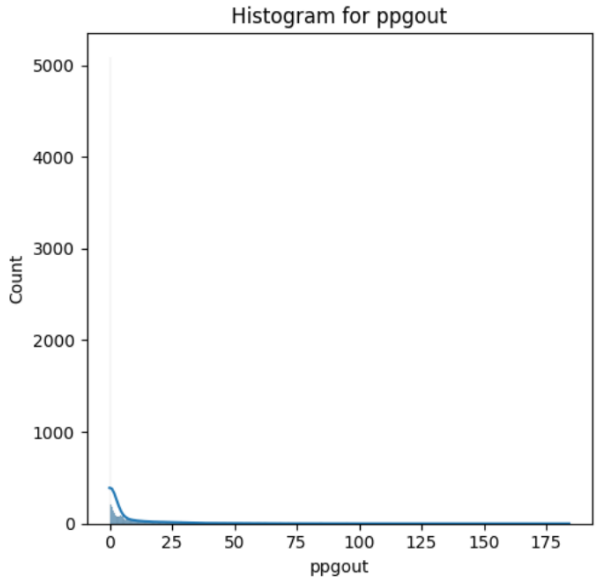

Boxplot for wchar
1e6
wchar
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00

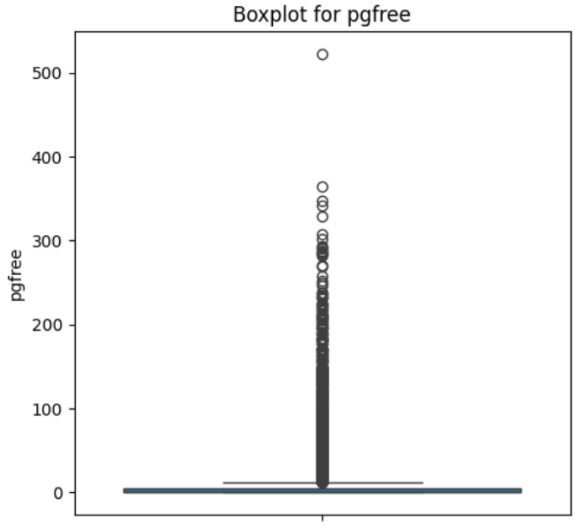

Histogram for wchar
Count
wchar
1e6
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
0
200
400
600
800
1000

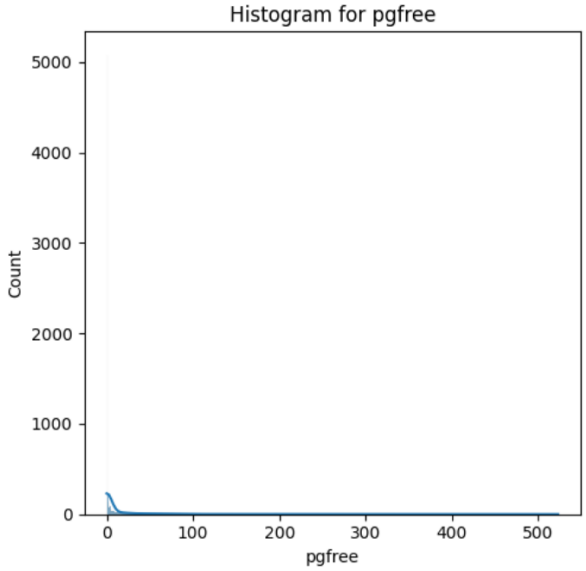

Boxplot for pgout
pgout
0
10
20
30
40
50
60
70
80

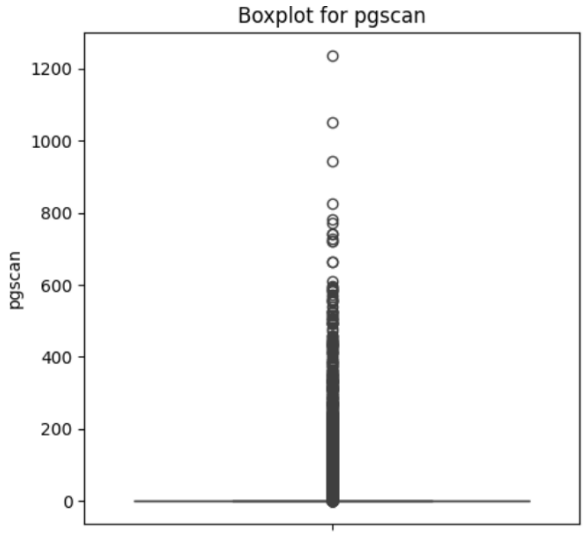

Histogram for pgout
Count
pgout
0
1000
2000
3000
4000
5000
0
10
20
30
40
50
60
70
80

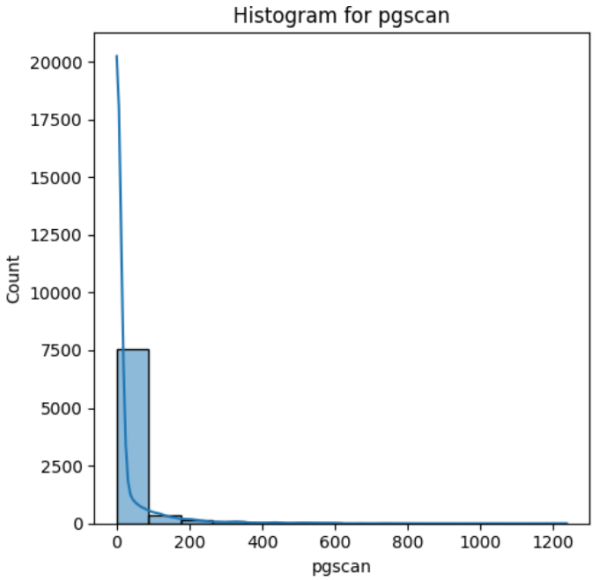

Boxplot for ppgout
ppgout
0
25
50
75
100
125
150
175

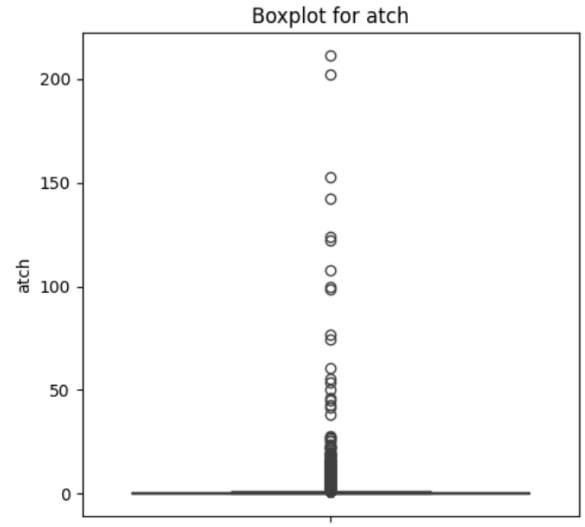

Histogram for ppgout
Count
ppgout
0
1000
2000
3000
4000
5000
0
25
50
75
100
125
150
175

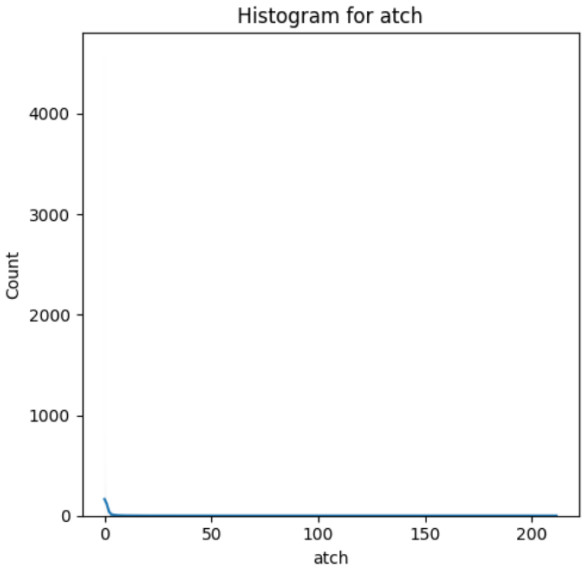

Boxplot for pgfree
pgfree
0
100
200
300
400
500

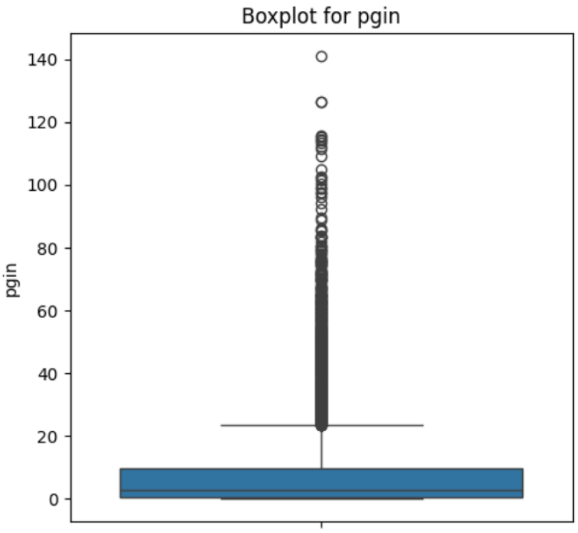

Histogram for pgfree
Count
pgfree
0
1000
2000
3000
4000
5000
0
100
200
300
400
500

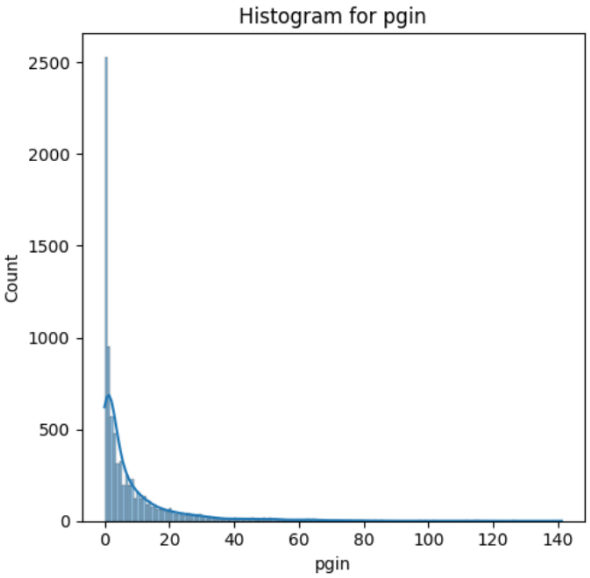

Boxplot for pgscan
pgscan
0
200
400
600
800
1000
1200

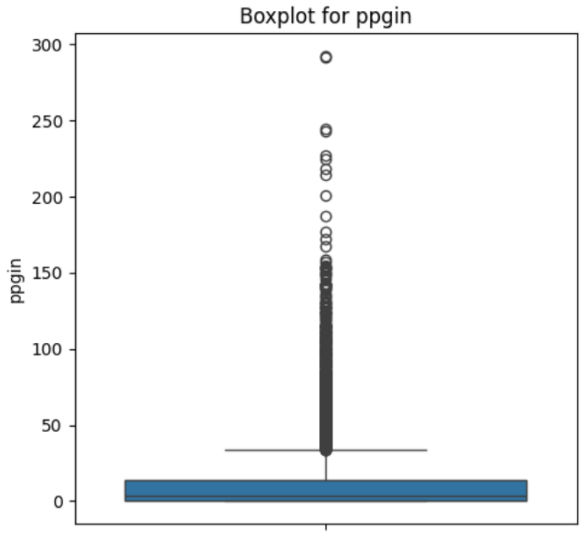

Histogram for pgscan
Count
pgscan
0
2500
5000
7500
10000
12500
15000
17500
20000
0
200
400
600
800
1000
1200

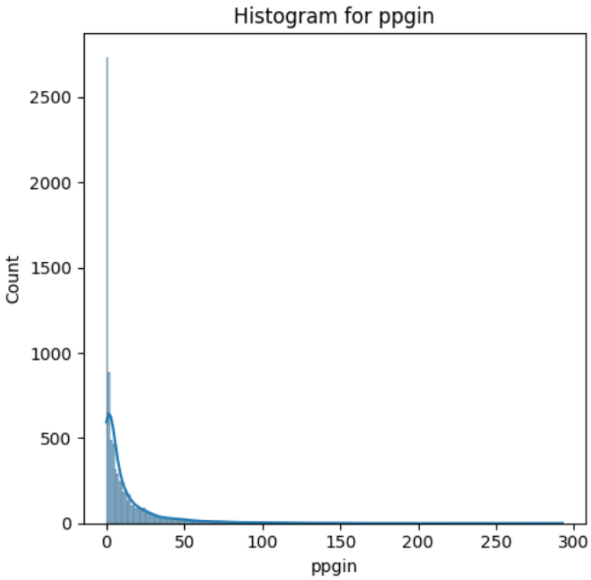

Boxplot for atch
atch
0
50
100
150
200

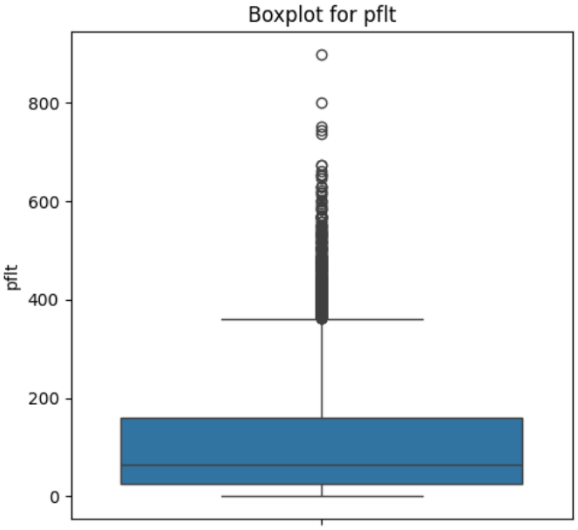

Histogram for atch
Count
atch
0
1000
2000
3000
4000
0
50
100
150
200

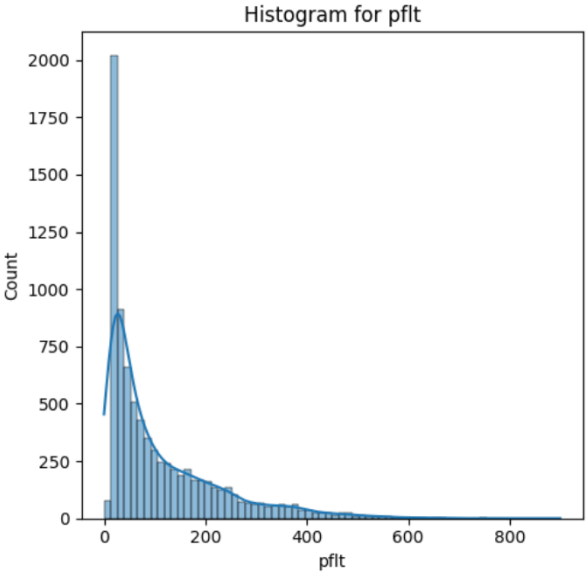

Boxplot for pgin
pgin
0
20
40
60
80
100
120
140

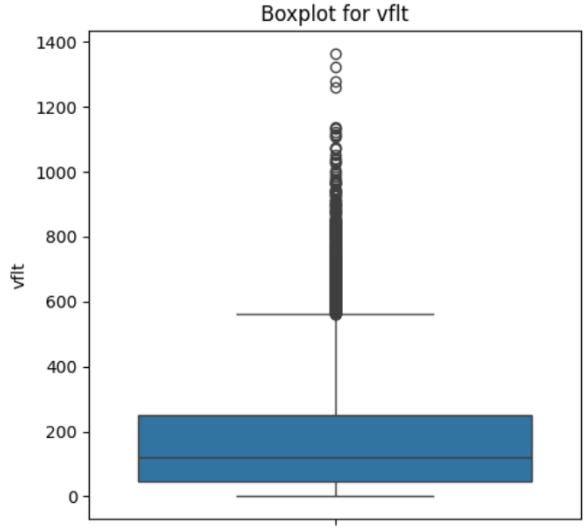

Histogram for pgin
Count
pgin
0
500
1000
1500
2000
2500
0
20
40
60
80
100
120
140

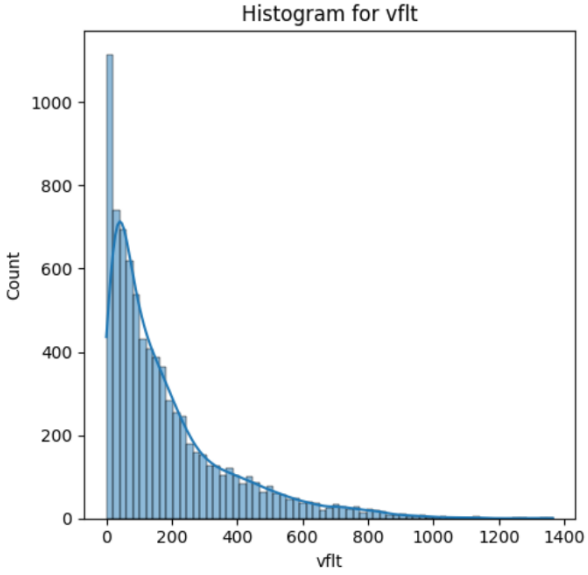

Boxplot for ppgin
ppgin
300
250
200
150
100
50
0


Histogram for ppgin
Count
2500
2000
1500
1000
500
0
0
50
100
150
200
250
300
ppgin


Boxplot for pflt
pflt
800
600
400
200
0


Histogram for pflt
Count
2000
1750
1500
1250
1000
750
500
250
0
0
200
400
600
800
pflt


Boxplot for vflt
vflt
1400
1200
1000
800
600
400
200
0


Histogram for vflt
Count
1000
800
600
400
200
0
0
200
400
600
800
1000
1200
1400
vflt

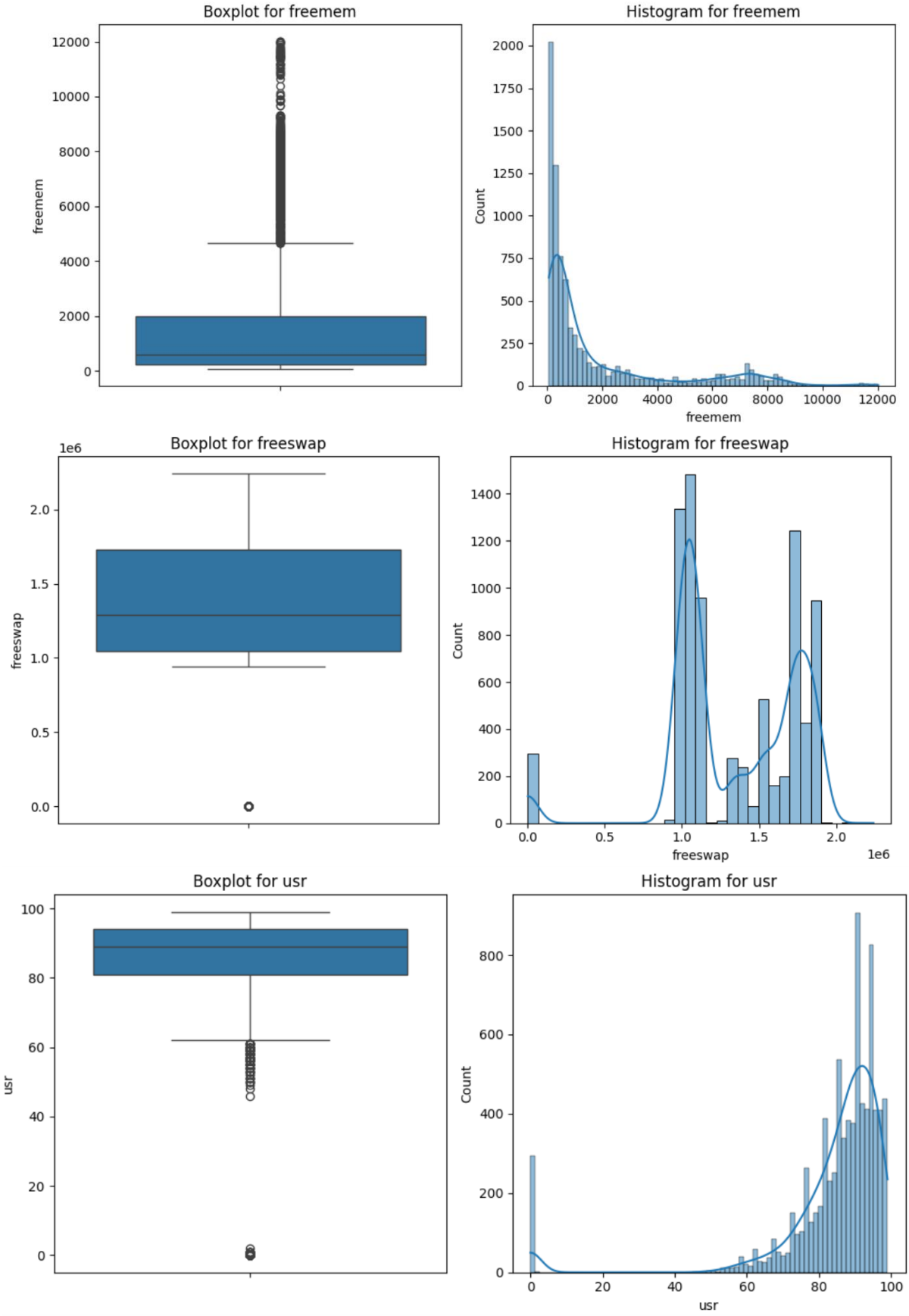
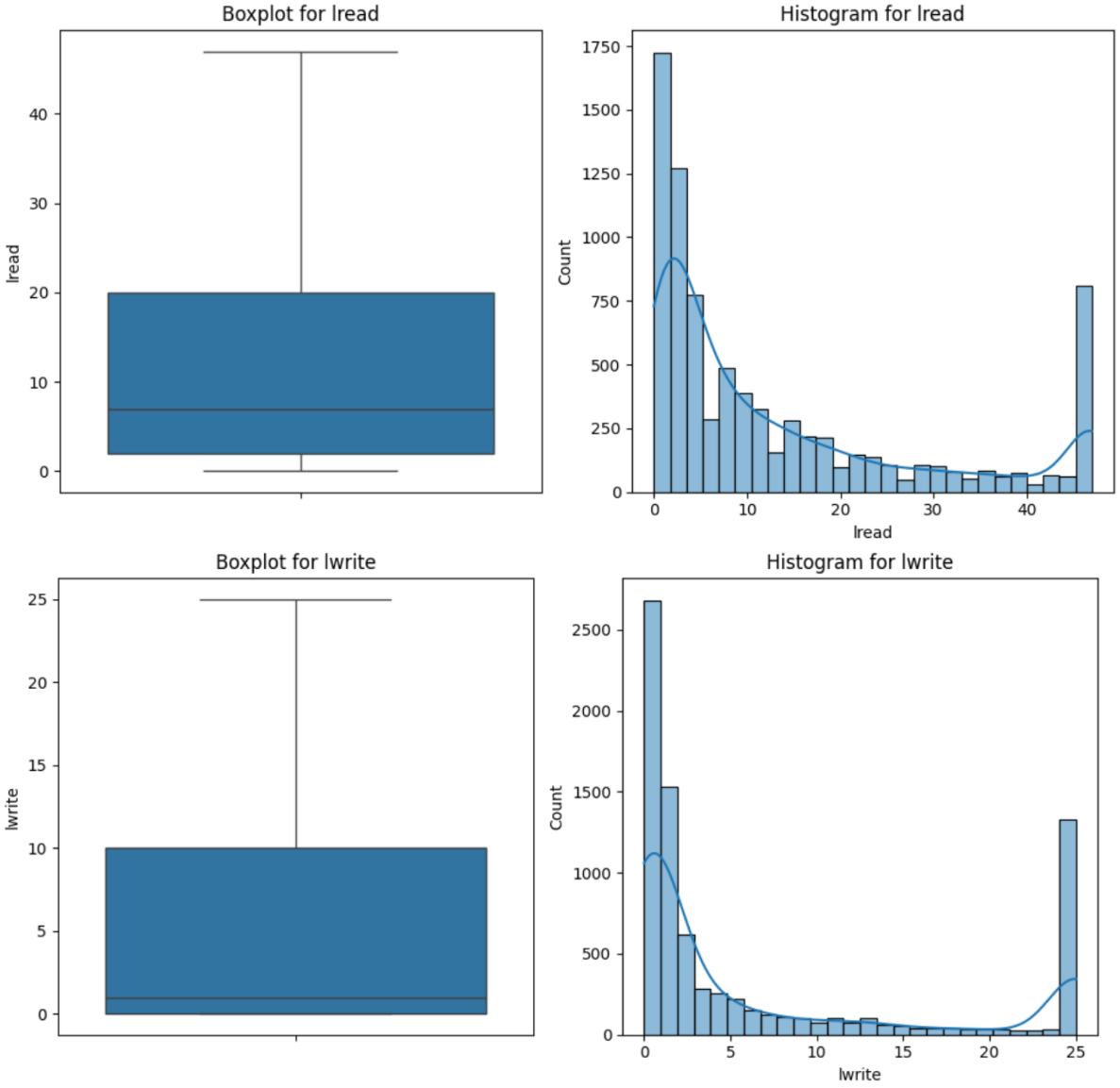
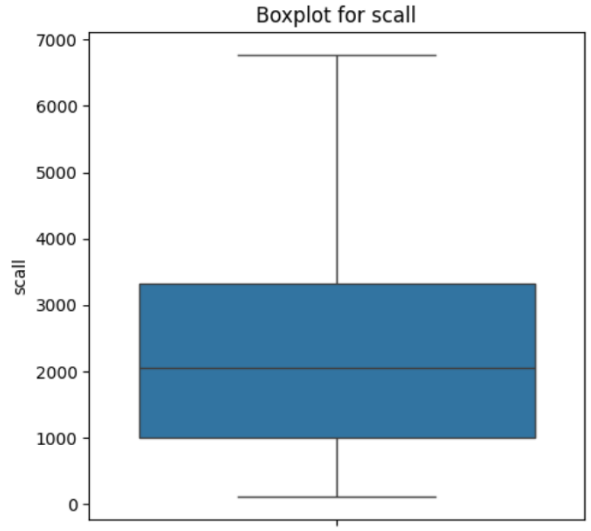
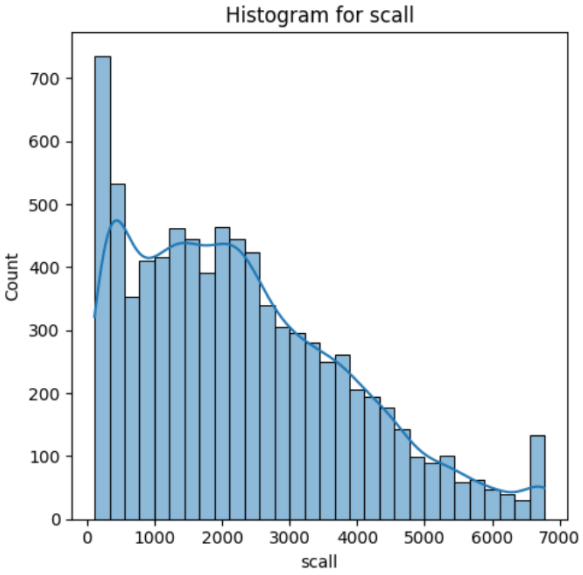


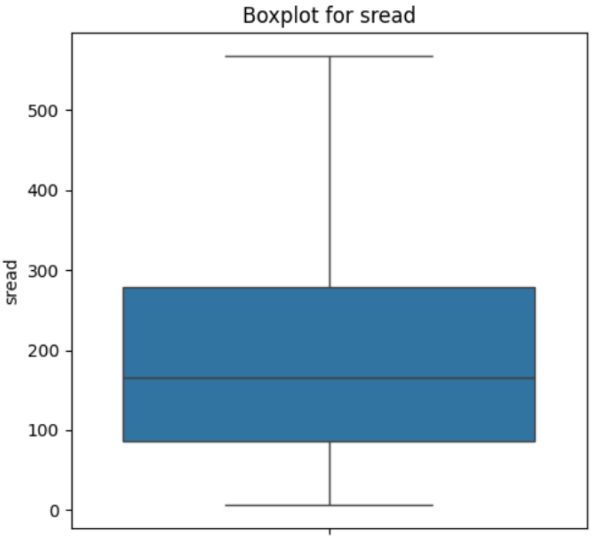
Fig-2 Boxplot Before Outlier Treatment

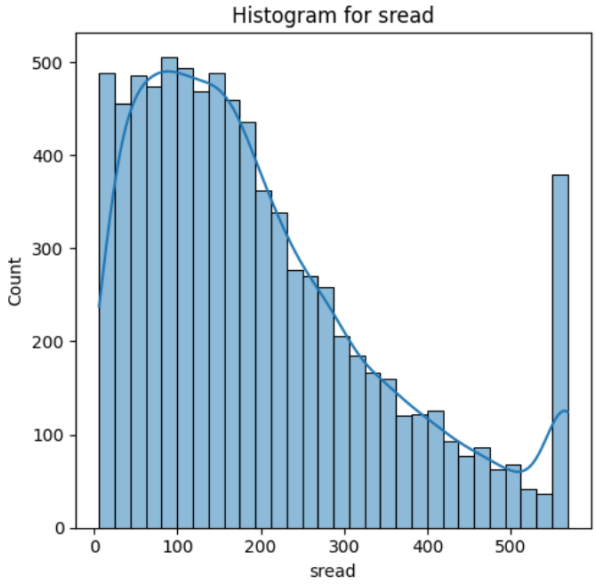
Now, we will perform outlier treatment using q1 and q3 to find IQR and using that to find out the upper and lower limit whiskers and finally bring all those outlier's points to these whiskers and obtain the below boxplots indicated that treatment has been done properly.

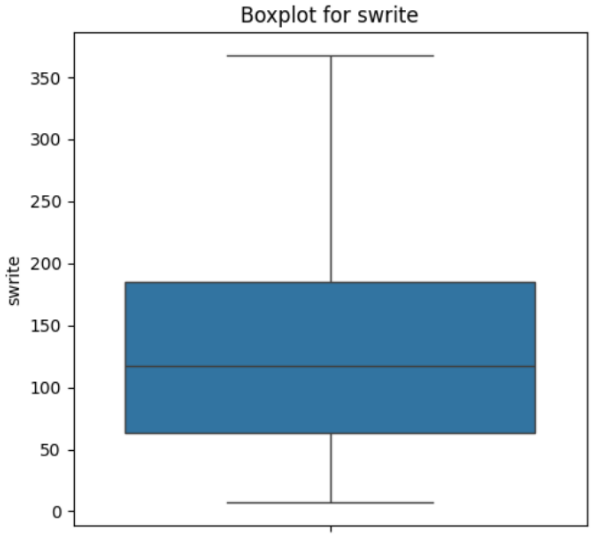


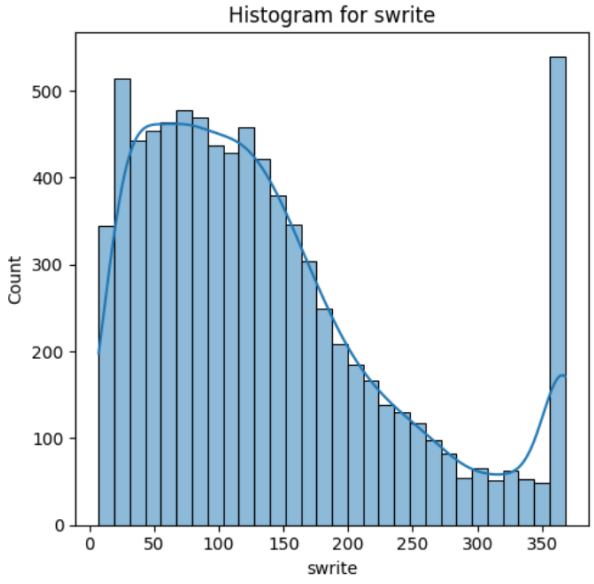

Boxplot for scall
scall
0
1000
2000
3000
4000
5000
6000
7000

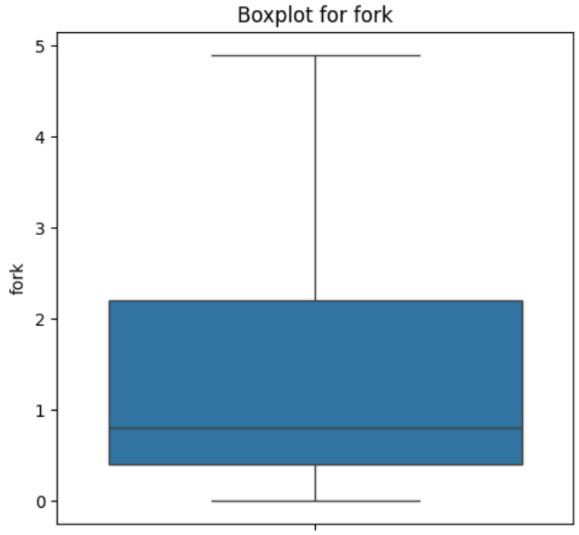

Histogram for scall
Count
scall
0
100
200
300
400
500
600
700
0
1000
2000
3000
4000
5000
6000
7000

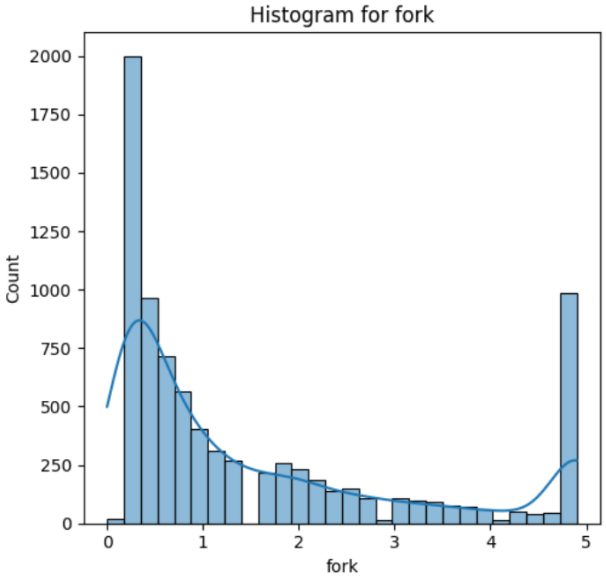

Boxplot for sread
sread
0
100
200
300
400
500

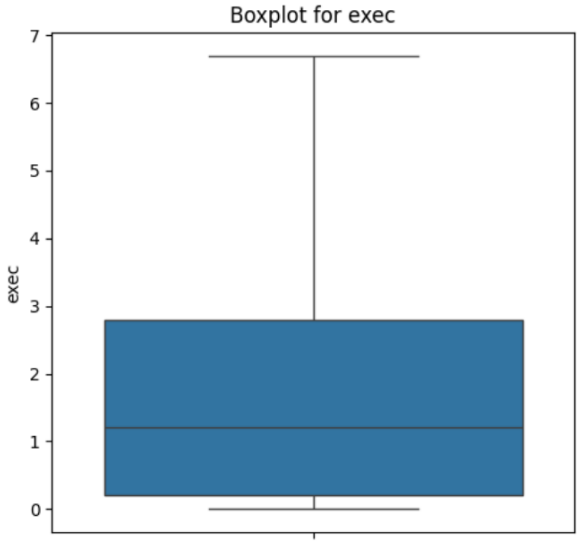

Histogram for sread
Count
sread
0
100
200
300
400
500
0
100
200
300
400
500

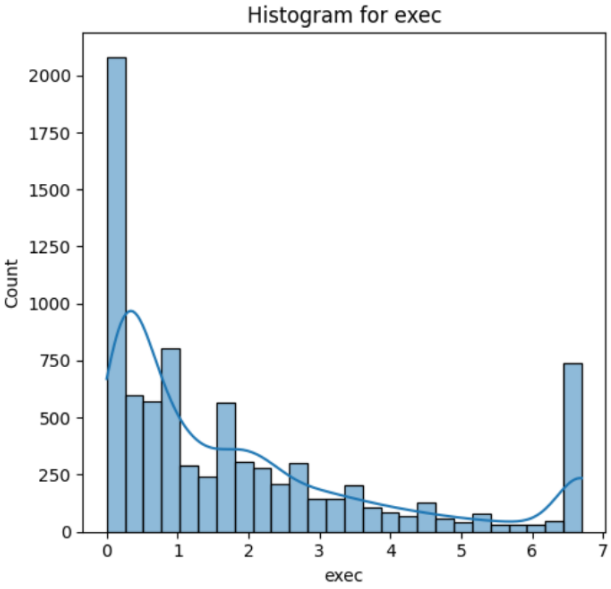

Boxplot for swrite
swrite
0
50
100
150
200
250
300
350

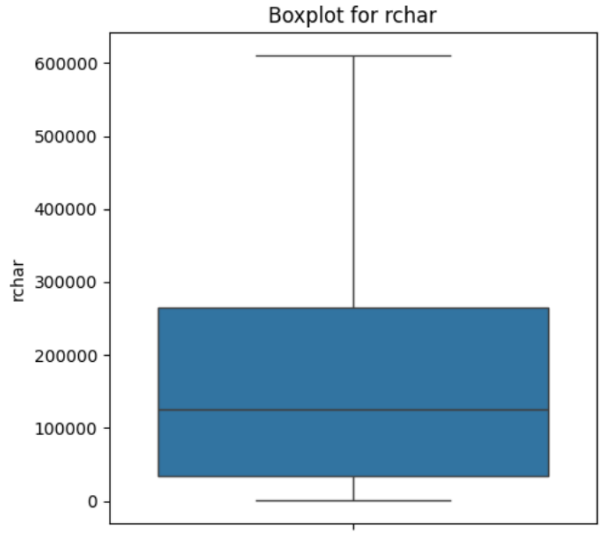

Histogram for swrite
Count
swrite
0
100
200
300
400
500
0
50
100
150
200
250
300
350

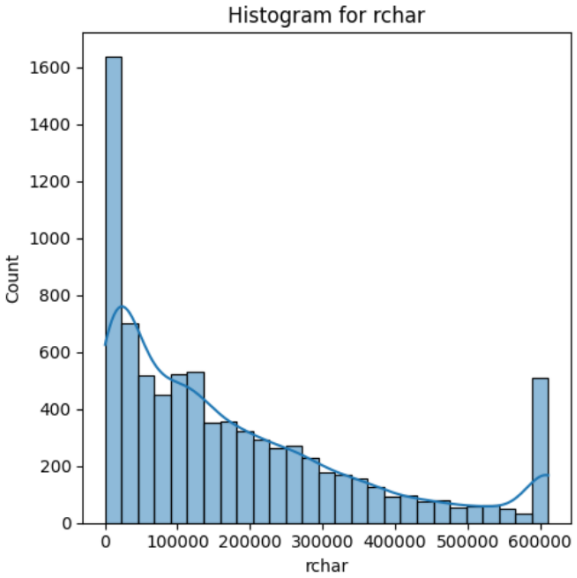

Boxplot for fork
fork
0
1
2
3
4
5

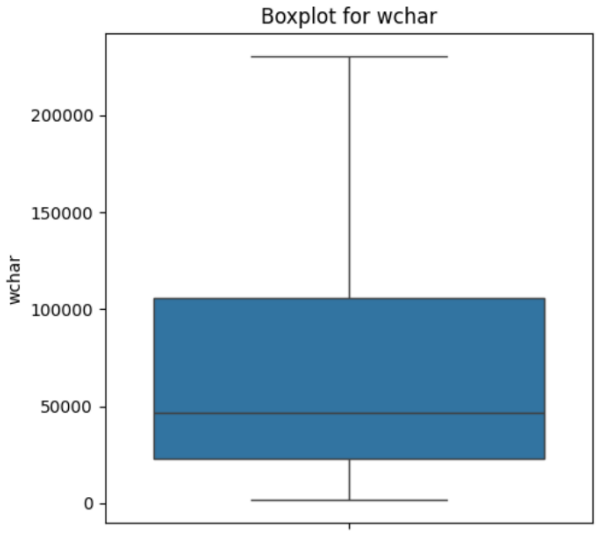

Histogram for fork
Count
fork
0
250
500
750
1000
1250
1500
1750
2000
0
1
2
3
4
5

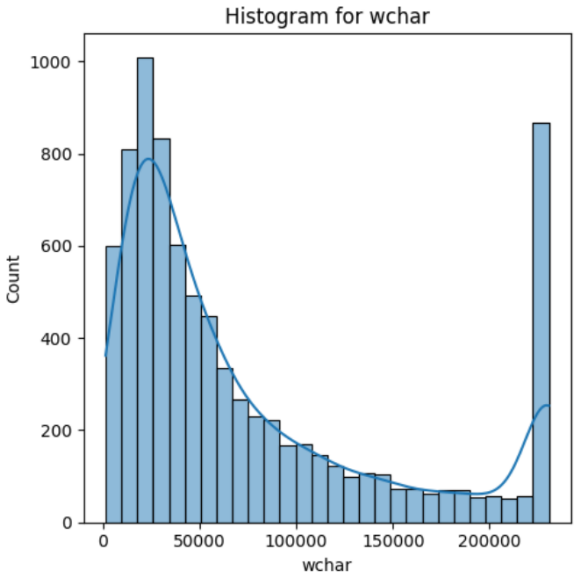

Boxplot for exec
exec
0
1
2
3
4
5
6
7

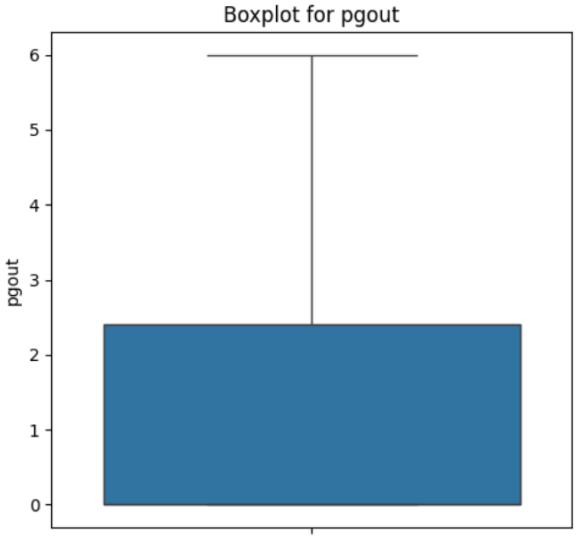

Histogram for exec
Count
exec
0
250
500
750
1000
1250
1500
1750
2000
0
1
2
3
4
5
6
7

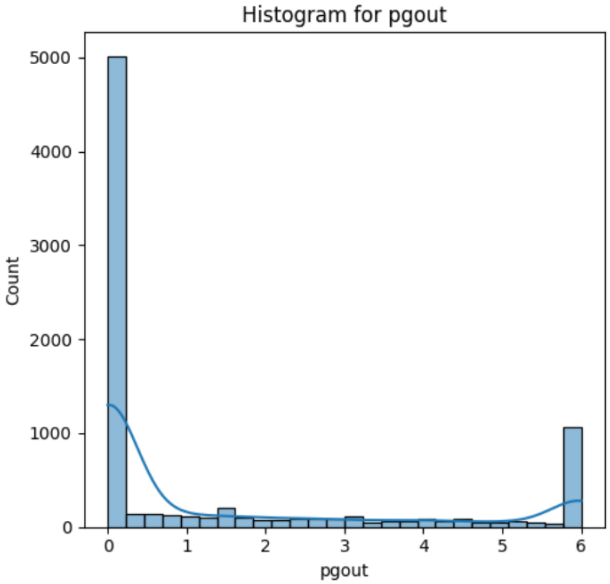

Boxplot for rchar
rchar
0
100000
200000
300000
400000
500000
600000

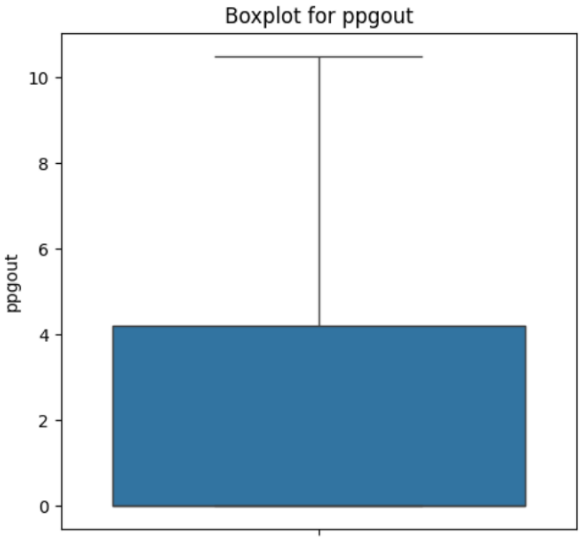

Histogram for rchar
Count
rchar
0
200
400
600
800
1000
1200
1400
1600
0
100000
200000
300000
400000
500000
600000

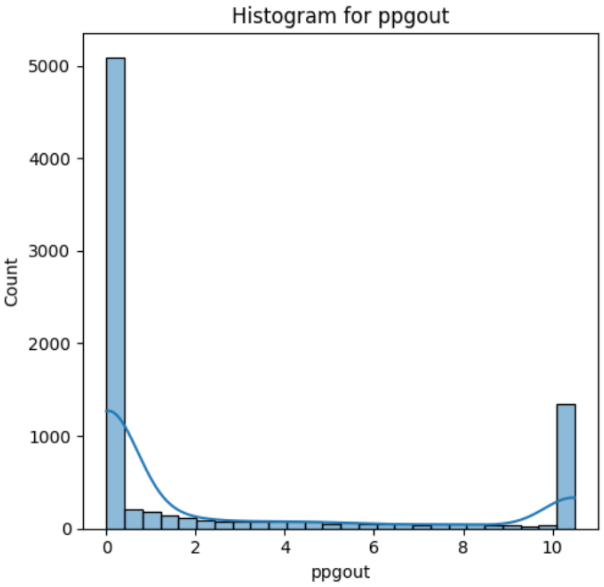

Boxplot for wchar
wchar
200000
150000
100000
50000
0

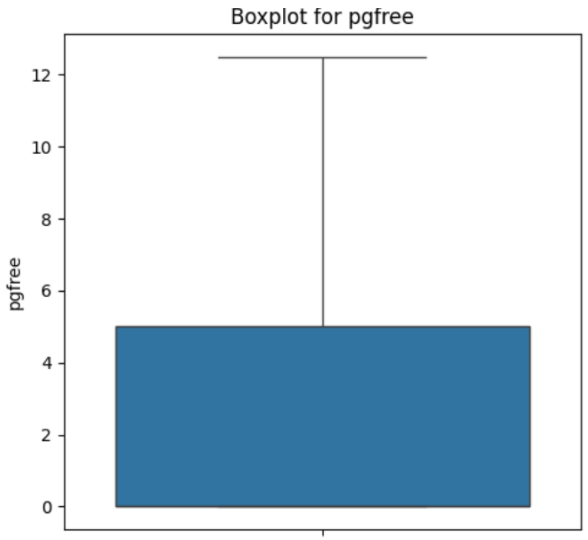

Histogram for wchar
Count
wchar
1000
800
600
400
200
0
0
50000
100000
150000
200000

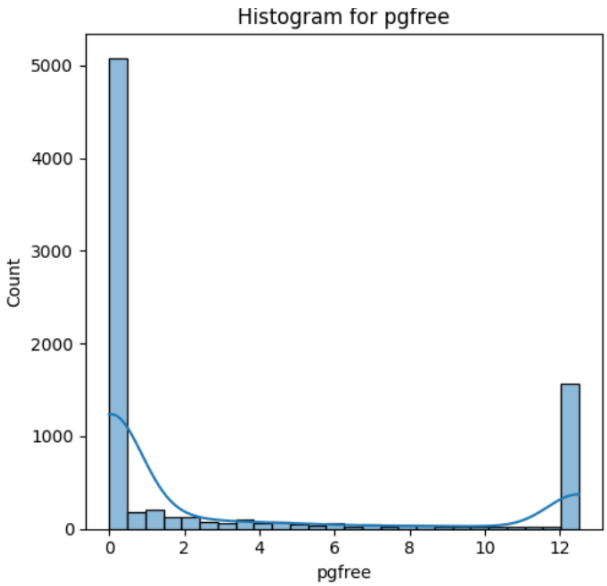

Boxplot for pgout
pgout
6
5
4
3
2
1
0

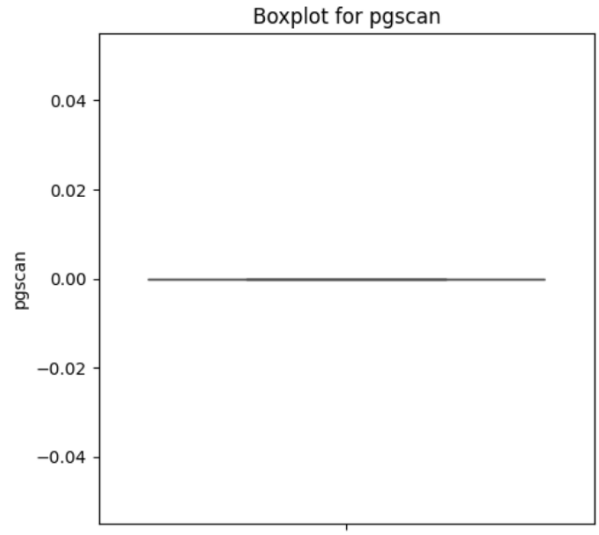

Histogram for pgout
Count
pgout
5000
4000
3000
2000
1000
0
0
1
2
3
4
5
6

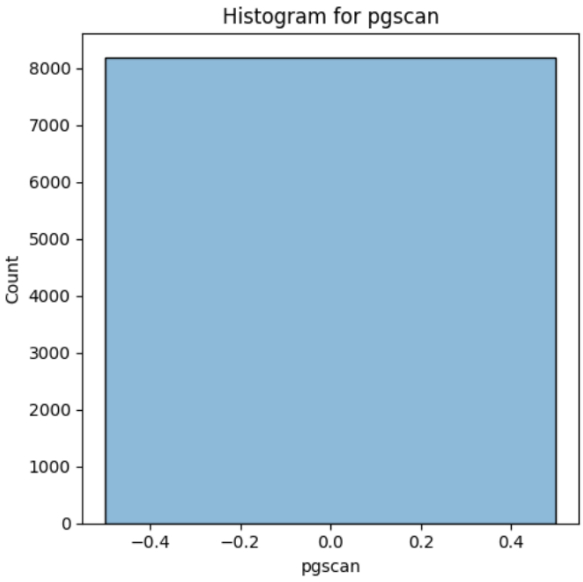

Boxplot for ppgout
ppgout
10
8
6
4
2
0

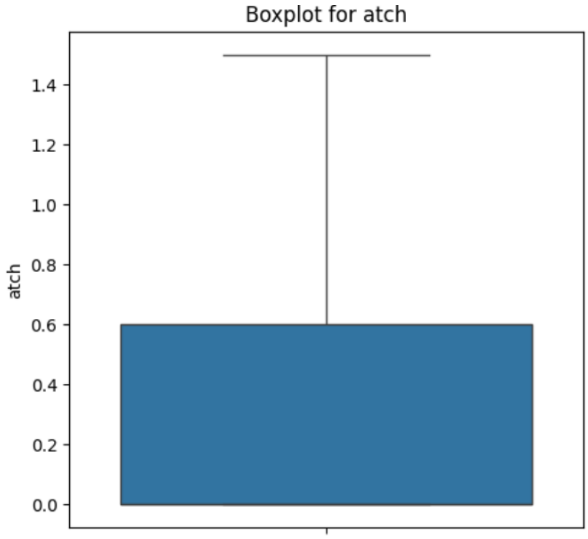

Histogram for ppgout
Count
ppgout
5000
4000
3000
2000
1000
0
0
2
4
6
8
10

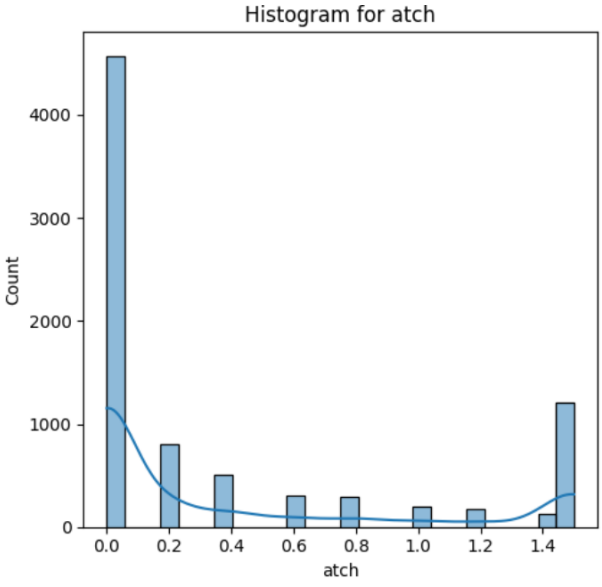

Boxplot for pgfree
pgfree
0
2
4
6
8
10
12

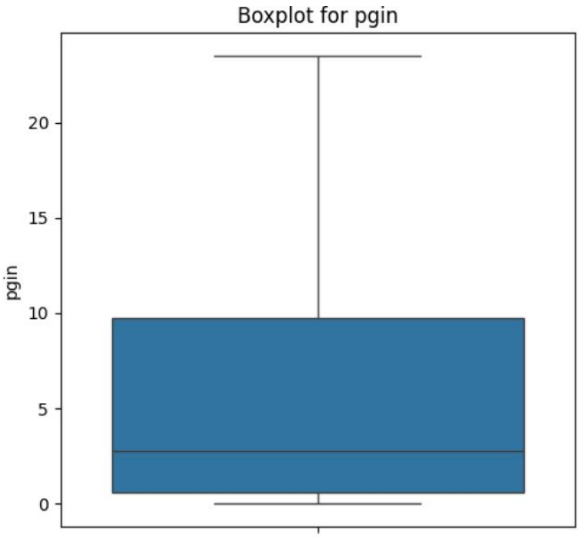

Histogram for pgfree
Count
0
1000
2000
3000
4000
5000
0
2
4
6
8
10
12
pgfree

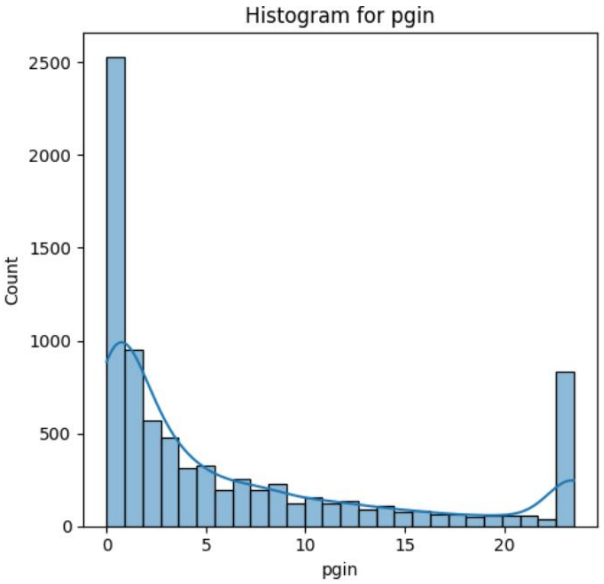

Boxplot for pgscan
pgscan
0.04
0.02
0.00
−0.02
−0.04

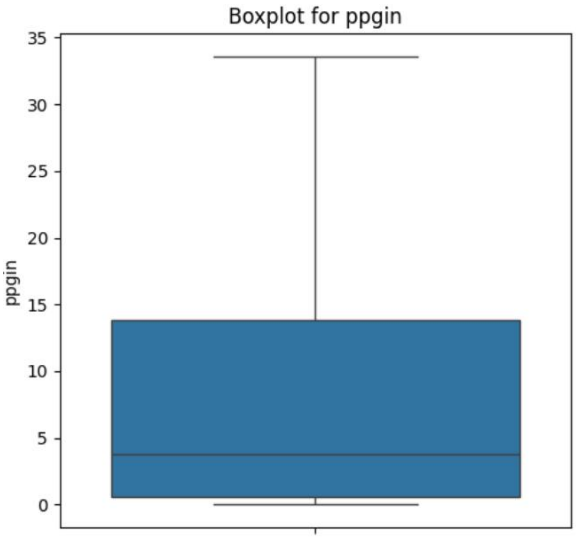

Histogram for pgscan
Count
0
1000
2000
3000
4000
5000
6000
7000
8000
−0.4
−0.2
0.0
0.2
0.4
pgscan

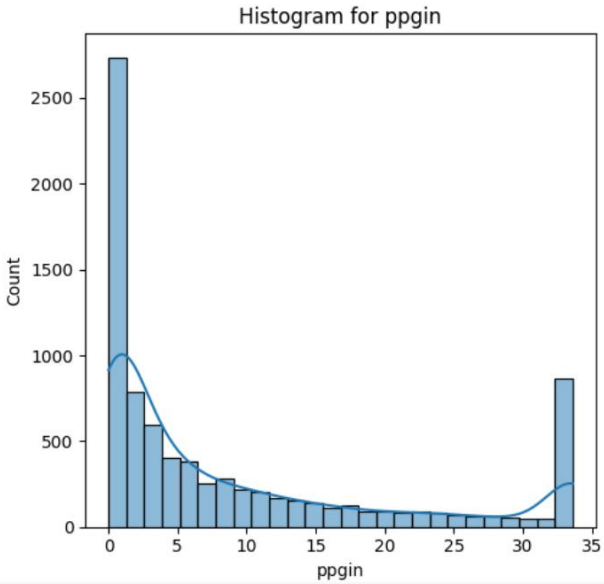

Boxplot for atch
atch
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4

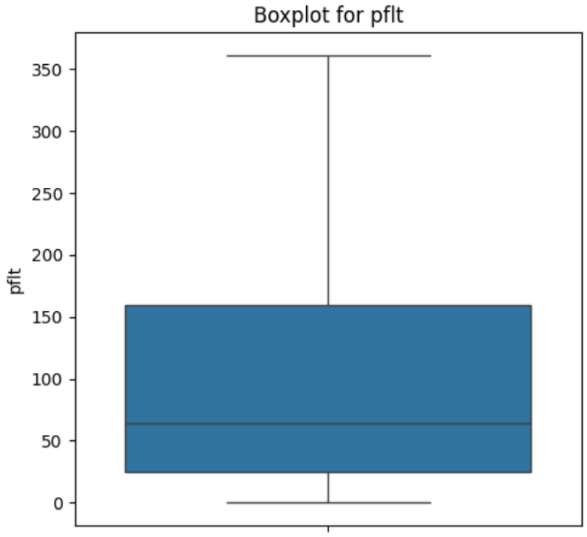

Histogram for atch
Count
0
1000
2000
3000
4000
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
atch

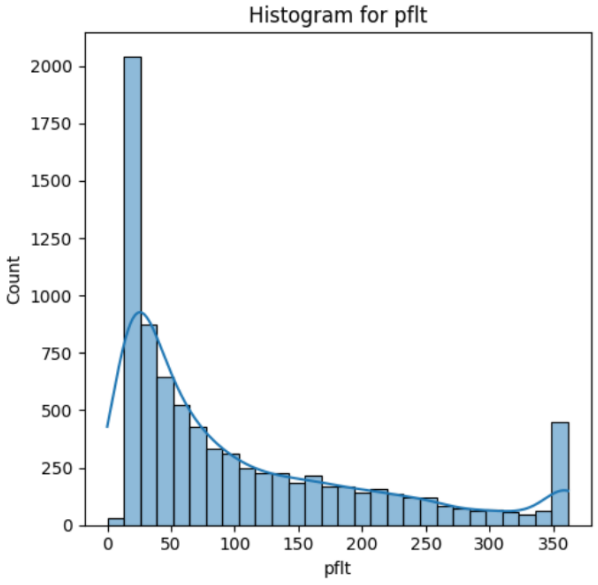

Boxplot for pgin
pgin
0
5
10
15
20

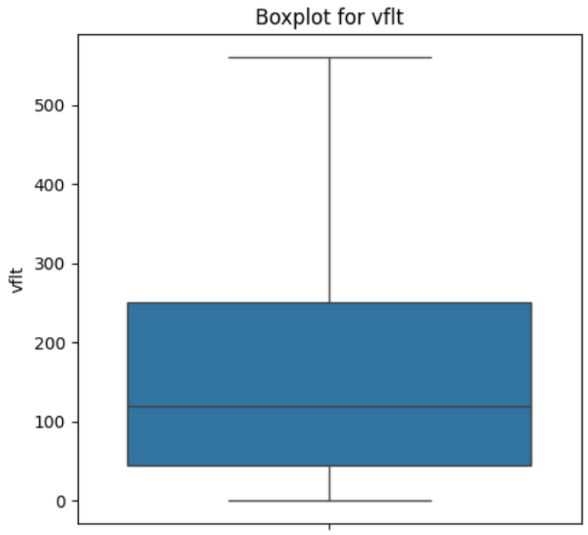

Histogram for pgin
Count
pgin
0
500
1000
1500
2000
2500
0
5
10
15
20

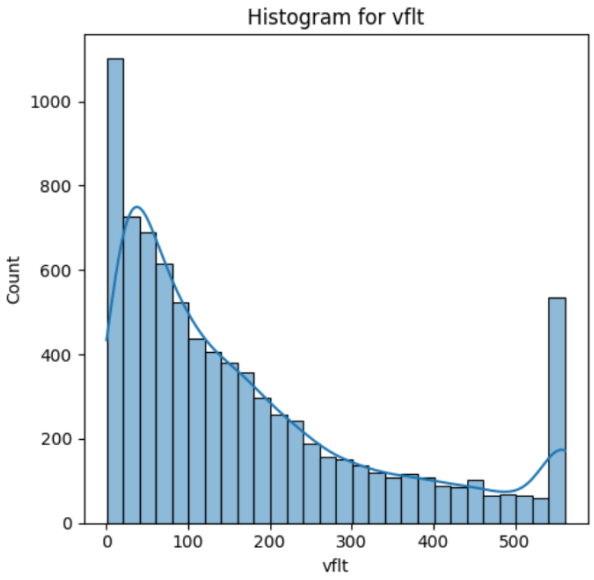

Boxplot for ppgin
ppgin
0
5
10
15
20
25
30
35

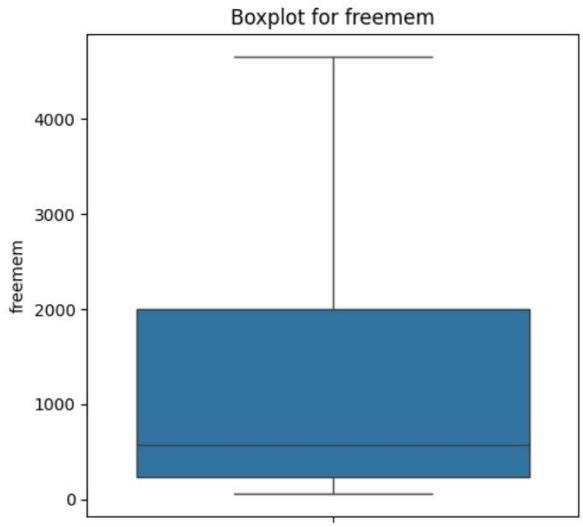

Histogram for ppgin
Count
ppgin
0
500
1000
1500
2000
2500
0
5
10
15
20
25
30
35

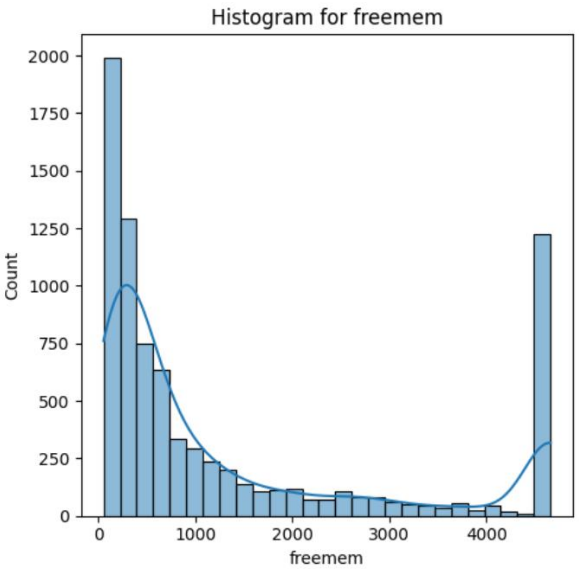

Boxplot for pflt
pflt
0
50
100
150
200
250
300
350


Histogram for pflt
Count
pflt
0
250
500
750
1000
1250
1500
1750
2000
0
50
100
150
200
250
300
350


Boxplot for vflt
vflt
0
100
200
300
400
500


Histogram for vflt
Count
vflt
0
200
400
600
800
1000
0
100
200
300
400
500


Boxplot for freemem
freemem
0
1000
2000
3000
4000


Histogram for freemem
Count
freemem
0
250
500
750
1000
1250
1500
1750
2000
0
1000
2000
3000
4000

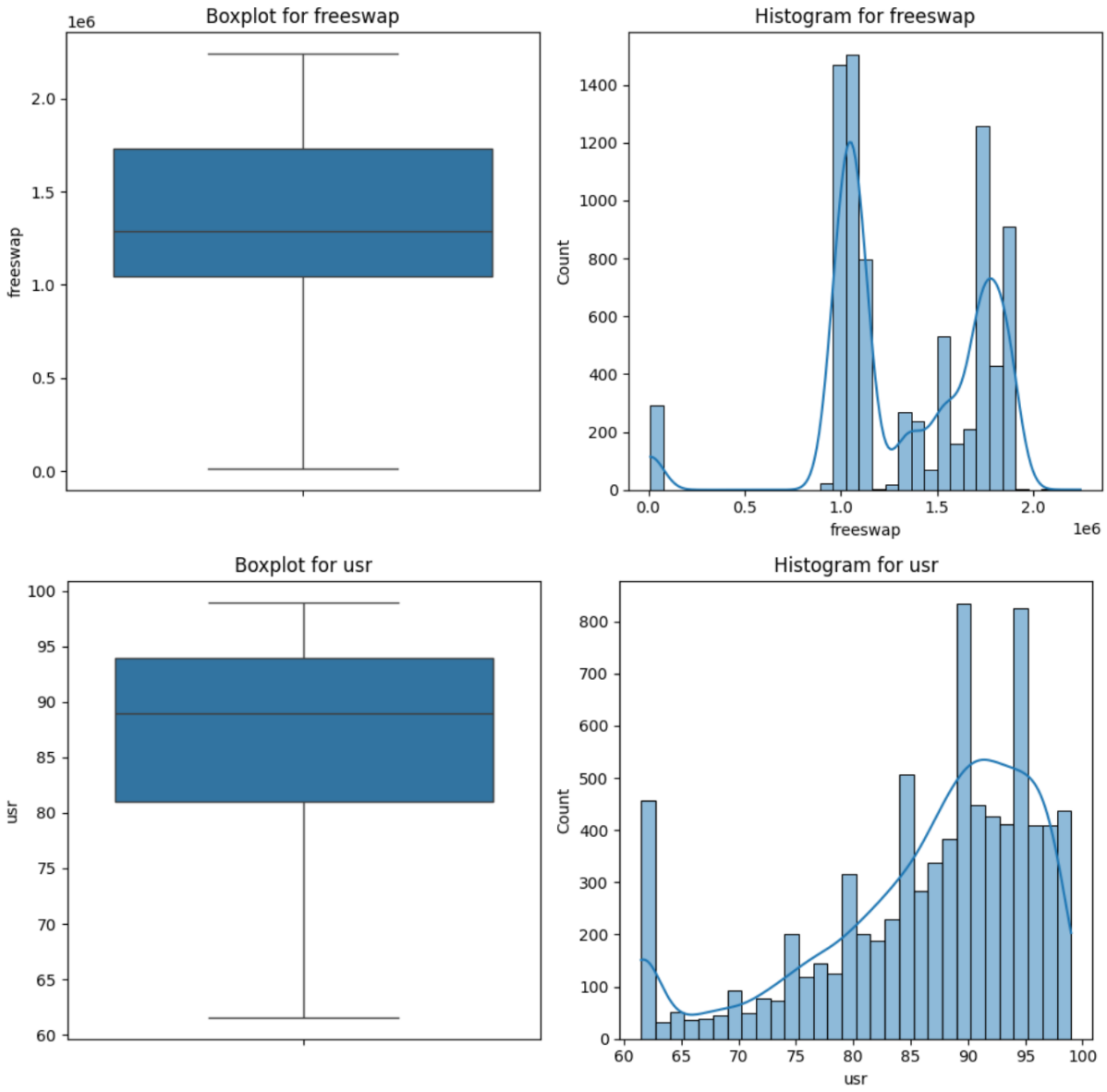


Fig-3 Boxplot After Outlier Treatment

From the above boxplots after the outlier treatment, we can drop the variable column named "pgscan" from our dataset permanently as it contains nothing but zero in its entire column.

Observation:

- It's evident that normal distribution isn't characteristic of any column within the dataset.
- A rightward skewness is prevalent across most columns, indicating outliers are present.
- The dependent variable 'Usr' exhibits a skewness towards the left.

Now, we will perform Multi-variate Analysis on the numerical data to see and find any correlations between the variables and find any hidden pattern the below figure shows the pair plot, heatmap.

From the below figure-4 and figure-5 we can say the following observations

- A comparison between the read operations per second and the CPUs operating in user mode reveals two distinct patterns.
- It's noted that with an increase in write operations, a mere 2% of CPU activity is dedicated to user mode.
- This suggests a trend where elevated read/write operations correlate with a decrease in CPU activity in user mode.

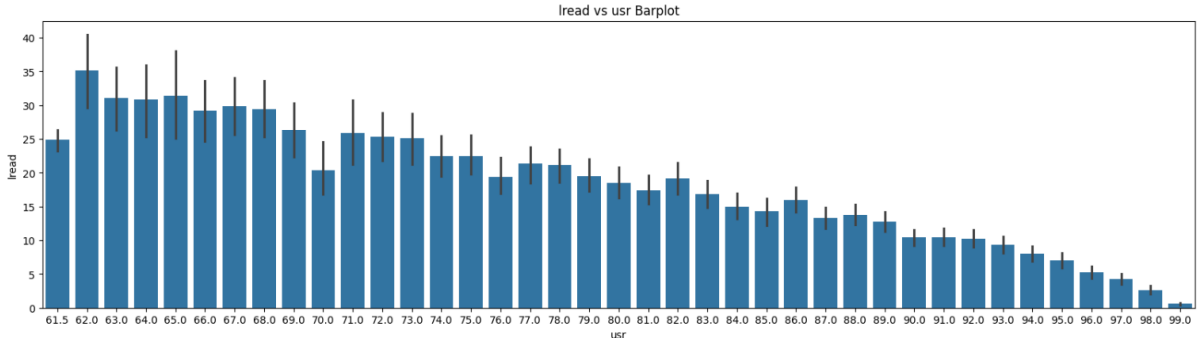


Fig-4 Bar plot for lread vs usr

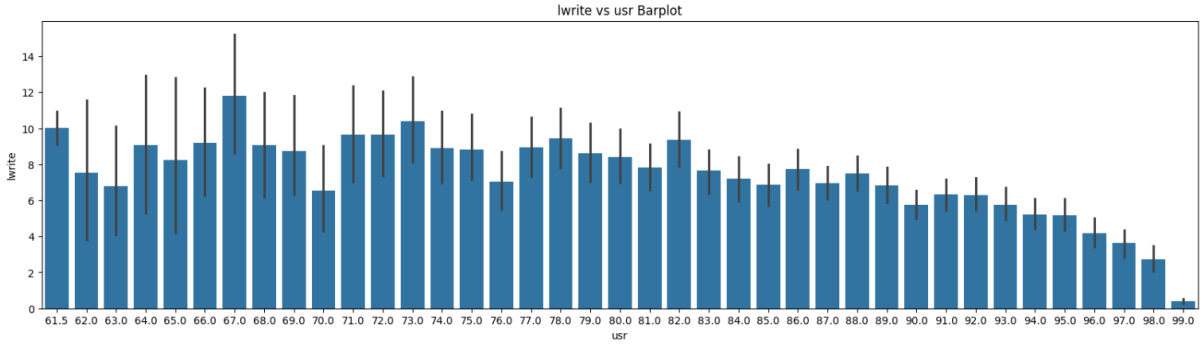


Fig-5 Bar plot for lwrite vs usr

Now we, we create a pair plot and a heatmap for numerical variables using seaborn library as shown below figure-6 and figure-7.

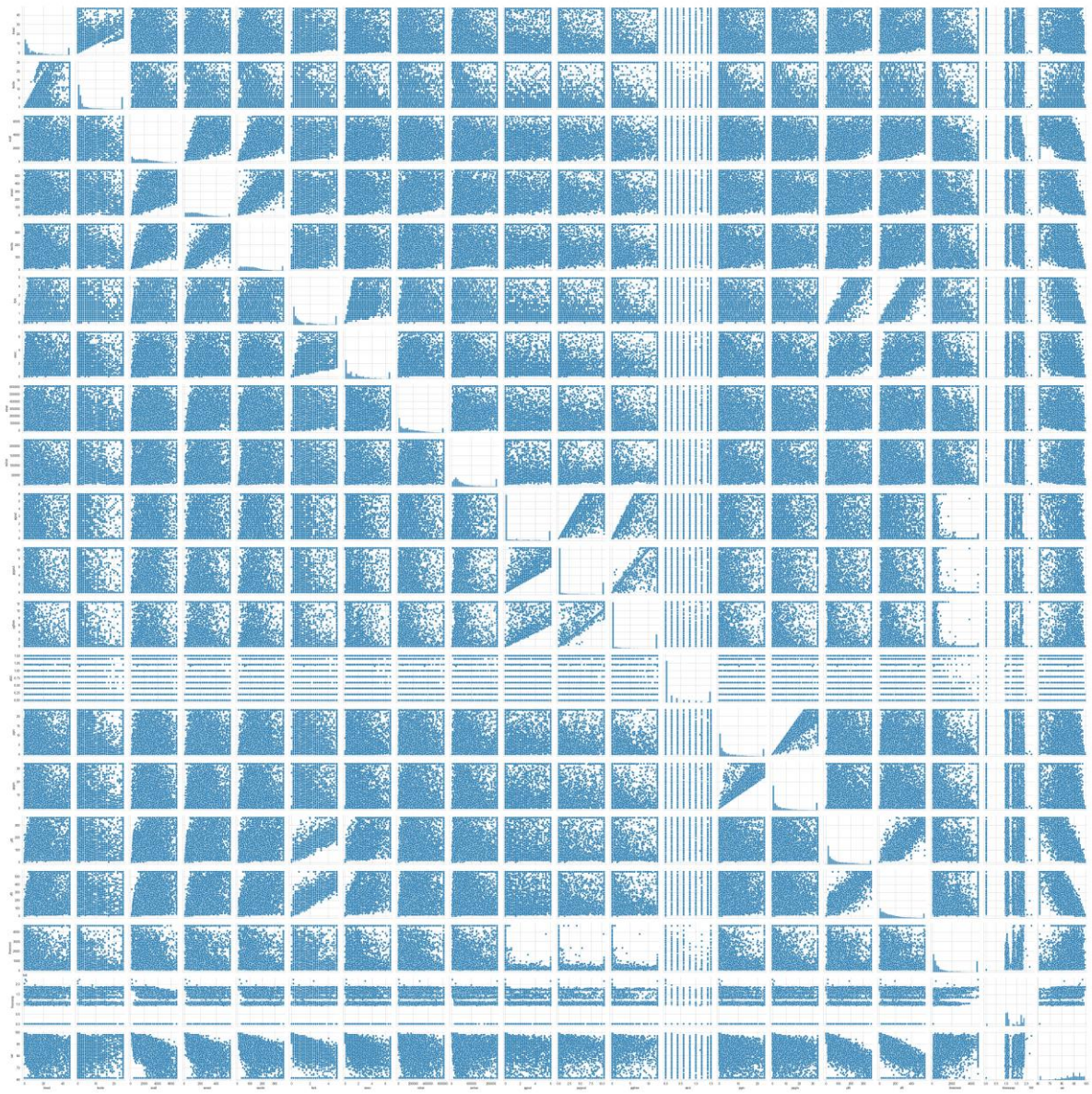


Fig-6 Pair plot for Numerical variable

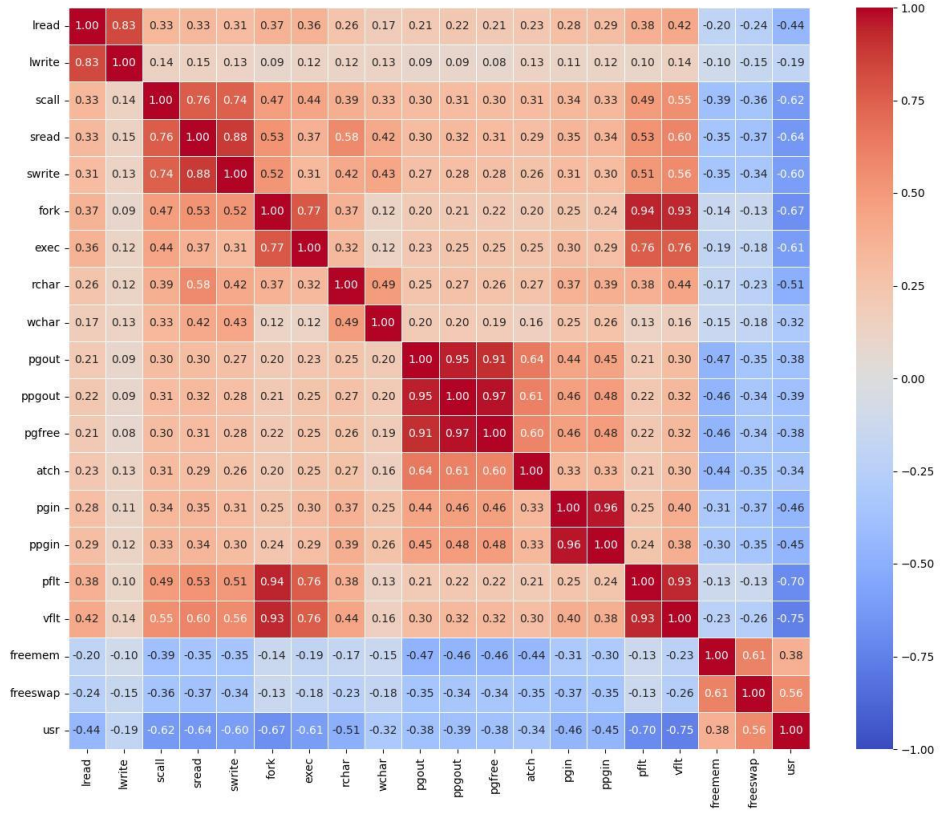


Fig-7 Heatmap for Numerical variable

From the Multivariate Analysis on the numerical dataset, we can observe the following things: -

1. The variables exec, Pfit, and Vfit demonstrate a strong correlation with fork, exhibiting correlation coefficients of 0.76, 0.93, and 0.94 respectively.
2. There's a notable correlation between swrite and sread, with a coefficient of 0.88.
3. The analysis highlights evident correlations within the data.
4. The page fault metrics, Pfit and vflt, show a significant correlation with the fork variable.
5. A robust positive correlation is observed between the outcome variable usr and the predictive variables freemem and freeswap.
6. A linear relationship is apparent between vflt, pflt, and fork; an uptick in fork processes typically leads to an increase in page faults.
7. In parallel, the frequency of page out requests per second is closely linked to the volume of pages being paged out every second.

Now, we will combine the two datasets into one and apply data encoding on "runsqzs" categorical variable and convert it into numerical variable.

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	...	pgfree	atch	pgin	ppgin	pflt	vflt	freemem	freeswap	usr
0	1.0	0.0	2147.0	79.0	68.0	0.2	0.2	40671.0	53995.0	0.0	...	0.0	0.0	1.6	2.6	16.00	26.40	4659.125	1730946.0	95.0
1	0.0	0.0	170.0	18.0	21.0	0.2	0.2	448.0	8385.0	0.0	...	0.0	0.0	0.0	0.0	15.63	16.83	4659.125	1869002.0	97.0
2	15.0	3.0	2162.0	159.0	119.0	2.0	2.4	125473.5	31950.0	0.0	...	0.0	1.2	6.0	9.4	150.20	220.20	702.000	1021237.0	87.0
3	0.0	0.0	160.0	12.0	16.0	0.2	0.2	125473.5	8670.0	0.0	...	0.0	0.0	0.2	0.2	15.60	16.80	4659.125	1863704.0	98.0
4	5.0	1.0	330.0	39.0	38.0	0.4	0.4	125473.5	12185.0	0.0	...	0.0	0.0	1.0	1.2	37.80	47.60	633.000	1760253.0	90.0

runqsz_Not_CPU_Bound
0
1
1
1
1

Table-8 Combined and Encoded dataset

And finally we will check for the presence zero value in our dataset. That is the value zero is enter in the variable columns as shown below figure.

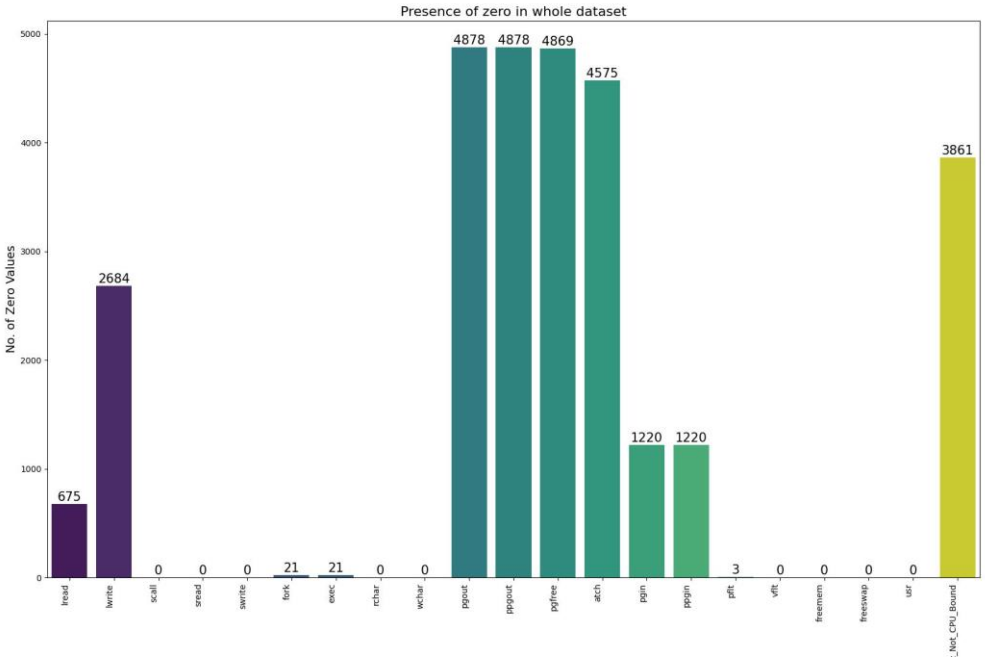


Fig-8 Presence of Zero in Dataset

Now, we will calculated the percentage of zero presence in our dataset as shown in the below table.

lread	8.24
lwrite	32.76
scall	0.00
sread	0.00
swrite	0.00
fork	0.26
exec	0.26
rchar	0.00
wchar	0.00
pgout	59.55
ppgout	59.55
pgfree	59.44
atch	55.85
pgin	14.89
ppgin	14.89
pflt	0.04
vflt	0.00
freemem	0.00
freeswap	0.00
usr	0.00
runqsz_Not_CPU_Bound	47.13

Table-9 Percentage of zero presence

From the above table and figure we will drop the variables that has more than 50% of zero value presence in data and we obtained the final dataset that will be used for linear regression.

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgin	ppgin	pflt	vflt	freemem	freeswap	usr	runqsz_Not_CPU_Bound
0	1.0	0.0	2147.0	79.0	68.0	0.2	0.2	40671.0	53995.0	1.6	2.6	16.00	26.40	4659.125	1730946.0	95.0	0
1	0.0	0.0	170.0	18.0	21.0	0.2	0.2	448.0	8385.0	0.0	0.0	15.63	16.83	4659.125	1869002.0	97.0	1
2	15.0	3.0	2162.0	159.0	119.0	2.0	2.4	125473.5	31950.0	6.0	9.4	150.20	220.20	702.000	1021237.0	87.0	1
3	0.0	0.0	160.0	12.0	16.0	0.2	0.2	125473.5	8670.0	0.2	0.2	15.60	16.80	4659.125	1863704.0	98.0	1
4	5.0	1.0	330.0	39.0	38.0	0.4	0.4	125473.5	12185.0	1.0	1.2	37.80	47.60	633.000	1760253.0	90.0	1

Table-10 Final Dataset

Now, we will divide our final dataset into independent and dependent variables and save it in “X” for independent variables and “y” for dependent variable. After that we will split our this two dataset into training and testing dataset using train_test_split function where, 70% of data will be in training dataset and 30% will be in testing dataset with random state as 1.

1.2 Linear Regression using StatsModels:

Now, using olsmode we will fit the x_train and y_train into it and we obtain the summary as shown below.

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.793
Model:	OLS	Adj. R-squared:	0.793
Method:	Least Squares	F-statistic:	1371.
Date:	Sun, 09 Jun 2024	Prob (F-statistic):	0.00
Time:	11:08:38	Log-Likelihood:	-16697.
No. Observations:	5734	AIC:	3.343e+04
Df Residuals:	5717	BIC:	3.354e+04
Df Model:	16		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	84.0958	0.315	267.117	0.000	83.479	84.713
lread	-0.0629	0.009	-6.962	0.000	-0.081	-0.045
lwrite	0.0494	0.013	3.740	0.000	0.023	0.075
scall	-0.0007	6.32e-05	-10.490	0.000	-0.001	-0.001
sread	4.76e-05	0.001	0.047	0.963	-0.002	0.002
swrite	-0.0051	0.001	-3.558	0.000	-0.008	-0.002
fork	0.0447	0.132	0.338	0.735	-0.215	0.304
exec	-0.3141	0.052	-6.058	0.000	-0.416	-0.212
rchar	-4.957e-06	4.88e-07	-10.152	0.000	-5.91e-06	-4e-06
wchar	-6.262e-06	1.03e-06	-6.056	0.000	-8.29e-06	-4.23e-06
pgin	0.0211	0.029	0.737	0.461	-0.035	0.077
ppgin	-0.0806	0.020	-4.107	0.000	-0.119	-0.042
pflt	-0.0336	0.002	-16.849	0.000	-0.037	-0.030
vflt	-0.0060	0.001	-4.193	0.000	-0.009	-0.003
freemem	-0.0004	4.79e-05	-7.828	0.000	-0.000	-0.000
freeswap	8.78e-06	1.91e-07	45.993	0.000	8.41e-06	9.15e-06
runqsz_Not_CPU_Bound	1.5202	0.126	12.065	0.000	1.273	1.767

Omnibus:	1078.874	Durbin-Watson:	2.022
Prob(Omnibus):	0.000	Jarque-Bera (JB):	2284.270
Skew:	-1.102	Prob(JB):	0.00
Kurtosis:	5.169	Cond. No.	7.65e+06

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 7.65e+06. This might indicate that there are
strong multicollinearity or other numerical problems.

Table-11 Ols summary before VIF treatment

Table-11 presents preliminary findings from the initial model, which has not yet been validated against its assumptions. The model's R-squared and adjusted R-squared values are 0.793 and 0.793, respectively, suggesting that it accounts for 79.3% of the variance, which is indicative of a robust model. However, the presence of multicollinearity is apparent, necessitating further evaluation.

To address the multicollinearity, the Variance Inflation Factor (VIF) method was applied. Variables with a VIF exceeding 10 were identified and sequentially removed to observe any resultant changes in the R-squared and adjusted R-squared values.

Upon the exclusion of the “fork”,”sread” and “pgin” variable, which was contributing to multicollinearity, the model’s efficiency remained largely unaffected, with the R-squared and adjusted R-squared values remain unchanged. These three variables were incrementally removed. This process led to the development of a model exhibiting minimal or no multicollinearity.

OLS Regression Results

Dep. Variable:	usr	**R-squared:**	0.793
Model:	OLS	**Adj. R-squared:**	0.793
Method:	Least Squares	**F-statistic:**	1688.
Date:	Sun, 09 Jun 2024	**Prob (F-statistic):**	0.00
Time:	11:08:44	**Log-Likelihood:**	-16697.
No. Observations:	5734	**AIC:**	3.342e+04
Df Residuals:	5720	**BIC:**	3.352e+04
Df Model:	13		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	84.1010	0.311	270.687	0.000	83.492	84.710
lread	-0.0630	0.009	-6.984	0.000	-0.081	-0.045
lwrite	0.0492	0.013	3.739	0.000	0.023	0.075
scall	-0.0007	5.98e-05	-11.093	0.000	-0.001	-0.001
swrite	-0.0050	0.001	-4.685	0.000	-0.007	-0.003
exec	-0.3086	0.050	-6.216	0.000	-0.406	-0.211
rchar	-4.981e-06	4.38e-07	-11.381	0.000	-5.84e-06	-4.12e-06
wchar	-6.249e-06	1.03e-06	-6.066	0.000	-8.27e-06	-4.23e-06
ppgin	-0.0675	0.007	-10.295	0.000	-0.080	-0.055
pflt	-0.0334	0.002	-18.185	0.000	-0.037	-0.030
vflt	-0.0056	0.001	-4.529	0.000	-0.008	-0.003
freemem	-0.0004	4.79e-05	-7.820	0.000	-0.000	-0.000
freeswap	8.777e-06	1.89e-07	46.540	0.000	8.41e-06	9.15e-06
runqsz_Not_CPU_Bound	1.5197	0.126	12.065	0.000	1.273	1.767

Omnibus:	1079.031	**Durbin-Watson:**	2.022
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	2283.863
Skew:	-1.102	**Prob(JB):**	0.00
Kurtosis:	5.168	**Cond. No.**	7.55e+06

Table-12 Ols Model post VIF treatment

Fundamentals of Linear Regression: These fundamental principles are crucial prerequisites that must be satisfied prior to interpreting the model's estimates or employing the model for prediction purposes. In the context of Linear Regression, it's imperative to verify the adherence to the following principles:

→ Linearity
→ Independence
→ Homoscedasticity
→ Normal Distribution of Error Terms
→ Absence of Pronounced Multicollinearity

Evaluating Linearity and Independence:

Linearity is characterized by a direct proportional relationship between two variables, necessitating that predictor variables maintain a linear association with the outcome variable.

To assess linearity, one can plot the predicted values against the residuals. If the plot doesn't exhibit any discernible pattern and resembles a straight line, the model is considered linear. Conversely, if there's a discernible pattern, it suggests non-linearity in the model.

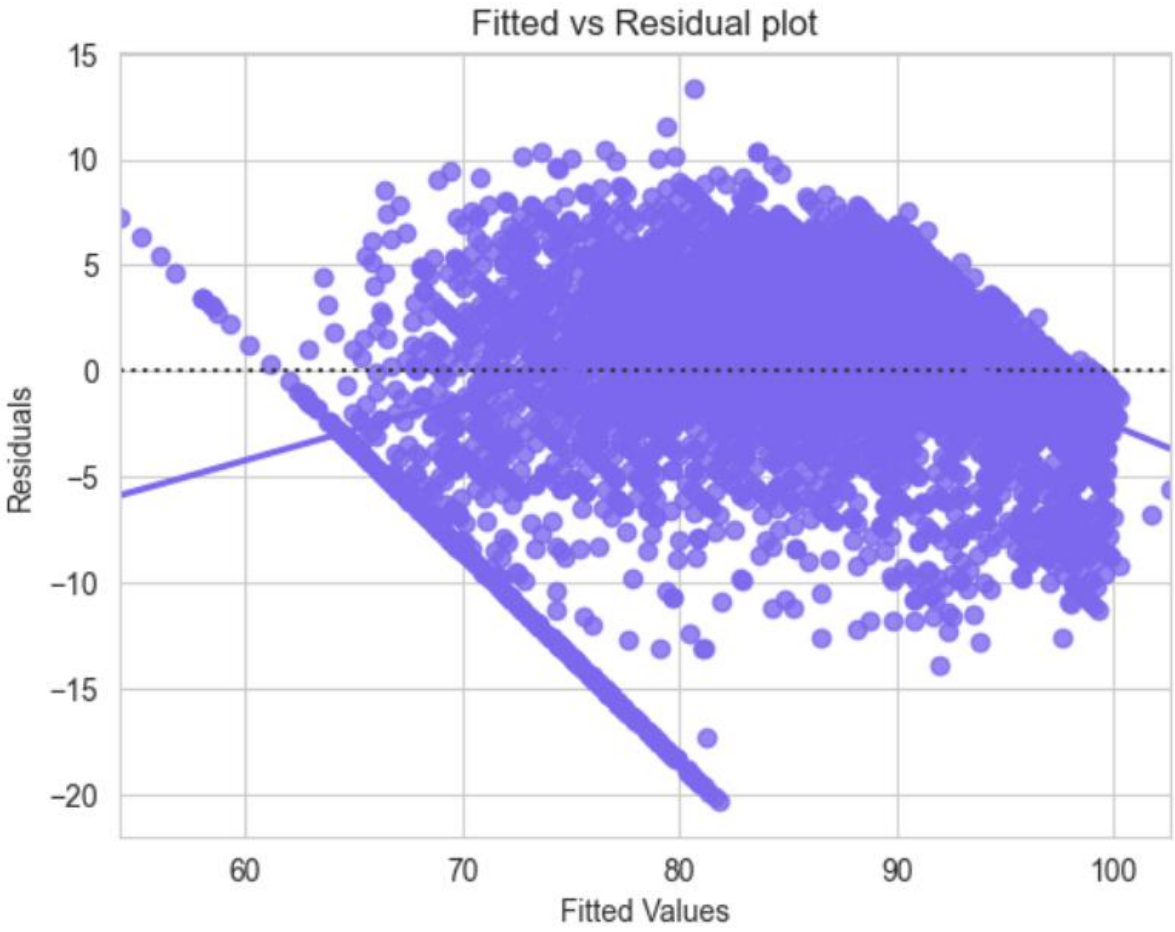


Fig-9 Linearity and Independence Graph

Evaluating Normality:

- The residuals or error terms are expected to follow a normal distribution.
- Deviations from normality can lead to unstable confidence intervals, complicating the estimation of coefficients through least squares minimization.
- The visual analysis indicates that residuals are normally distributed, albeit with a slight leftward skew.
- This skewness may stem from the inclusion of outliers as valid values in the model.

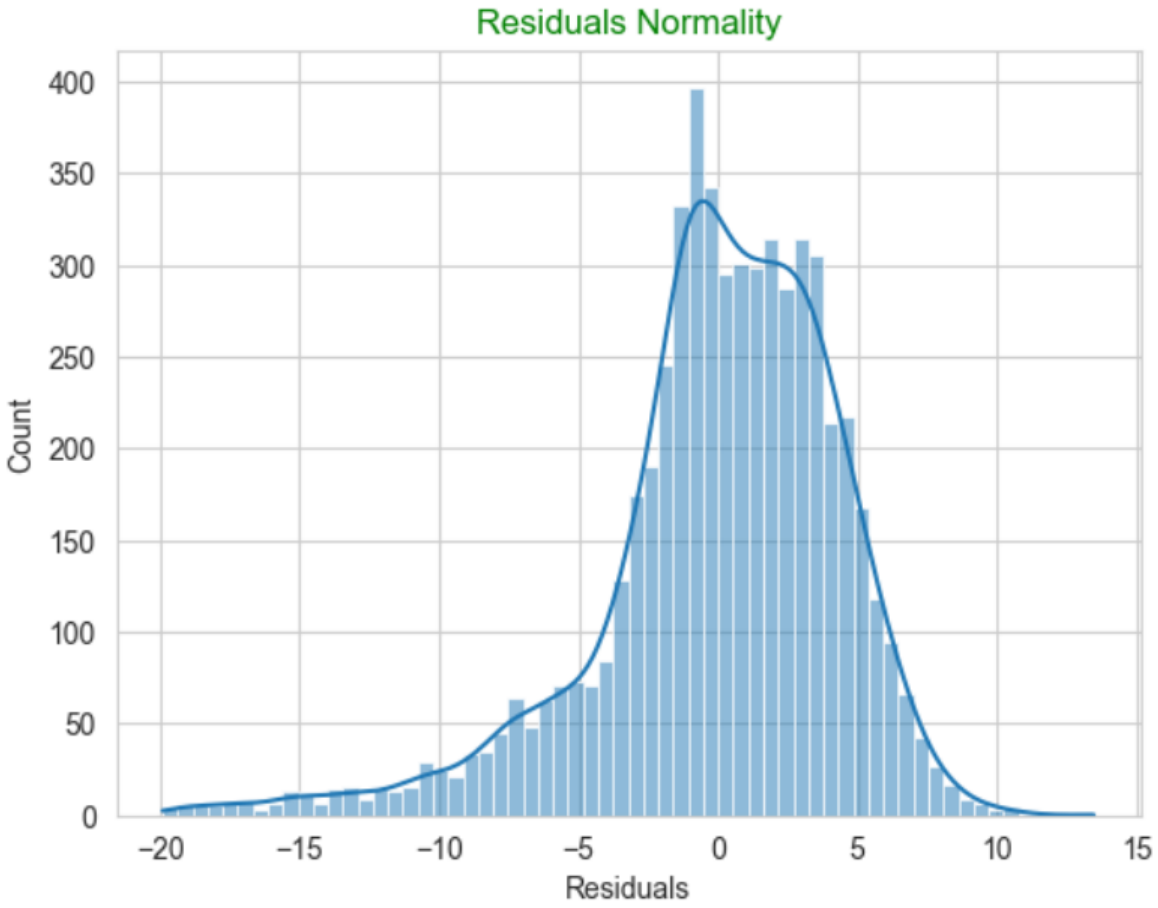


Fig-10 Residual Normality

- The QQ plot is a graphical tool for assessing the normality of residuals, which ideally should align closely with a straight line. While most data points adhere to this line, perfect alignment is rare, particularly without domain-specific adjustments. Nevertheless, the current QQ plot is largely satisfactory.

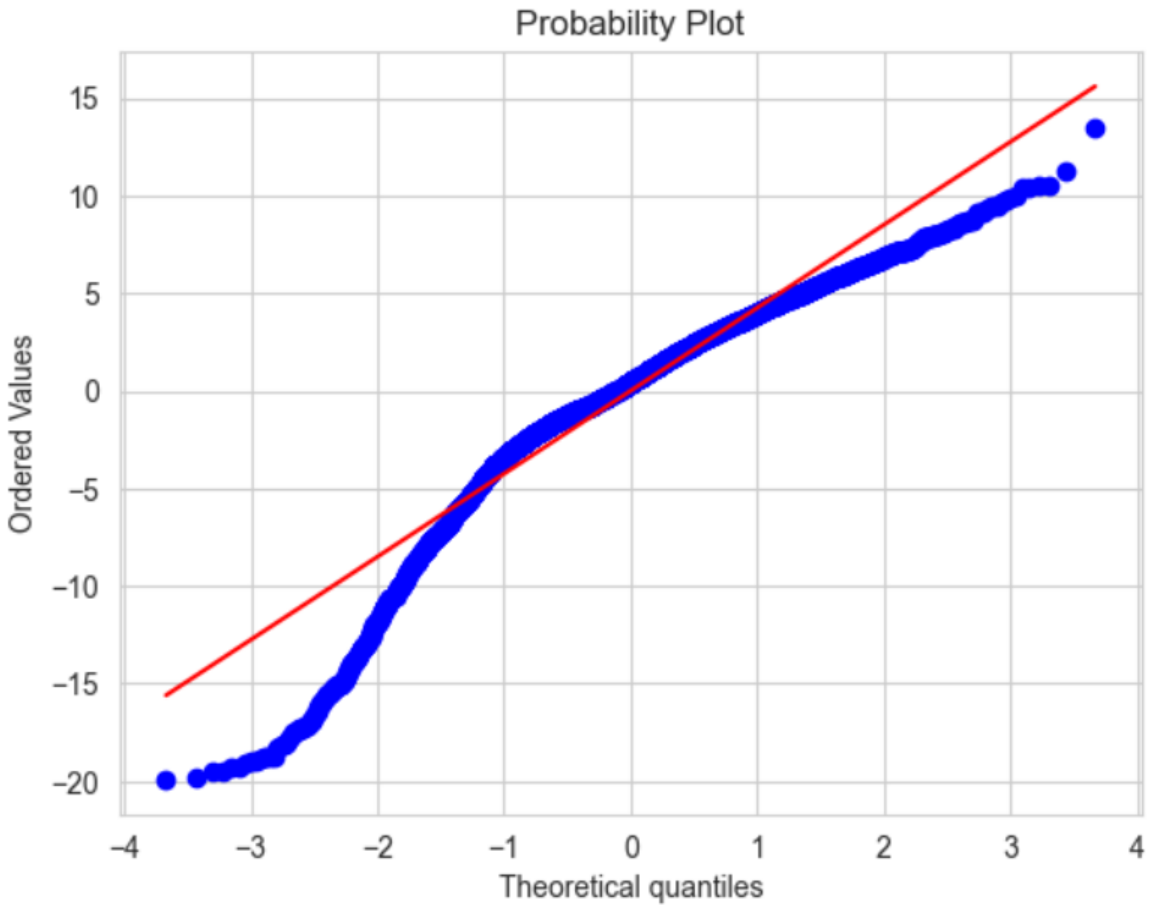


Fig-11 Probability plot

- The Shapiro-Wilk test is another method for testing normality, with the following hypotheses:
  - **Null hypothesis**: Data follows a normal distribution.
  - **Alternate hypothesis**: Data does not follow a normal distribution.

  And we obtained,
  ShapiroResult(statistic=0.936675488948822, pvalue=4.344025239406933e-44)

- Given that the p-value is less than 0.05, the Shapiro test suggests that the residuals do not conform to normality. However, for practical purposes, this distribution may be considered approximately normal.

Assessing Homoscedasticity:

- The Goldfeld-Quandt test is employed to test for homoscedasticity, with the following hypotheses:
  - **Null hypothesis**: Residuals are homoscedastic.
  - **Alternate hypothesis**: Residuals are heteroscedastic.

  We obatined,

  ```
  Pvalue  = 0.0015699789061589338
  ```

- A p-value greater than 0.05 indicates that the residuals are homoscedastic.

Model Performance - RMSE Scores:

- The Root Mean Square Error (RMSE) for the training set is
  4.387920340877025

, while for the test set, it is
4.610896886509368.

- The similarity in RMSE values between the training and test datasets suggests that the model is not overfitting.
- Therefore, the "OLS" model is deemed suitable for both predictive and inferential purposes.
- And, the equation we found for this is written below:

usr = 83.15783465735694 + -0.06287515640697153 * (lread) + 0.05009401389469448 * (lwrite) + -1.424125934471 47e-07 * (scall) + -0.003356603974680 5686 * (swrite) + -0.2790046231599548 * (exec) + -4.8572305667963736e-06 * (rchar) + -6.431881855263365e-06 * (wchar) + -0.06858535962774243 * (ppgin) + -0.03322952936106653 * (pflt) + -0.005908210604829442 * (vflt) + -0.0003434836221386 2534 * (freemem) + 8.933858939901896e-06 * (freeswap) + 1.559599966991291 * (runqsz_Not_CPU_Bound)

1.3 Linear Regression using Sklearn :

Applying the Linear Regression model using Sklearn on the same training and testing dataset we obtained after performing the VIF treatment. We got the coefficents for each variables as written below:

```
The            coefficient            for            const            is            0.0
The         coefficient         for         lread         is         -0.06287515641358427
The         coefficient         for         lwrite         is         0.05009401390671614
The         coefficient         for         scall         is         -1.424125933896222e-07
The         coefficient         for         swrite         is         -0.0033566039746875807
The         coefficient         for         exec         is         -0.2790046231598889
The         coefficient         for         rchar         is         -4.857230566785805e-06
The         coefficient         for         wchar         is         -6.43188185516163e-06
The         coefficient         for         ppgin         is         -0.06858535962777663
The         coefficient         for         pflt         is         -0.03322952936107273
The         coefficient         for         vflt         is         -0.005908210604827568
The         coefficient         for         freemem         is         -0.00034348362213795785
The         coefficient         for         freeswap         is         8.933858939906642e-06
The coefficient for runqsz_Not_CPU_Bound is 1.559599966991288
```

Table-13 Coefficients of variables

- The intercept for our model is 83.1578346572953

- R-squared:
  Training: 0.7932223244636325
  Testing: 0.7645419860579308

- 79% and 76% of the variation in the usr is explained by the predictors in the model for train and test set.

- RMSE:

  Training: 4.450707631846364

  Testing: 4.6841331106664192

Regularization Techniques in Linear Regression: Regularization methods like Ridge and Lasso are instrumental in refining linear regression models. They introduce a penalty term to the cost function, which is contingent on the coefficients' magnitudes. Given that these penalties are influenced by the scale of the input features, normalizing the data is advisable to ensure uniform influence across all features.

Ridge Regularization Analysis:

- The optimal alpha value determined through Grid Search CV is 10.
- The Ridge model exhibits commendable performance with a training score of 0.793218 and a testing score of 0.76447.
- The RMSE for the training set is approximately 4.4501, while for the test set, it is around 4.6847.

The Actual vs. Predicted plot for a sample of 100 records shows minimal deviation, indicating the model's accuracy in predictions.

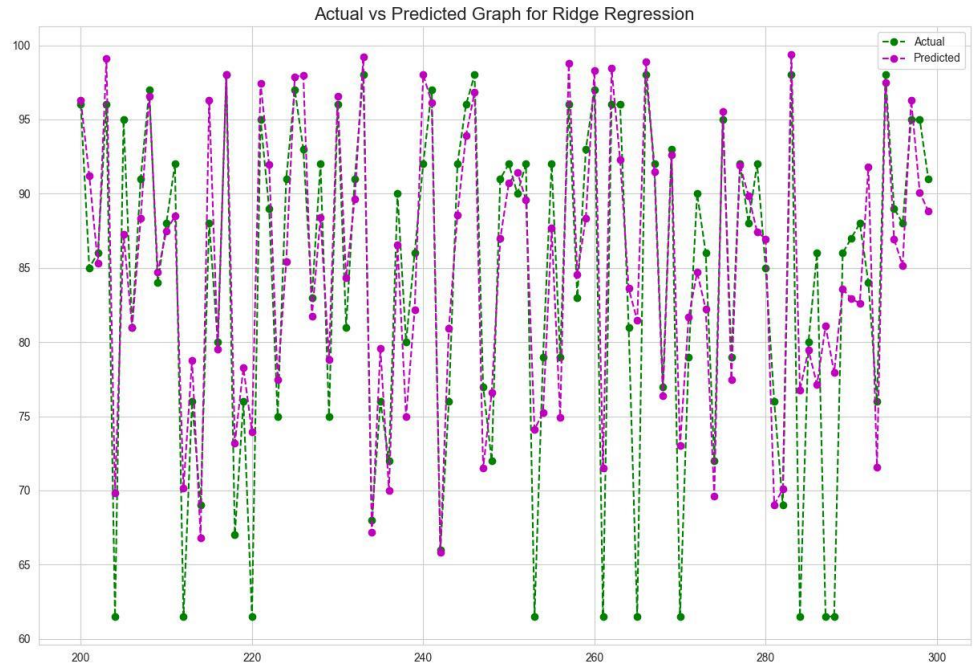


Fig-12 Actual vs Predicted graph for Rigid Method

Lasso Regularization Analysis:

- The best alpha value, ascertained via Grid Search CV, is 0.01.
- The Lasso model scores are close, with a training score of 0.79322 and a testing score of 0.76454.
- The RMSE on the training set is nearly 4.4507 , and on the test set, it is about 4.6841.

The plot for the Lasso model, similar to the Ridge model, demonstrates a slight discrepancy between actual and predicted values, signifying the model's precision.

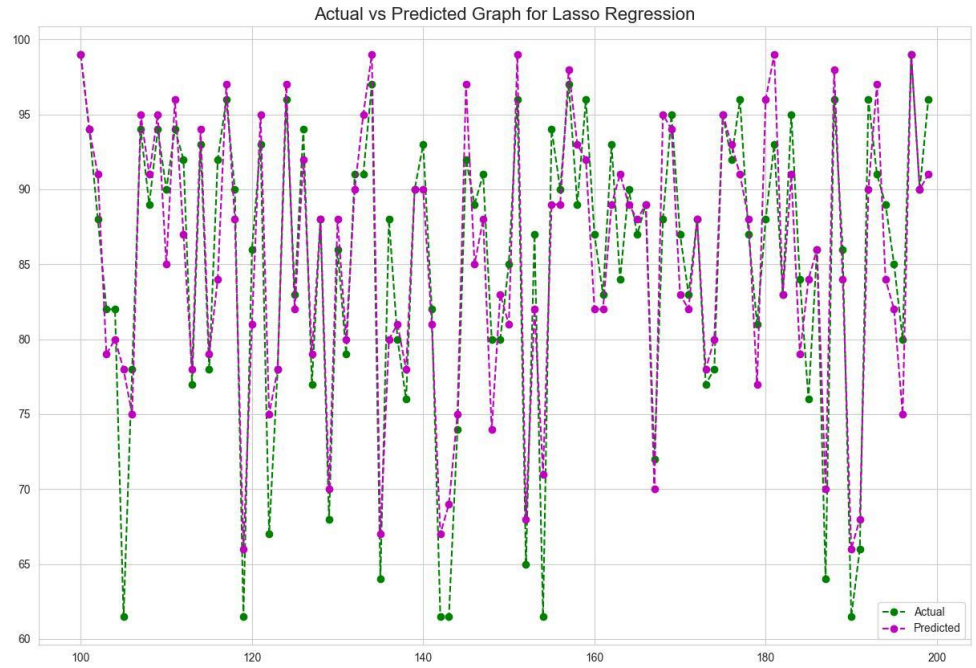


Fig-13 Actual vs Predicted graph for Lasso Method

Final Model Equation:

usr = 83.15783465735694 + -0.06287515640697153 * (lread) + 0.050094013894694 48 * (lwrite) + -1.424125934471 47e-07 * (scall) + -0.003356603974680 5686 * (swrite) + -0.2790046231599548 * (exec) + -4.8572305667963736e-06 * (rchar) + -6.431881855263365e-06 * (wchar) + -0.06858535962774243 * (ppgin) + -0.03322952936106653 * (pflt) + -0.005908210604829442 * (vflt) + -0.0003434836221386 2534 * (freemem) + 8.933858939901896e-06 * (freeswap) + 1.559599966991291 * (runqsz_Not_CPU_Bound)

1.4 Conclusions and Business Insights:

The initial data exploration revealed a rightward skew and numerous outliers. Despite a significant presence of zeros in many columns, these were retained initially. Missing values in two columns were imputed with their respective medians. Correlations were then examined via a heatmap.

Outliers were capped at the 95th and 5th percentiles, followed by a data split into training and testing sets. The model's accuracy was first gauged using a linear regression model from `sklearn`. Further outlier capping at the 10th and 90th percentiles yielded a slight accuracy improvement.

Columns with over 50% zeros were removed, which did not impact accuracy. Incorporating the square of the `freeswap` variable significantly enhanced training accuracy to nearly 79%. The

Variance Inflation Factor (VIF) was then calculated to assess multicollinearity, leading to the iterative removal of columns until an acceptable model was achieved. Despite some remaining

1. An increase in `scall`, `rchar`, and `pflt` leads to a decrease in the percentage of time CPUs operate in user mode. Monitoring these metrics is crucial as their escalation can reduce user mode CPU time.

2. The impact of `freeswap` on user mode CPU time is nuanced and depends on the combined effect of the terms 8.933858939901896e-06 ×(freeswap) and while removing `freeswap` could simplify interpretation, it was retained to avoid a significant drop in model accuracy.

Problem-2: Contraceptive Prevalence Survey.

In your role as a statistician at the Republic of Indonesia Ministry of Health, you have been entrusted with a dataset containing information from a Contraceptive Prevalence Survey. This dataset encompasses data from 1473 married females who were either not pregnant or were uncertain of their pregnancy status during the survey.

Your task involves predicting whether these women opt for a contraceptive method of choice. This prediction will be based on a comprehensive analysis of their demographic and socio-economic attributes.

Data Description

1. Wife's age (numerical)
2. Wife's education (categorical) 1=uneducated, 2, 3, 4=tertiary
3. Husband's education (categorical) 1=uneducated, 2, 3, 4=tertiary
4. Number of children ever born (numerical)
5. Wife's religion (binary) Non-Scientology, Scientology
6. Wife's now working? (binary) Yes, No
7. Husband's occupation (categorical) 1, 2, 3, 4(random)
8. Standard-of-living index (categorical) 1=verlow, 2, 3, 4=high
9. Media exposure (binary) Good, Not good
10. Contraceptive method used (class attribute) No,Yes

2.1 Data- preprocessing and EDA:

First, we will look at first and last five rows using function head and tail respectively, of the dataset from excel file called Contraceptive method dataset that we loaded using read excel function. In fig-13 and fig-14 shows below shows the dataset.

	Wife_age	Wife_education	Husband_education	No_of_children_born	Wife_religion	Wife_Working	Husband_Occupation	Standard_of_living_index	Media_exposure	Contra
0	24.0	Primary	Secondary	3.0	Scientology	No	2	High	Exposed	
1	45.0	Uneducated	Secondary	10.0	Scientology	No	3	Very High	Exposed	
2	43.0	Primary	Secondary	7.0	Scientology	No	3	Very High	Exposed	
3	42.0	Secondary	Primary	9.0	Scientology	No	3	High	Exposed	
4	36.0	Secondary	Secondary	8.0	Scientology	No	3	Low	Exposed	

Table-14 First 5 rows

	Wife_age	Wife_ education	Husband_education	No_of_children_born	Wife_religion	Wife_Working	Husband_Occupation	Standard_of_living_index	Media_exposure	Con
1468	33.0	Tertiary	Tertiary	NaN	Scientology	Yes	2	Very High	Exposed	
1469	33.0	Tertiary	Tertiary	NaN	Scientology	No	1	Very High	Exposed	
1470	39.0	Secondary	Secondary	NaN	Scientology	Yes	1	Very High	Exposed	
1471	33.0	Secondary	Secondary	NaN	Scientology	Yes	2	Low	Exposed	
1472	17.0	Secondary	Secondary	1.0	Scientology	No	2	Very High	Exposed	

table-15 Last 5 rows

Now, we use shape function the dataset and we get that there are 1473 row and 10 columns. Then, we use info function and found out the data type of each column as shown in the below table. we will check for the duplicated rows are present or not using duplicate function and found out that in the dataset there are 80 duplicated rows present and using drop duplicate function we removed those rows from the dataset permanently.

We also used isnull function and found that there are some empty or null values in the dataset for wife age has 71 and number of children born has 21 null values, and we replaced those empty values with the mean of the respective variables and obtained the given below Info table.

```
<class 'pandas.core.frame.DataFrame'>
Index: 1391 entries, 0 to 1472
Data columns (total 10 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Wife_age                  1391 non-null   float64
 1   Wife_ education           1391 non-null   object
 2   Husband_education         1391 non-null   object
 3   No_of_children_born       1391 non-null   float64
 4   Wife_religion             1391 non-null   object
 5   Wife_Working              1391 non-null   object
 6   Husband_Occupation        1391 non-null   int64
 7   Standard_of_living_index  1391 non-null   object
 8   Media_exposure            1391 non-null   object
 9   Contraceptive_method_used 1391 non-null   object
dtypes: float64(2), int64(1), object(7)
```

Table-16 Info function Table

Now, using describe function we obtain the five summary points as shown below.

	count	mean	std	min	25%	50%	75%	max
Wife_age	1391.0	32.561913	8.092559	16.0	26.0	32.0	38.0	49.0
No_of_children_born	1391.0	3.286844	2.383479	0.0	1.0	3.0	5.0	16.0
Husband_Occupation	1391.0	2.176132	0.854041	1.0	1.0	2.0	3.0	4.0

Table- 17 Description Table

After this, using value counts function on each categorical variables we obtain the given below table.

```
Wife_Working
No     1041
Yes     350
Name: count, dtype: int64
```

```
Wife_ education
Tertiary      513
Secondary     398
Primary       330
Uneducated    150
Name: count, dtype: int64
```

```
Standard_of_living_index
Very High    616
High         419
Low          227
Very Low     129
Name: count, dtype: int64
```

```
Husband_education
Tertiary      825
Secondary     347
Primary       175
Uneducated     44
Name: count, dtype: int64
```

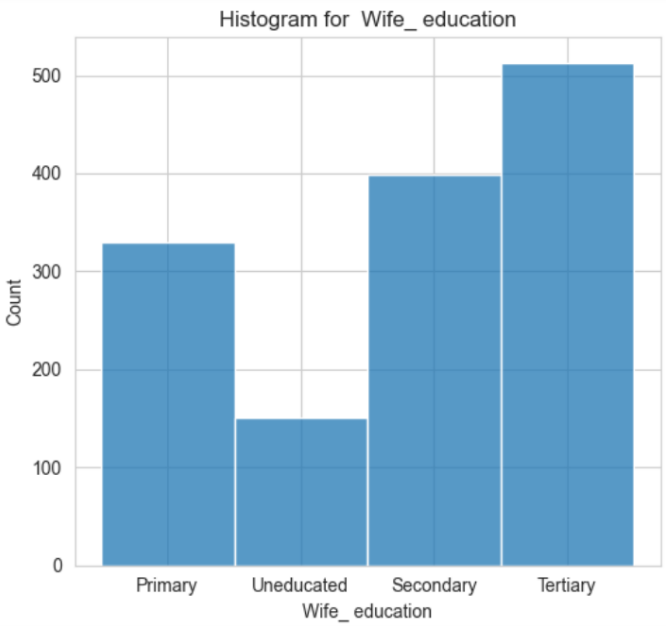
```
Media_exposure
Exposed        1282
Not-Exposed     109
Name: count, dtype: int64
```

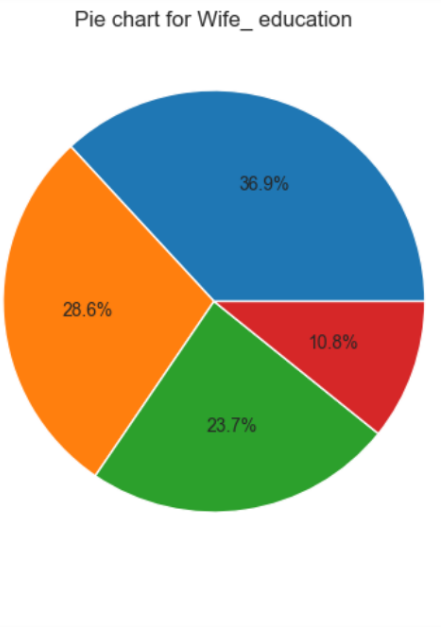
```
Wife_religion
Scientology        1184
Non-Scientology     207
Name: count, dtype: int64
```

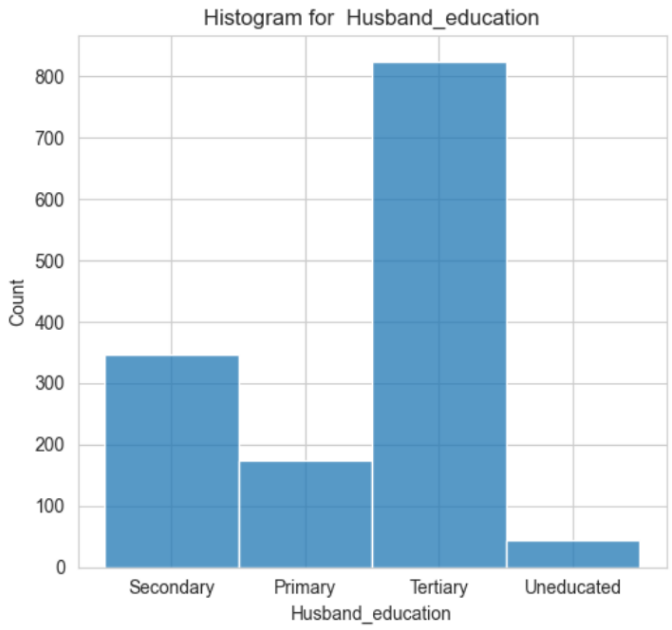
```
Contraceptive_method_used
Yes    777
No     614
Name: count, dtype: int64
```

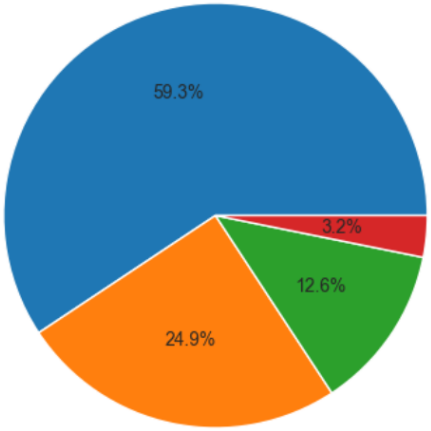
Table- 18 Categorical variable value counts

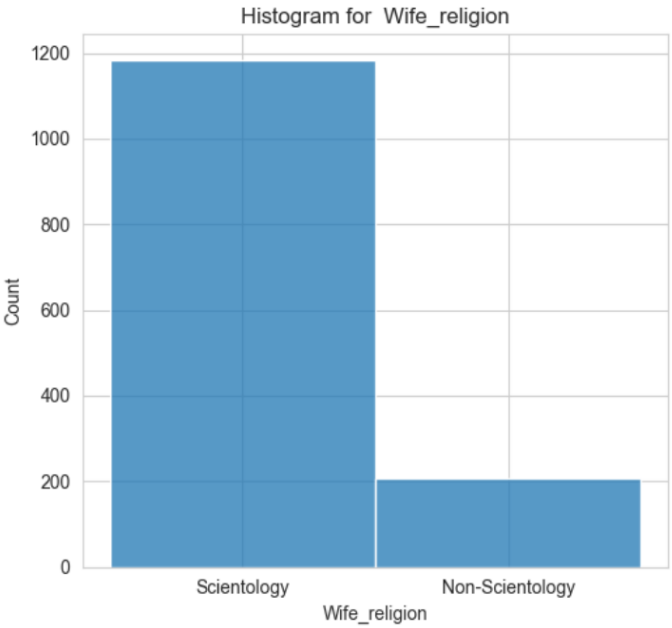
Univariate Analysis of each categorical variable is shown in the below figure.

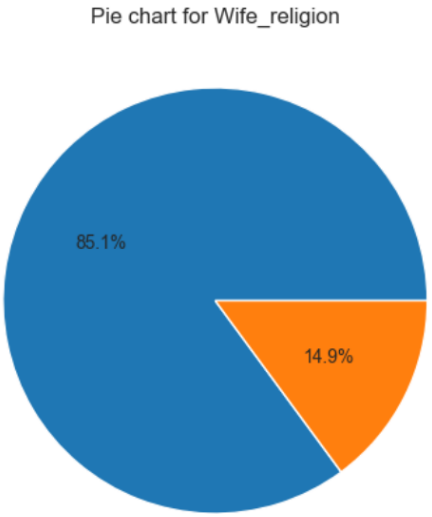

Histogram for Wife_ education
Count
0
100
200
300
400
500
Primary
Uneducated
Secondary
Tertiary
Wife_ education

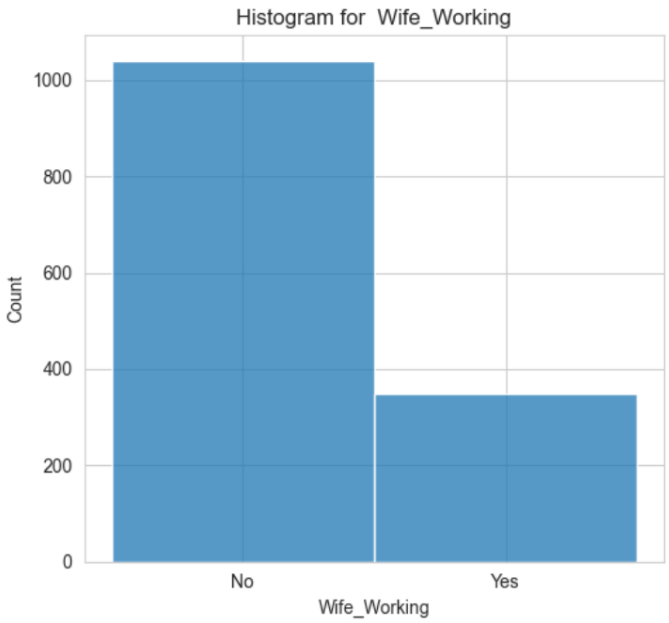

Pie chart for Wife_ education
36.9%
28.6%
10.8%
23.7%

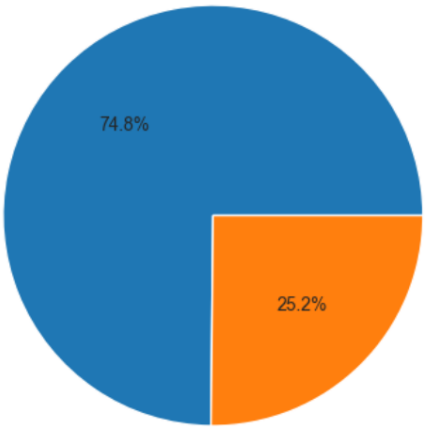

Histogram for Husband_education
Count
0
100
200
300
400
500
600
700
800
Secondary
Primary
Tertiary
Uneducated
Husband_education

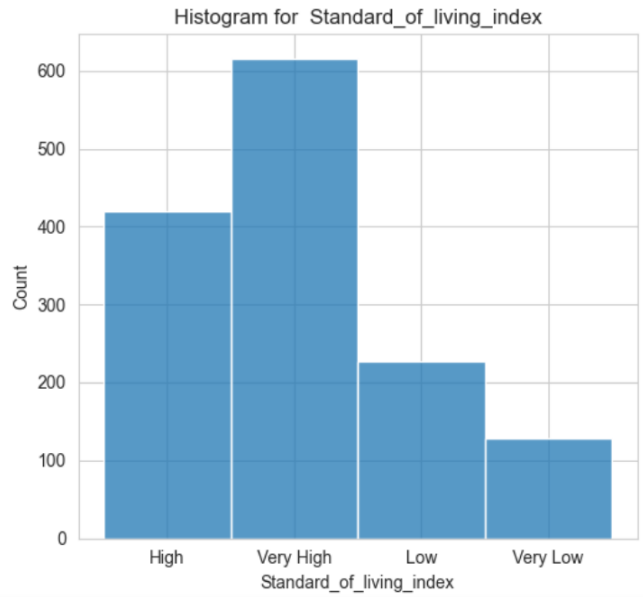

Pie chart for Husband_education
59.3%
3.2%
12.6%
24.9%

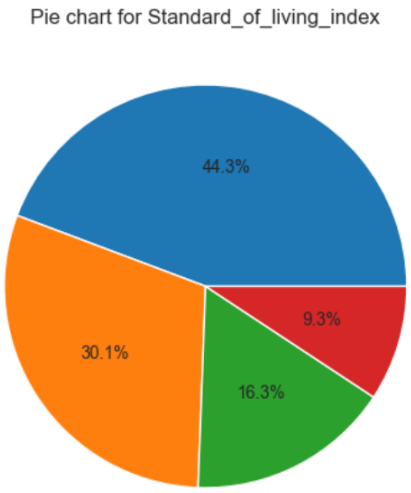

Histogram for Wife_religion
1200
1000
800
600
400
200
0
Count
Scientology
Non-Scientology
Wife_religion

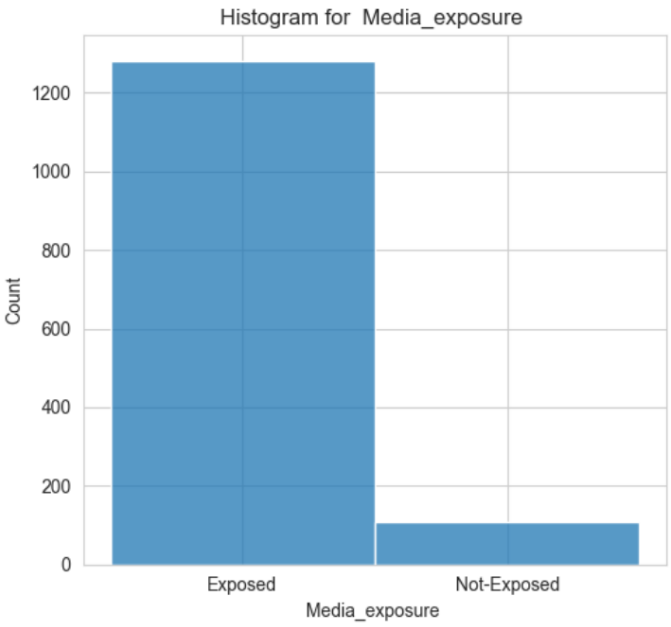

Pie chart for Wife_religion
85.1%
14.9%

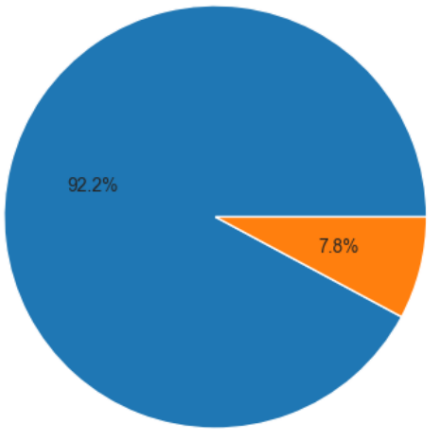

Histogram for Wife_Working
1000
800
600
400
200
0
Count
No
Yes
Wife_Working


Pie chart for Wife_Working
74.8%
25.2%


Histogram for Standard_of_living_index
Count
600
500
400
300
200
100
0
High
Very High
Low
Very Low
Standard_of_living_index


Pie chart for Standard_of_living_index
44.3%
9.3%
30.1%
16.3%


Histogram for Media_exposure
Count
1200
1000
800
600
400
200
0
Exposed
Not-Exposed
Media_exposure


Pie chart for Media_exposure
92.2%
7.8%

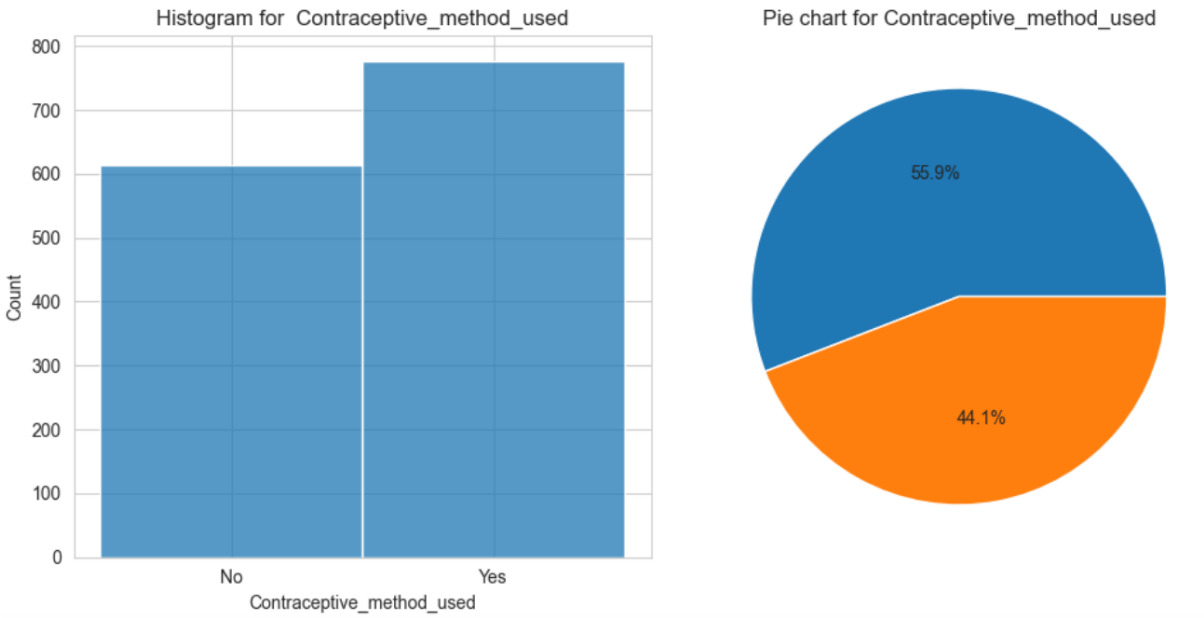


Fig-14 Count plots for categorical variable

Insights:

- A higher incidence of tertiary education is observed among both husbands and wives.
- The population of educated individuals surpasses that of the uneducated, for both genders.
- The number of uneducated wives exceeds that of uneducated husbands.
- Scientology is the predominant religion among the majority.
- The proportion of women not engaged in employment significantly outnumbers those who are employed.
- A substantial segment of the population resides in areas characterized by high or very high living standards.
- A total of 227 individuals are categorized under a 'Low' standard of living, while 129 fall into the 'Very Low' category.
- A considerable majority of women have utilized some form of contraception, yet a notable fraction has not.
- Exposure to media is widespread, with 1248 individuals having access, in contrast to 109 who do not.

Univariate Analysis of numerical variables and the boxplot and histogram has been shown below.

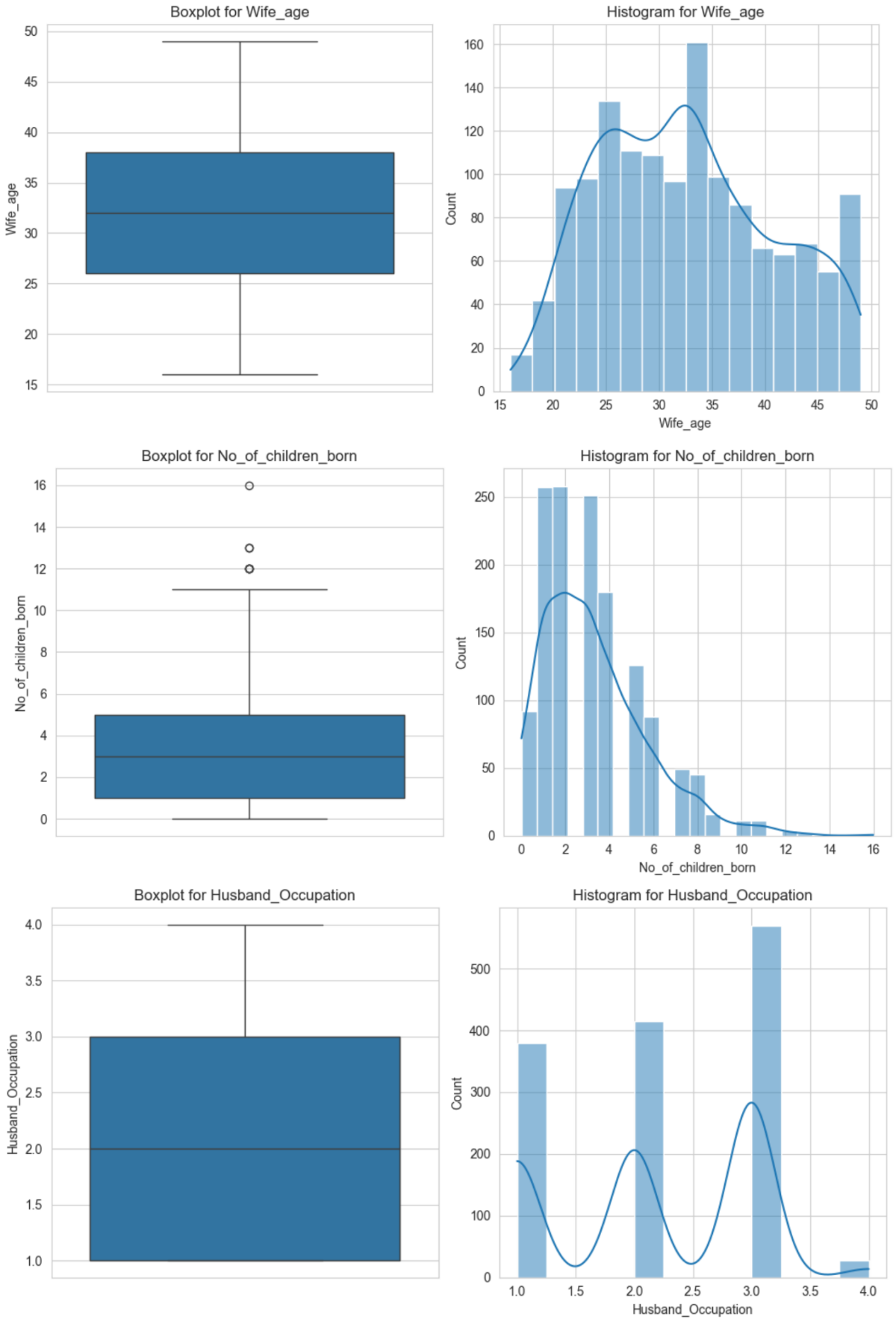


fig-15 Boxplot before outlier treatment

We will be using the same method we used in problem-1 for the outlier treatment and obtain the below graphs.

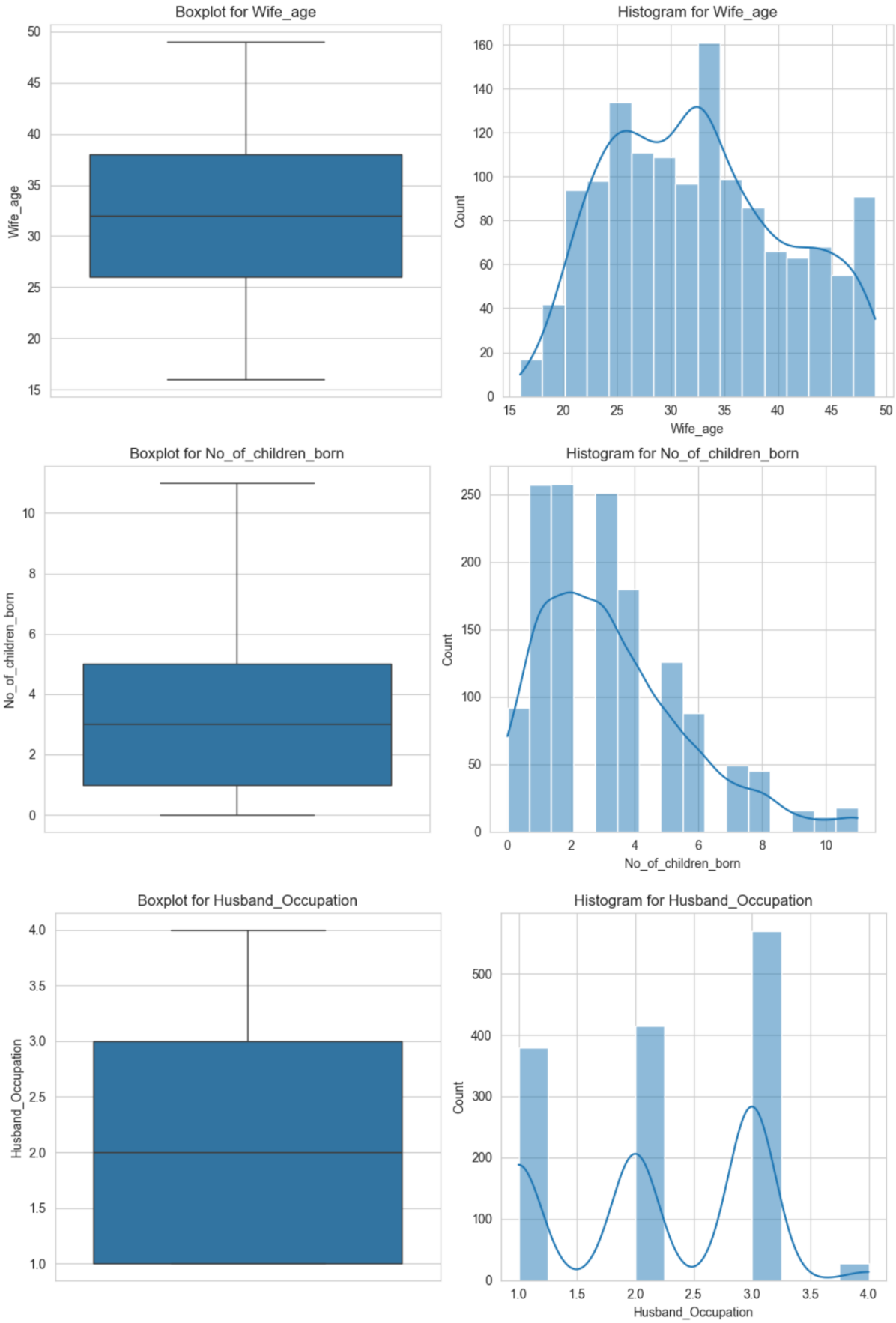


fig-16 Boxplot after outlier treatment

Observation for both before and after outlier treatment:

- It's noted that there's a correlation between the increase in the wife's age and the number of children born.
- Interestingly, a higher number of children are associated with cases where contraceptive methods were used, which is counterintuitive.
- The a fore mentioned anomaly might be attributed to incorrect data entries, as suggested by the presence of outliers in this data column.
- The distribution of the 'No_of_children_born' column is right-skewed, indicating outliers.
- The dataset encompasses women aged between 16 to 49 years.
- While the majority have either 1 or 4 children, there are instances where the number exceeds 10, with the maximum being 16 children.

Now, we perform Multivariate Analysis we will create count plot for all the variables for the given dependent variable and obtain also the pair plot and heatmap for the numerical variables in the dataset.

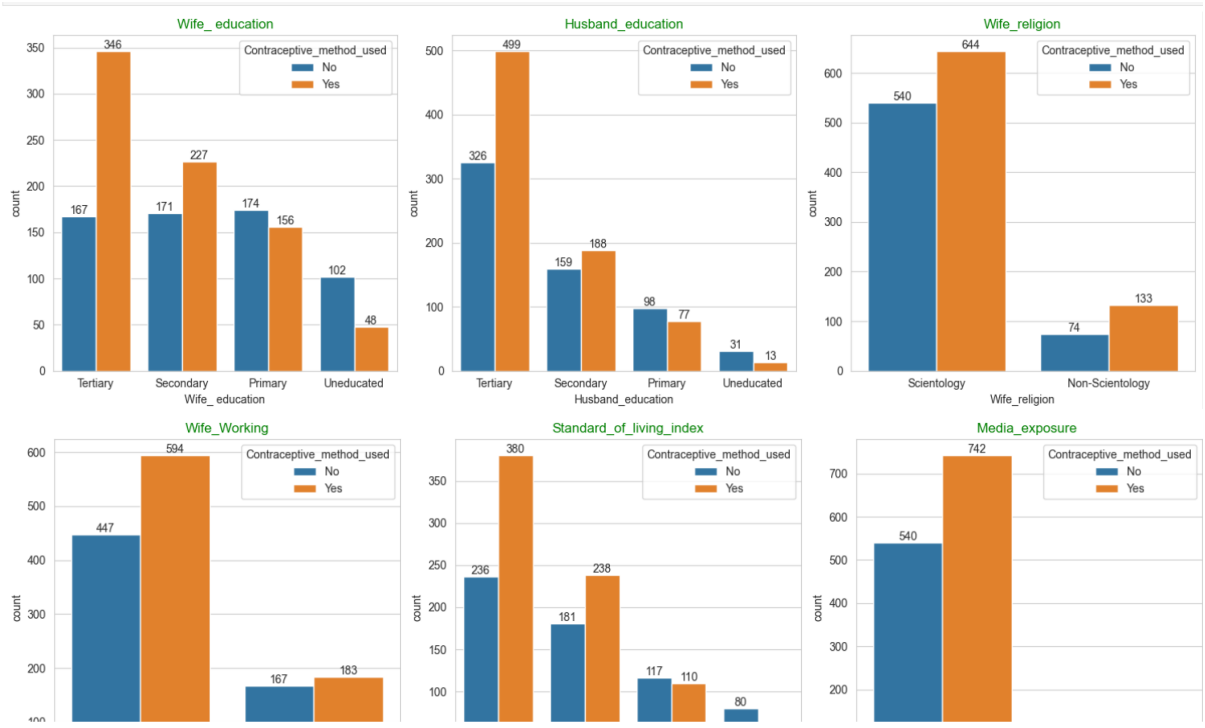


fig-17 Count plots for all variables vs Dependent variable.

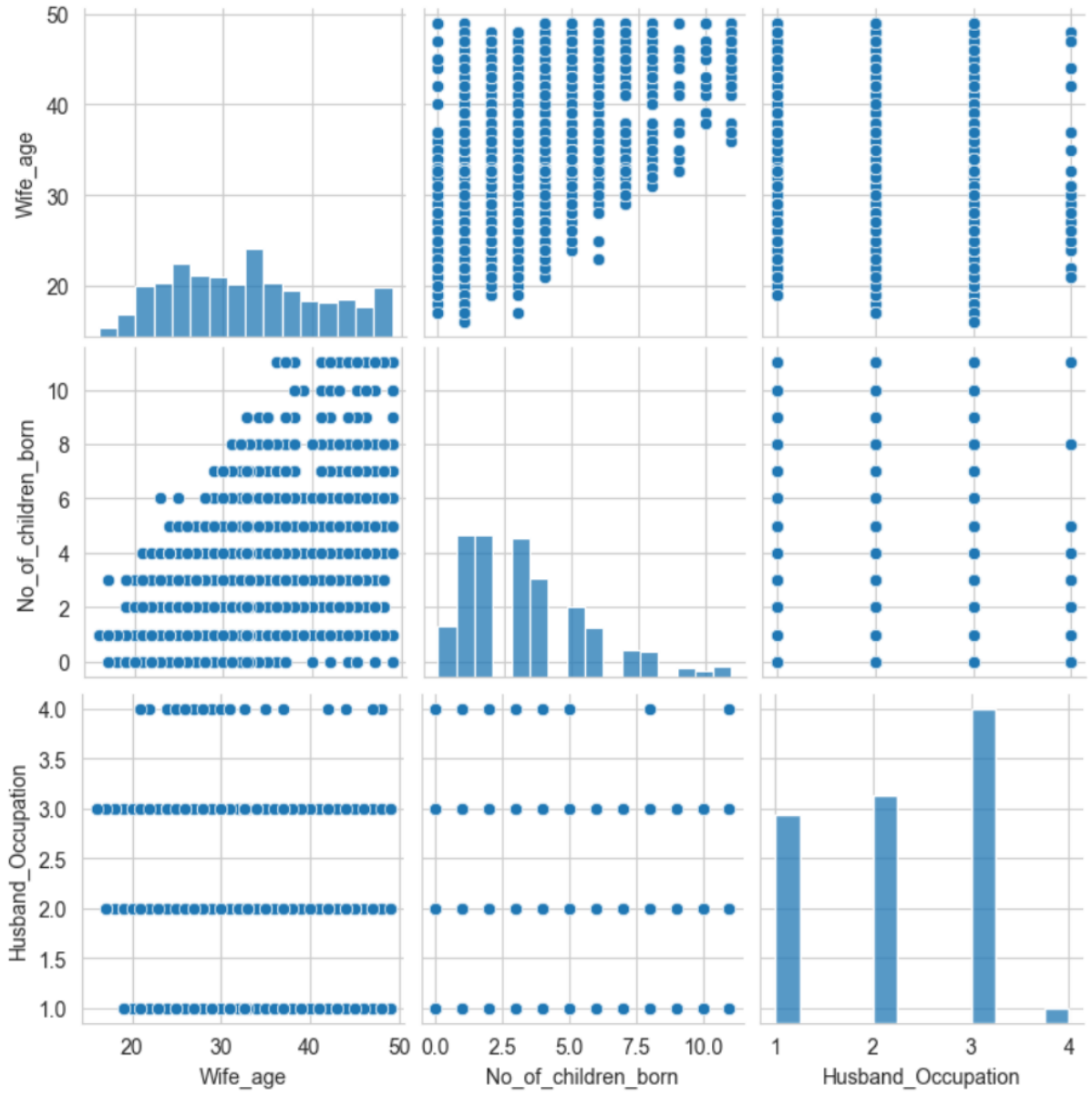


fig-18 Pair plot for numerical variables

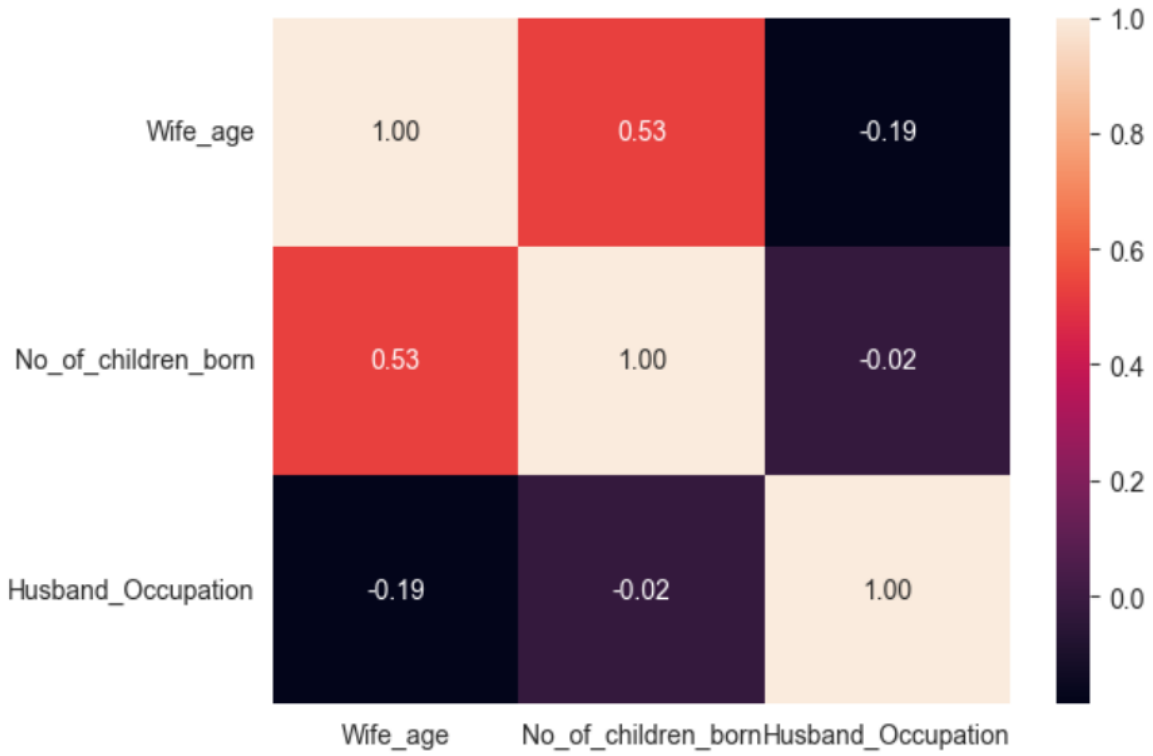


fig-19 Heatmap for numerical variables

Now, we will perform encoding on various variables in the dataset as per out requirement.

First of all we will do label encoding on variables namely wife education, standard of living index, husbands education, contraceptive method used. Then converting these labels into integer datatype. After this the remaining variables are encoded using get dummies function and those variables are also converted into integer dataset and the final dataset is obtained.

	Wife_age	Wife_education	Husband_education	No_of_children_born	Husband_Occupation	Standard_of_living_index	Contraceptive_method_used	Wife_religion_Scientology
0	24.0	2	3	3.0	2.0	3	0	1
1	45.0	1	3	10.0	3.0	4	0	1
2	43.0	2	3	7.0	3.0	4	0	1
3	42.0	3	2	9.0	3.0	3	0	1
4	36.0	3	3	8.0	3.0	2	0	1

Table-19 Final dataset

Now we will divide the data set into independent variables and dependent variables and the split the datasets into training and testing in same way we it for problem-1. After that we will scale the training and testing dataset of independent variables and now we will create models using Logistic, LDA and Cart method.

2.2 Logistic Regression Model:

Fitting the training data into the Logistic model from Sklearn.

We got the following Classification table for both Training and test Datasets as shown below

Train Data classification report----->

	precision	recall	f1-score	support
0	0.66	0.52	0.58	429
1	0.68	0.79	0.73	544
accuracy			0.67	973
macro avg	0.67	0.66	0.66	973
weighted avg	0.67	0.67	0.66	973

Test Data classification report ----->

	precision	recall	f1-score	support
0	0.64	0.45	0.53	185
1	0.65	0.80	0.72	233
accuracy			0.65	418
macro avg	0.64	0.63	0.62	418
weighted avg	0.65	0.65	0.63	418

Table-20 Train and Test Data classification report

Now we will see the confusion matrix for both the dataset as seen below.

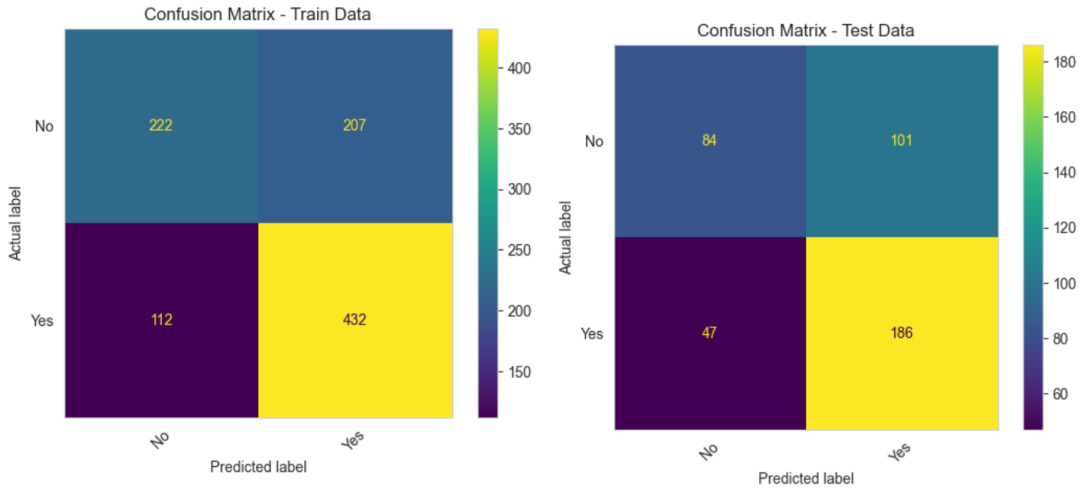


Fig-20 Confusion Matrix for both datasets

Now we will see the AUC-ROC of both the train and test dataset

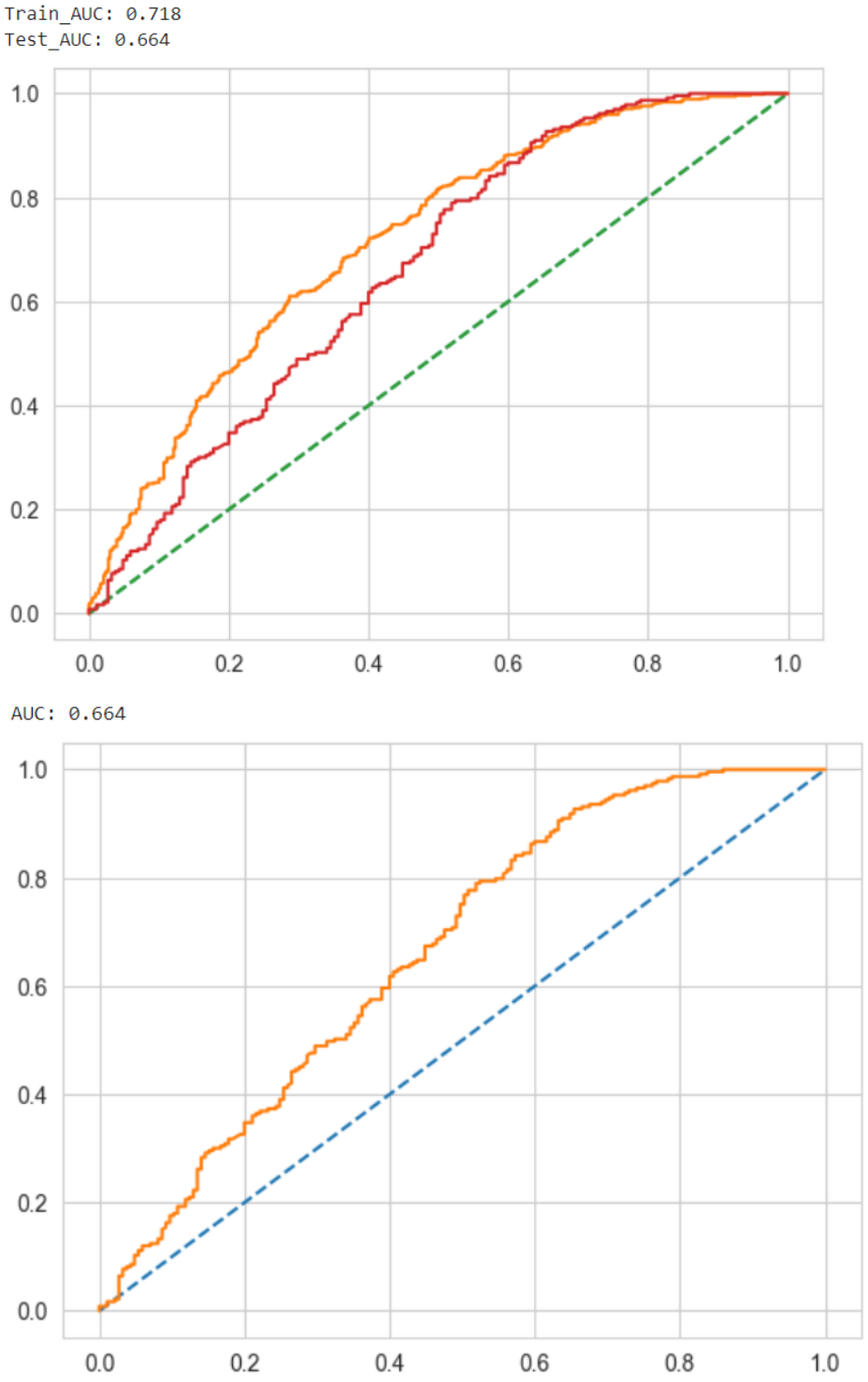


Fig-21 AUC-ROC for both datasets

For Logistic Regression we can observe the following:

- The training data's confusion matrix indicates 432true positives, 222 true negatives, 207 false positives, and 112 false negatives.
- For the test data, the confusion matrix reflects 186 true positives, 84 true negatives, 101 false positives, and 47 false negatives.

- AUC-ROC Curve Analysis:
    - The AUC score for the training data stands at 0.718.
    - The AUC score for the test data is 0.664.

Inferences for Contraceptive Method Usage Prediction:

- For 'Yes' (label 1):
    - Precision (65%): Of all the predictions for contraceptive use, 65% are correct.
    - Recall (80%): Of all the actual contraceptive users, 80% are correctly identified by the model.
- For 'No' (label 0):
    - Precision (64%): Of all the predictions for non-use of contraceptives, 64% are correct.
    - Recall (45%): Of all the actual non-users of contraceptives, 45% are correctly identified by the model.

The overall accuracy of the model is 65%, indicating that 65% of all predictions made by the model are correct. The consistency of accuracy, AUC, precision, and recall between the training and test data suggests that the model is well-calibrated, with no evidence of overfitting or underfitting. Consequently, the model is considered effective for classification purposes.

2.3 Linear Discriminant Analysis:

Now we will perform Linear Discriminant Analysis on the same training and testing dataset and get the following Classification reports.

Classification Report of the training data:

	precision	recall	f1-score	support
0	0.67	0.51	0.58	429
1	0.68	0.80	0.73	544
accuracy			0.67	973
macro avg	0.67	0.66	0.66	973
weighted avg	0.67	0.67	0.67	973

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.64	0.44	0.52	185
1	0.64	0.80	0.72	233
accuracy			0.64	418
macro avg	0.64	0.62	0.62	418
weighted avg	0.64	0.64	0.63	418

Table-21 Both Data classification report for LDA

The confusion matrix for LDA is as shown below.

Fig-22 Confusion Matrix for both datasets for LDA

And finally the AUC-ROC graph is shown below for the given LDA model:

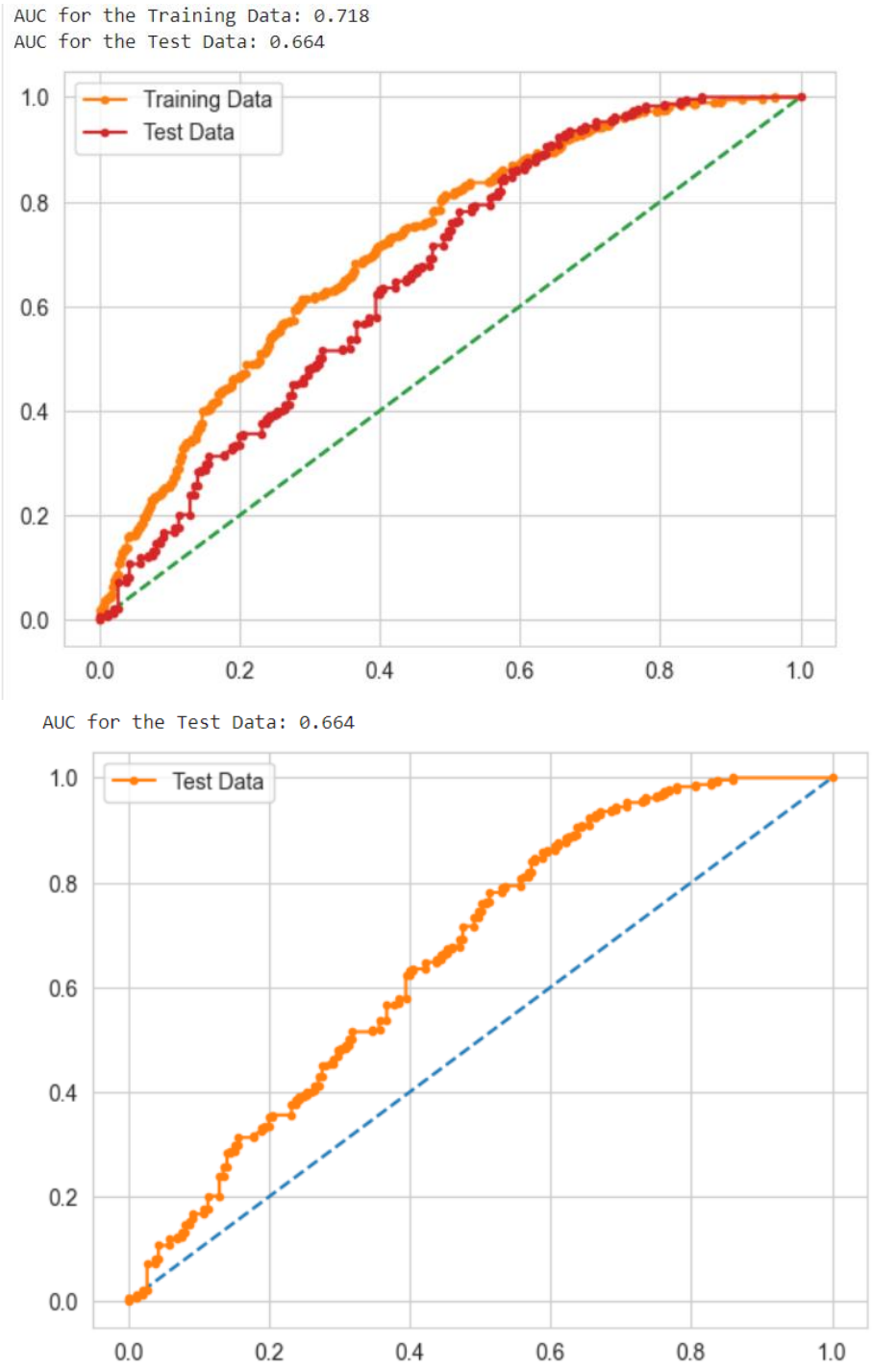


Fig-23 AUC-ROC for both datasets for LDA

For Linear Discriminant Analysis we can observe the following:

- For the prediction of Contraceptive method = Yes (label 1):
  - Precision (64%): This indicates that 64% of the women who were predicted to be using a contraceptive method are indeed using it.
  - Recall (80%): This suggests that the model has correctly identified 80% of the women who are using contraceptives.

- For the prediction of Contraceptive method = No (label 0):
    - Precision (64%): Here, 64% of the women predicted not to be using a contraceptive method are accurately identified.
    - Recall (44%): This means that 44% of the women who are not using contraceptives have been correctly predicted by the model.

The overall accuracy of the model stands at 64%, signifying that two-thirds of the predictions made by the model are correct. These metrics provide a comprehensive view of the model's performance in classifying contraceptive method usage.

2.4 CARTs Method:

While performing the Carts Method we extracted some important freatures when we made the Decision Tree model.

	Imp
Wife_age	0.287612
Wife_ education	0.119475
Husband_education	0.059052
No_of_children_born	0.267936
Husband_Occupation	0.077395
Standard_of_living_index	0.108327
Wife_religion_Scientology	0.028476
Wife_Working_Yes	0.043252
Media_exposure _Not-Exposed	0.008474

Table-22 Important Features before Pruning

As we all known that pruning is necessary for creating a decision tree model and for that we will use the Search grid function to obtain the required parameter for pruning a decision tree model. And we obtained the parameters as follows:

{'ccp_alpha': 0.001, 'criterion': 'entropy', 'max_depth': 5, ' max_features': 'auto',

'min_samples_leaf': 5}

After pruning the model we again extract the important features of the dataset and can be seen below:

	Imp
Wife_age	0.287612
Wife_ education	0.119475
Husband_education	0.059052
No_of_children_born	0.267936
Husband_Occupation	0.077395
Standard_of_living_index	0.108327
Wife_religion_Scientology	0.028476
Wife_Working_Yes	0.043252
Media_exposure _Not-Exposed	0.008474

Table-23 Important Features After Pruning

We obtained the AUC-ROC for Both the dataset as follows.

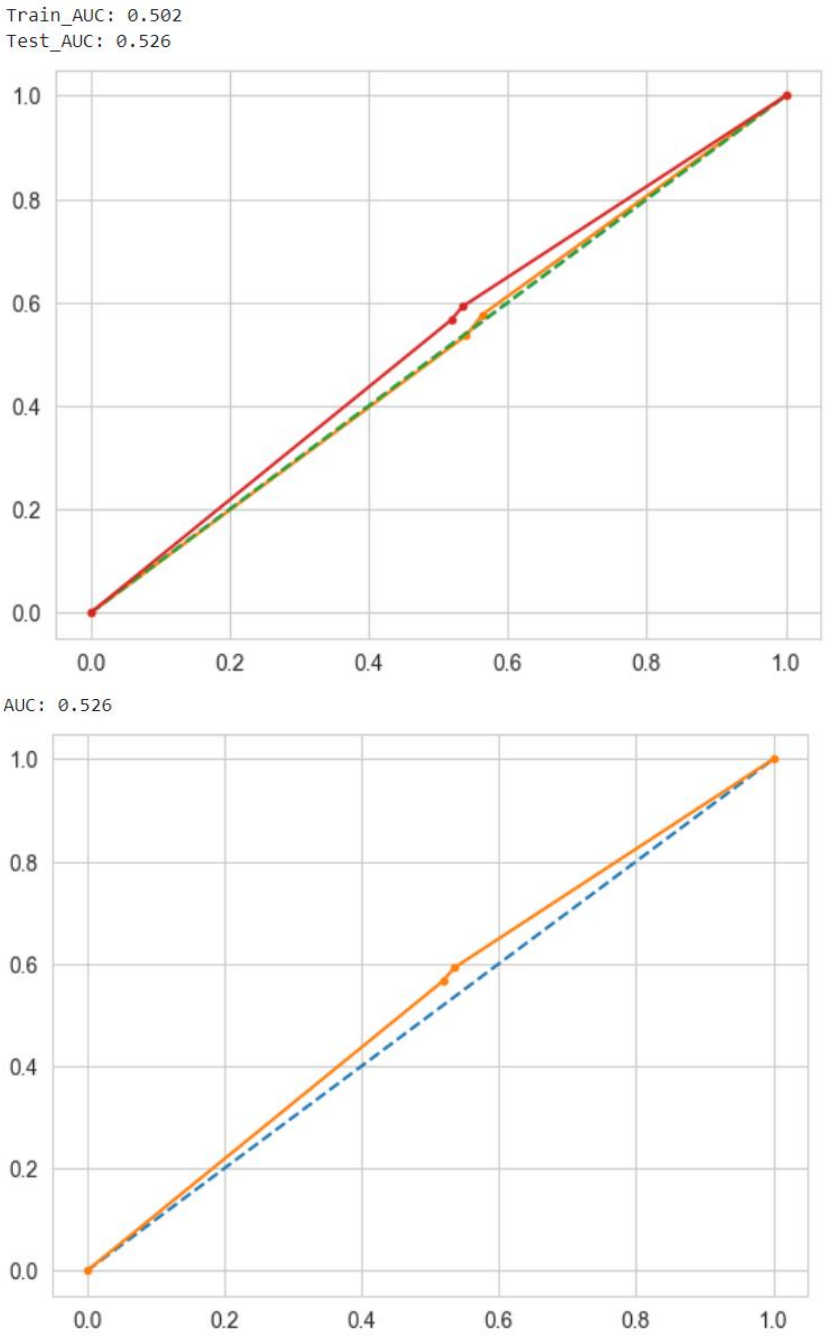


Fig-24 AUC-ROC for CARTs Method

The Classification report for the training and testing dataset are:

Classification Report of the test data:

	precision	recall	f1-score	support
0	0.68	0.48	0.56	181
1	0.67	0.83	0.74	237
accuracy			0.68	418
macro avg	0.68	0.65	0.65	418
weighted avg	0.68	0.68	0.66	418

Classification Report of the train data:

	precision	recall	f1-score	support
0	0.74	0.56	0.64	433
1	0.71	0.84	0.77	540
accuracy			0.72	973
macro avg	0.72	0.70	0.70	973
weighted avg	0.72	0.72	0.71	973

Table-24 Classification report for Decision Tree

The Confusion Matrix for both the dataset is as follows:

Confusion Matrix of the train data:

```
array([[244, 189],
       [ 85, 455]], dtype=int64)
```

Confusion Matrix of the test data:

```
array([[ 86,  95],
       [ 40, 197]], dtype=int64)
```

Table-25 Confusion Matrix for Decision Tree

For CARTs Method (Decision Tree) we can observe the following:

- For the prediction of Contraceptive_method_used = Yes (label 1):
  - Precision (67%): This indicates that 67% of the women who were predicted to be using a contraceptive method are indeed using it.
  - Recall (83%): This suggests that the model has correctly identified 83% of the women who are actually using contraceptives.
- For the prediction of Contraceptive_method_used = No (label 0):
  - Precision (68%): Here, 68% of the women predicted not to be using a contraceptive method are accurately identified.

- Recall (48%): This means that 48% of the women who are not using contraceptives have been correctly predicted by the model.

The overall accuracy of the model stands at 68%, signifying that approximately two-thirds of the predictions made by the model are correct. These metrics provide a comprehensive view of the model's performance in classifying contraceptive method usage.

2.5 Conclusion and Insights:

Model Building Process:

- Comprehensive data analysis was conducted as a preliminary step.
- A train-test split was implemented to evaluate model performance.
- Various algorithms were applied to build different models.
- Models were assessed using metrics like accuracy, precision, recall, and AUC score.

Feature Importance in Models:

- Linear Discriminant Analysis (LDA) highlighted the 'Number of children born' as a highly influential variable due to its significant coefficient magnitude.
- In contrast, 'Wife age' was deemed less critical in LDA, indicating a smaller effect on the model's classification ability.
- The CART model assigned the highest importance to 'Wife age', which is notably contrary to LDA's assessment.
- 'Media exposure' emerged as the least impactful feature, suggesting that for decision trees to split effectively, either a balanced dataset is necessary or the feature must hold substantial information.

Model Performance Metrics:

- In terms of accuracy, recall, and precision, the CART model is superior.
- The most influential features for the CART model are 'Wife age', 'No_of_children_born', 'Wife education', and 'Husband education'.
- The recall for the CART model is 83%, indicating a high rate of correctly identified positive cases.
- The AUC (Area Under the Curve) for the CART model is 50.2% for train data and 52.6% for test data, which is higher than the other models.

Recommendations:

- Factors such as wife's age, number of children born, wife's education, and husband's education play a crucial role in contraceptive use.
- Educational campaigns targeting uneducated women about contraceptive methods could be beneficial.
- Higher contraceptive use is noted among those with a very high standard of living.

- Media exposure is significant in influencing contraceptive method usage.
- Univariate analysis revealed a near-normal distribution for 'Wife age' without outliers, and a pattern in education levels where tertiary education was most common, and lack of education was least.
- Multivariate analysis was employed to examine the interaction between continuous and categorical variables, with missing values addressed through appropriate imputation.
- A dual approach to outliers in 'Number of children born' was adopted: one treating them as legitimate and the other capping them at plausible values. The latter approach yielded a more effective model.
- Logistic Regression served as the initial algorithm, delivering satisfactory results despite a limited dataset. Metrics such as accuracy, precision, recall, and F1-score were computed.
- Both LDA and CART (Decision Tree) were explored for their feature importance insights and accuracy. LDA successfully differentiated the target variable, while the initial CART model suffered from overfitting.
- Pruning and hyperparameter tuning via Grid Search CV were applied to the CART model, which mitigated overfitting to some extent but not conclusively, suggesting that CART may not be optimal for the given data.
- Logistic Regression and LDA outperformed the Decision Tree model. Selecting either Logistic Regression or LDA could yield favourable outcomes, especially with a larger dataset.