PREDICTIVE MODELING

2024

PGP DSBA PROGRAM

by: ABHISHEK K HIREMATH

Problem 1 – Linear Regression:

The comp-activ databases is a collection of a computer systems activity measures. The data was collected from a Sun SPARCstation 20/712 with 128 Mbytes of memory running in a multi-user university department. Users would typically be doing a large variety of tasks ranging from accessing the internet, editing files or running very CPU-bound programs.
As you are a budding data scientist you thought to find out a linear equation to build a model to predict.
'usr'(Portion of time (%) that CPUs run in user mode) and to find out how each attribute affects the system to be in 'usr' mode using a list of system attributes.

Introduction:

The aim is to establish a linear equation for predicting 'usr' (the percentage of time CPUs operate in user mode). Also, to analyse various system attributes to understand their influence on the system's 'usr' mode.

DICTIONARY:

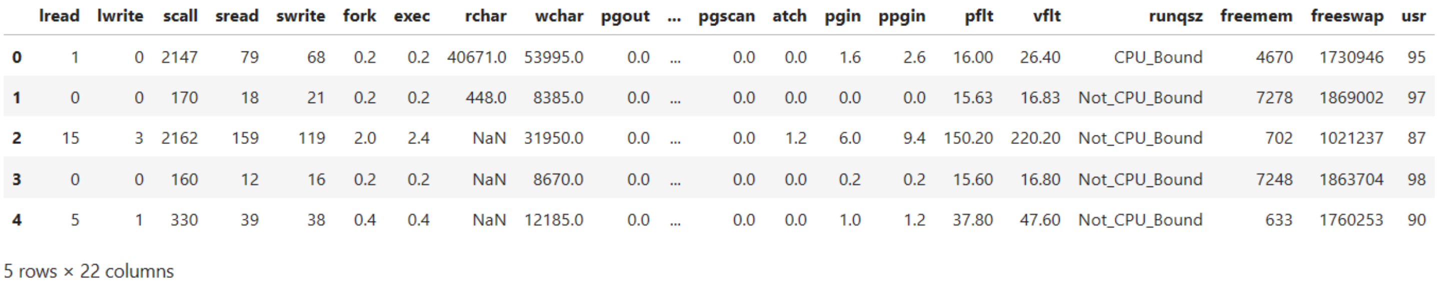
Column	Description
lread	Reads (transfers per seconds) between system memory and user memory
lwrite	writes (transfers per second) between system memory and user memory
scall	Number of systems calls of all types per second
sread	Number of systems read calls per seconds.
swrite	Number of systems write calls per seconds.
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 transferred 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.

1.1 Read the data and do exploratory data analysis. Describe the data briefly. (Check the Data types, shape, EDA, 5point summary). Perform Univariate, Bivariate Analysis, Multivariate Analysis.

As an initial set up we have imported all the necessary library to work on the given problem. We have loaded the data file and the Dataset has 8192 rows and 22 columns.

It is always a good practice to view a sample of the rows. A simple way to do that is to view first 5 rows and last 5 rows.

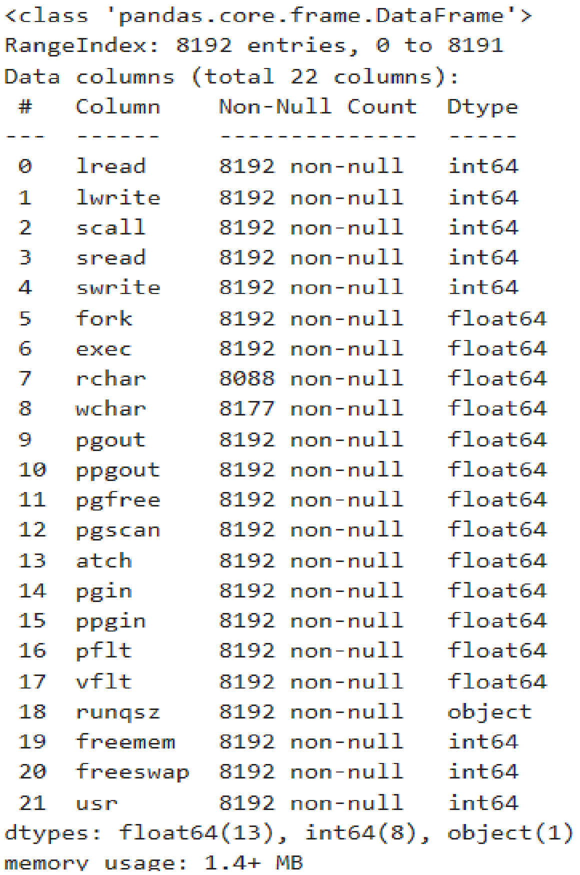


	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	...	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	freeswap	usr
0	1	0	2147	79	68	0.2	0.2	40671.0	53995.0	0.0	...	0.0	0.0	1.6	2.6	16.00	26.40	CPU_Bound	4670	1730946	95
1	0	0	170	18	21	0.2	0.2	448.0	8385.0	0.0	...	0.0	0.0	0.0	0.0	15.63	16.83	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

Fig: -1; Dataset head

Data types of variables:



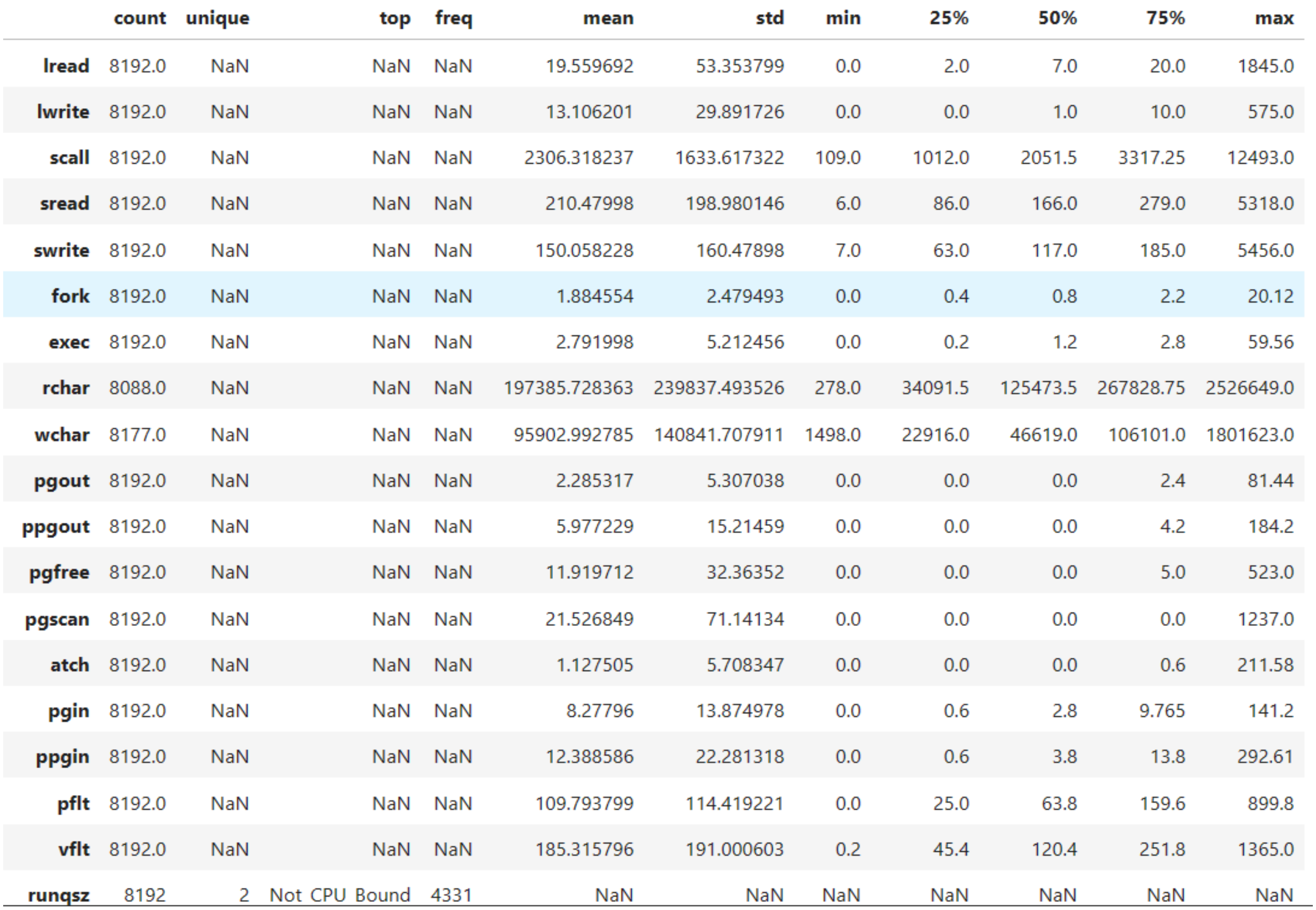
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8192 entries, 0 to 8191
Data columns (total 22 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   lread     8192 non-null   int64  
 1   lwrite    8192 non-null   int64  
 2   scall     8192 non-null   int64  
 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)
memory usage: 1.4+ MB
```

Fig: - 2: Dataset info

Observation:

There are 13 float data type, 8 integer data type and 1 object data type variables in the dataset.

Statistical summary of the dataset:



	count	unique	top	freq	mean	std	min	25%	50%	75%	max
lread	8192.0	NaN	NaN	NaN	19.559692	53.353799	0.0	2.0	7.0	20.0	1845.0
lwrite	8192.0	NaN	NaN	NaN	13.106201	29.891726	0.0	0.0	1.0	10.0	575.0
scall	8192.0	NaN	NaN	NaN	2306.318237	1633.617322	109.0	1012.0	2051.5	3317.25	12493.0
sread	8192.0	NaN	NaN	NaN	210.47998	198.980146	6.0	86.0	166.0	279.0	5318.0
swrite	8192.0	NaN	NaN	NaN	150.058228	160.47898	7.0	63.0	117.0	185.0	5456.0
fork	8192.0	NaN	NaN	NaN	1.884554	2.479493	0.0	0.4	0.8	2.2	20.12
exec	8192.0	NaN	NaN	NaN	2.791998	5.212456	0.0	0.2	1.2	2.8	59.56
rchar	8088.0	NaN	NaN	NaN	197385.728363	239837.493526	278.0	34091.5	125473.5	267828.75	2526649.0
wchar	8177.0	NaN	NaN	NaN	95902.992785	140841.707911	1498.0	22916.0	46619.0	106101.0	1801623.0
pgout	8192.0	NaN	NaN	NaN	2.285317	5.307038	0.0	0.0	0.0	2.4	81.44
ppgout	8192.0	NaN	NaN	NaN	5.977229	15.21459	0.0	0.0	0.0	4.2	184.2
pgfree	8192.0	NaN	NaN	NaN	11.919712	32.36352	0.0	0.0	0.0	5.0	523.0
pgscan	8192.0	NaN	NaN	NaN	21.526849	71.14134	0.0	0.0	0.0	0.0	1237.0
atch	8192.0	NaN	NaN	NaN	1.127505	5.708347	0.0	0.0	0.0	0.6	211.58
pgin	8192.0	NaN	NaN	NaN	8.27796	13.874978	0.0	0.6	2.8	9.765	141.2
ppgin	8192.0	NaN	NaN	NaN	12.388586	22.281318	0.0	0.6	3.8	13.8	292.61
pflt	8192.0	NaN	NaN	NaN	109.793799	114.419221	0.0	25.0	63.8	159.6	899.8
vflt	8192.0	NaN	NaN	NaN	185.315796	191.000603	0.2	45.4	120.4	251.8	1365.0
runqsz	8192	2	Not_CPU_Bound	4331	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Fig: -3: Statistical summary

Key points to observe from the above table are:

1. The values are of different scale for all the attributes.
2. In the above table, few variables are having more that 50% of the data with zero values which are pgout, ppgout, pgfree, atch. However, pgscan is having more than 75% of the data with zero values. We will check the dependency of these variables in later section and will take necessary action.
3. The minimum usr is 0 and maximum is 94 while the mean value is 83.97.
4. And, there are no Duplicate values.

Univariate Analysis:

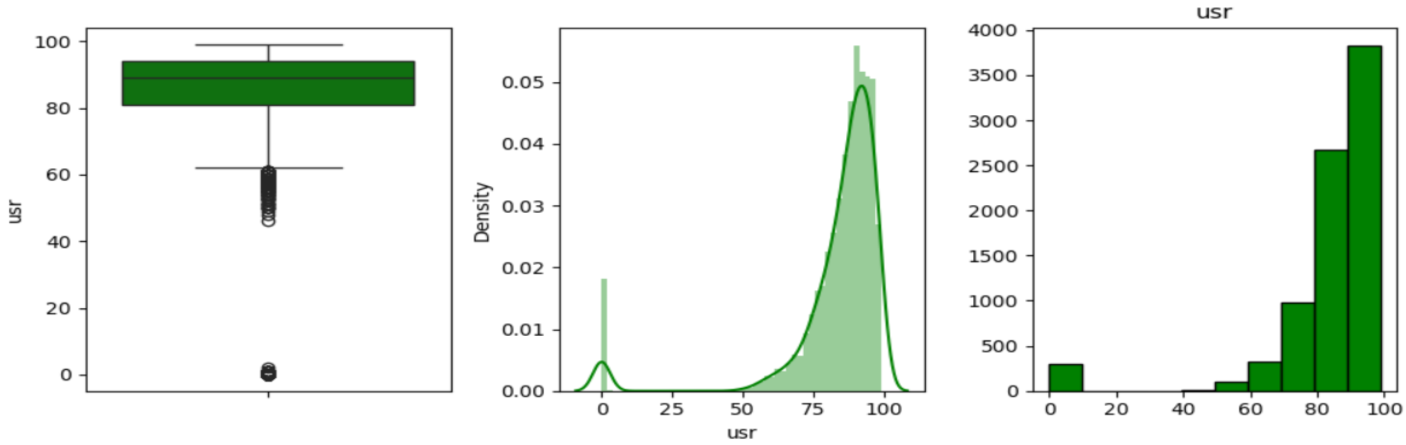


Fig: -4: Dependent variable usr

- The CPU runs in user mode 80% - 95% of the times or it stays idle. The transfer for read and write is very quick.

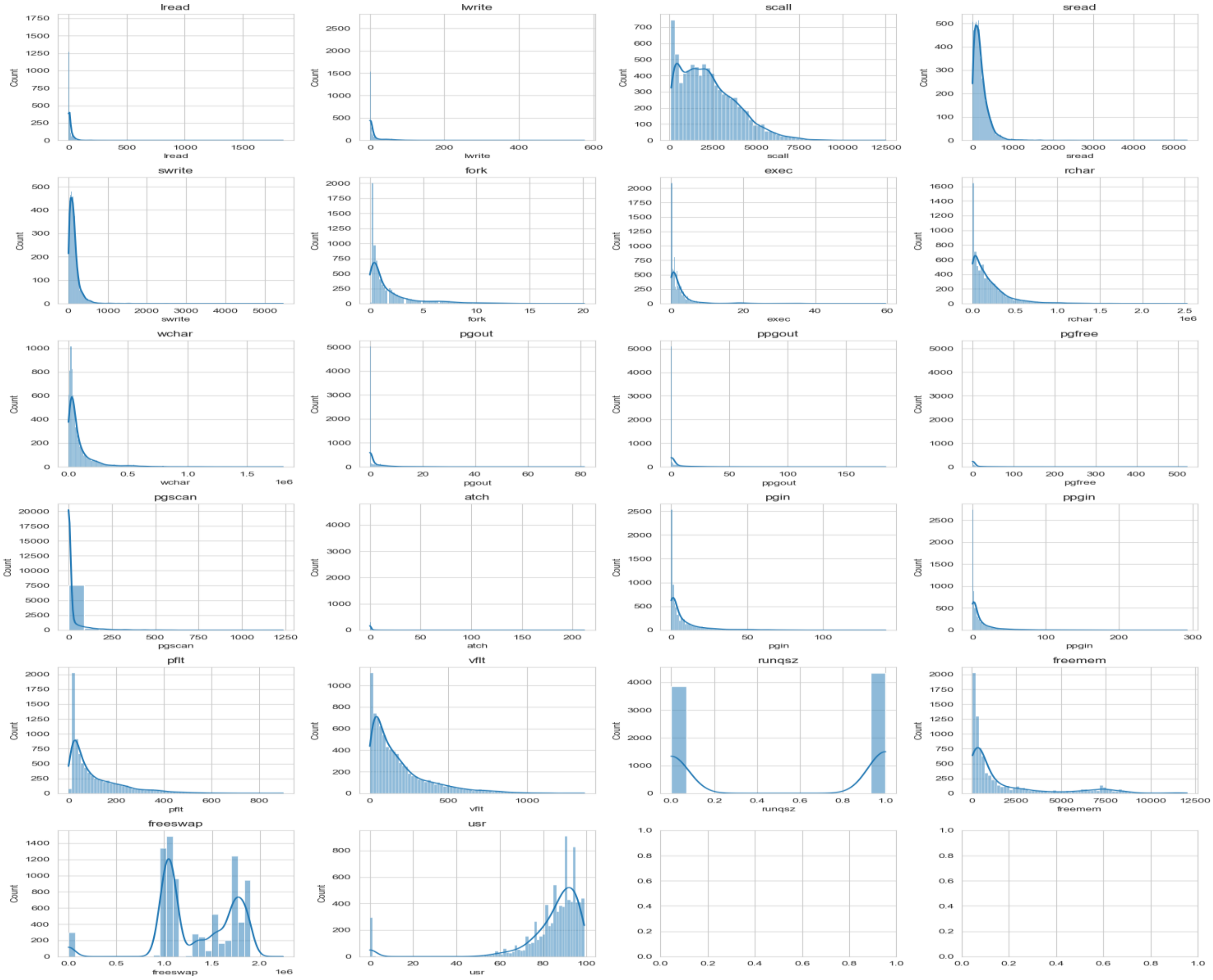


Fig: -5: distribution plot for variables

The histplot bear proof of the inherent skewness in the attributes. Most of the data is concentrated towards the starting part with a disproportionate amount concentrated on later part of the chart.

- The System read-write rate is under 5% which means this is also quick.
- Sread and Swrite is distributed nearly to 900

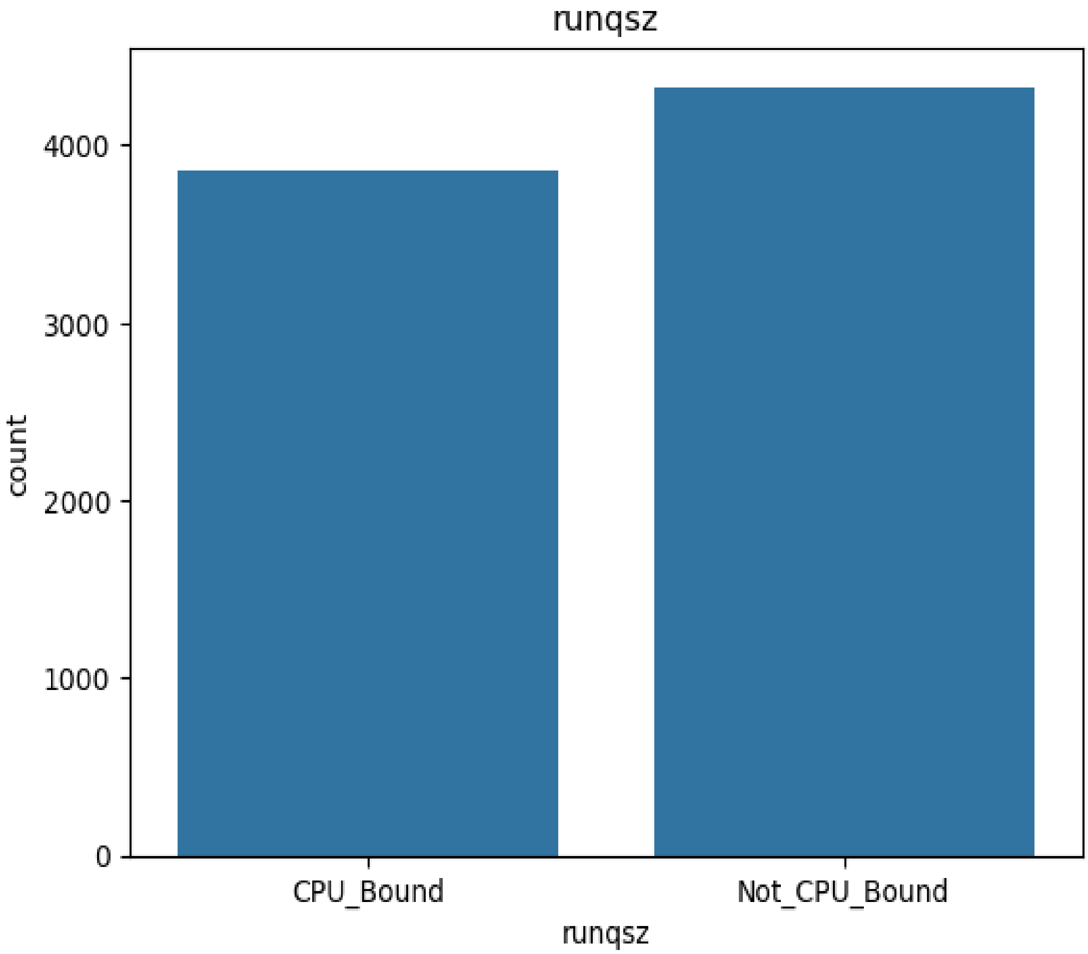


Fig: -6: countplot for runqsz

The above plot for the variables "runqsz" indicate there is a balance between the two values (CPU_Bond and Not_CPU_Bound). Further indicating absence of Imbalanced data for this particular variable

Bivariate Analysis:

We cannot clearly analyse from the below given pairplot chart because of higher number of attributes present in the data, however we can clearly make the observation that there are few attributes highly correlated in each other. Hence, to visualize the same we can obtain the below correlation plot.

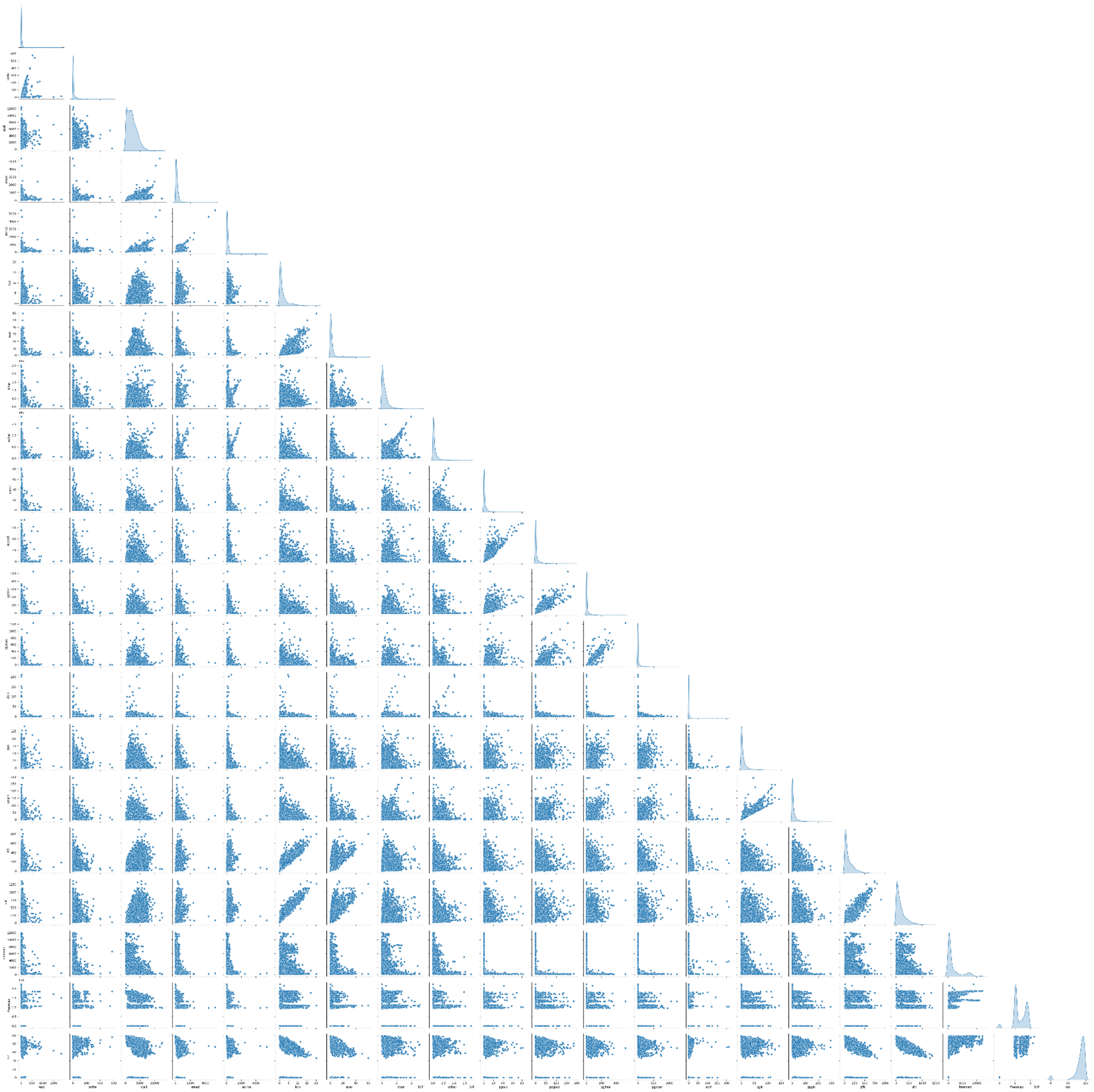


Fig: -7: Pairplot

From the below Heatmap, we can easily make out the following observations:

a) We have a higher positive correlation between "pflt" and "vflt" with "fork", "pgfree" against "ppgout", "pgscan" against "pgfree", "ppgin" against "pgin" and "vflt" against "pflt" with more than 0.9 in each of these cases. This indicates that there is strong linear dependence present in the variables.

b) There are also few attributes present with negative correlation, e.g., "vflt" against "usr" with at -0.42 which indicates strong linear dependence present in it. "scall" against "freemem", "pflt" against "usr" and "scall" against "freeswap" are having moderate linear dependence present in it.

c) It can be inferred from the that there are ample number of independent variables that are correlated with each other. A total of 15 combinations of variables can be noted to have correlations above 0.5(50%). Another thing can be noted with respect to the target variable is that all the variables except the runqsz(categorical), freemem and freeswap are negatively correlated and freeswap is said to have a strong positive correlation with the target(usr).

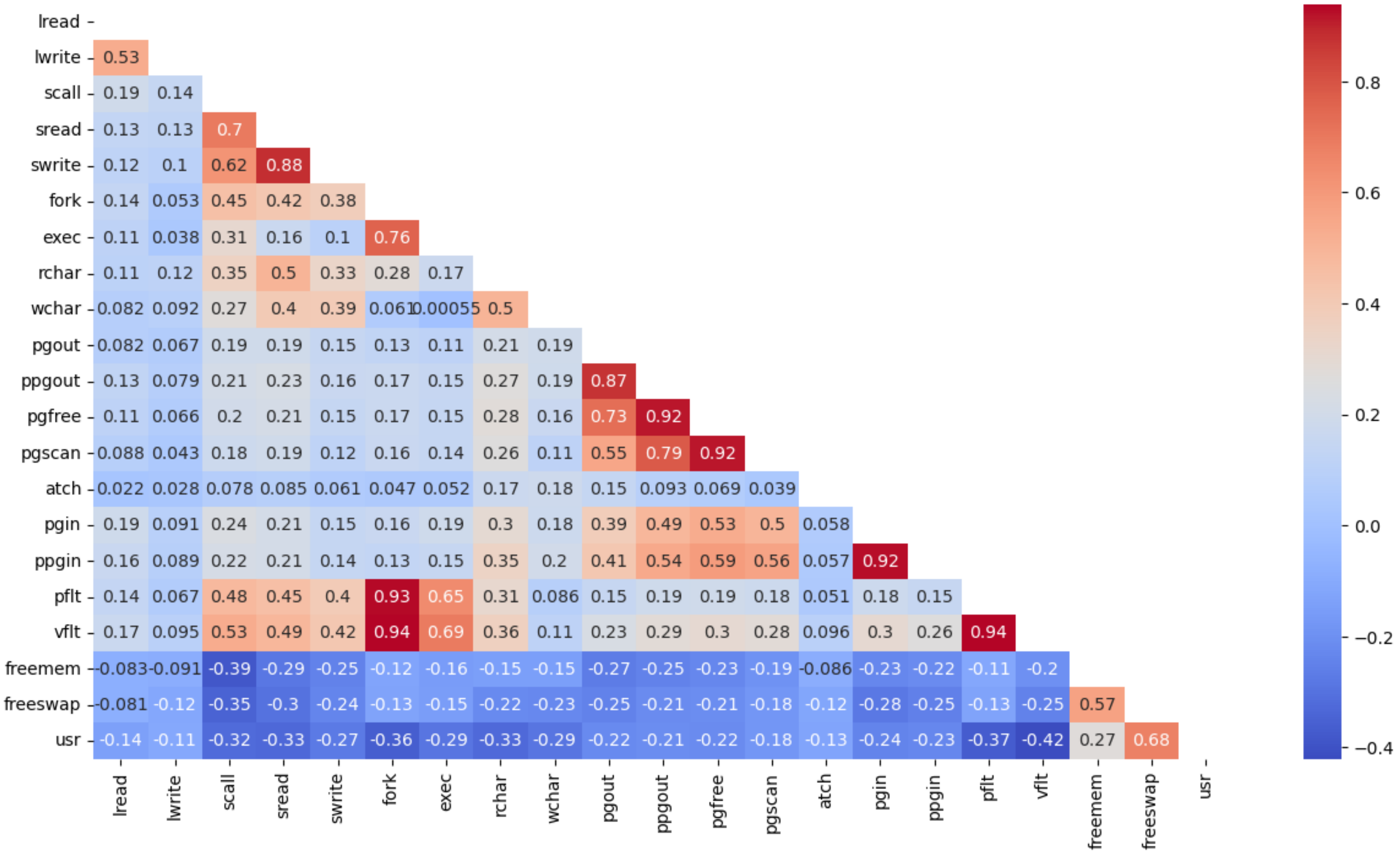


Fig: -8: Heatmap for Correlation analysis

1.2 Impute null values if present, also check for the values which are equal to zero. Do they have any meaning or do we need to change them or drop them? Check for the possibility of creating new features if required. Also check for outliers and duplicates if there.

- There are **some missing values** in variables **'rchar' (104 values) & 'wchar' (15 values)**, which were **treated** by **replacing them with Median**
- Upon checking for **0 values**, we found them in many variables. However, upon further looking at these, it is to be noted that these all are valid values these are related to the activities in the computer. Hence, **we do not need to drop them**
- There are **no duplicate rows present** in the data
- The new feature is not necessarily required here as these do not have any significant output due to presence of 0s.

- Multiple approach for building the model has been implemented like building the model with zeros, without zeros, with log transformation in the presence of zeros, penalizing the variables coefficient using Regularization methods.

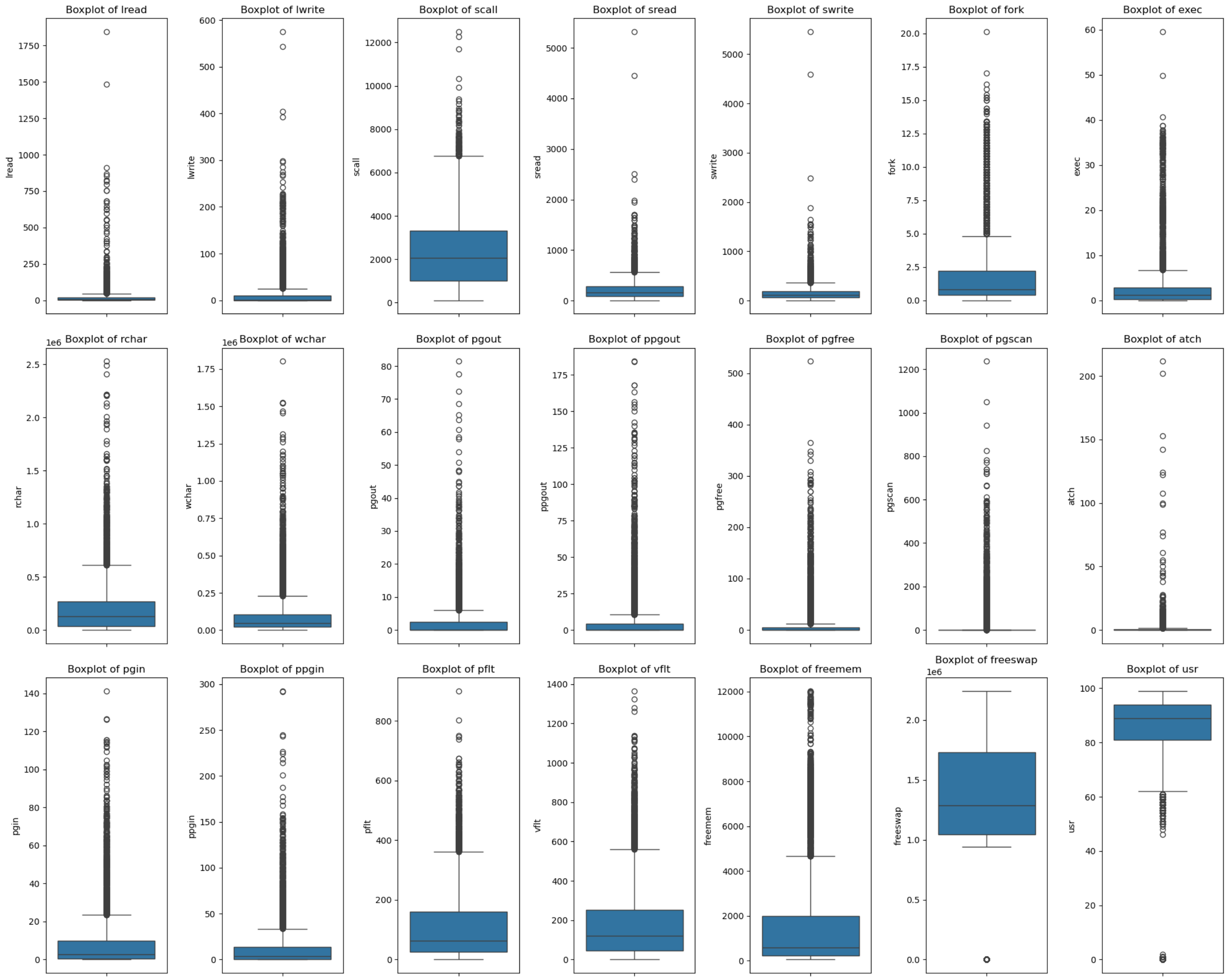


Fig: -9: Boxplot of all the numeric attributes before treating outliers

- Based on the above given Figure 8, it is evident that we have outliers across all the features.
- Let us do the treatment of the outlier basis limiting to Upper Range and Lower Range. Let us see the impact of this for all the features.

Treatment of Outlier basis limiting to Upper Range and Lower Range:

Though outliers can bring in error slightly more than what the non-outlier model would bring in, it is still best suited to go ahead with outliers if the outlying values turn out to be legitimate.

There is no change in multicollinearity Hence no need to drop the variable or change it we should change it because after change the variable it could change whole meaning of the variable. So, we should keep it as they are.

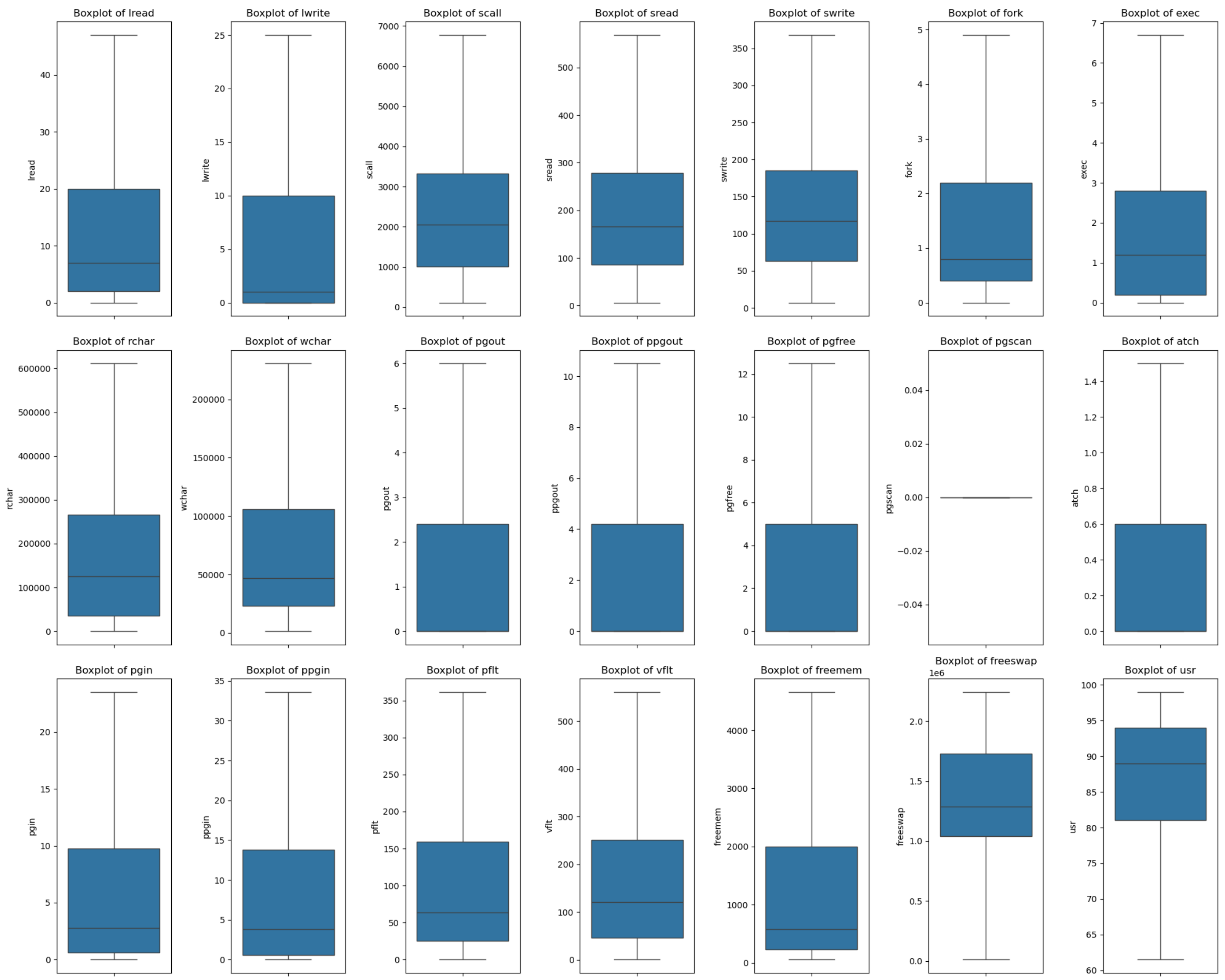
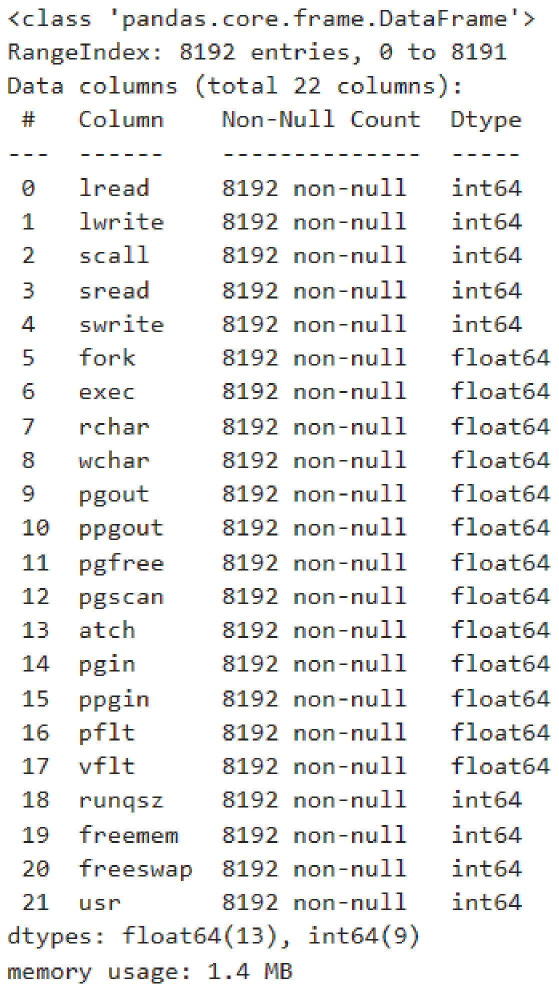


Fig: -10: Boxplot of numerical attributes after treating outlier

We can observe from the above plot that the outliers in the dataset have been treated.

1.3 Encode the data (having string values) for Modelling. Split the data into train and test (70:30). Apply Linear regression using scikit learn. Perform checks for significant variables using appropriate method from stats model. Create multiple models and check the performance of Predictions on Train and Test sets using R-square, RMSE & Adj R-square. Compare these models and select the best one with appropriate reasoning.

- **One Hot encoding** is done on the only '**Object**' types variable i**.e. 'runqsz**'.
- A new column is created, with 1 indicating that variable as CPU_bound and 0 as CPU_not_bound and this is how the extended variable's data looks.
- It is the most common format of dummy variable coding in which each category of the nominal variable is represented by either 1 or 0.



```
<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     8192 non-null   float64
 8   wchar     8192 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   int64  
 19  freemem   8192 non-null   int64  
 20  freeswap  8192 non-null   int64  
 21  usr       8192 non-null   int64  
dtypes: float64(13), int64(9)
memory usage: 1.4 MB
```

Fig: -11: Data info after encoding the data.

Splitting the data into Train and Test:

The dataset is split into a training and a test data set according to a random 70:30 allocation. Training data contains 5734 observations since 70% of 8192 is approximately 5734. A simple random sample (without replacement) of size 5734 is taken from the first 8192 positive integers, and the training dataset is formed by selecting the rows of data corresponding to these random numbers. The remaining (8192-5734=) 2458 rows will constitute the test dataset.

Checking the dimensions of the training and test data:

X_train (5734, 17)

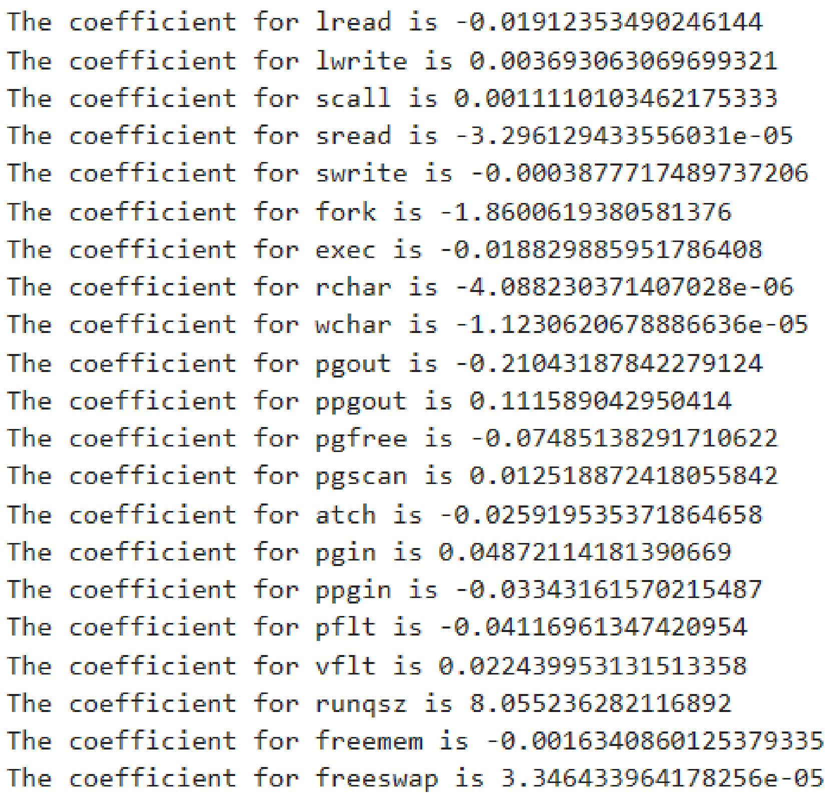
X_test (2458, 17)

y_train (5734, 1)

y_test (2458, 1)

Thus, the training dataset train df is created on which the candidate models will be built. The test dataset is test df on which the models will be validated by comparing their predictive ability.

Application of Linear Regression Model Using Sci-kit learn:



```
The coefficient for lread is -0.01912353490246144
The coefficient for lwrite is 0.003693063069699321
The coefficient for scall is 0.0011110103462175333
The coefficient for sread is -3.296129433556031e-05
The coefficient for swrite is -0.0003877717489737206
The coefficient for fork is -1.8600619380581376
The coefficient for exec is -0.018829885951786408
The coefficient for rchar is -4.088230371407028e-06
The coefficient for wchar is -1.1230620678886636e-05
The coefficient for pgout is -0.21043187842279124
The coefficient for ppgout is 0.111589042950414
The coefficient for pgfree is -0.07485138291710622
The coefficient for pgscan is 0.012518872418055842
The coefficient for atch is -0.025919535371864658
The coefficient for pgin is 0.04872114181390669
The coefficient for ppgin is -0.03343161570215487
The coefficient for pflt is -0.04116961347420954
The coefficient for vflt is 0.022439953131513358
The coefficient for runqsz is 8.055236282116892
The coefficient for freemem is -0.0016340860125379335
The coefficient for freeswap is 3.346433964178256e-05
```

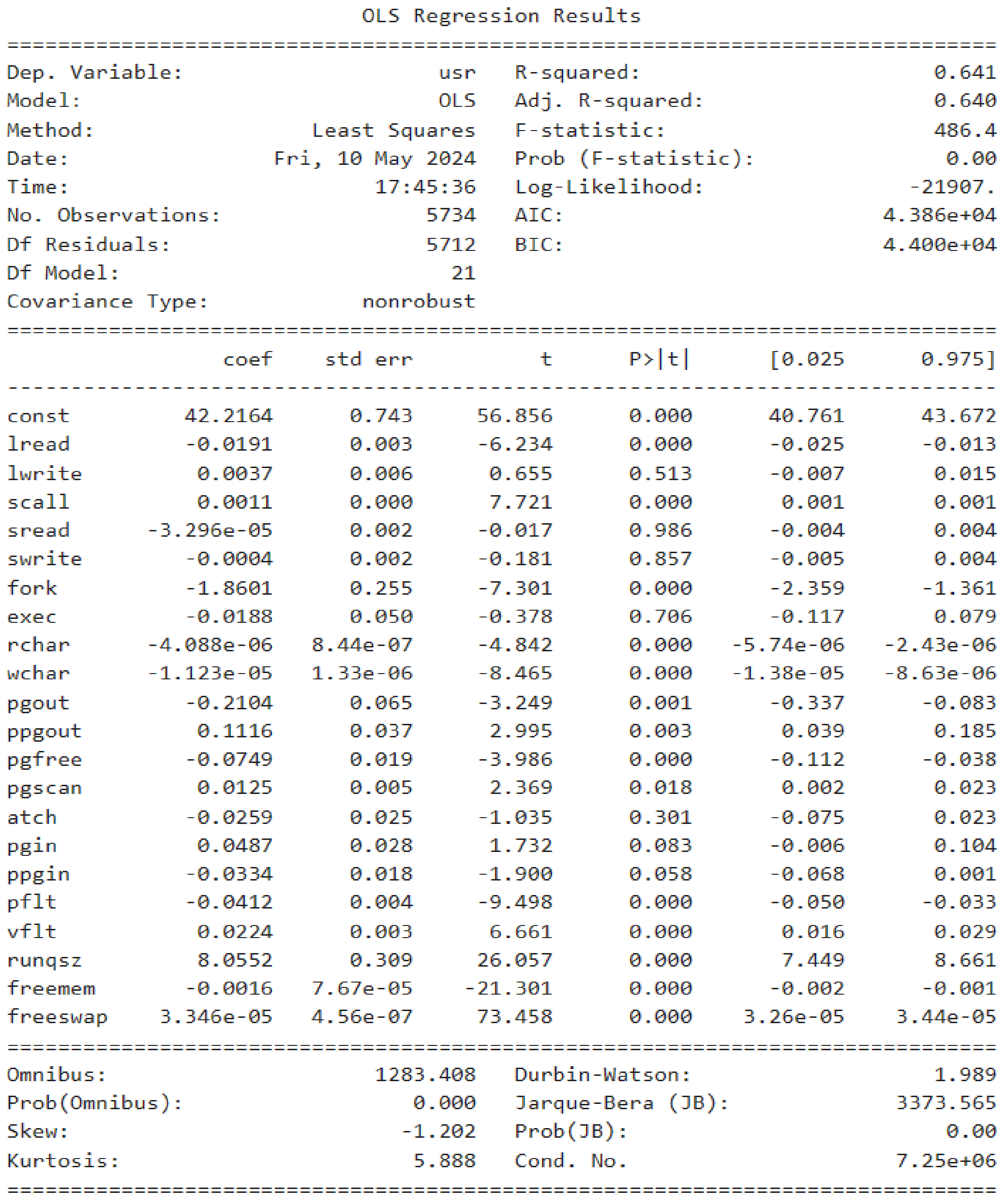
- The Intercept of our Model is: 42.21642790179804

- Basis the above co-efficient from the model, we have got R-Square value on training data at 0.6413678769409354. Similarly, the R-Square value on training data is at 0.6359429828421388.

In multiple regression, if one or more pairs of explanatory variables is highly correlated among themselves, then the phenomenon is known as multi-collinearity. Multi-collinearity is not desirable. It leads to inflated standard errors of the estimates of the regression coefficients, which in turn affects significance of the regression parameters. Often the signs of the regression coefficients may also change. As a result, the regression model becomes non-reliable or lacks interpretability.
The first thing one should do in multiple linear regression is, to check if multi-collinearity is present in the data.
Hence, to remove any multicollinearity we should now build the model using statsmodels (OLS).

Let us check the summary of first model after fitting the train data:

To proceed with this, we have first performed Linear Regression Model using OLS method on the main dataset i.e. taking all the variables. Then, we have reduced some variables due to high multicollinearity and then built the model using reduced number of variables.



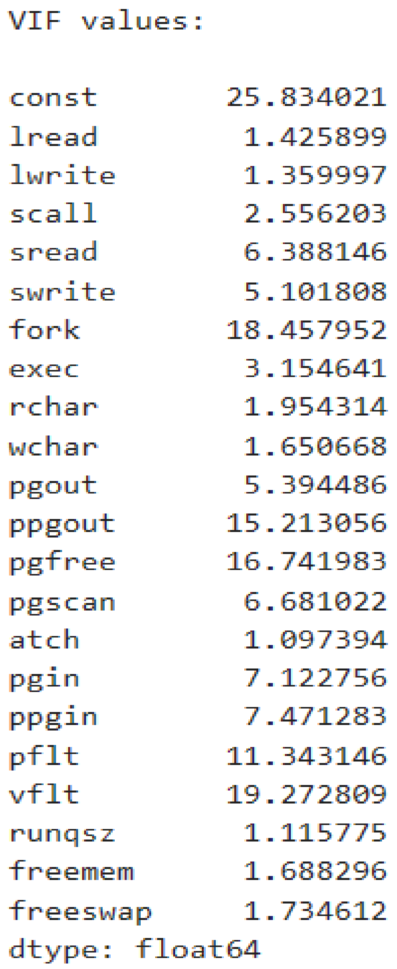
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.641
Model:                            OLS   Adj. R-squared:                  0.640
Method:                 Least Squares   F-statistic:                     486.4
Date:                Fri, 10 May 2024   Prob (F-statistic):               0.00
Time:                        17:45:36   Log-Likelihood:                -21907.
No. Observations:                5734   AIC:                         4.386e+04
Df Residuals:                    5712   BIC:                         4.400e+04
Df Model:                          21
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         42.2164      0.743     56.856      0.000      40.761      43.672
lread         -0.0191      0.003     -6.234      0.000      -0.025      -0.013
lwrite         0.0037      0.006      0.655      0.513      -0.007       0.015
scall          0.0011      0.000      7.721      0.000       0.001       0.001
sread      -3.296e-05      0.002     -0.017      0.986      -0.004       0.004
swrite        -0.0004      0.002     -0.181      0.857      -0.005       0.004
fork          -1.8601      0.255     -7.301      0.000      -2.359      -1.361
exec          -0.0188      0.050     -0.378      0.706      -0.117       0.079
rchar      -4.088e-06   8.44e-07     -4.842      0.000   -5.74e-06   -2.43e-06
wchar      -1.123e-05   1.33e-06     -8.465      0.000   -1.38e-05   -8.63e-06
pgout         -0.2104      0.065     -3.249      0.001      -0.337      -0.083
ppgout         0.1116      0.037      2.995      0.003       0.039       0.185
pgfree        -0.0749      0.019     -3.986      0.000      -0.112      -0.038
pgscan         0.0125      0.005      2.369      0.018       0.002       0.023
atch          -0.0259      0.025     -1.035      0.301      -0.075       0.023
pgin           0.0487      0.028      1.732      0.083      -0.006       0.104
ppgin         -0.0334      0.018     -1.900      0.058      -0.068       0.001
pflt          -0.0412      0.004     -9.498      0.000      -0.050      -0.033
vflt           0.0224      0.003      6.661      0.000       0.016       0.029
runqsz         8.0552      0.309     26.057      0.000       7.449       8.661
freemem       -0.0016   7.67e-05    -21.301      0.000      -0.002      -0.001
freeswap    3.346e-05   4.56e-07     73.458      0.000    3.26e-05    3.44e-05
==============================================================================
Omnibus:                     1283.408   Durbin-Watson:                   1.989
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3373.565
Skew:                          -1.202   Prob(JB):                         0.00
Kurtosis:                       5.888   Cond. No.                     7.25e+06
==============================================================================
```

Fig: - 12: Initial model of ols Regression Results

The data was split into X and y for Independent and dependent variables respectively, it was further distributed into respective train and test with a test size of 30% and a random state as 7 to obtain the model shown in the fig - 12. The random state 7 will be used for all the models.

From the fig - 12, the initial model (assumptions unchecked) infers few key observations. The R-squared and adjusted R-squared can be noted as 0.641 and 0.640 respective. This can be termed as a good observation as the model explains 64% of variance. However, the model is yet to be evaluated for its assumptions as we can see the presence of multicollinearity.

Since there is a presence of multicollinearity, the approach to solve the same was employed by VIF (Variance Inflation Factor). The Vif was calculated for the initial model and the variables having VIF larger than 10 was noted down and eliminated one by one and check for R-squared and adj. R-squared was conducted to make note of any changes in them. however, we need to check and remove the multicollinearity in the predictor variables. To do that let us check the VIF of the 1st model.



```
VIF values:

const       25.834021
lread        1.425899
lwrite       1.359997
scall        2.556203
sread        6.388146
swrite       5.101808
fork        18.457952
exec         3.154641
rchar        1.954314
wchar        1.650668
pgout        5.394486
ppgout      15.213056
pgfree      16.741983
pgscan       6.681022
atch         1.097394
pgin         7.122756
ppgin        7.471283
pflt        11.343146
vflt        19.272809
runqsz       1.115775
freemem      1.688296
freeswap     1.734612
dtype: float64
```

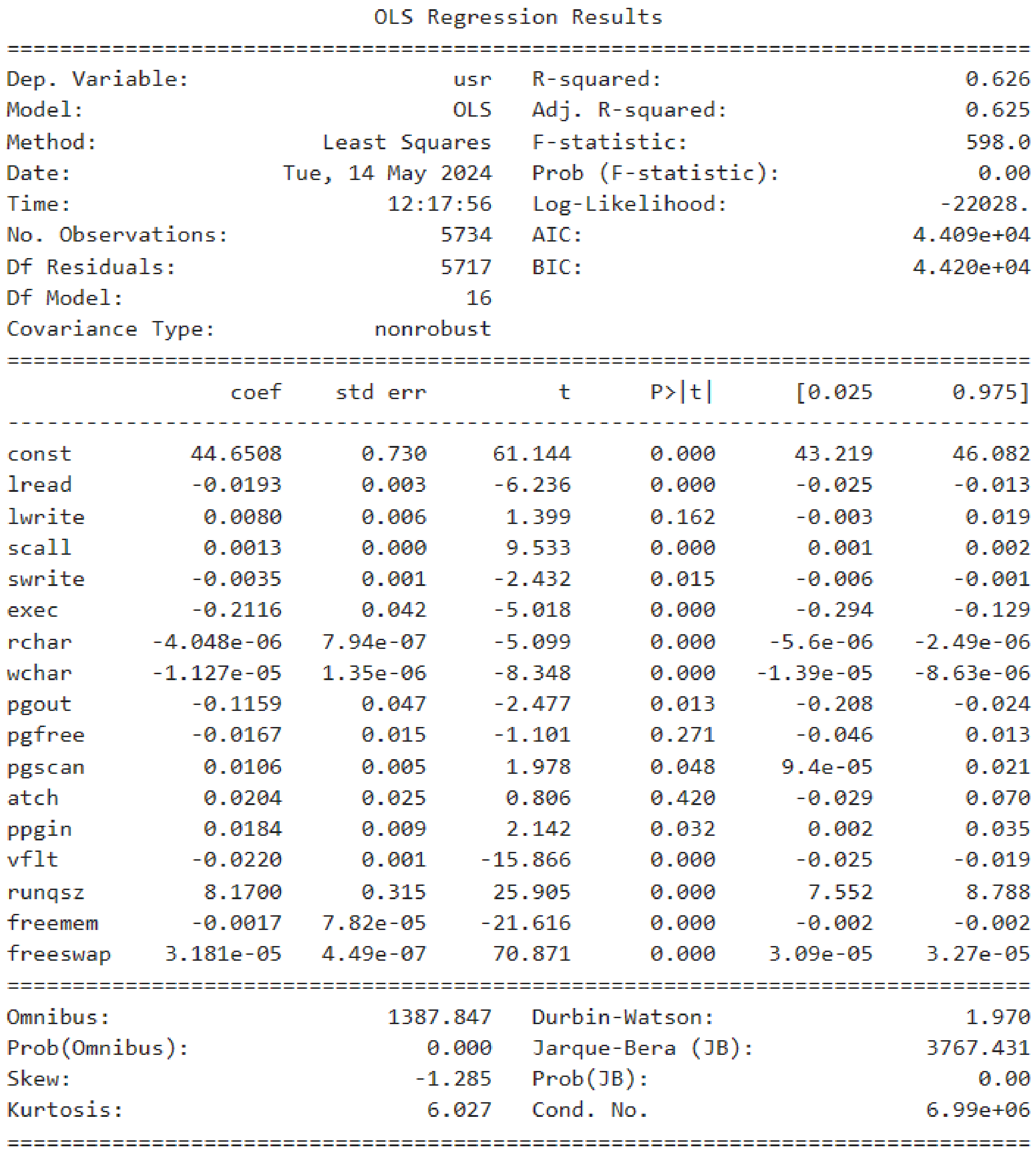
Fig: - 13: VIF values

We observe that among all the continuous predictors, “vflt”,” ppgout”, “pgin”, “ppgin”, “fork”, “pflt” have sufficiently high VIF (more than 10) indicating it is substantially correlated with the other predictor variables. Apart from these, there are other predictor variables having moderate correlation, for instance, “sread”, “lread”, “lwrite” and “swrite” which has VIF more than 5.
We will drop the variable that has the least impact on the adjusted R-squared of the model one by one.

Let's drop multicollinear columns one by one and observe the effect on our predictive model.

Since there is not significant effect on adj. R-squared after dropping the “vflt”, “ppgin”, “fork”, “sread”, “lread”, “swrite” and “exec” columns, we can remove it from the training set.

Noting down the R-squared and the Adjusted R-squared we can say dropping the variable "sread", "pflt" and "Pgin" would bring less or no decrease in the model efficiency.



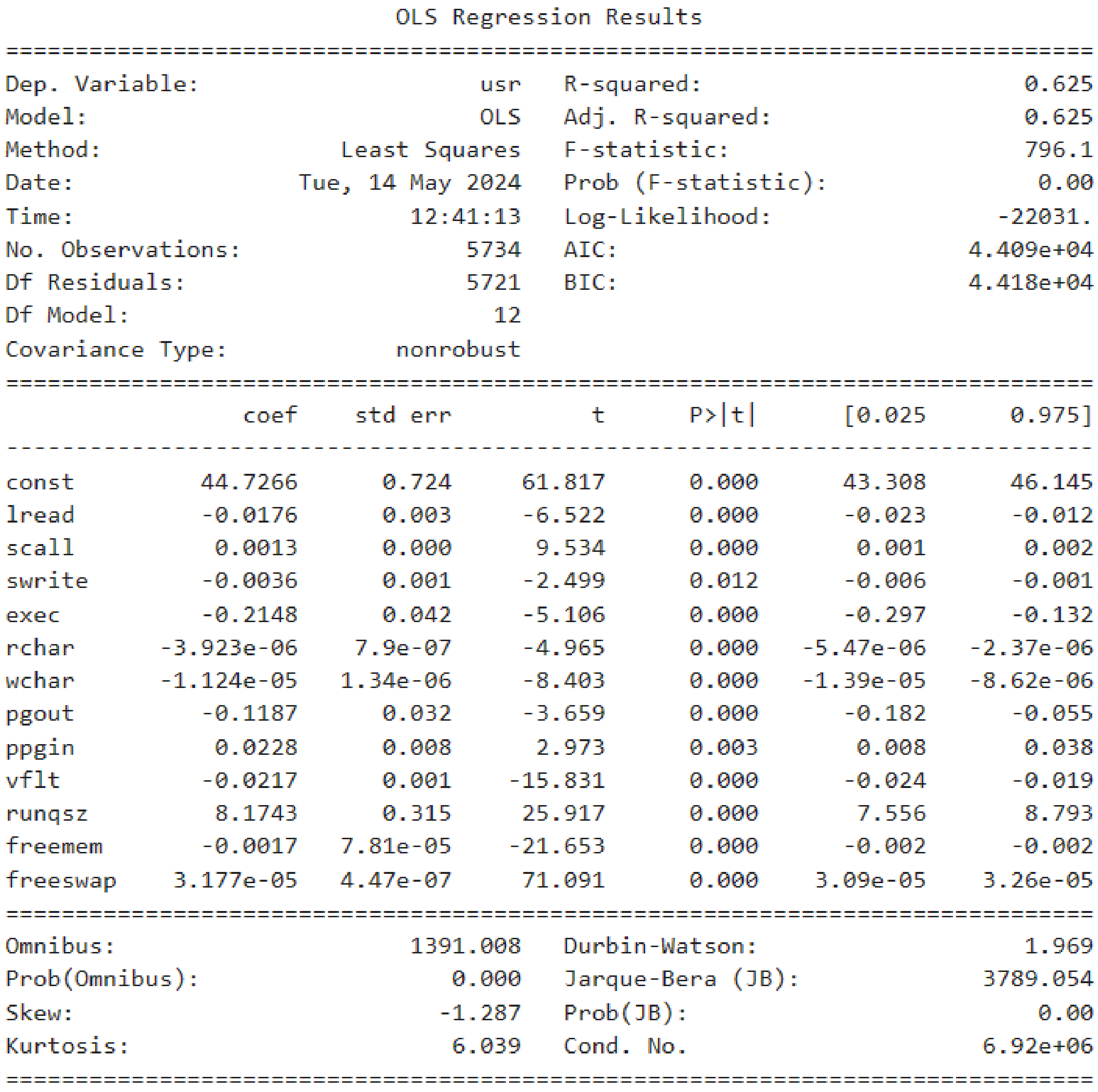
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.626
Model:                            OLS   Adj. R-squared:                  0.625
Method:                 Least Squares   F-statistic:                     598.0
Date:                Tue, 14 May 2024   Prob (F-statistic):               0.00
Time:                        12:17:56   Log-Likelihood:                -22028.
No. Observations:                5734   AIC:                         4.409e+04
Df Residuals:                    5717   BIC:                         4.420e+04
Df Model:                          16
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         44.6508      0.730     61.144      0.000      43.219      46.082
lread         -0.0193      0.003     -6.236      0.000      -0.025      -0.013
lwrite         0.0080      0.006      1.399      0.162      -0.003       0.019
scall          0.0013      0.000      9.533      0.000       0.001       0.002
swrite        -0.0035      0.001     -2.432      0.015      -0.006      -0.001
exec          -0.2116      0.042     -5.018      0.000      -0.294      -0.129
rchar      -4.048e-06   7.94e-07     -5.099      0.000    -5.6e-06   -2.49e-06
wchar      -1.127e-05   1.35e-06     -8.348      0.000   -1.39e-05   -8.63e-06
pgout         -0.1159      0.047     -2.477      0.013      -0.208      -0.024
pgfree        -0.0167      0.015     -1.101      0.271      -0.046       0.013
pgscan         0.0106      0.005      1.978      0.048     9.4e-05       0.021
atch           0.0204      0.025      0.806      0.420      -0.029       0.070
ppgin          0.0184      0.009      2.142      0.032       0.002       0.035
vflt          -0.0220      0.001    -15.866      0.000      -0.025      -0.019
runqsz         8.1700      0.315     25.905      0.000       7.552       8.788
freemem       -0.0017   7.82e-05    -21.616      0.000      -0.002      -0.002
freeswap    3.181e-05   4.49e-07     70.871      0.000    3.09e-05    3.27e-05
==============================================================================
Omnibus:                     1387.847   Durbin-Watson:                   1.970
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3767.431
Skew:                          -1.285   Prob(JB):                         0.00
Kurtosis:                       6.027   Cond. No.                     6.99e+06
==============================================================================
```

Fig: - 14: OLS model post VIF treatment

- Noting down the R-squared and the Adjusted R-squared we can say dropping the variable "sread", "pflt" and "Pgin" would bring less or no decrease in the model efficiency.

- If we now look at the VIF values, all the existing columns are now having less than 2 VIF scores which means we now do not have problems with multicollinearity.

- Based on the OLS regression summary table, we can see that “lwrite” and “atch” is having p-value at 0.420 and 0.162 which is >0.05 which means, this variable is not significant in building our model and we can drop it.



```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.625
Model:                            OLS   Adj. R-squared:                  0.625
Method:                 Least Squares   F-statistic:                     796.1
Date:                Tue, 14 May 2024   Prob (F-statistic):               0.00
Time:                        12:41:13   Log-Likelihood:                -22031.
No. Observations:                5734   AIC:                         4.409e+04
Df Residuals:                    5721   BIC:                         4.418e+04
Df Model:                          12                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         44.7266      0.724     61.817      0.000      43.308      46.145
lread         -0.0176      0.003     -6.522      0.000      -0.023      -0.012
scall          0.0013      0.000      9.534      0.000       0.001       0.002
swrite        -0.0036      0.001     -2.499      0.012      -0.006      -0.001
exec          -0.2148      0.042     -5.106      0.000      -0.297      -0.132
rchar      -3.923e-06    7.9e-07     -4.965      0.000   -5.47e-06   -2.37e-06
wchar      -1.124e-05   1.34e-06     -8.403      0.000   -1.39e-05   -8.62e-06
pgout         -0.1187      0.032     -3.659      0.000      -0.182      -0.055
ppgin          0.0228      0.008      2.973      0.003       0.008       0.038
vflt          -0.0217      0.001    -15.831      0.000      -0.024      -0.019
runqsz         8.1743      0.315     25.917      0.000       7.556       8.793
freemem       -0.0017   7.81e-05    -21.653      0.000      -0.002      -0.002
freeswap    3.177e-05   4.47e-07     71.091      0.000    3.09e-05    3.26e-05
==============================================================================
Omnibus:                     1391.008   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3789.054
Skew:                          -1.287   Prob(JB):                         0.00
Kurtosis:                       6.039   Cond. No.                     6.92e+06
==============================================================================
```

Fig: -15: Final OLS Model

The R-squared and adj. R-squared remain un-altered post dropping **lwrite** as it is clearly visible in the Fig-15 consisting of final OLS model. In this OLS model the assumption of multicollinearity was eliminated and the feature importance was taken into consideration.

The final **R-squared and Adj. R-squared** remain **0.625**, explaining **62.5%** of the **variance**. There are few more assumptions that needs to be satisfied so as to conclude the satisfaction of the result the model will predict.

Assumptions of Linear Regression :

These assumptions are essential conditions that should be met before we draw inferences regarding the model estimates or use the model to make a prediction.

For Linear Regression, we need to check if the following assumptions

hold: -

a. Linearity

b. Independence

c. Normality of error terms

d. Homoscedasticity

e. No strong Multicollinearity

Let us check the assumptions against the model that we have finalized as 3rd version shown in Fig-15.

Linearity and Independence test:

- Linearity describes a straight-line relationship between two variables, predictor variables must have a linear relation with the dependent variable.

- A plot of fitted values vs residuals, if they don't follow any pattern (the curve is a straight line), then we say the model is linear otherwise model is showing signs of non-linearity.

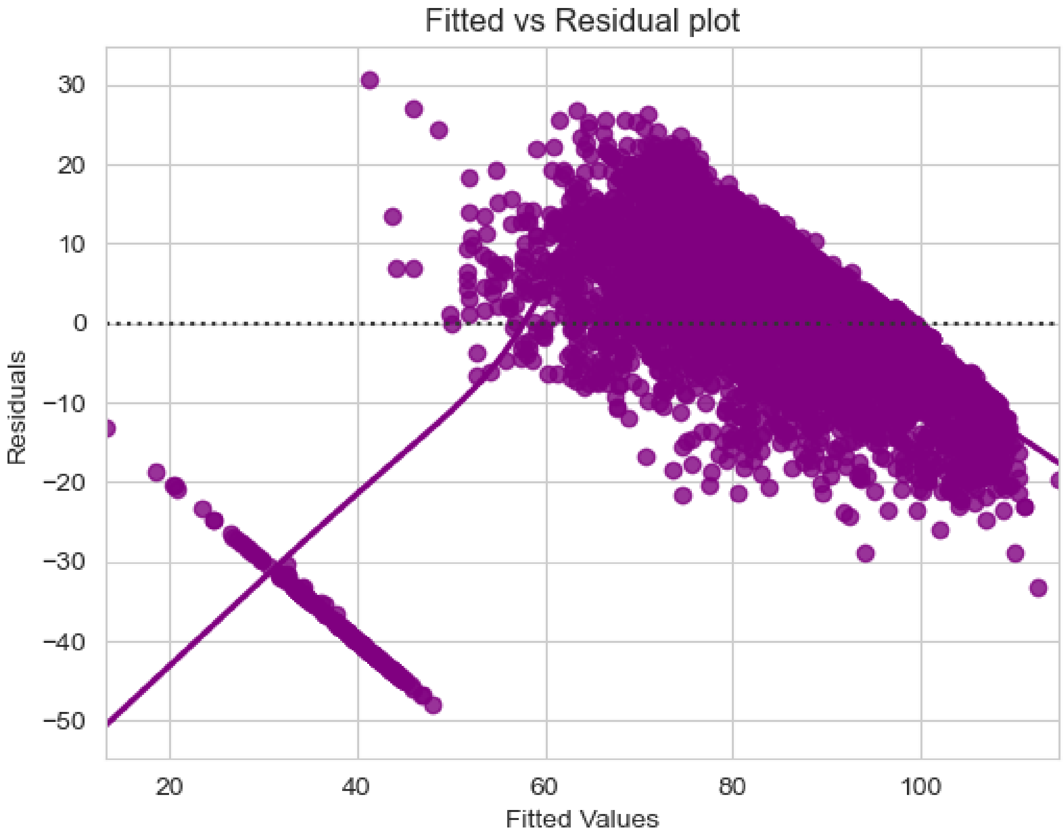


Fig: -16: Fitted Vs. Residual Plot for Linearity

The above plot (Figure 16) of fitted values vs residuals follow any pattern (the curve is not straight line), then we say the model is almost not linear. Achieving linear model can be a challenging task as perfect linear sometimes can be impossible. However, a perfect nonlinear like the above plot is achievable as they represent randomly distributed.

Test for Normality:

- Error terms/residuals should be normally distributed.

- If the error terms are not normally distributed, confidence intervals may become too wide or narrow. Once confidence interval becomes unstable, it leads to difficulty in estimating coefficients based on minimization of least squares.

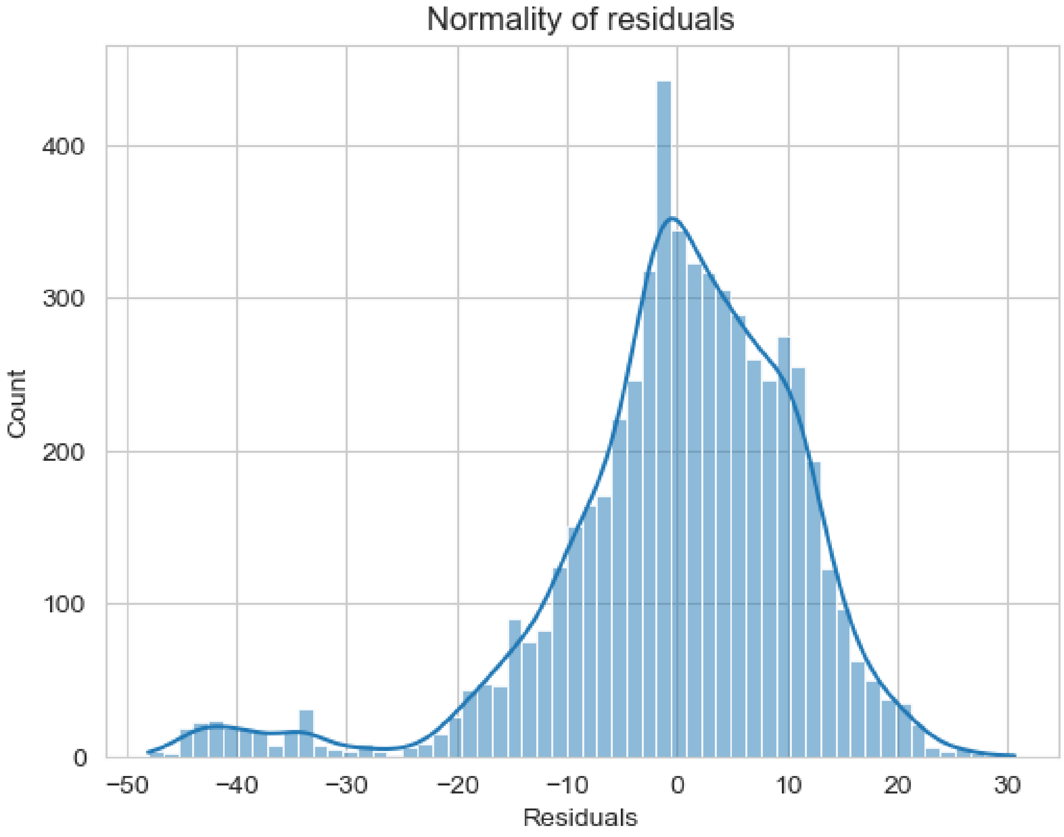


Fig: - 17: Normality of residuals

The above visual representation tells that the errors are normally distributed. It also suggests to some extent it is skewed towards its left. Since the model is built without outlier treatment considering the outliers are a legitimate value, the slight skewness could be a result of it. However, we will try solving the data with other approaches.

The **QQ plot** of residuals can be used to visually check the normality assumption. The normal probability plot of residuals should approximately follow a straight line.

Most of the points are lying on the straight line in QQ plot. There can be few exceptions as suggested earlier getting a full perfect model can be highly challenging especially without domain intervention. However, the above QQ plot could satisfy the need.

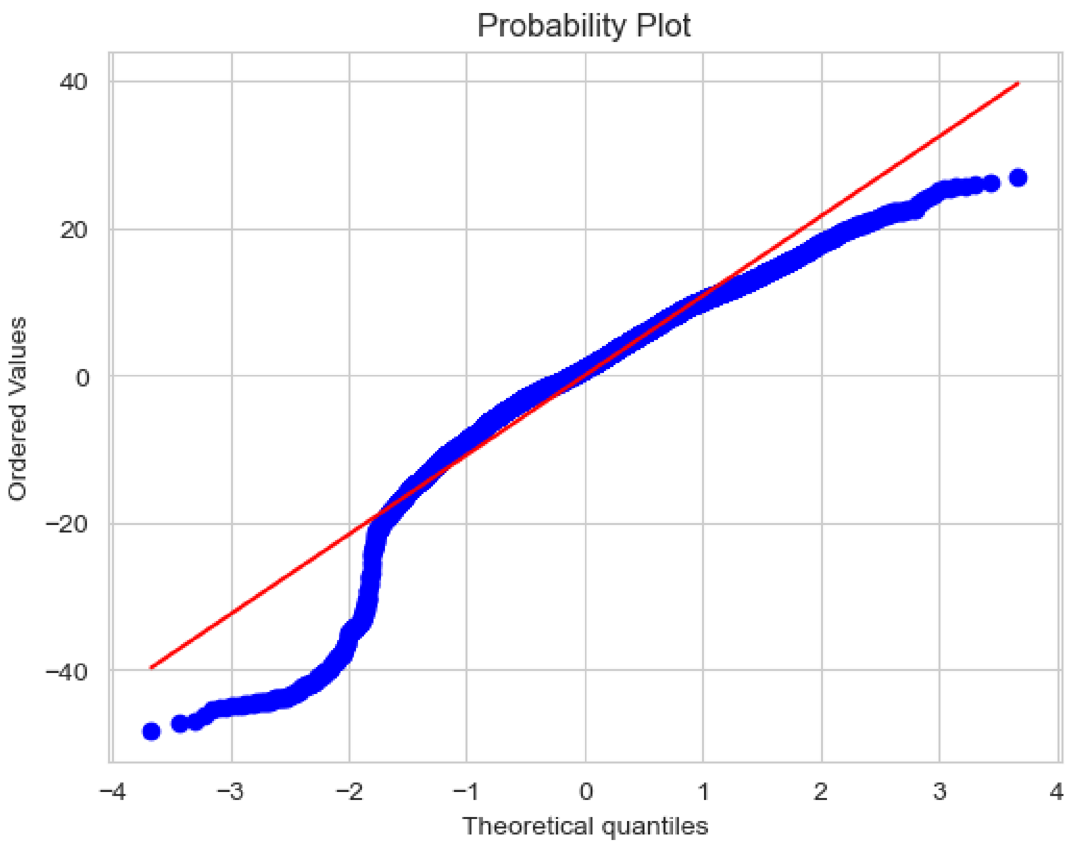
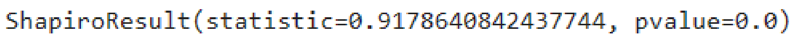


Fig: - 18: QQ Plot for normality

The Shapiro-Wilk test can also be used for checking the normality. The null and alternate hypotheses of the test are as follows:

- Null hypothesis - Data is normally distributed.
- Alternate hypothesis - Data is not normally distributed.

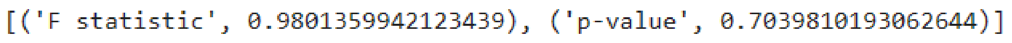


```
ShapiroResult(statistic=0.9178640842437744, pvalue=0.0)
```

- Since p-value < 0.05, the residuals are not normal as per Shapiro test.
- Strictly speaking - the residuals are not normal. However, as an approximation, we might be willing to accept this distribution as close to being normal

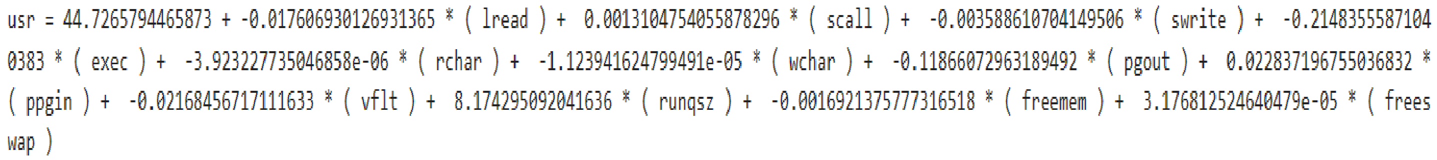
The null and alternate hypotheses of the **goldfeldquandt test** are as follows:

- Null hypothesis: Residuals are homoscedastic
- Alternate hypothesis: Residuals have heteroscedasticity.



```
[('F statistic', 0.9801359942123439), ('p-value', 0.7039810193062644)]
```

- Since p-value > 0.05 we can say that the **residuals are homoscedastic.**



```
usr = 44.7265794465873 + -0.017606930126931365 * ( lread ) +  0.0013104754055878296 * ( scall ) +  -0.003588610704149506 * ( swrite ) +  -0.2148355587104
0383 * ( exec ) +  -3.923227735046858e-06 * ( rchar ) +  -1.123941624799491e-05 * ( wchar ) +  -0.11866072963189492 * ( pgout ) +  0.022837196755036832 *
( ppgin ) +  -0.021684567171116 33 * ( vflt ) +  8.17429509204163 6 * ( runqsz ) +  -0.0016921375777316518 * ( freemem ) +  3.176812524640479e-05 * ( frees
wap )
```

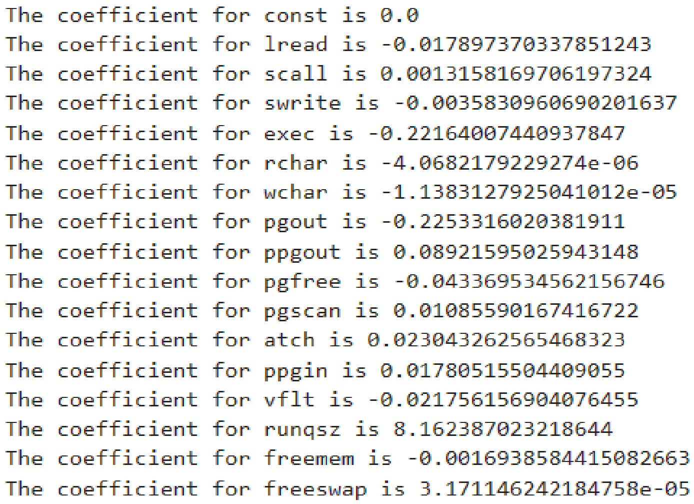
We have also applied the same to the test data and below is the summary of the same:

- The Root Mean Square Error (RMSE) of the model is for the training set is 11.2831235.

The Root Mean Square Error (RMSE) of the model is for the testing set is 11. 329079759988172

- The Mean Absolute Error (MAE) on the train data is 8.150261.
- The Mean Absolute Error (MAE) on the test data is 8.185690.

Using Linear Model from Sci-kit learn library, now we get the final model with the below coefficients:



```
The coefficient for const is 0.0
The coefficient for lread is -0.017897370337851243
The coefficient for scall is 0.0013158169706197324
The coefficient for swrite is -0.0035830960690201637
The coefficient for exec is -0.22164007440937847
The coefficient for rchar is -4.0682179229274e-06
The coefficient for wchar is -1.1383127925041012e-05
The coefficient for pgout is -0.2253316020381911
The coefficient for ppgout is 0.08921595025943148
The coefficient for pgfree is -0.043369534562156746
The coefficient for pgscan is 0.01085590167416722
The coefficient for atch is 0.023043262565468323
The coefficient for ppgin is 0.01780515504409055
The coefficient for vflt is -0.021756156904076455
The coefficient for runqsz is 8.162387023218644
The coefficient for freemem is -0.0016938584415082663
The coefficient for freeswap is 3.171146242184758e-05
```

The coefficient of determination R^2 of the prediction on Train set 0.626193156572201.

The coefficient of determination R^2 of the prediction on Test set 0. 0.6181741187969207.

RMSE Score:

Training Score: 11.272062678395427

Test Score: 11.317997252568398

Let us see the regression plot of the test and train data with actual and predicted values below:

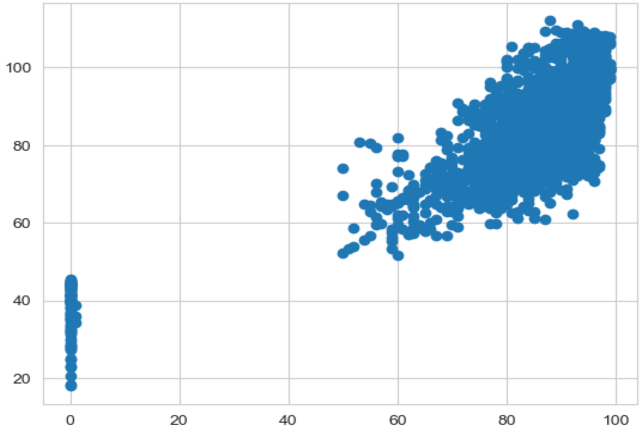


Fig: - 19: regression plot of test data on actual vs. predicted usr

The plot represents the **Actual vs Predicted** graph of a randomly chosen 100 records. The blue represents the Actual records and the red represents the Predicted records. We can now see a very minimal deviation when compared each other representing the model is doing a perfect job in predicting the values.

- We can see that RMSE on the train and test sets are comparable. So, our model is not suffering from overfitting.
- Hence, we can conclude the model "OLS" is good for prediction as well as inference purposes.

1.4 Inference: Basis on these predictions, what are the business insights and recommendations. Please explain and summaries the various steps performed in this project. There should be proper business interpretation and actionable insights present.

a) A unit increase in the Number of writes (transfers per second) between system memory and user memory will result in a 0.0396 unit decrease in the usr, all other variables remaining constant.

b) The usr of a runaqz of non-CPU bound will be 1.7334 units higher than a runsqz of CPU Bound, all other variables remaining constant.

c) Most of the co-efficient are negative, for instance, Number of writes (transfers per second) between system memory and user memory will have negative impact in usr.

d) We can see that RMSE on the train and test sets are comparable. So, our model is not suffering from overfitting.

e) MAE indicates that our current model is able to predict usr within a mean error of 8.18 units on the test data.

f) Hence, we can conclude the model is good for prediction as well as inference purposes. The comparability of RMSE for train data and test data suggests that our final model is unbiased and hence a good fit

Insights

a) There is a Decrement in Portion of time (%) that cpus run in user mode by a larger factor if the Number of characters transferred per second by system write calls increases.

b) There is a Decrement in Portion of time (%) that cpus run in user mode by a larger factor if the Number of page-in requests per second increases.

c) More the number of disk blocks available for page swapping's the Portion of time (%) that cpus run in user mode goes down a little.

d) 1 unit increase in number of disk blocks available for page swapping (freeswap) leads to a very less unit decrease in portion of time that CPUs run in user mode (usr).

e) 1 unit increase in number of memory pages available to user processes (freemem) leads to a 0.00169 times decrease in portion of time that CPUs run in user mode (usr).

Problem 2 – Logistic Regression, LDA and CART:

Executive Summary:

Republic of Indonesia Ministry of Health, has entrusted us 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.

The problem is to predict do/don't they use a contraceptive method of choice based on their demographic and socio-economic characteristics.

The data collected has so many variables thus making it difficult to find useful details without using Data Science Techniques. You are tasked to perform detailed EDA and identify Optimum Principal Components that explains the most variance in data. Use Sklearn only.

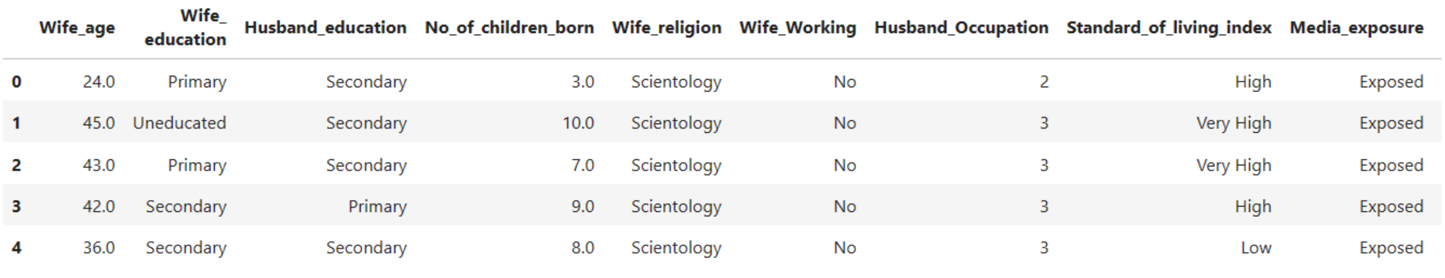
Data Dictionary:

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=very low, 2, 3, 4=high
9. Media exposure (binary) Good, not good
10. Contraceptive method used (class attribute) No, Yes

2.1 Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, check for duplicates and outliers and write an inference on it. Perform Univariate and Bivariate Analysis and Multivariate Analysis.

Performing the Exploratory Data Analysis (EDA)

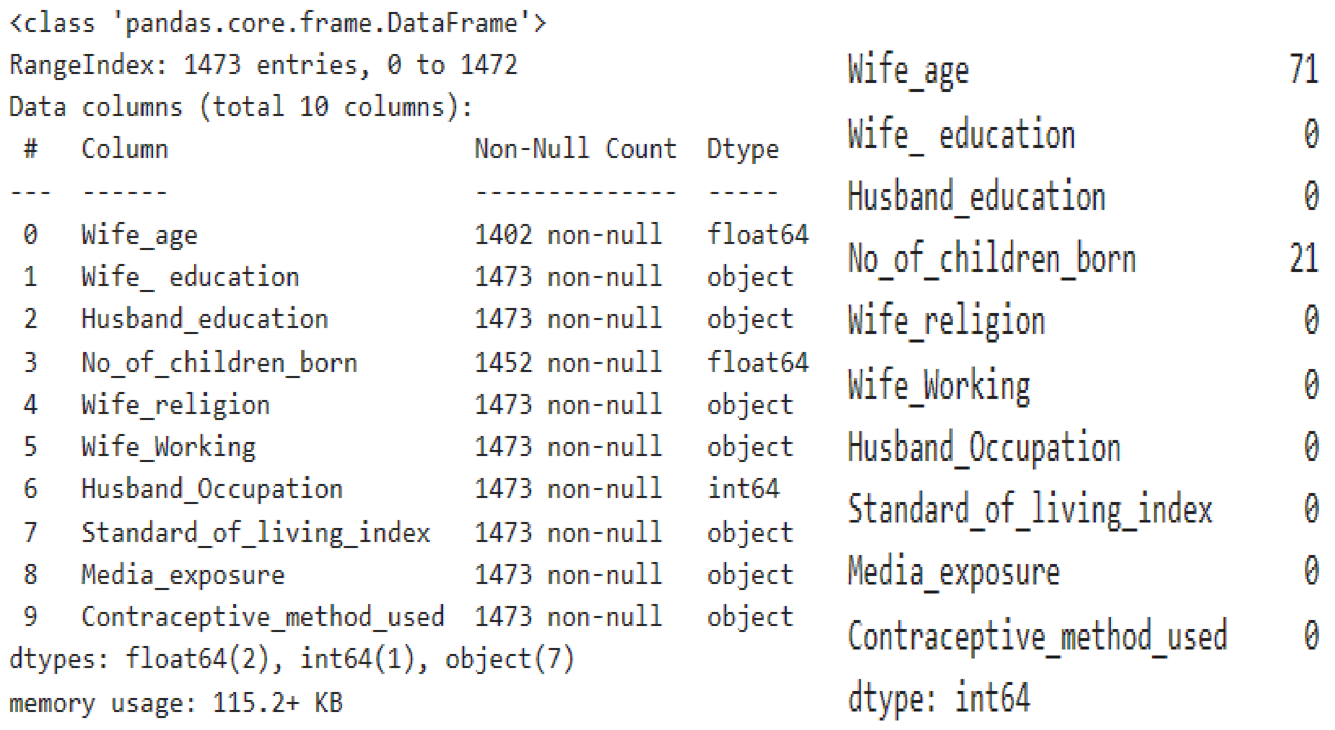
- The data has **1473** rows and **10** columns.
- There are 7 object type data types, 1 Int & 2 float data types
- First 5 values of the data set are as below:-



	Wife_age	Wife_ education	Husband_education	No_of_children_born	Wife_religion	Wife_Working	Husband_Occupation	Standard_of_living_index	Media_exposure
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

Fig: -20: Dataset head

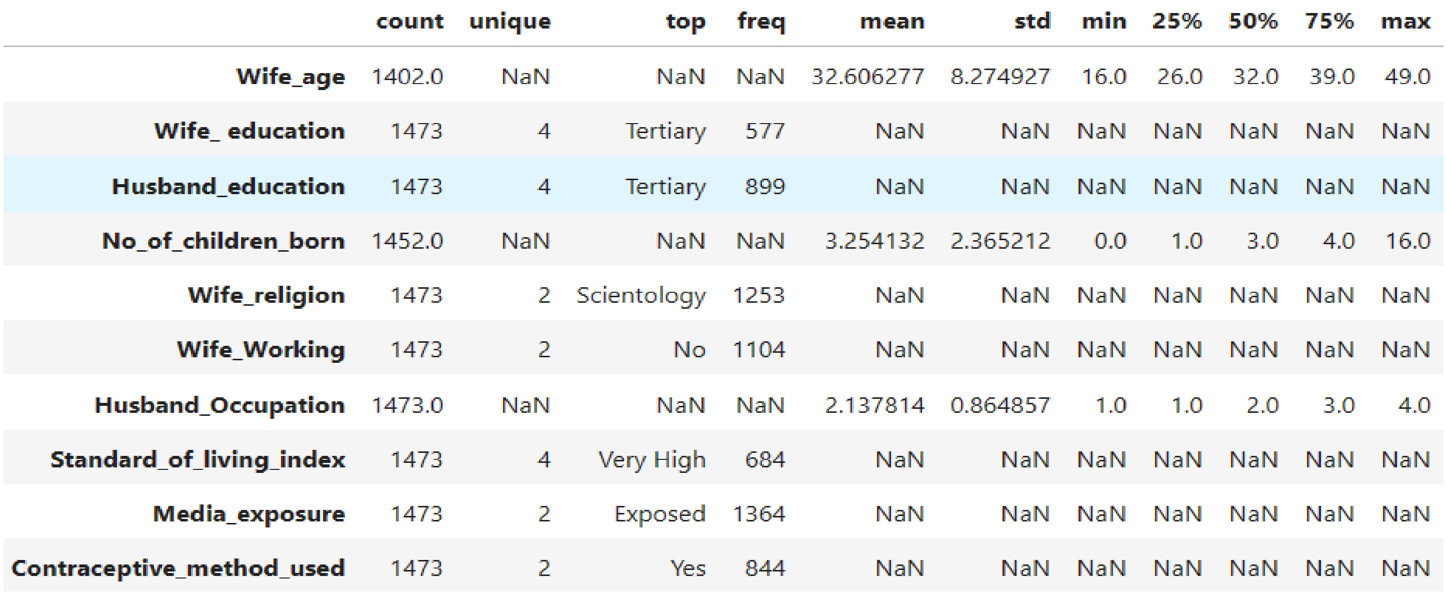
- **Missing** values in **'Wife age'** and **'No_of_children_born'** treated using median imputation.
- **85 duplicate** rows present in the dataset.
- **Majority of women follow Scientology** and **are not working**.
- **Tertiary education** is the **most common** level for **both husbands and wives.**
- **Most husbands** work in **level 3 occupations**.
- **Majority** of **women** have **used contraceptives**.
- **High standard of living** and **media exposure** suggest **urban residency**.
- **Most families have 1 or 2 children**, but some have over 15.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1473 entries, 0 to 1472
Data columns (total 10 columns):
 #   Column                     Non-Null Count  Dtype  
---  ------                     --------------  -----  
 0   Wife_age                   1402 non-null   float64
 1   Wife_ education            1473 non-null   object 
 2   Husband_education          1473 non-null   object 
 3   No_of_children_born        1452 non-null   float64
 4   Wife_religion              1473 non-null   object 
 5   Wife_Working               1473 non-null   object 
 6   Husband_Occupation         1473 non-null   int64  
 7   Standard_of_living_index   1473 non-null   object 
 8   Media_exposure             1473 non-null   object 
 9   Contraceptive_method_used  1473 non-null   object 
dtypes: float64(2), int64(1), object(7)
memory usage: 115.2+ KB
```

```
Wife_age                     71
Wife_ education               0
Husband_education             0
No_of_children_born          21
Wife_religion                 0
Wife_Working                  0
Husband_Occupation            0
Standard_of_living_index      0
Media_exposure                0
Contraceptive_method_used     0
dtype: int64
```

Fig: -21: Dataset info and Null Values



	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Wife_age	1402.0	NaN	NaN	NaN	32.606277	8.274927	16.0	26.0	32.0	39.0	49.0
Wife_ education	1473	4	Tertiary	577	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Husband_education	1473	4	Tertiary	899	NaN	NaN	NaN	NaN	NaN	NaN	NaN
No_of_children_born	1452.0	NaN	NaN	NaN	3.254132	2.365212	0.0	1.0	3.0	4.0	16.0
Wife_religion	1473	2	Scientology	1253	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Wife_Working	1473	2	No	1104	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Husband_Occupation	1473.0	NaN	NaN	NaN	2.137814	0.864857	1.0	1.0	2.0	3.0	4.0
Standard_of_living_index	1473	4	Very High	684	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Media_exposure	1473	2	Exposed	1364	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Contraceptive_method_used	1473	2	Yes	844	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Fig: -22: Dataset Describe

To proceed further, we have replaced the missing values with median values of the columns for Wife_age and Husband_Occupation.

Also, there are 85 duplicated rows from the data set.

Univariate Analysis:

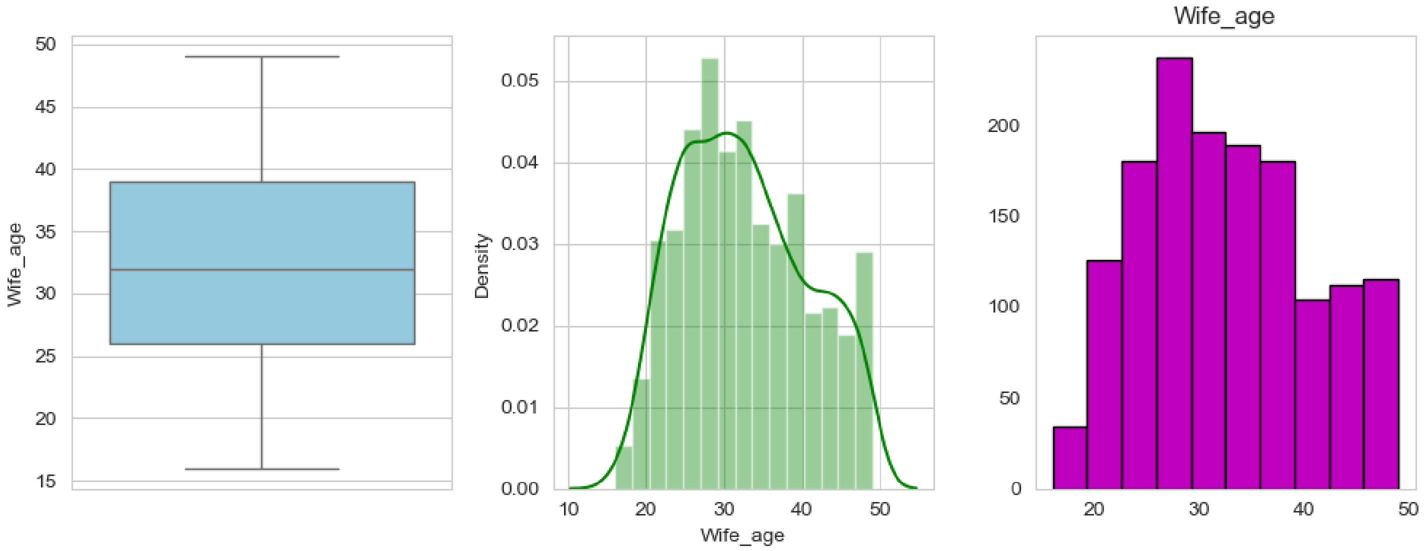


Fig: -23: Age of wife

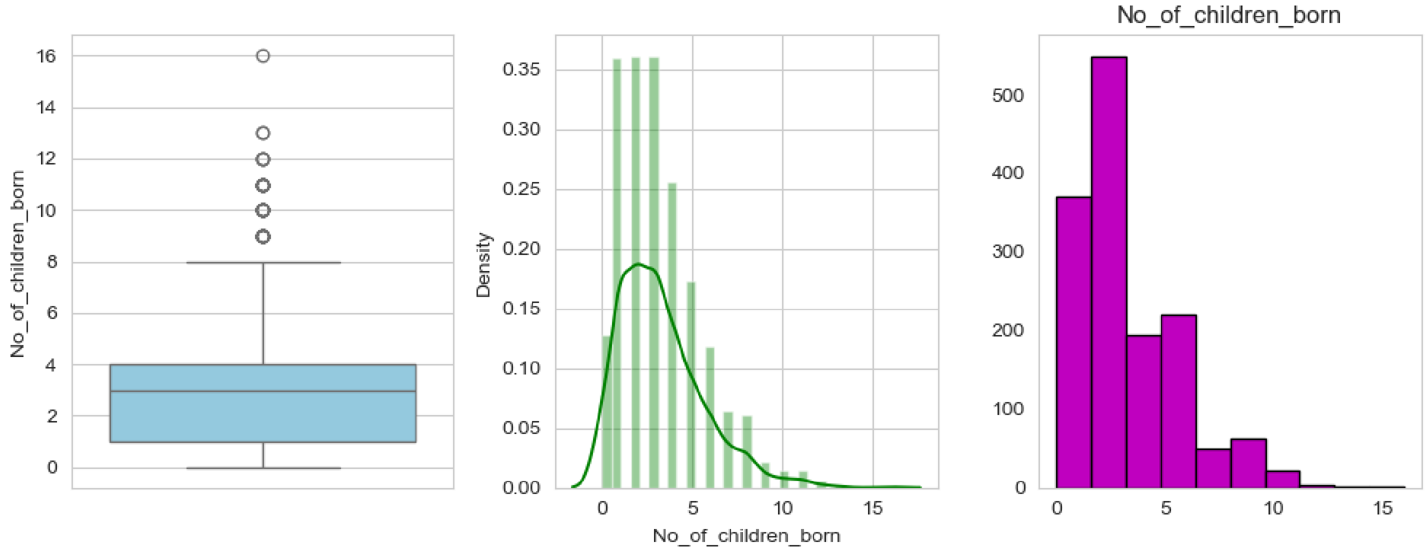


Fig: -24: No of Children

Observations:

a) The average age of wife is 32 with similar median age. The data consists of minimum age of 16 with maximum age of 49. The distribution is slightly right skewed which is understandable given the problem statement we have.

b) 25% of the women are having 1 or no child. Moreover, 25% of the women having 5 or more children.

c) Average number of children born is at ~3 with median similar in this case too.

d) It seems, we have few outliers for Number of Children born as the maximum count is at 16.

Let us now check the categorical variables below:

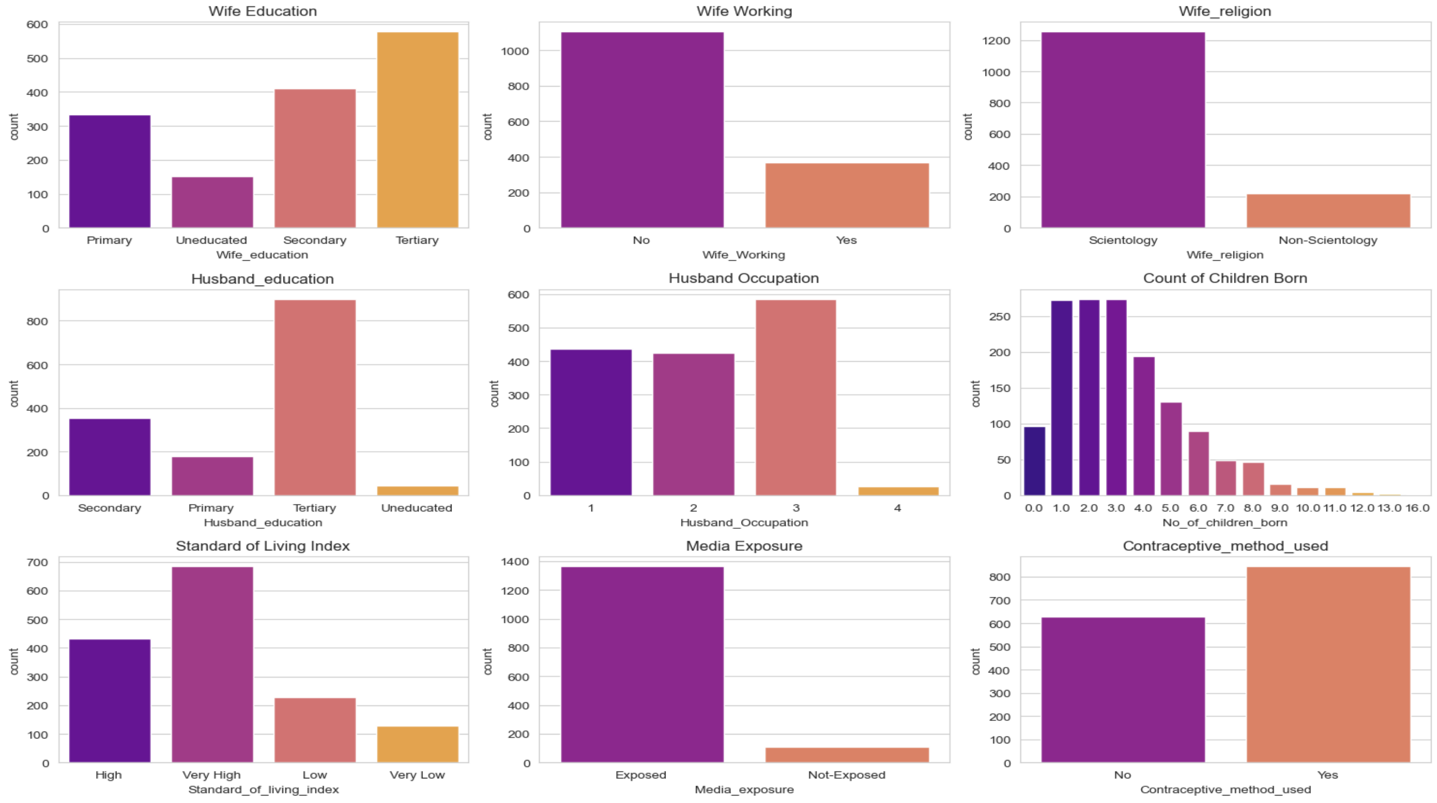


Fig: -25: countplot Categorical Variables

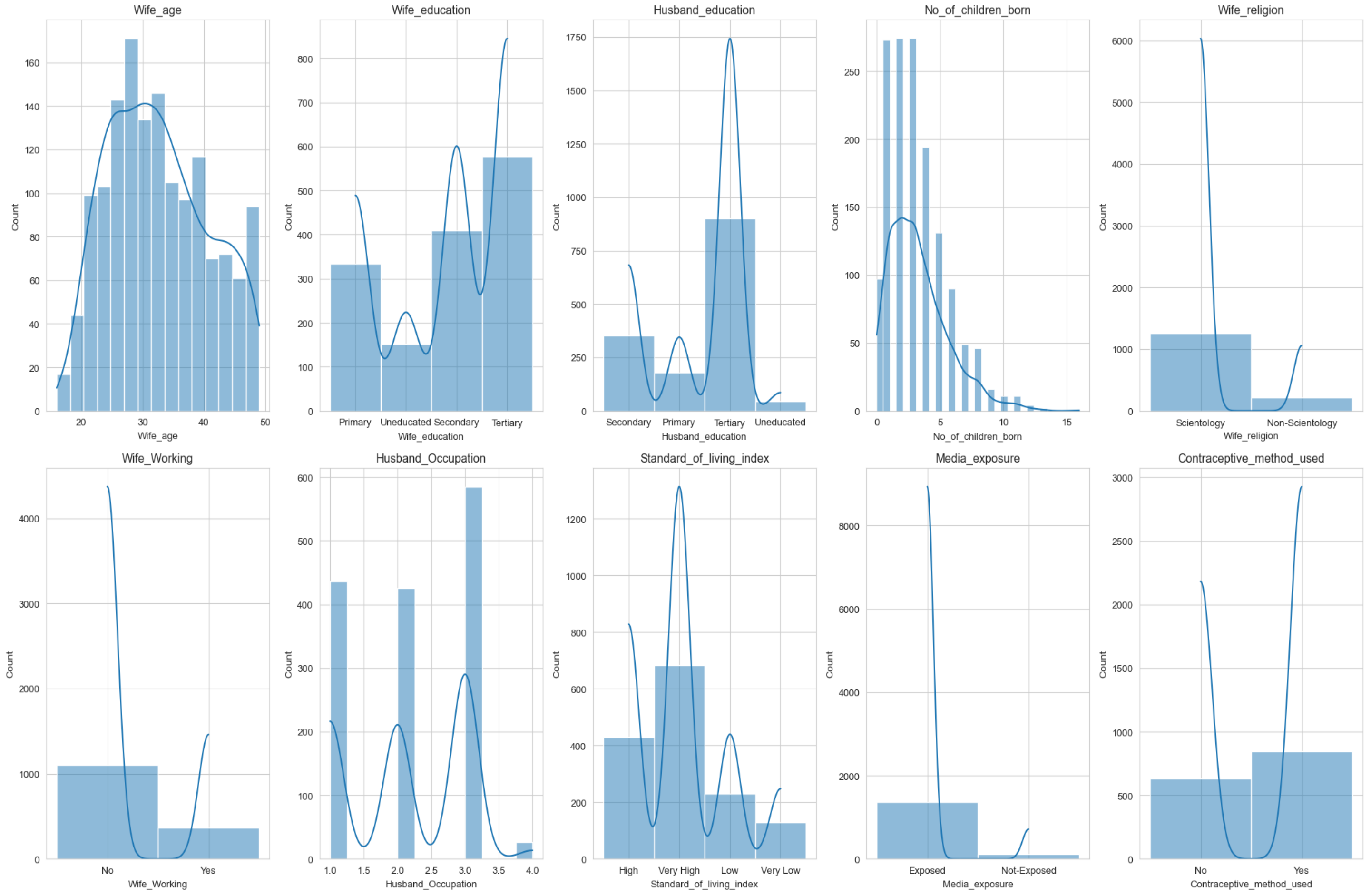


Fig: -26: Histplot for Categorical Variables

Observation:

a) ~34% of the wives have not attended secondary or higher education whereas it is only ~15% in case of Husbands.

b) Only 25% of the wives are working.

c) More than 75% of the data having high standard of living index.

d) More than 90% of the sample is having exposure to media.

e) As per the data provided, there are 55% of the people used contraceptive method.

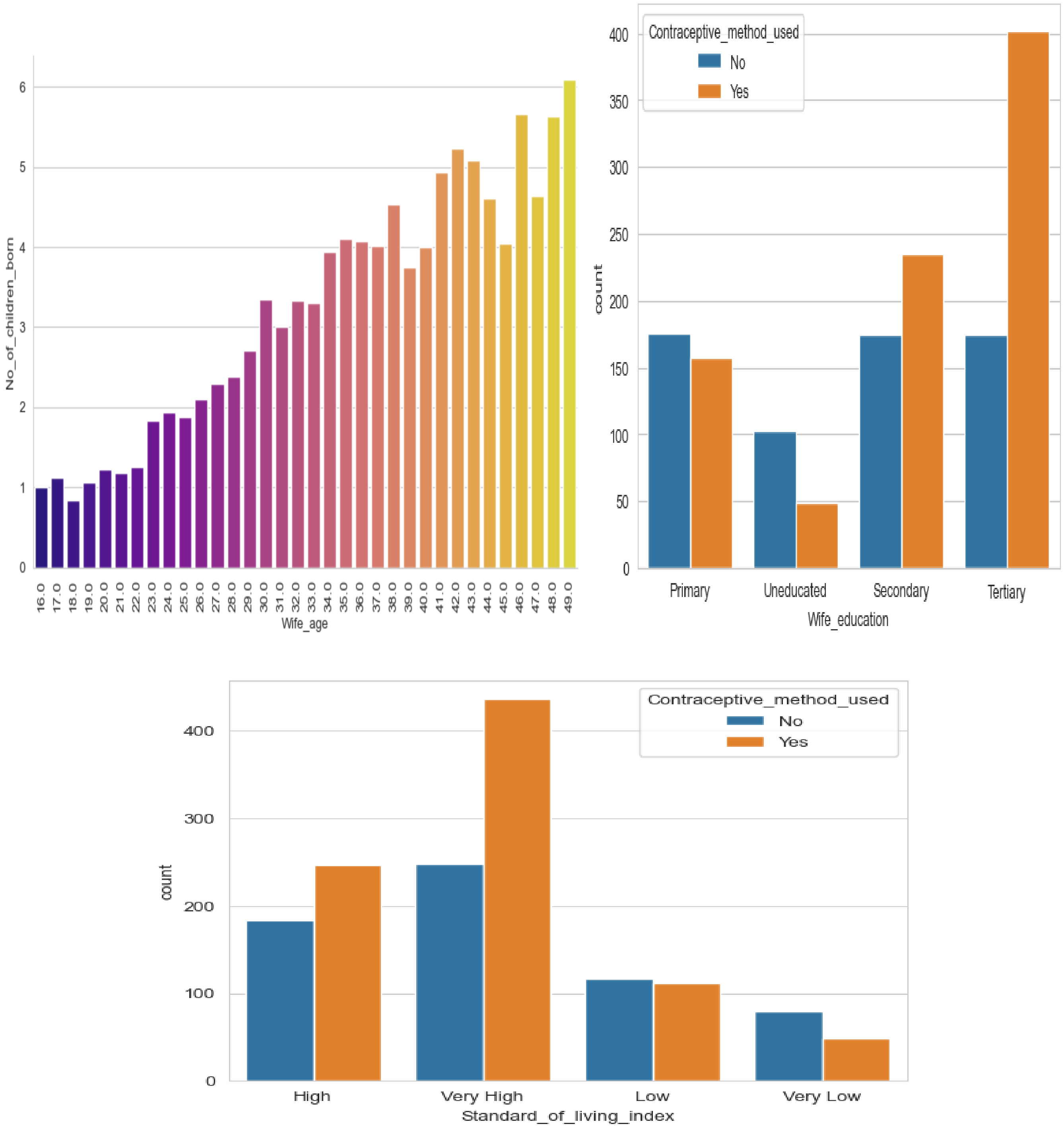


Fig: -27: Multivariate analysis

1. Higher Living Standards: In areas with high and very high standards of living, a larger number of people do not use contraceptives compared to those who do.
2. Lower Living Standards: As the standard of living decreases to low and very low, the use of contraceptives significantly drops, with very few people using them in the very low category.
3. Contraceptive Trends: The trend suggests that contraceptive use is less common in areas with lower standards of living.
4. Data Representation: The graph uses blue bars to indicate ‘No’ use and orange bars for ‘Yes’ use, clearly differentiating the two categories across different living standards.
5. Higher Education, Higher Usage: There is a clear trend showing that as the education level increases, the usage of contraceptive methods also increases. Nearly all tertiary-educated wives use contraceptives.
6. Lower Education, Lower Usage: Wives with primary or no education have a much lower usage of contraceptives, with over 150 primary-educated wives not using any methods.
7. Significant Gap: The gap between users and non-users of contraceptives is most pronounced at the secondary education level, where approximately 200 reported not using any methods.
8. Education as a Factor: The data suggests that education plays a significant role in contraceptive use, with higher education correlating with higher usage rates.

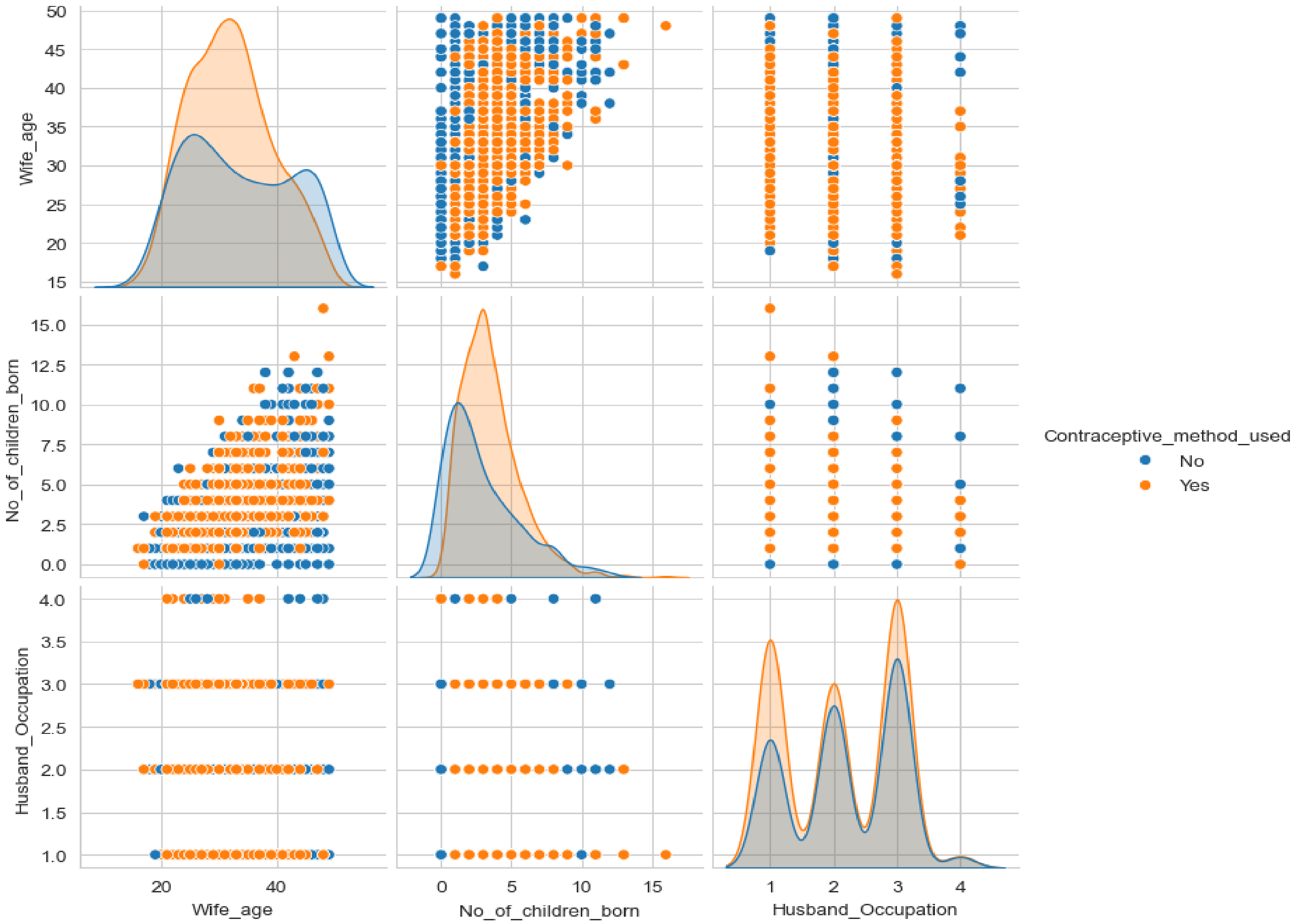


Fig: -28: pairplot

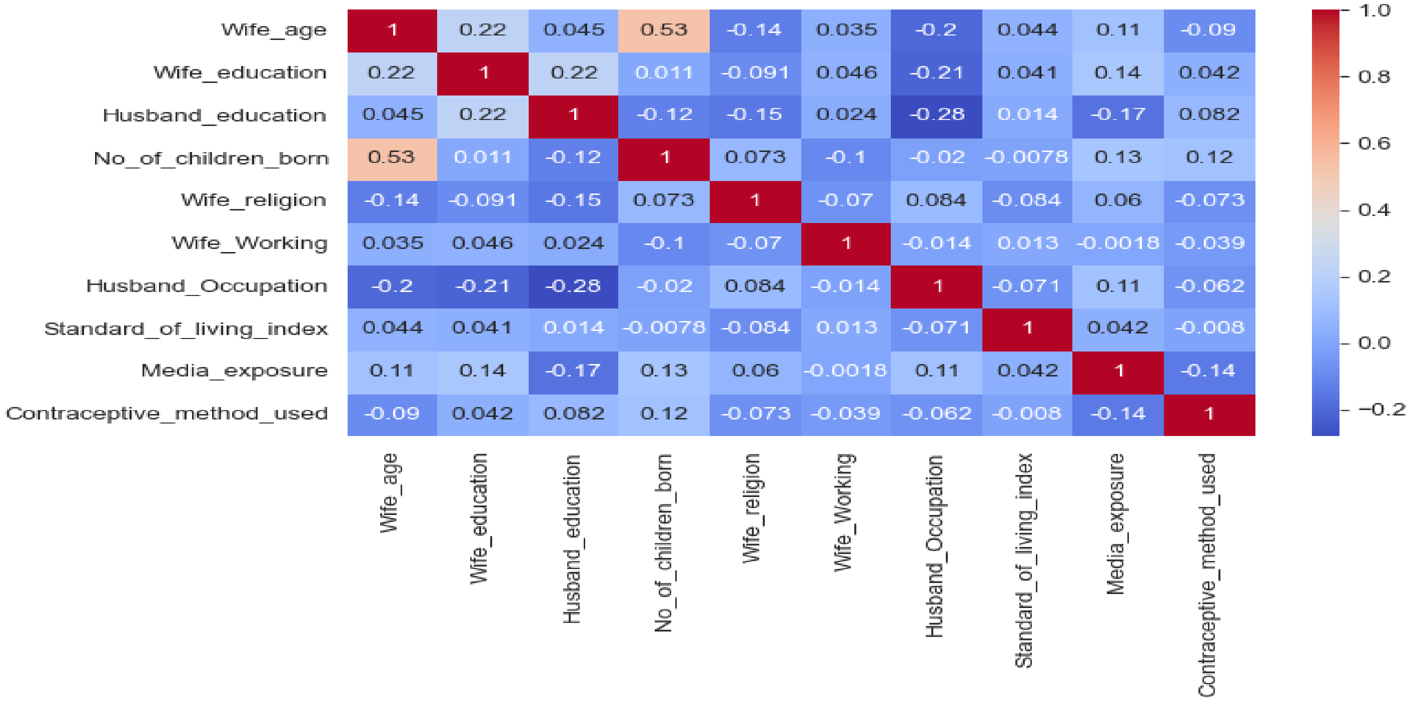


Fig: -29: Heatmap

- The pairplot & heatmap does not indicate any major trend/correlation between the variables.
- Some of the variables available in the pairplot, do not have the classes well separated. They will not be considered as good predictors.

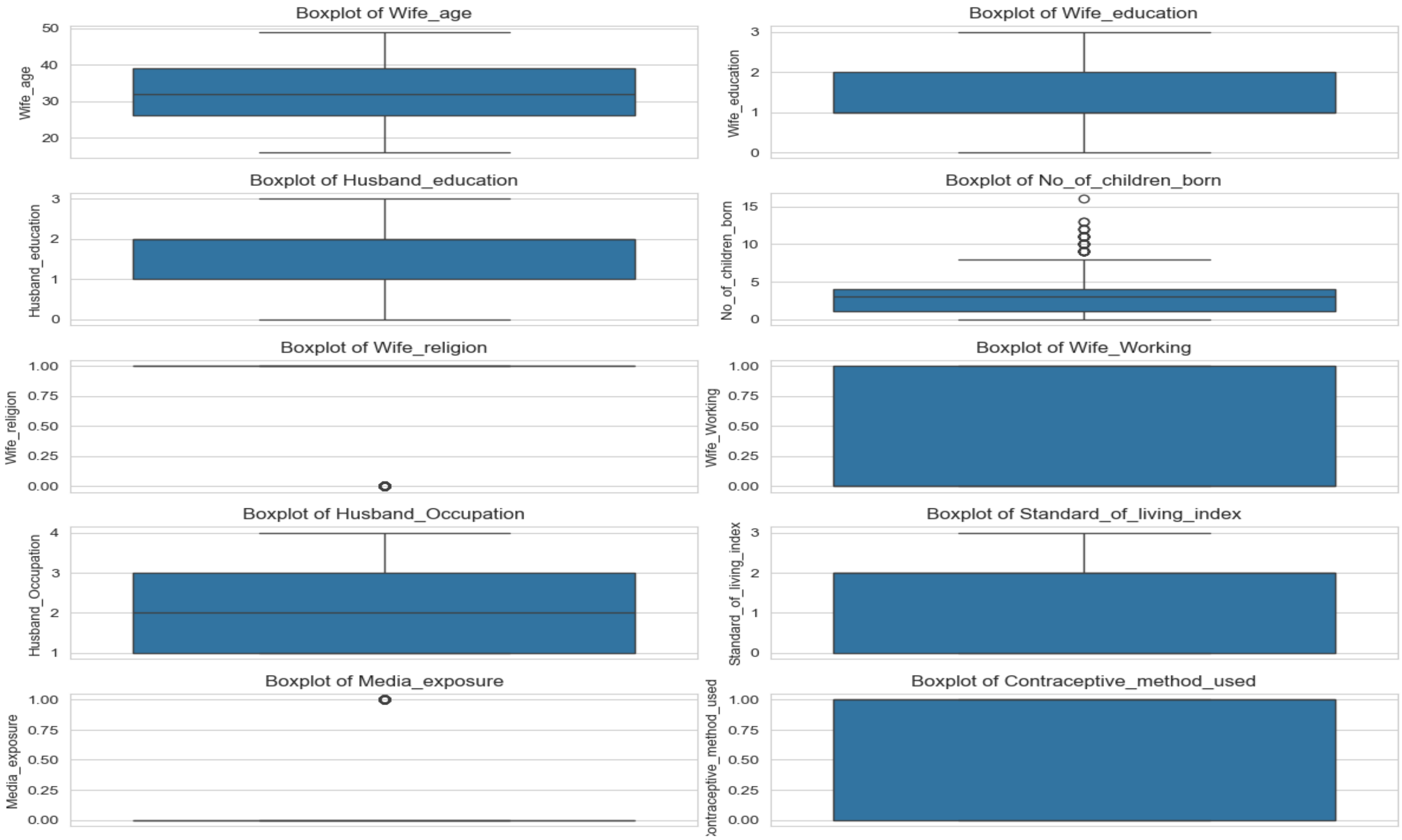


Fig: -30: Boxplot before treating outlier

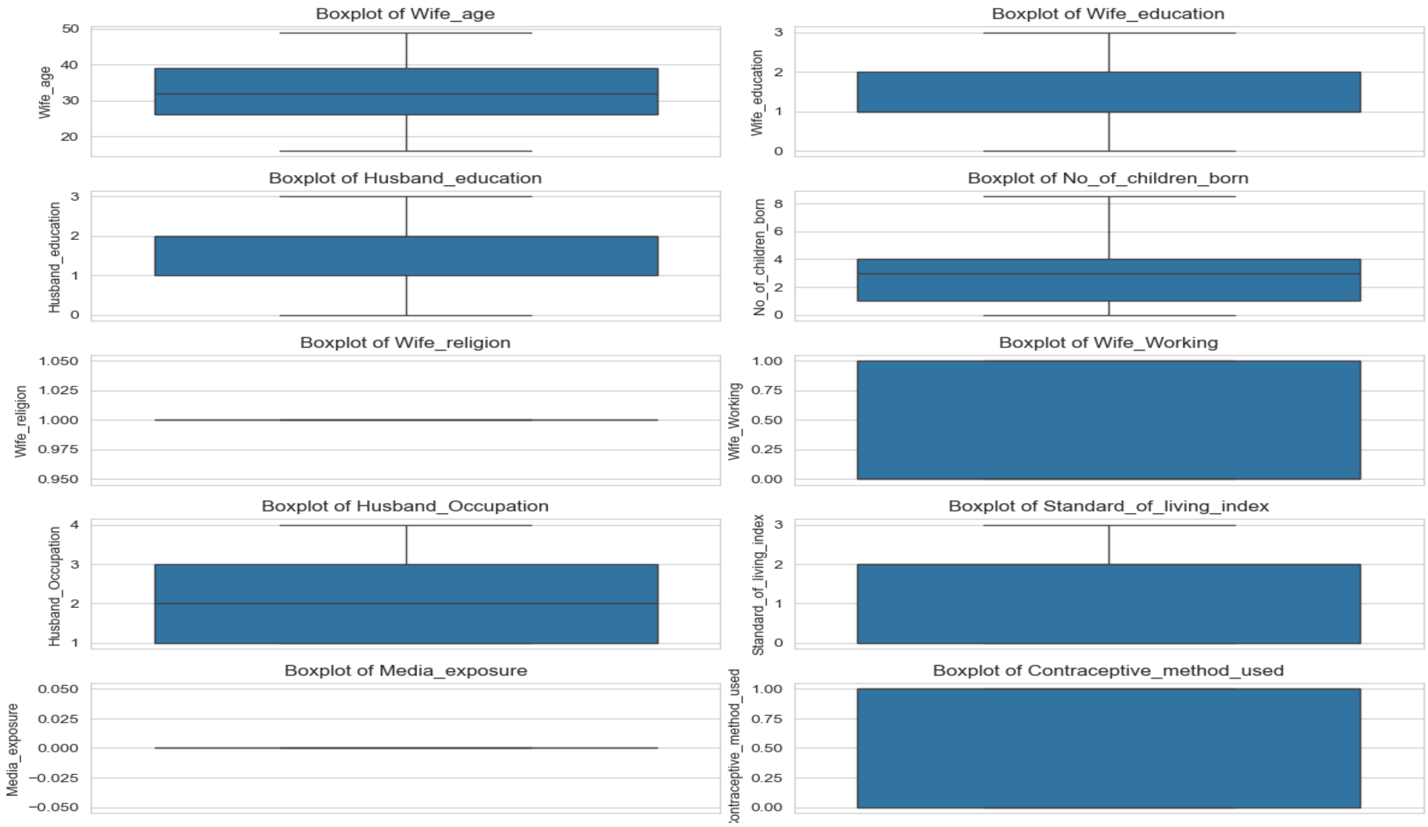


Fig: -31: Boxplot after treating outlier

2.2 Do not scale the data. Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Logistic Regression and LDA (linear discriminant analysis) and CART.

Treating the Categorical Nominal/ordinal Variables

Before the problem of fitting a multiple logistic regression, model is taken up the case of categorical predictor needs to be explicitly analyzed.

There are two types of categorical data

- Ordinal: Order based like 'good','bad','worst' or Clothing sizes
- Nominal: Without any order or ranks like city names, Genders, etc.

Here, for now let us convert these into categorical codes basis a custom formula as per the guidance mentioned in data dictionary. Below are the codes which will replace the string mentioned against each categorical variable.

Table 1: Categorical Codes Vs. String to transform the categorical variable

Wife_ education	Codes
Primary	2
Secondary	3
Tertiary	4
Uneducated	1

Husband_education	Codes
Primary	2
Secondary	3
Tertiary	4
Uneducated	1

Standard_of_living_index	Codes
Primary	2
Secondary	3
Tertiary	4
Uneducated	1

Wife_religion	Codes
Non-Scientology	0
Scientology	1

Wife_working	Codes
No	0
Yes	1

Media_exposure	Codes
Exposed	0
Not-Exposed	1

Splitting the data into Train and Test

After the EDA and all adjustments and transformations were performed on the full data, it was randomly split into training and test sets in 70:30 ratio

Training data contains 971 observations since 70% of 1388 is approximately 971. A simple random sample (without replacement) of size 971 is taken from the first 1388 positive integers, and the training dataset is formed by selecting the rows of Data corresponding to these random numbers. The remaining (1388-971=) 417 rows will constitute the test dataset.

Checking the dimensions of the training and test data:

X_train (971, 9)

X_test (417, 9)

y_train (971, 1)

y_test (417, 1)

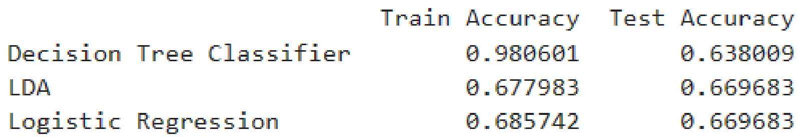
It is always a good idea to check that the success proportion of response is similar in both training and test data.

If we further check the ratio of response variables in train and test data below, it is evident that the model split is fine.

As a next step, we have built the model basis

a) Logistic Regression,
b) Decision Tree Classifier, and
c) LDA

Below is the initial accuracy score of these three models for Train and Test data:



```
                          Train Accuracy  Test Accuracy
Decision Tree Classifier        0.980601       0.638009
LDA                             0.677983       0.669683
Logistic Regression             0.685742       0.669683
```

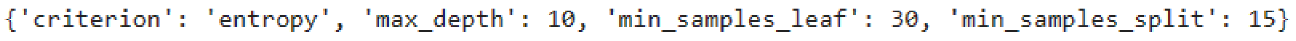
Fig: -32: Outcome of the model basis the applied method

From the above figure it I evident that, the model outcome for Decision Tree seems overfitting as the score for Train data is at 98% whereas, for Test data, it comes down to 63.8%. We might need to tweak the model to create a robust model which works better in Test data as well.

Moreover, the model for LDA and Logistic Regression seems working in similar fashion for both train and test data as the score is almost same.

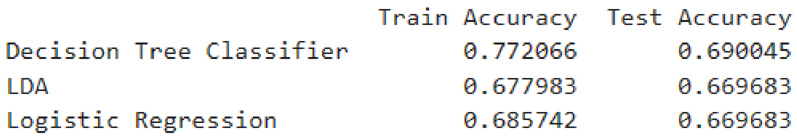
Optimization has been done for the decision tree classifier model by optimizing the hyper parameter of the GreedsearchCV to select the best model.

The input which will be best fit for the model are ascertained as below:



```
{'criterion': 'entropy', 'max_depth': 10, 'min_samples_leaf': 30, 'min_samples_split': 15}
```

Basis this input, if we recalculate the model we get the below accuracy score:



```
                            Train Accuracy  Test Accuracy
Decision Tree Classifier          0.772066       0.690045
LDA                               0.677983       0.669683
Logistic Regression               0.685742       0.669683
```

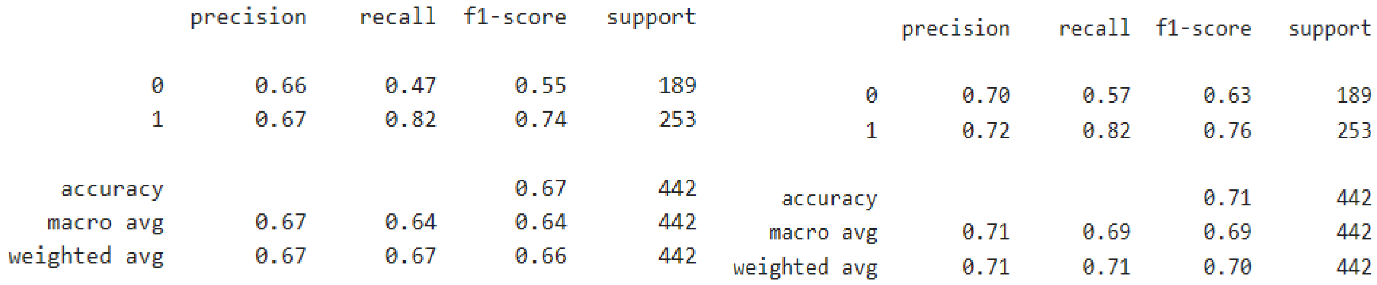
Now, it is evident the score has improved for Decision Tree Classifier model between train and test data; however, the overall score has come down to 77.2% on train dataset from 98% of first model.

2.3 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model Final Model: Compare Both the models and write inference which model is best/optimized.

Once a satisfactory model is developed on the training data, the same can be applied to estimate accuracy on both training and test data.

Let us check one by one the Confusion Matrix and Classification report of all these models and let us finalize the model that we want to go ahead with finally.

Regression Logistics Model output:



```
              precision    recall  f1-score   support

           0       0.66      0.47      0.55       189
           1       0.67      0.82      0.74       253

    accuracy                           0.67       442
   macro avg       0.67      0.64      0.64       442
weighted avg       0.67      0.67      0.66       442
```

```
              precision    recall  f1-score   support

           0       0.70      0.57      0.63       189
           1       0.72      0.82      0.76       253

    accuracy                           0.71       442
   macro avg       0.71      0.69      0.69       442
weighted avg       0.71      0.71      0.70       442
```

Fig: -33: Classification report for train and test data

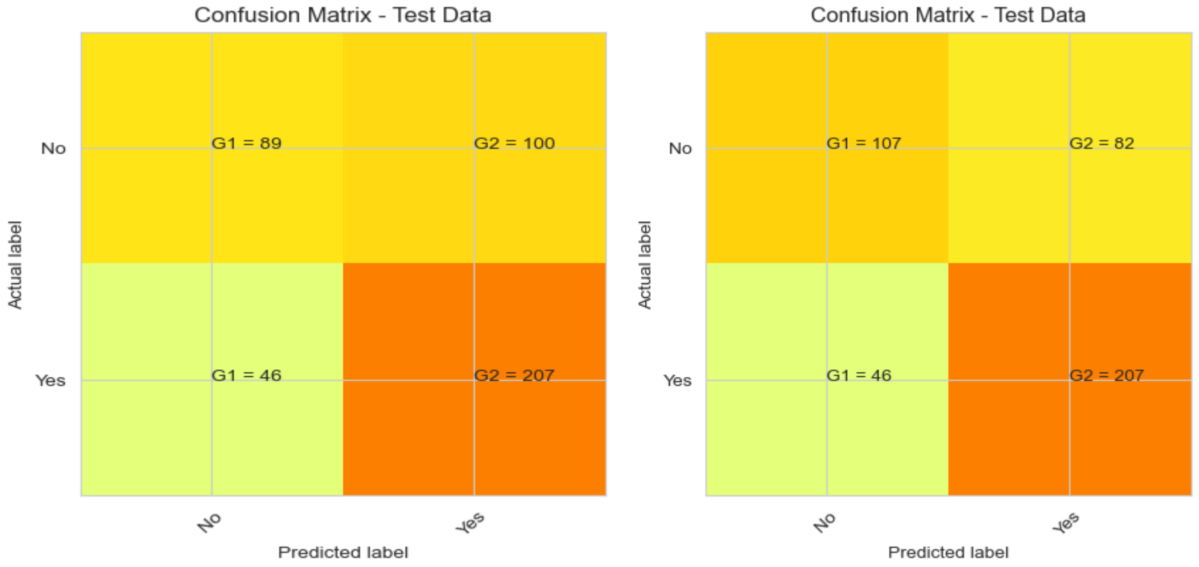


Fig: -34: Confusion matrix for test and train data

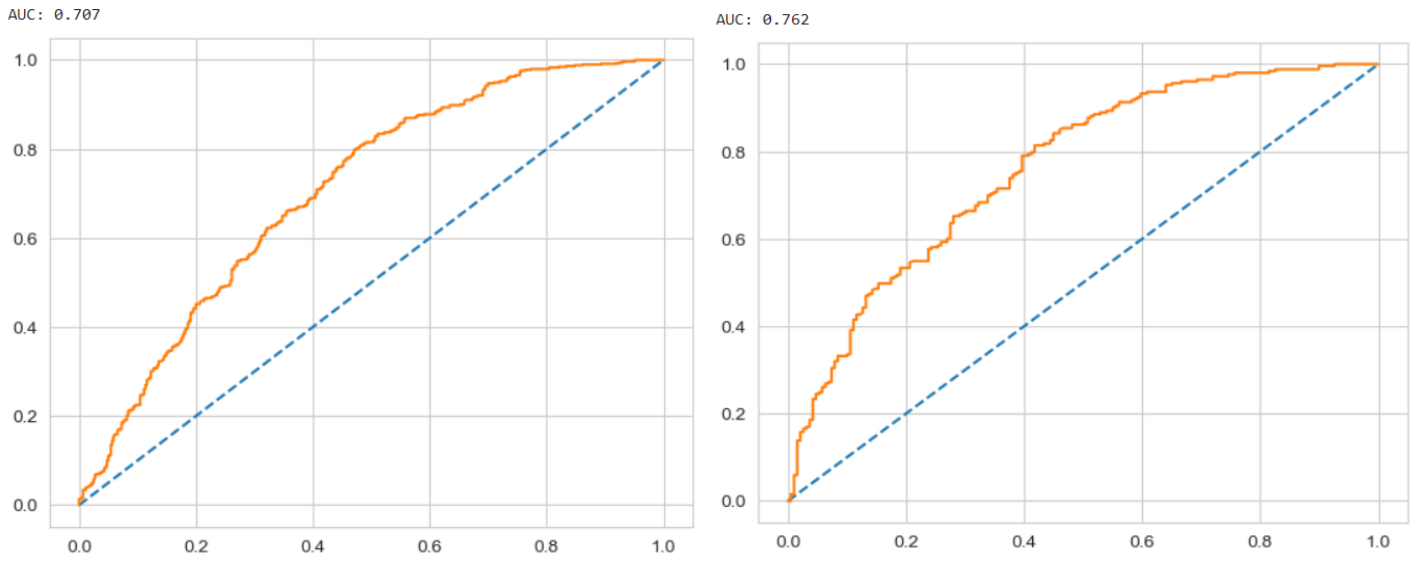
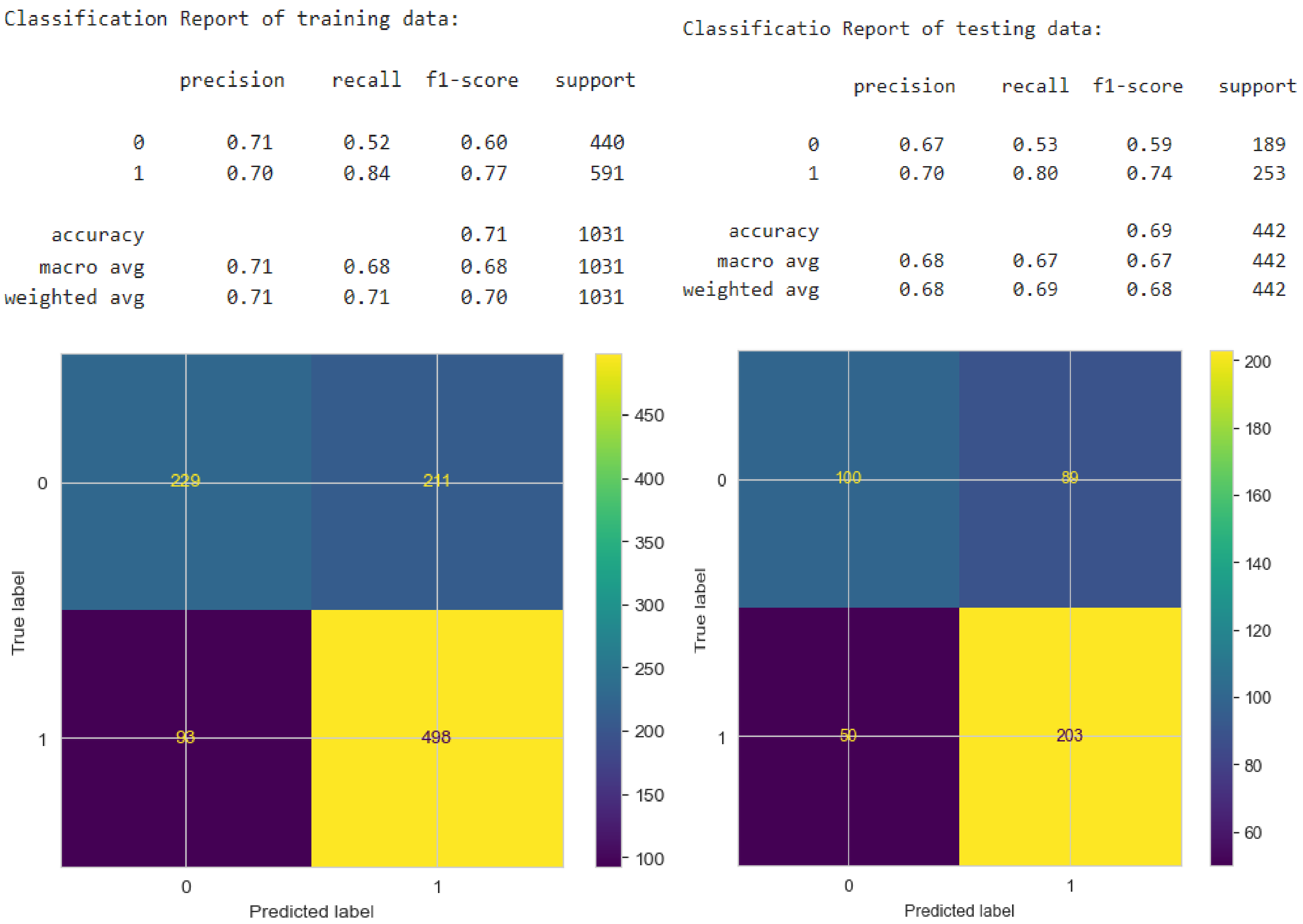


Fig: -35: AUC curve for train and test data

In this exercise we need to analyse precision score as we want to minimize false negative. The above classification report helps us to understand that the precision score is good against recall.

Regression Logistics Model output:



```
Classification Report of training data:

              precision    recall  f1-score   support

           0       0.71      0.52      0.60       440
           1       0.70      0.84      0.77       591

    accuracy                           0.71      1031
   macro avg       0.71      0.68      0.68      1031
weighted avg       0.71      0.71      0.70      1031
```

```
Classificatio Report of testing data:

              precision    recall  f1-score   support

           0       0.67      0.53      0.59       189
           1       0.70      0.80      0.74       253

    accuracy                           0.69       442
   macro avg       0.68      0.67      0.67       442
weighted avg       0.68      0.69      0.68       442
```

Fig: -36: Classification and Confusion matrix for train and test data

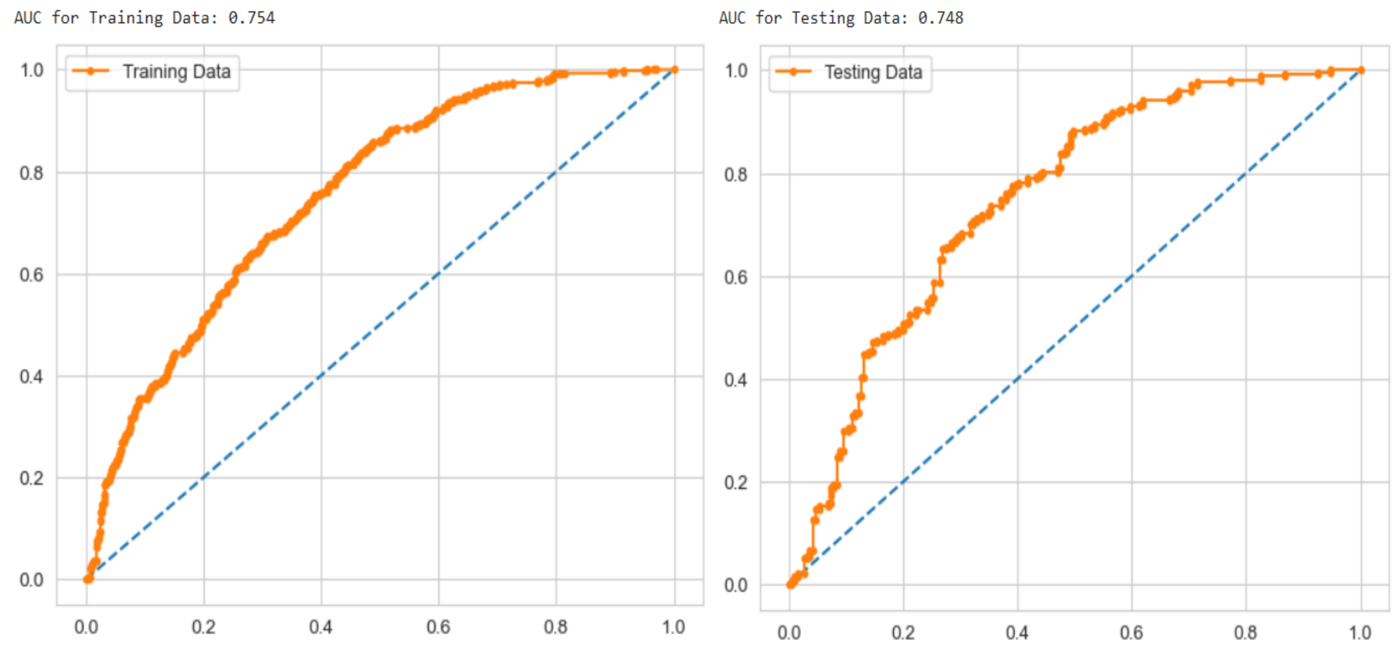
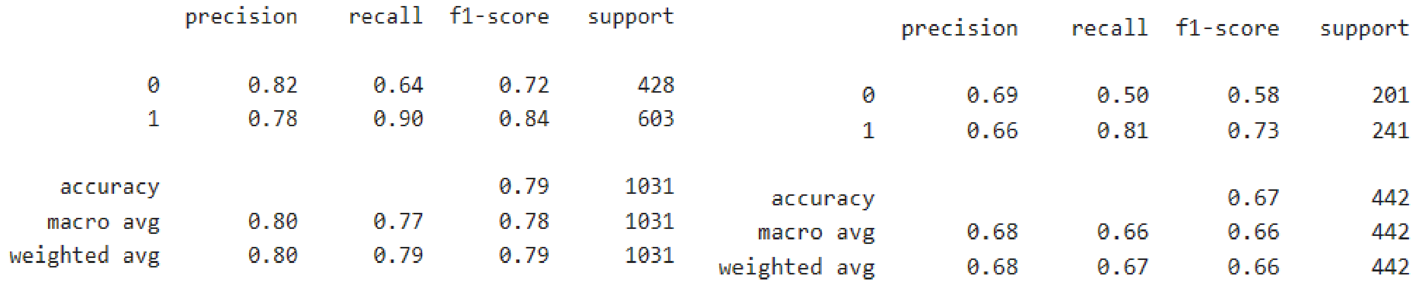


Fig: -37: AUC curve for train and test data

There is not much difference in various accuracy parameters in LDA Vs. Logistic regressions. However, the AUCROC is higher in case of LDA model as compared to Logistic regression model and hence we can say that LDA model is the more powerful model between both of these models.

Let us evaluate same information for Decision Tree Classifier as well:



```
              precision    recall  f1-score   support

           0       0.82      0.64      0.72       428
           1       0.78      0.90      0.84       603

    accuracy                           0.79      1031
   macro avg       0.80      0.77      0.78      1031
weighted avg       0.80      0.79      0.79      1031
```

```
              precision    recall  f1-score   support

           0       0.69      0.50      0.58       201
           1       0.66      0.81      0.73       241

    accuracy                           0.67       442
   macro avg       0.68      0.66      0.66       442
weighted avg       0.68      0.67      0.66       442
```

Fig: -38: LDA classification Matrix for Train and test data

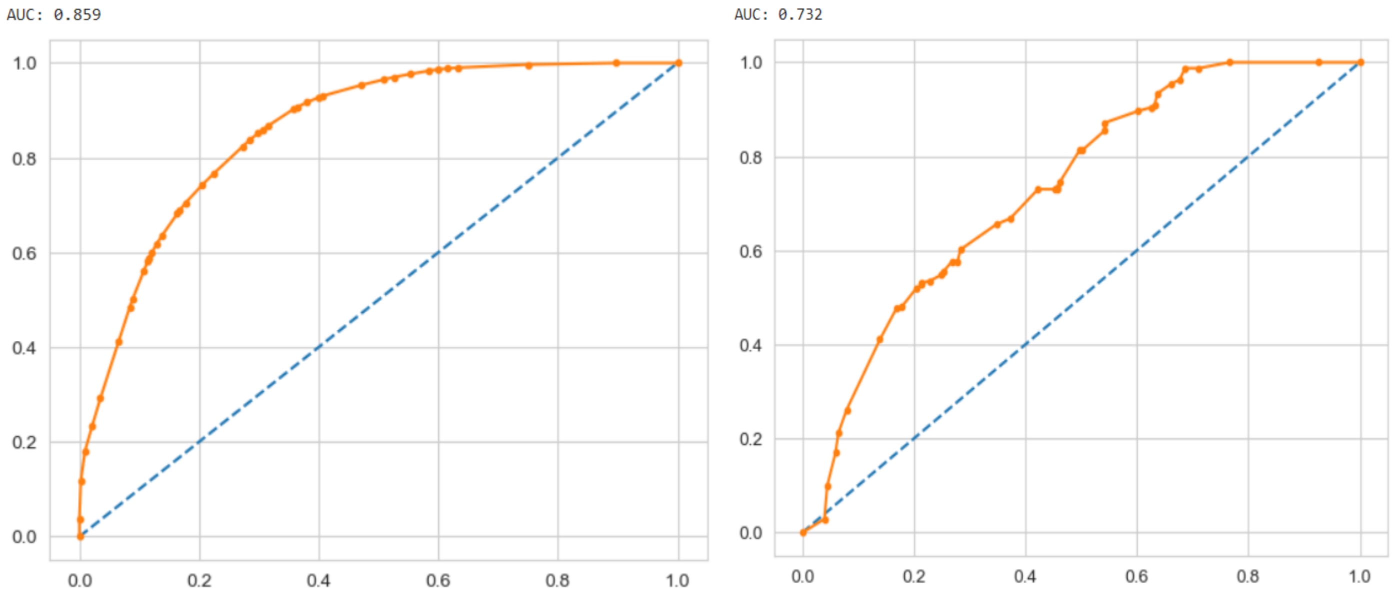


Fig: - 39: AUC ROC curve

Accuracy on the Training Data: 82%

Accuracy on the Test Data: 69%

AUC on the Training Data: 85.9%

AUC on the Test: 73.2%

Accuracy, AUC, Precision and Recall for test data is almost inline with training data. This proves no overfitting or underfitting has happened, and overall, the model is a good model for classification.

Out of the three model, we are finalizing the Decision Tree Classifier as the best model to be use for the classification.

Also, here analyzing the metric precision is more important because, we don't want to miss out on those customers who has not taken any contraceptive method whereas the model predicts it that those women has taken contraceptive method, having a predictive power to catch those would help the health ministry to be more proactive in their approach and plan their scheme, from the confusion matrix of test data we can see that our model has miss classified 201(False Positive) women.

Business Insights and Recommendations

Importance of feature based on best model

- It was observed for Linear Discriminant Analysis that the Variable Number of children born has the largest magnitude further helping in classifying.

- Linear discriminant Analysis focuses on the magnitude and coefficients are considered to do so. The higher the coefficients larger the magnitude and important the feature. Lower the coefficients lower the magnitude and least important the feature.

- In LDA, Wife age had the lesser magnitude and further specifying less importance I the model building.

- The CART model provides information on feature importance where Wife age is given the upper most importance and surprisingly the opposite of what LDA model suggested

- The least important came out to be Media exposure. To note the way Decision tree splits it would be better if there was absence of under sampling or else they have to carry crucial information.

Actionable Insights and recommendations

- The Univariate analysis was performed to analyse the pattern displayed by them. It can be noted the age of Wife has almost normal distribution with absence of outliers. Count of Education for both husband and wife have similar pattern where tertiary has the highest count and uneducated being the least.

- Multivariate Analysis was performed to check the distribution of continuous alongside the category. The missing value were identified and treated with proper imputation techniques.

- A two-fold approach was considered. The first one was to considered the outliers in Number of children born were legit and another was to consider them as exaggerated values where they will be capped to the nearest legit value. However, when the model was built for both the approach the second model provided better results and was considered.

- Logistic Regression was the first algorithm considered. The results provided by the model was pretty satisfying considering the number of records (less). The model accuracy, precision, recall and f1-score was calculated.

- Similarly, Linear Discriminant Analysis and CART (Decision tree) was built to check for their approach (feature importance) and accuracy. The LDA could divide the target as it is meant to and the accuracy was considerable.

- The decision tree model was built initially without tuning any parameters. The result was overfitting and variance was the cause of such results. Another model was built by pruning the tree and hyper tuning the parameters. Grid search CV was considered as the multi fold approach. The results did reduce the overfitting slightly but not significantly further proving that decision tree might not be the right option for the data provided.

- The Logistic Regression and LDA model provided better results compared to that of Decision tree. Considering either of Logistic regression or LDA could provide the desirable results. It is also important to note that the results can be astonishing if the quantity of data is more.
- **Focus on promoting contraceptive usage** among **women** with a **high and very high standard** of living, as they are more likely to use them.
- **Target women aged 25 to 35** with a **good education level**, as they are more likely to use contraceptives.
- **Encourage husbands** to be **involved in family planning decisions**, as their **education level plays a significant role** in the use of contraceptives.
- **Investigate** the reasons behind women with no children using contraceptives, as this could provide valuable insights.
- **Leverage media exposure** to **promote contraceptive usage and awareness**, as it plays a key role in shaping opinions.
- **The Republic of Indonesia Ministry of Health** should **initiate outreach programs to educate women** who do not use contraceptives about their benefits, usage, and potential side effects.
- **Investigate** why wives with 8, 10, 11, and 12 years of education are not using contraceptives, and address any barriers or misconceptions they may have.