

---

# PREDICTIVE MODELLING PROJECT BUSINESS REPORT

---

DSBA

*Written by*  
***Priyamvada Singh***

Dated: **05-03-2023**  
(Format: dd-mm-yyyy)

## Contents

DSBA .....	0
<b>Problem 1: Linear Regression</b> .....	2
<b>1.1 Read the data and do exploratory data analysis. Describe the data briefly. (Check the Data types, shape, EDA, 5-point summary). Perform Univariate, Bivariate Analysis, Multivariate Analysis.</b> .....	2
<b>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.</b> .....	13
<b>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 statsmodel. Create multiple models and check the performance of Predictions on Train and Test sets using Rsquare, RMSE &amp; Adj Rsquare. Compare these models and select the best one with appropriate reasoning.</b> .....	16
<b>1.4 Inference: Basis on these predictions, what are the business insights and recommendations. Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.</b> .....	25
<b>Problem 2: Logistic Regression, LDA and CART</b> .....	26
<b>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.</b> .....	26
<b>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.</b> .....	32
<b>2.4 Inference: Basis on these predictions, what are the insights and recommendations.</b> .....	44
<b>Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.</b> .....	44

## 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.

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

i. Head of the data (the first five rows):

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	...	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	reesv
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	1730
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	1869
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	1021
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	1863
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	1760

5 rows × 22 columns

- ii. The compactiv dataset has 8192 rows and 22 columns.
- iii. Null values: 'rchar' has 104 null values while 'wchar' has 15 null values.
- iv. No duplicates found in the dataset.
- v. The info on the dataset is as follows: 13 *float64* datatype, 8 *int64* datatype, 1 *object* datatype variables.
- vi. The 5-point-summary (min, 25%, 50%, 75%, max) of each variable is given below through data description:

	count	mean	std	min	25%	50%	75%	max
<b>lread</b>	8192.0	1.955969e+01	53.353799	0.0	2.0	7.0	20.000	1845.00
<b>lwrite</b>	8192.0	1.310620e+01	29.891726	0.0	0.0	1.0	10.000	575.00
<b>scall</b>	8192.0	2.306318e+03	1633.617322	109.0	1012.0	2051.5	3317.250	12493.00
<b>sread</b>	8192.0	2.104800e+02	198.980146	6.0	86.0	166.0	279.000	5318.00
<b>swrite</b>	8192.0	1.500582e+02	160.478980	7.0	63.0	117.0	185.000	5456.00
<b>fork</b>	8192.0	1.884554e+00	2.479493	0.0	0.4	0.8	2.200	20.12
<b>exec</b>	8192.0	2.791998e+00	5.212456	0.0	0.2	1.2	2.800	59.56
<b>rchar</b>	8088.0	1.973857e+05	239837.493526	278.0	34091.5	125473.5	267828.750	2526649.00
<b>wchar</b>	8177.0	9.590299e+04	140841.707911	1498.0	22916.0	46619.0	106101.000	1801623.00
<b>pgout</b>	8192.0	2.285317e+00	5.307038	0.0	0.0	0.0	2.400	81.44
<b>ppgout</b>	8192.0	5.977229e+00	15.214590	0.0	0.0	0.0	4.200	184.20
<b>pgfree</b>	8192.0	1.191971e+01	32.363520	0.0	0.0	0.0	5.000	523.00
<b>pgscan</b>	8192.0	2.152685e+01	71.141340	0.0	0.0	0.0	0.000	1237.00
<b>atch</b>	8192.0	1.127505e+00	5.708347	0.0	0.0	0.0	0.600	211.58
<b>pgin</b>	8192.0	8.277960e+00	13.874978	0.0	0.6	2.8	9.765	141.20
<b>ppgin</b>	8192.0	1.238859e+01	22.281318	0.0	0.6	3.8	13.800	292.61
<b>pflt</b>	8192.0	1.097938e+02	114.419221	0.0	25.0	63.8	159.600	899.80
<b>vflt</b>	8192.0	1.853158e+02	191.000603	0.2	45.4	120.4	251.800	1365.00
<b>freemem</b>	8192.0	1.763456e+03	2482.104511	55.0	231.0	579.0	2002.250	12027.00
<b>freeswap</b>	8192.0	1.328126e+06	422019.426957	2.0	1042623.5	1289289.5	1730379.500	2243187.00
<b>usr</b>	8192.0	8.396887e+01	18.401905	0.0	81.0	89.0	94.000	99.00

- vii. The null values in 'rchar' and 'wchar' have been replaced with median of the respective features.  
viii. Value counts of the target variable 'runqsz' is given below:

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

These values have been replaced as 0 and 1 to give them numeric values. Not\_CPU\_Bound has been assigned 0 and CPU\_Bound has been assigned 1.

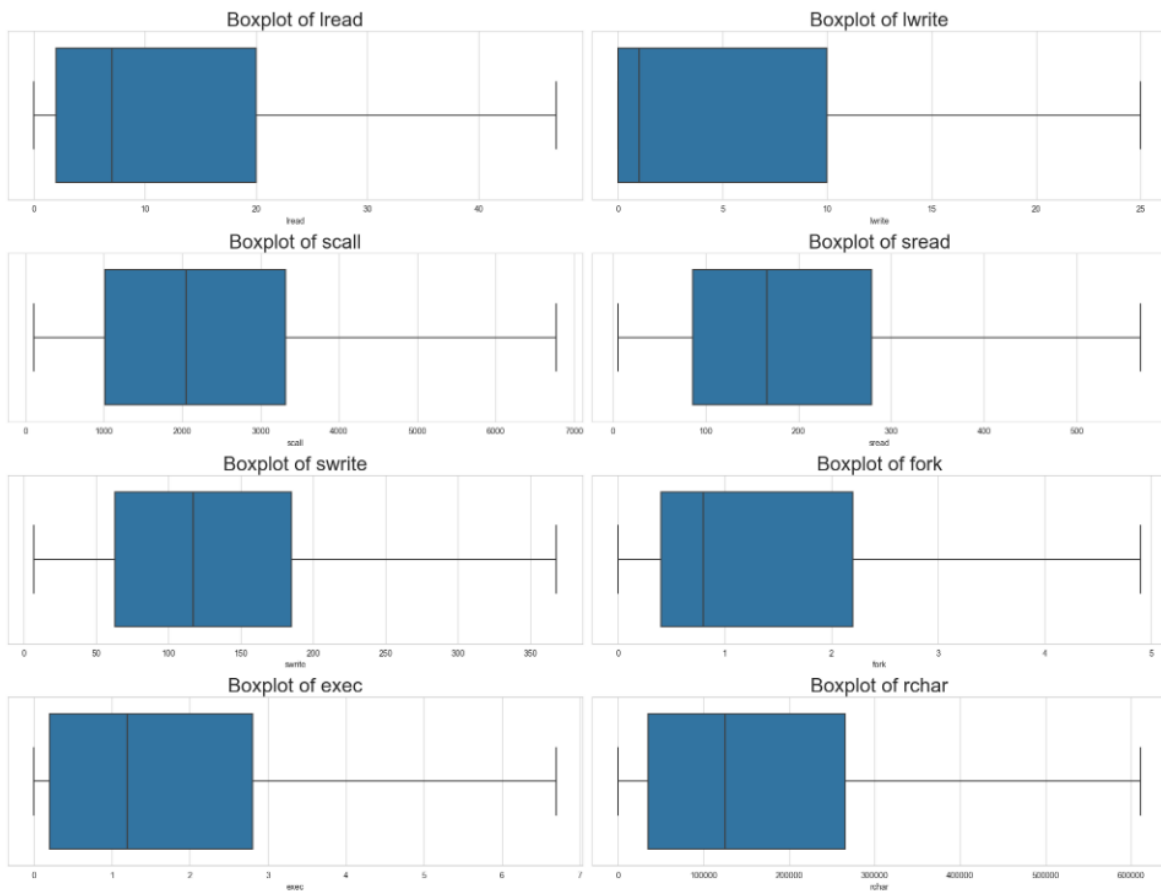
**ix. Univariate analysis:**

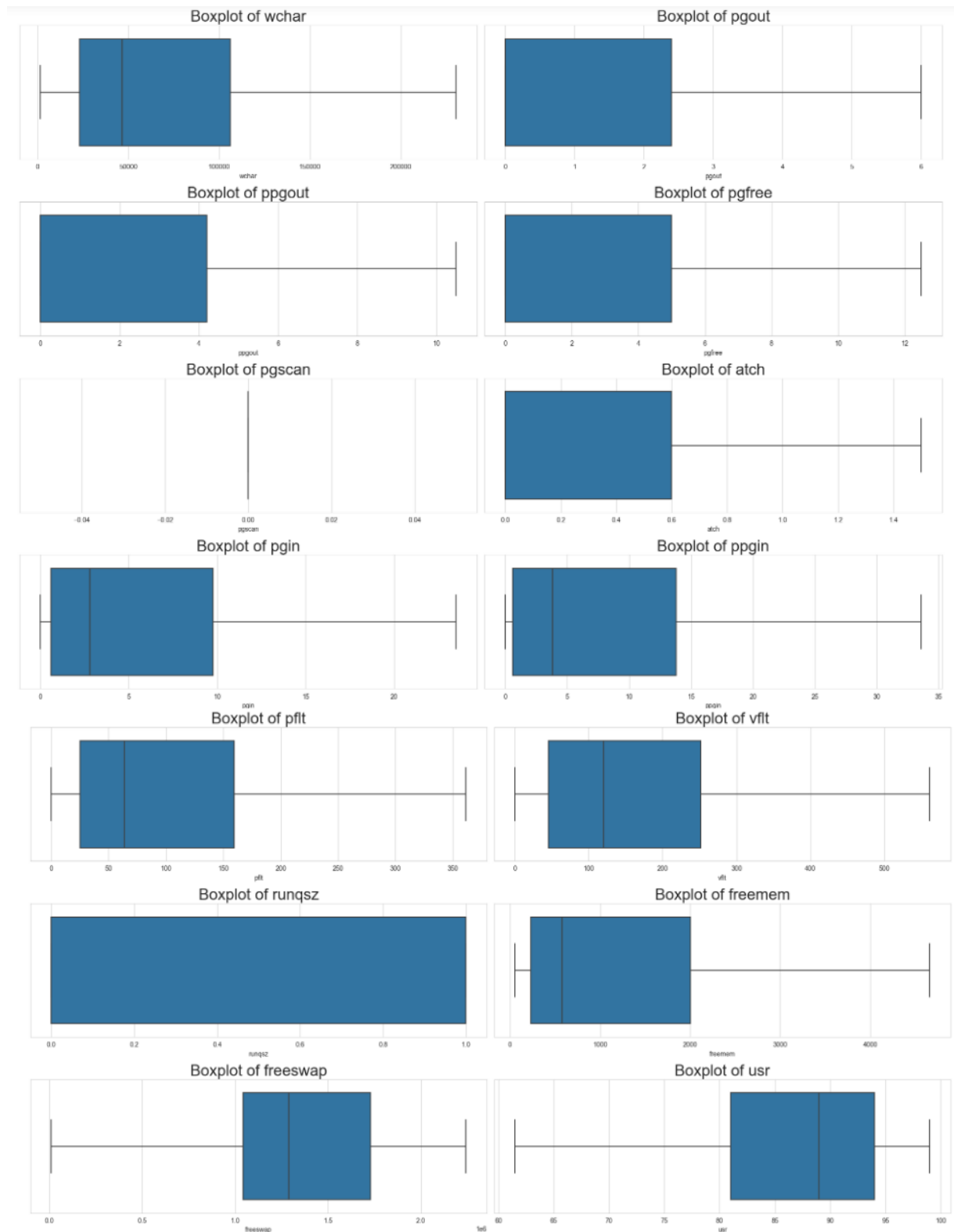
- Before analysing the dataset, I have checked for outliers and treated them with the help of the IQR method. I have replaced outliers with lower and upper range values that the formulae upper range=Q3+(1.5\*IQR) and lower range=Q1-(1.5\*IQR) give as output.

- Following are the 0 values present in each feature:

```
lread: 675
lwrite: 2684
scall: 0
sread: 0
swrite: 0
fork: 21
exec: 21
rchar: 0
wchar: 0
pgout: 4878
ppgout: 4878
pgfree: 4869
pgscan: 8192
atch: 4575
pgin: 1220
ppgin: 1220
pflt: 3
vflt: 0
runqsz: 4331
freemem: 0
freeswap: 0
```

- After treating outliers, there are 8192 zero values in 'pgscan', hence I dropped this column.
- Following is the boxplot distribution of each variable via boxplot:

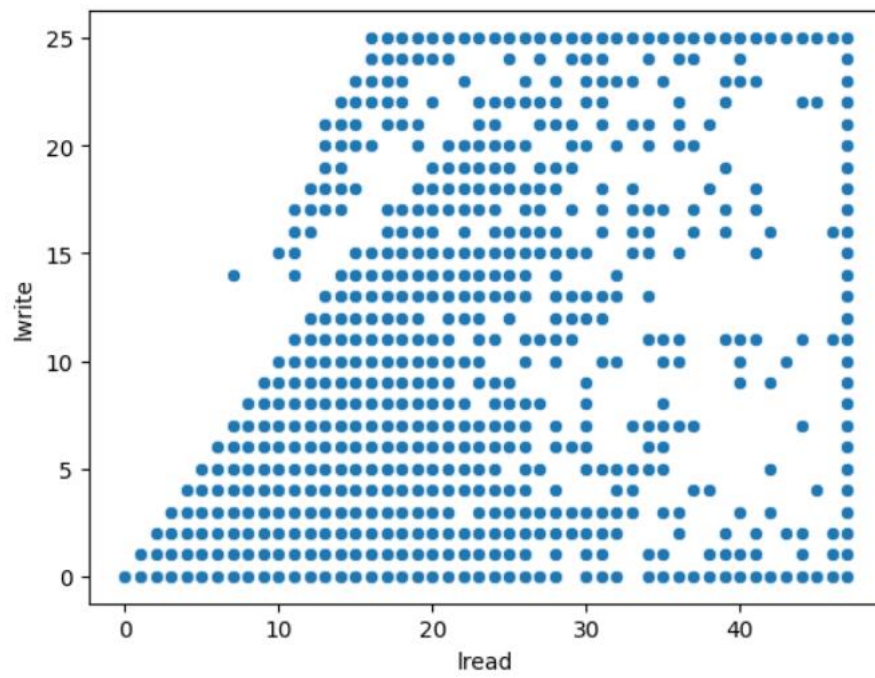




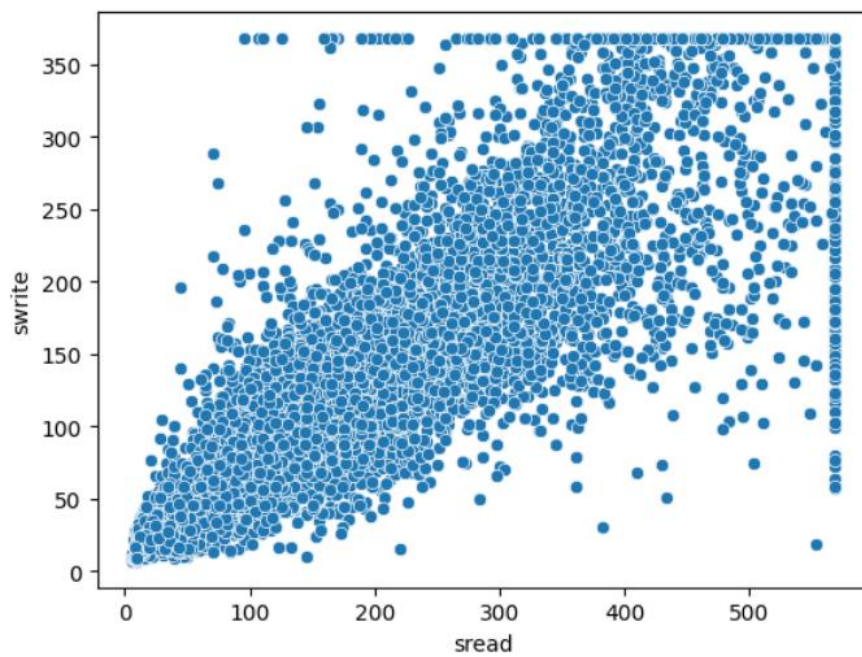
All the variable distributions in the above figures are right-skewed, except the last two, i.e., 'freeswap' and 'usr', which are left-skewed.

#### x. Bivariate Analysis:

- 'lread' versus 'lwrite'

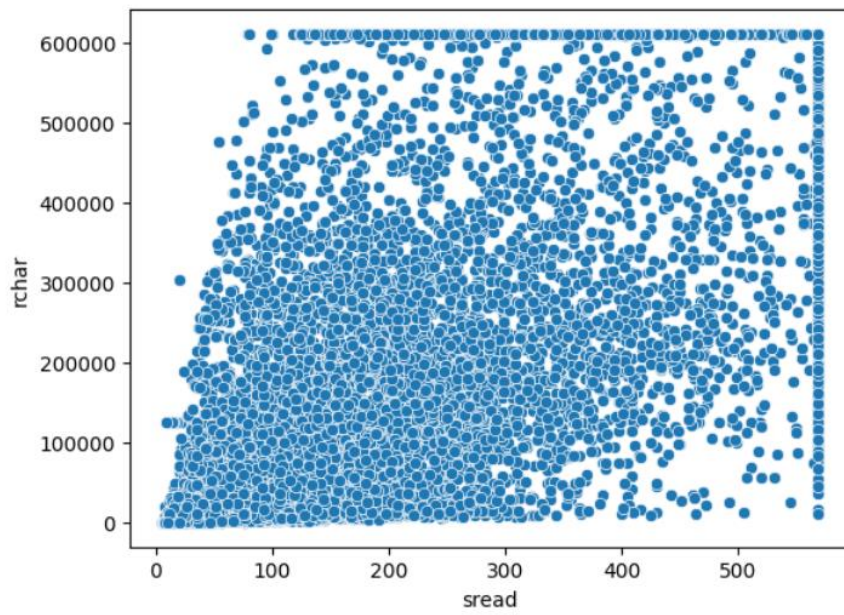


- 'sread' versus 'swrite'

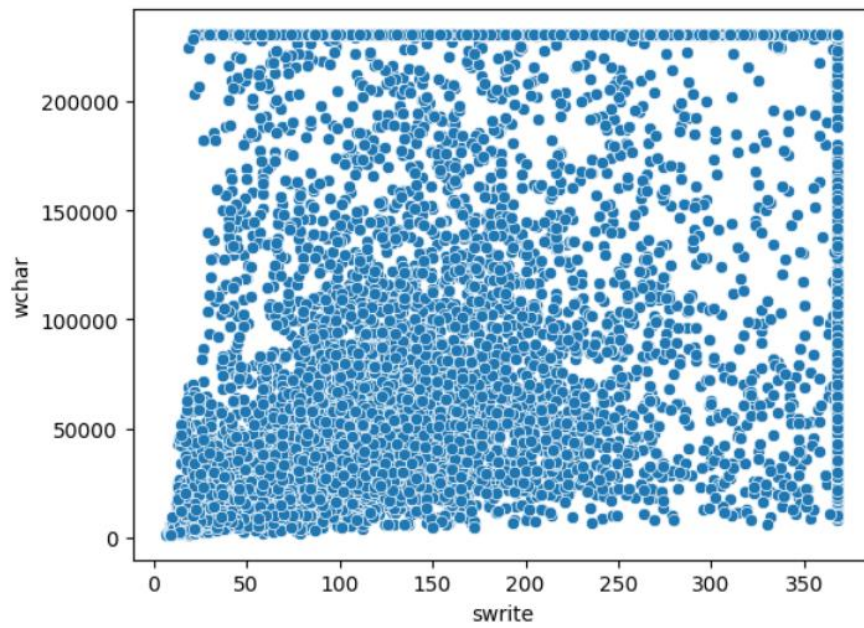




- 'sread' versus 'rchar'

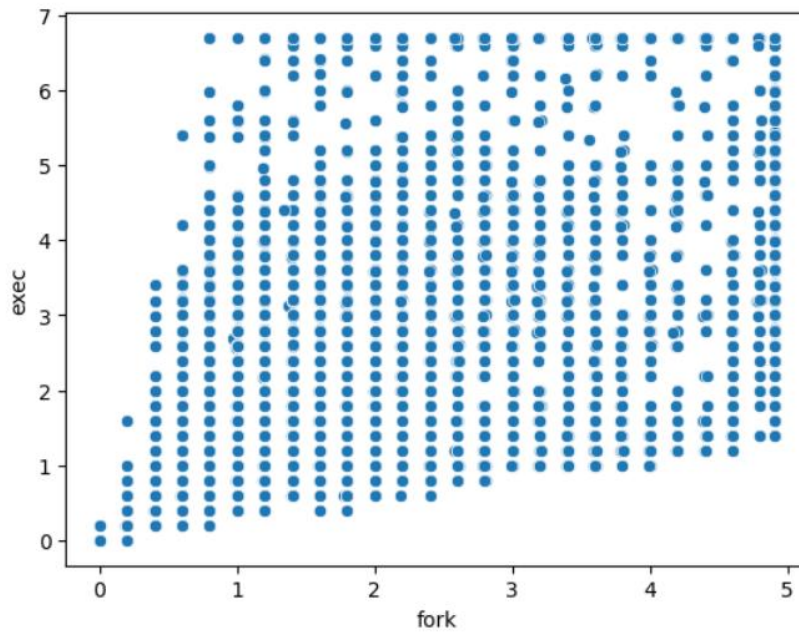


- 'swrite' versus 'wchar'

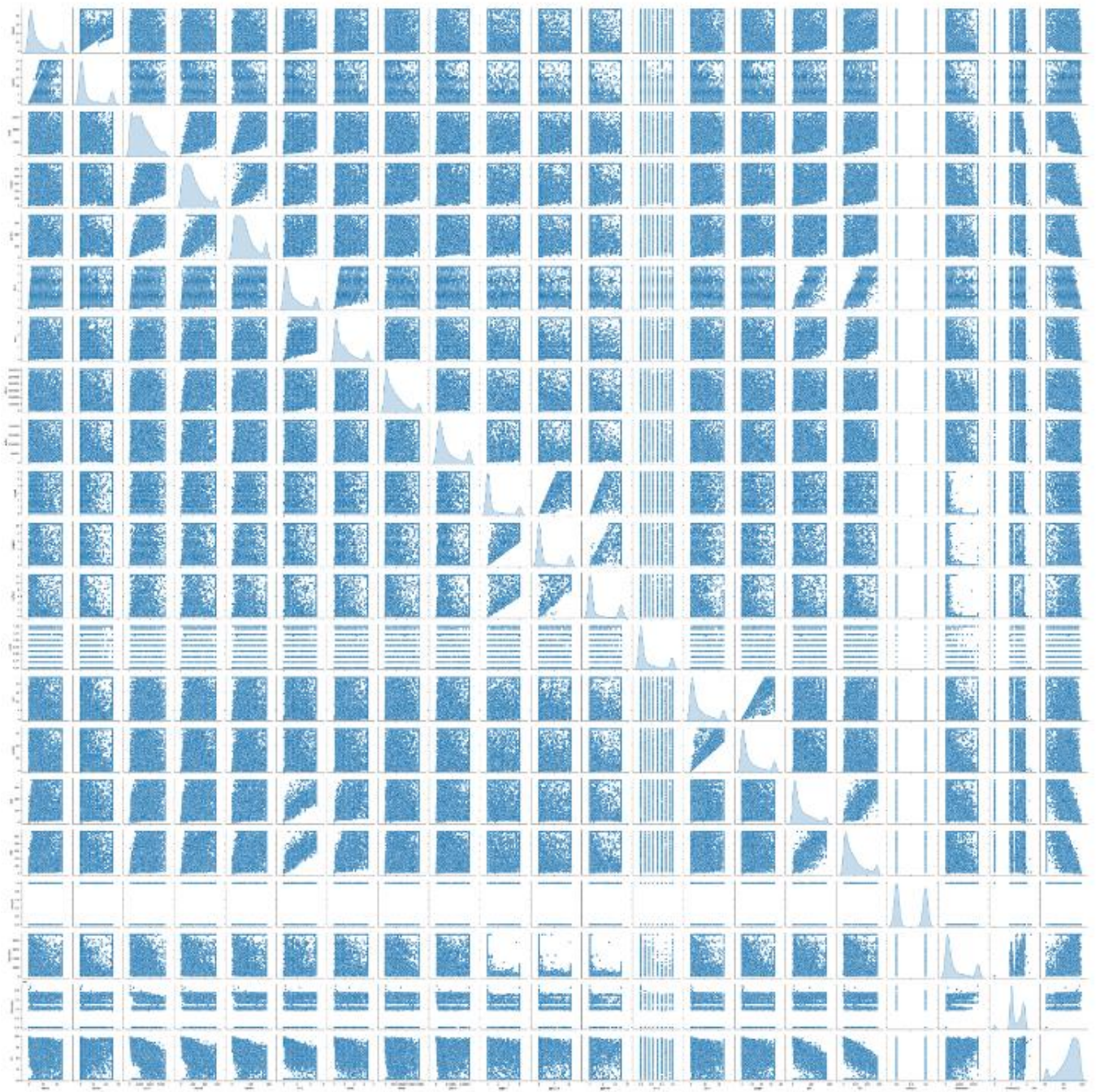


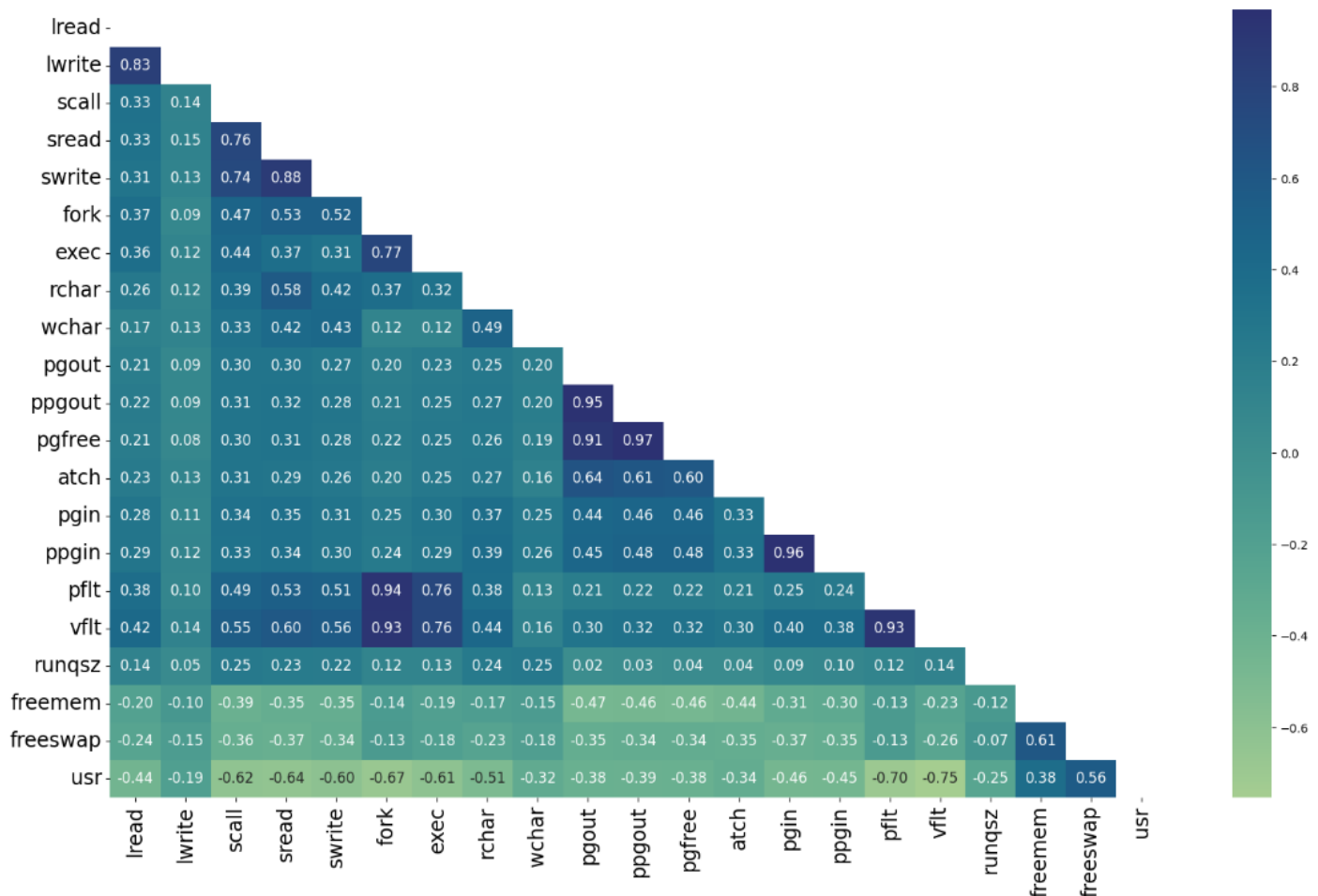


- 'fork' versus 'exec'



- The above scatterplots indicate some correlation between variables but to be sure, we need to look at the pairplot as well as the correlation heatmap of the dataset. Sample screenshot given below:

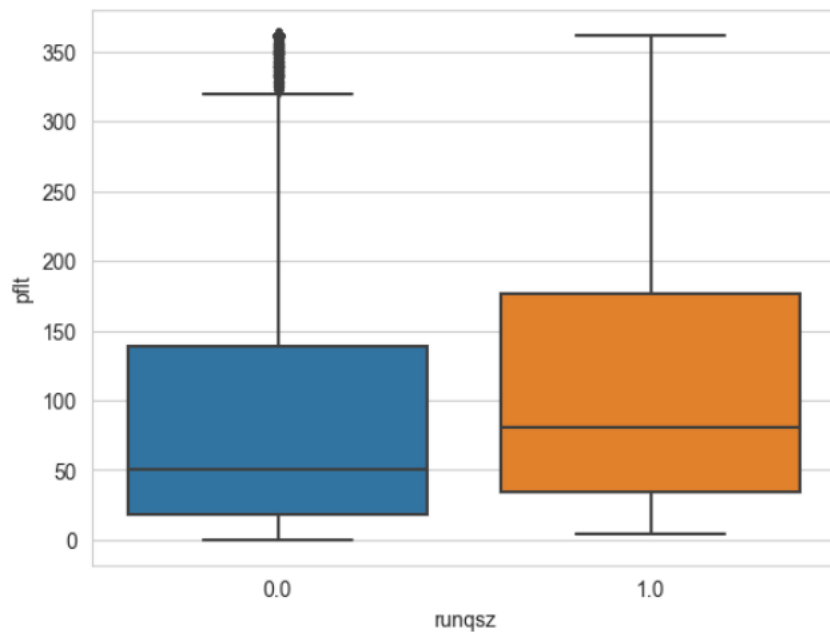




- Following pairs show a strong correlation from the above graphs:
  - a. pgin, ppgin – 0.96
  - b. pflt, vflt – 0.93
  - c. fork, pflt – 0.94
  - d. fork, vflt – 0.93
  - e. sread, swrite – 0.88
  - f. lread, lwrite – 0.83
  - g. fork, exec – 0.77
  - h. ppgout, pgfree – 0.97
  - i. ppgout, pgout – 0.95
  - j. scall, sread – 0.76
  - k. scall, swrite – 0.74
  
- Since 'usr' is our target variable, we can spot many strong inverse correlations between 'usr' and other variables like 'vflt', 'pflt', 'fork', and 'sread'.
  
- To reduce 'usr', we need to investigate the cause of increase in pflt.
- To investigate this further, I plotted a boxplot by keeping 'runqsz' in the x-axis and 'pflt'.
- The below boxplot shows that the average 'pflt' is higher in CPU\_Bound (1) records than in Not\_CPU\_Bound (0) records.

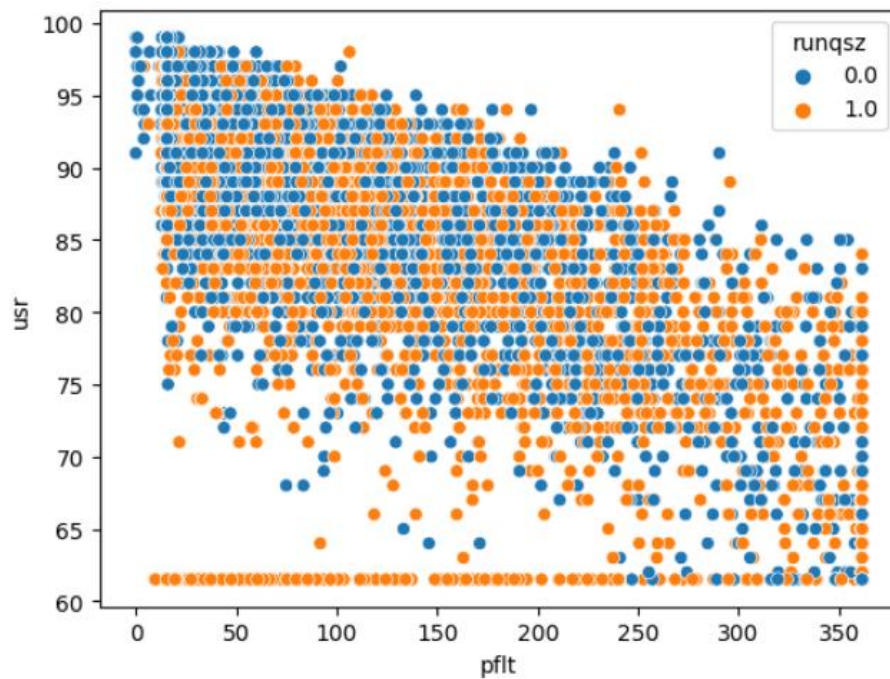


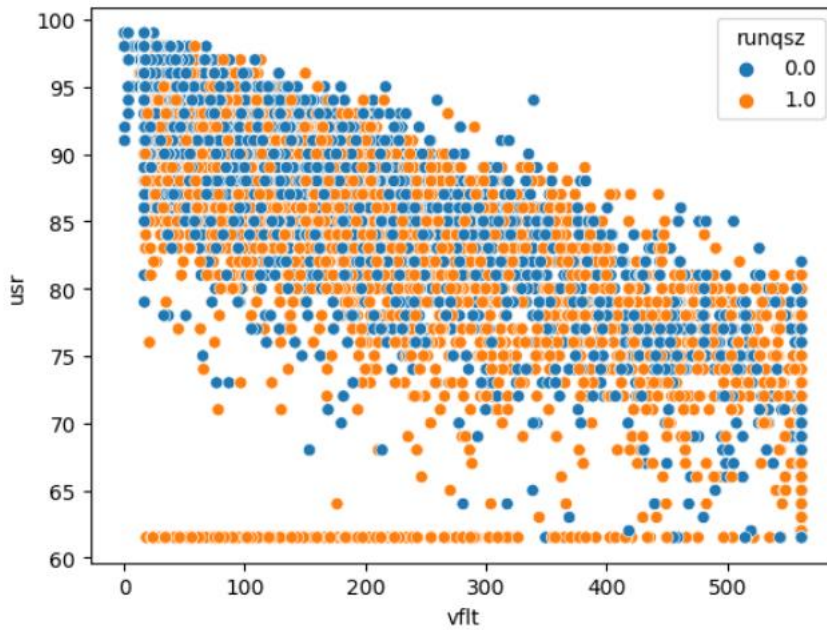
```
runqsz
0.0    93.738677
1.0    118.981735
Name: pflt, dtype: float64
```



xi. **Multivariate Analysis:**

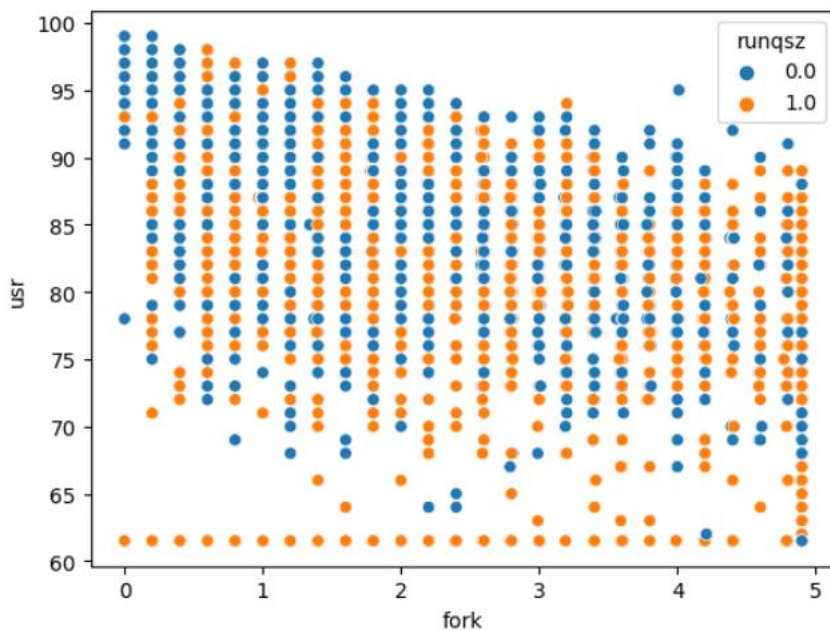
- Building on the bivariate analysis, I went one step further to analyse 'usr' against 'pflt'/'vflt', with 'runqsz' as hue:





Both, 'pflt' and 'vflt' against 'usr' show an inversely correlated pattern for both, 0 and 1 classes of 'runqsz' with little difference in pattern.

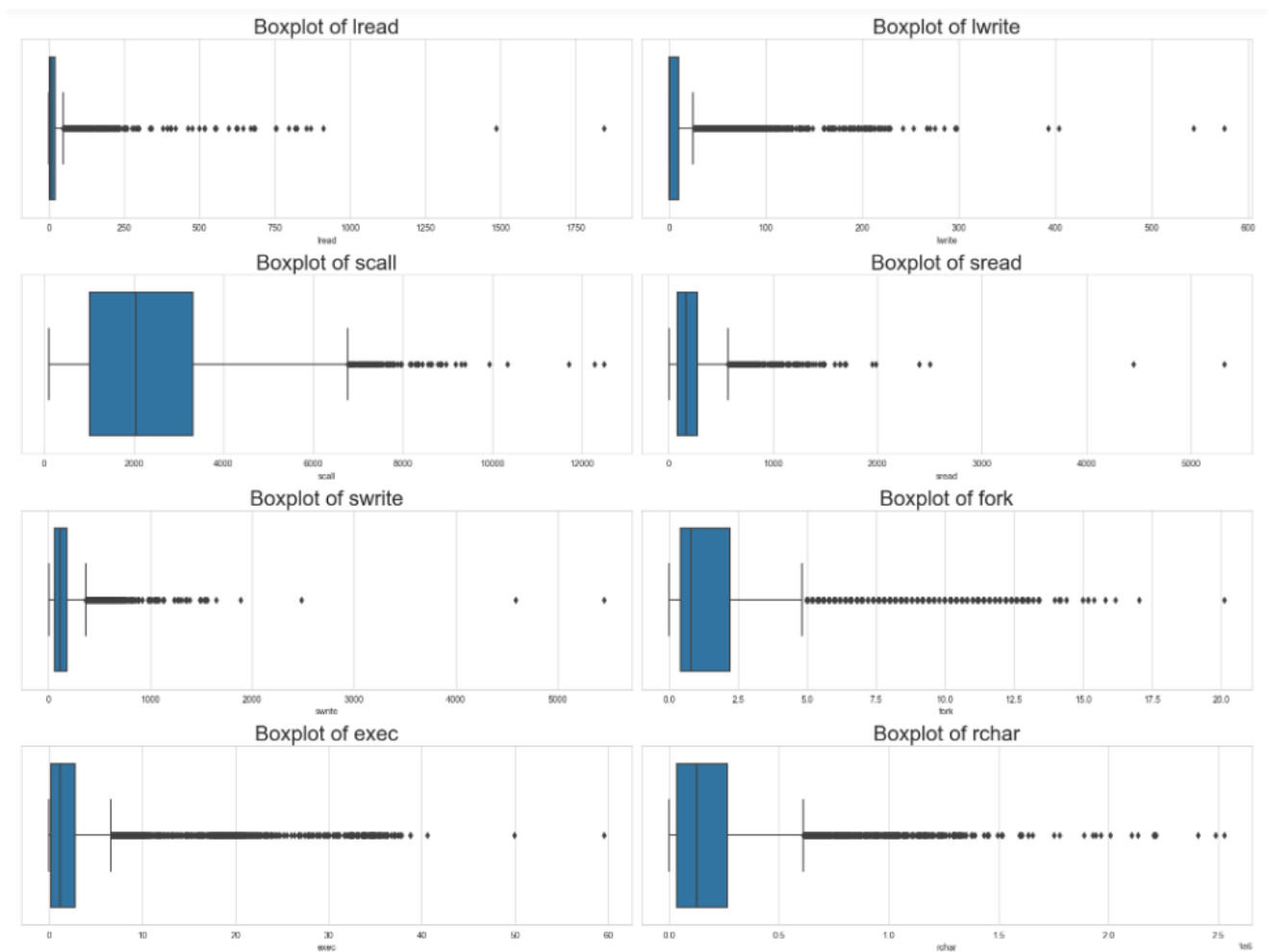
- Scatterplot with 'fork' and 'exec' on the axis and 'runqsz' as hue:

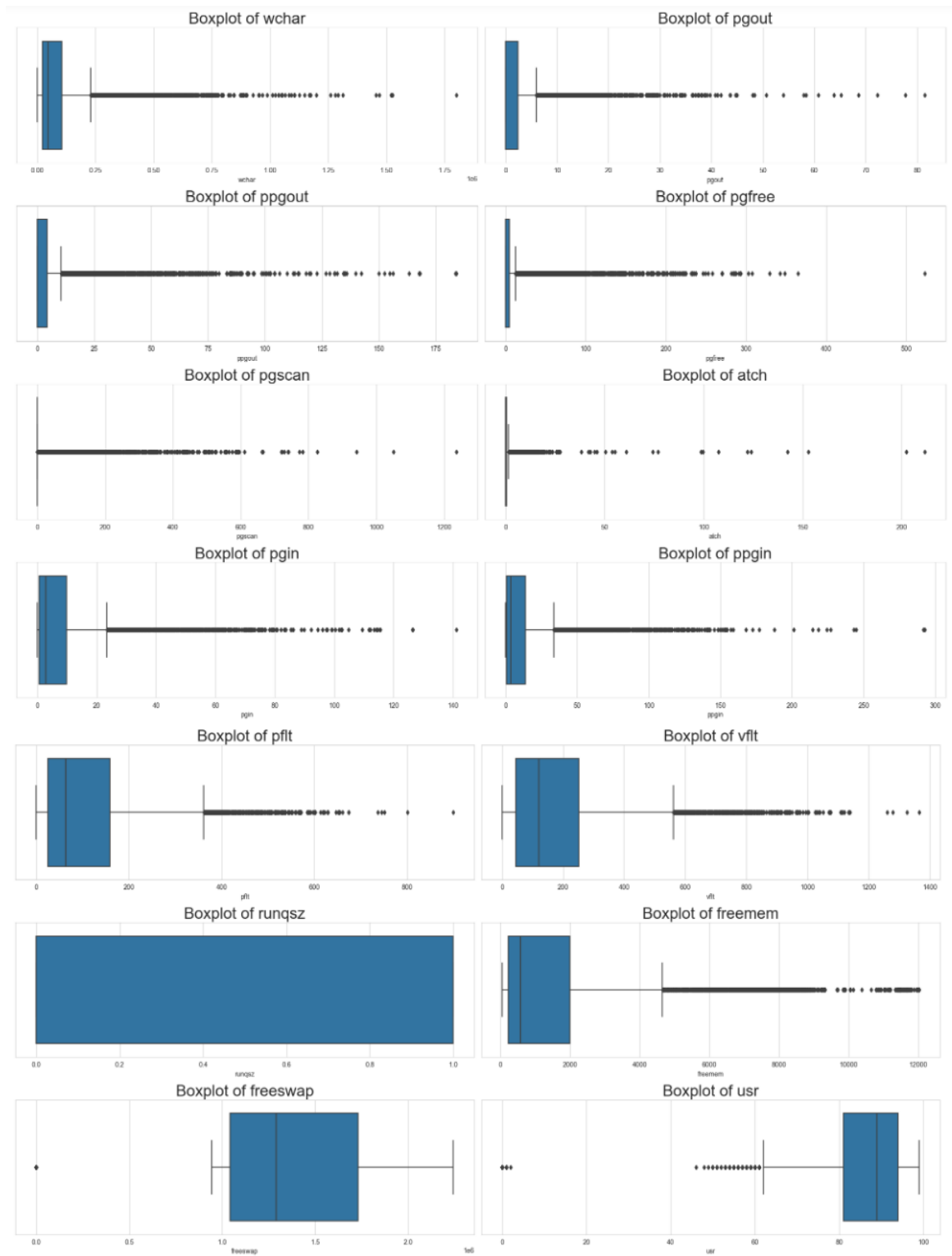


Even for 'usr' versus 'fork', no pattern relating to 'runqsz' classes found.

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.

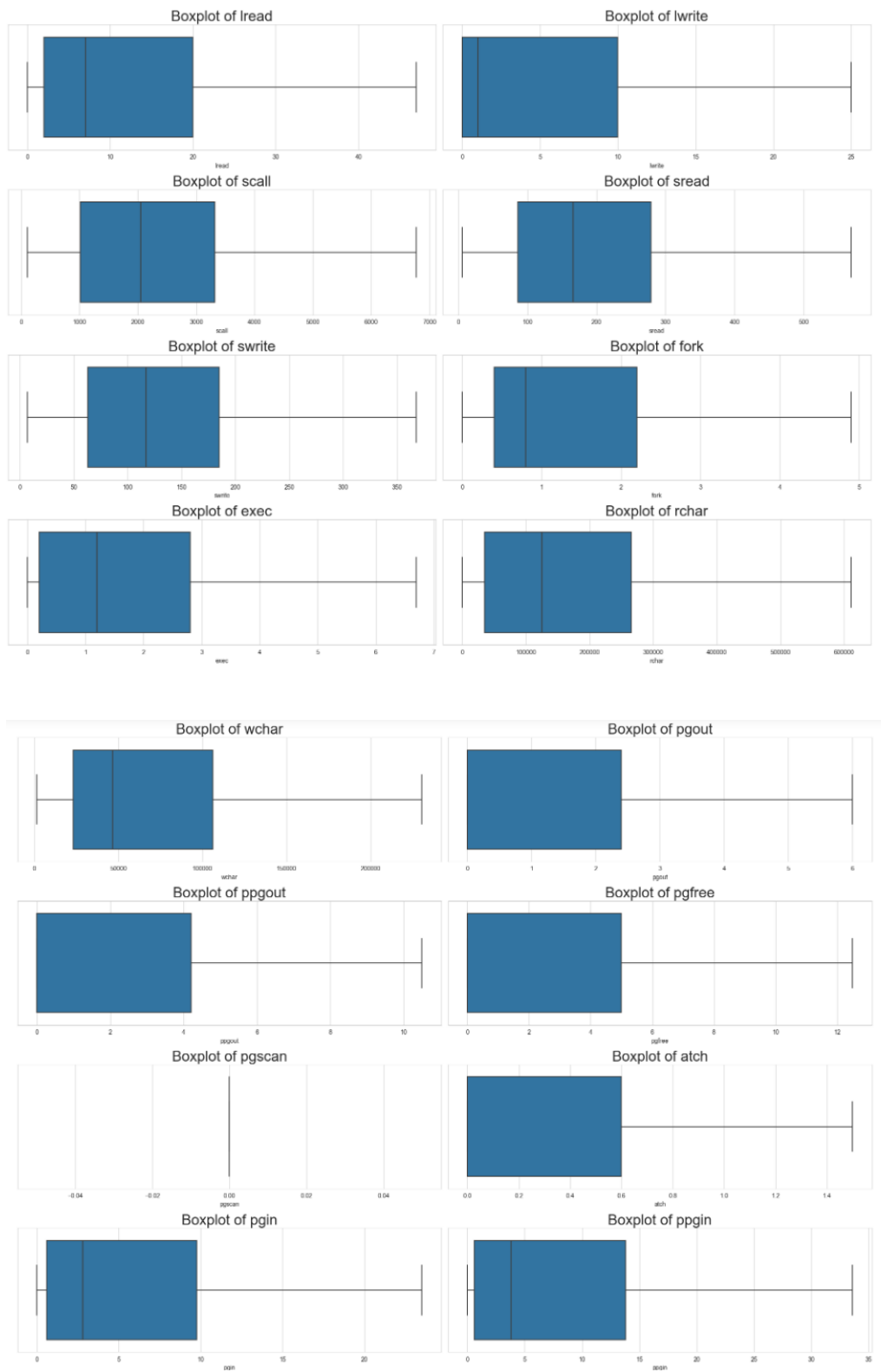
- Null values already imputed with median for columns 'rchar' and 'wchar'.
- No duplicates were found.
- 'pgscan' variable has been dropped because all the values in this columns came to a 0 after treating outliers. Even pre-outlier treatment, about 78% of the 'pgscan' values were 0. Due to this reason, I made the decision to drop it.
- No new features created/required yet.
- Outliers have been checked and treated. Snapshot given below:

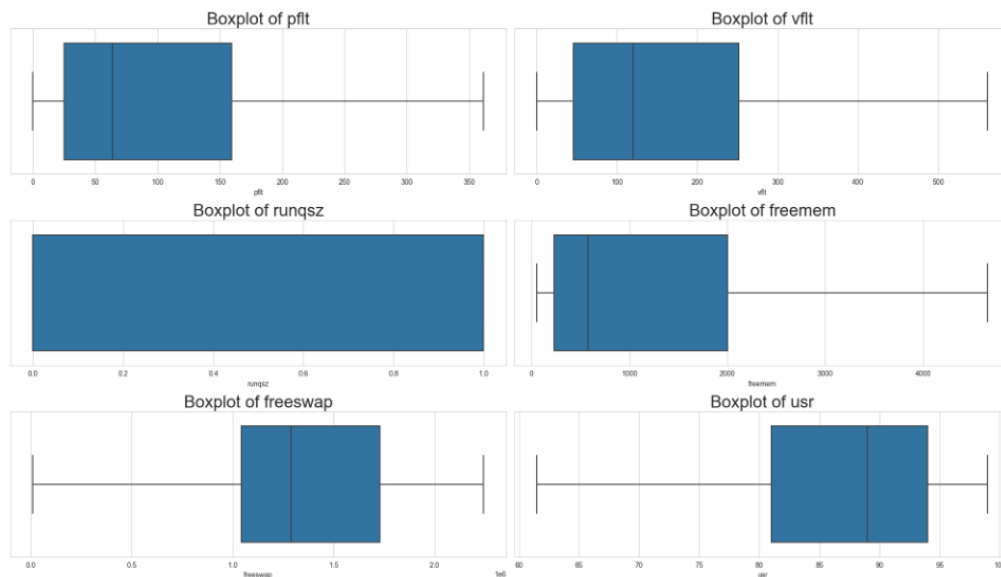




vi. After the treatment of outliers, the boxplots look like this:







- vii. 'runqsz' has been converted to numeric with values 0 and 1 already where 'CPU\_Bound' is assigned the value of 1 and 'Not\_CPU\_Bound' is assigned 0.

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 statsmodel. Create multiple models and check the performance of Predictions on Train and Test sets using Rsquare, RMSE & Adj Rsquare. Compare these models and select the best one with appropriate reasoning.

- i. Initial model without any adjustments:

```

=====
                        OLS Regression Results
=====
Dep. Variable:          usr      R-squared:          0.793
Model:                  OLS      Adj. R-squared:    0.792
Method:                 Least Squares      F-statistic:      1093.
Date:                  Sun, 05 Mar 2023      Prob (F-statistic): 0.00
Time:                  11:28:40      Log-Likelihood:   -16686.
No. Observations:      5734      AIC:              3.341e+04
Df Residuals:          5713      BIC:              3.355e+04
Df Model:              20
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	85.0494	0.298	285.844	0.000	84.466	85.633
lread	-0.0633	0.009	-7.147	0.000	-0.081	-0.046
lwrite	0.0456	0.013	3.517	0.000	0.020	0.071
scall	-0.0007	6.35e-05	-11.350	0.000	-0.001	-0.001
sread	0.0026	0.001	2.513	0.012	0.001	0.005
swrite	-0.0064	0.001	-4.423	0.000	-0.009	-0.004
fork	-0.0896	0.133	-0.672	0.502	-0.351	0.172
exec	-0.2494	0.051	-4.861	0.000	-0.350	-0.149
rchar	-4.983e-06	4.87e-07	-10.237	0.000	-5.94e-06	-4.03e-06
wchar	-5.196e-06	1.04e-06	-4.992	0.000	-7.24e-06	-3.16e-06
pgout	-0.4979	0.089	-5.575	0.000	-0.673	-0.323
ppgout	-0.0675	0.081	-0.837	0.403	-0.226	0.091
pgfree	0.1456	0.049	2.966	0.003	0.049	0.242
atch	0.5881	0.143	4.107	0.000	0.307	0.869
pgin	0.0518	0.029	1.778	0.076	-0.005	0.109
ppgin	-0.0846	0.020	-4.178	0.000	-0.124	-0.045
pflt	-0.0324	0.002	-16.409	0.000	-0.036	-0.029
vflt	-0.0062	0.001	-4.399	0.000	-0.009	-0.003
runqsz	-1.7242	0.126	-13.649	0.000	-1.972	-1.477
freemem	-0.0005	5.12e-05	-9.155	0.000	-0.001	-0.000
freeswap	9.223e-06	1.91e-07	48.401	0.000	8.85e-06	9.6e-06

```

=====
Omnibus:              980.750      Durbin-Watson:      2.022
Prob(Omnibus):        0.000      Jarque-Bera (JB):   1932.140
Skew:                 -1.040      Prob(JB):           0.00
Kurtosis:             4.940      Cond. No.           7.16e+06
=====

```

- In the initial model, we notice that the p-value for 'fork' and 'ppgout' is very high, followed by 'pgin' and 'sread'. The R-squared value is 0.793. Now we will go ahead and check the VIF (variance inflation factor) values for each variable to see how much multicollinearity has been brought by each feature.

VIF values:

```

const      25.629887
lread      5.222496
lwrite     4.230336
scall      2.987824
sread      6.555928
swrite     5.666273
fork       13.195160
exec       3.216047
rchar      2.088006
wchar      1.583353
pgout      11.215199
ppgout     30.947431
pgfree     17.468614
atch       1.848297
pgin       14.475734
ppgin      14.673035
pflt       11.703237
vflt       15.370510
runqsz     1.151214
freemem    1.974160
freeswap   1.847552
dtype: float64

```

- Since there are multiple features with higher than 10 VIF value, I have dropped one variable at a time and checked the R-squared for the model.
- R-squared and Adj. R-squared remained the same when 'ppgout', 'pgin' and 'fork' were dropped. This indicates that we can start by dropping these variables to check if the VIF values become suitable to build a better model.
- The value of R-squared and Adj. R-squared dropped by 0.009 when 'pflt' was dropped. This decline is relatively steep when compared to other variables. Hence, 'pflt' is concluded to be an important variable for the model.
- After dropping 'ppgout', this is the model as output:

OLS Regression Results						
=====						
Dep. Variable:	usr	R-squared:	0.793			
Model:	OLS	Adj. R-squared:	0.792			
Method:	Least Squares	F-statistic:	1150.			
Date:	Sun, 05 Mar 2023	Prob (F-statistic):	0.00			
Time:	11:28:41	Log-Likelihood:	-16687			
No. Observations:	5734	AIC:	3.341e+04			
Df Residuals:	5714	BIC:	3.355e+04			
Df Model:	19					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]
-----						
const	85.0682	0.297	286.732	0.000	84.487	85.650
lread	-0.0634	0.009	-7.167	0.000	-0.081	-0.046
lwrite	0.0457	0.013	3.530	0.000	0.020	0.071
scall	-0.0007	6.35e-05	-11.347	0.000	-0.001	-0.001
sread	0.0026	0.001	2.505	0.012	0.001	0.005
swrite	-0.0065	0.001	-4.425	0.000	-0.009	-0.004
fork	-0.0870	0.133	-0.653	0.514	-0.348	0.174
exec	-0.2502	0.051	-4.877	0.000	-0.351	-0.150
rchar	-4.983e-06	4.87e-07	-10.237	0.000	-5.94e-06	-4.03e-06
wchar	-5.234e-06	1.04e-06	-5.033	0.000	-7.27e-06	-3.2e-06
pgout	-0.5473	0.067	-8.159	0.000	-0.679	-0.416
pgfree	0.1124	0.029	3.878	0.000	0.056	0.169
atch	0.5903	0.143	4.123	0.000	0.310	0.871
pgin	0.0531	0.029	1.823	0.068	-0.004	0.110
ppgin	-0.0856	0.020	-4.237	0.000	-0.125	-0.046
pflt	-0.0324	0.002	-16.409	0.000	-0.036	-0.029
vflt	-0.0062	0.001	-4.419	0.000	-0.009	-0.003
runqsz	-1.7235	0.126	-13.644	0.000	-1.971	-1.476
freemem	-0.0005	5.11e-05	-9.186	0.000	-0.001	-0.000
freeswap	9.218e-06	1.9e-07	48.401	0.000	8.84e-06	9.59e-06
=====						
Omnibus:	980.533	Durbin-Watson:	2.023			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1932.446			
Skew:	-1.039	Prob(JB):	0.00			
Kurtosis:	4.941	Cond. No.	7.14e+06			
=====						

VIF values after dropping 'ppgout':

```

const      25.484056
lread      5.220098
lwrite     4.229441
scall      2.987783
sread      6.555363
swrite     5.666264
fork       13.188017
exec       3.214984
rchar      2.088006
wchar      1.580437
pgout      6.324897
pgfree     6.097389
atch       1.847637
pgin       14.437958
ppgin      14.617568
pflt       11.703237
vflt       15.362860
runqsz     1.151159
freemem    1.972182
freeswap   1.845464
dtype: float64

```

- VIF values still seem to be high. Hence, dropping other variables one after another as shown below:

After dropping 'fork':

```

=====
                        OLS Regression Results
=====
Dep. Variable:          usr      R-squared:                0.793
Model:                  OLS      Adj. R-squared:           0.792
Method:                 Least Squares      F-statistic:           1214.
Date:                  Sun, 05 Mar 2023      Prob (F-statistic):      0.00
Time:                  11:28:42      Log-Likelihood:        -16687.
No. Observations:      5734      AIC:                   3.341e+04
Df Residuals:          5715      BIC:                   3.354e+04
Df Model:              18
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	85.0919	0.294	289.002	0.000	84.515	85.669
lread	-0.0636	0.009	-7.188	0.000	-0.081	-0.046
lwrite	0.0462	0.013	3.575	0.000	0.021	0.072
scall	-0.0007	6.28e-05	-11.368	0.000	-0.001	-0.001
sread	0.0026	0.001	2.540	0.011	0.001	0.005
swrite	-0.0066	0.001	-4.629	0.000	-0.009	-0.004
exec	-0.2599	0.049	-5.289	0.000	-0.356	-0.164
rchar	-4.982e-06	4.87e-07	-10.235	0.000	-5.94e-06	-4.03e-06
wchar	-5.221e-06	1.04e-06	-5.022	0.000	-7.26e-06	-3.18e-06
pgout	-0.5468	0.067	-8.154	0.000	-0.678	-0.415
pgfree	0.1126	0.029	3.884	0.000	0.056	0.169
atch	0.5937	0.143	4.150	0.000	0.313	0.874
pgin	0.0533	0.029	1.831	0.067	-0.004	0.110
ppgin	-0.0850	0.020	-4.209	0.000	-0.125	-0.045
pflt	-0.0329	0.002	-18.092	0.000	-0.036	-0.029
vflt	-0.0067	0.001	-5.385	0.000	-0.009	-0.004
runqsz	-1.7225	0.126	-13.637	0.000	-1.970	-1.475
freemem	-0.0005	5.11e-05	-9.196	0.000	-0.001	-0.000
freeswap	9.205e-06	1.89e-07	48.624	0.000	8.83e-06	9.58e-06

```

=====
Omnibus:                981.667      Durbin-Watson:           2.023
Prob(Omnibus):          0.000      Jarque-Bera (JB):        1940.233
Skew:                   -1.039      Prob(JB):                0.00
Kurtosis:               4.949      Cond. No.                7.07e+06
=====

```

VIF values after dropping 'fork':

```

const      25.101797
lread      5.215916
lwrite     4.214692
scall      2.929090
sread      6.540038
swrite     5.467054
exec       2.948582
rchar      2.087974
wchar      1.579880
pgout      6.324301
pgfree     6.096861
atch       1.845154
pgin       14.435714
ppgin      14.578592
pflt       9.929178
vflt       11.857791
runqsz     1.150974
freemem    1.971834
freeswap   1.823381
dtype: float64

```

After dropping 'pgin':

```

=====
                        OLS Regression Results
=====
Dep. Variable:          usr      R-squared:                0.793
Model:                  OLS      Adj. R-squared:           0.792
Method:                  Least Squares      F-statistic:            1285.
Date:                    Sun, 05 Mar 2023    Prob (F-statistic):      0.00
Time:                    11:28:43           Log-Likelihood:          -16689.
No. Observations:        5734              AIC:                    3.341e+04
Df Residuals:            5716              BIC:                    3.353e+04
Df Model:                 17
Covariance Type:         nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	85.1390	0.293	290.213	0.000	84.564	85.714
lread	-0.0641	0.009	-7.241	0.000	-0.081	-0.047
lwrite	0.0466	0.013	3.603	0.000	0.021	0.072
scall	-0.0007	6.28e-05	-11.328	0.000	-0.001	-0.001
sread	0.0026	0.001	2.549	0.011	0.001	0.005
swrite	-0.0067	0.001	-4.649	0.000	-0.009	-0.004
exec	-0.2571	0.049	-5.234	0.000	-0.353	-0.161
rchar	-5.046e-06	4.86e-07	-10.393	0.000	-6e-06	-4.09e-06
wchar	-5.195e-06	1.04e-06	-4.997	0.000	-7.23e-06	-3.16e-06
pgout	-0.5433	0.067	-8.102	0.000	-0.675	-0.412
pgfree	0.1099	0.029	3.797	0.000	0.053	0.167
atch	0.6002	0.143	4.196	0.000	0.320	0.881
ppgin	-0.0502	0.007	-7.276	0.000	-0.064	-0.037
pflt	-0.0331	0.002	-18.270	0.000	-0.037	-0.030
vflt	-0.0064	0.001	-5.229	0.000	-0.009	-0.004
runqsz	-1.7181	0.126	-13.602	0.000	-1.966	-1.471
freemem	-0.0005	5.11e-05	-9.201	0.000	-0.001	-0.000
freeswap	9.179e-06	1.89e-07	48.609	0.000	8.81e-06	9.55e-06

```

=====
Omnibus:                986.106      Durbin-Watson:           2.022
Prob(Omnibus):           0.000      Jarque-Bera (JB):        1956.098
Skew:                    -1.042      Prob(JB):                 0.00
Kurtosis:                 4.961      Cond. No.                 7.05e+06
=====

```

VIF values after dropping 'pgin':

```

const      24.910077
lread      5.211756
lwrite     4.213626
scall      2.927634
sread      6.539854
swrite     5.466379
exec       2.945778
rchar      2.076999
wchar      1.579597
pgout      6.319010
pgfree     6.081600
atch       1.844025
ppgin      1.704801
pflt       9.876458
vflt       11.742820
runqsz     1.150570
freemem    1.971808
freeswap   1.813811
dtype: float64

```

After dropping 'vflt':

```

=====
                        OLS Regression Results
=====
Dep. Variable:          usr      R-squared:                0.792
Model:                  OLS      Adj. R-squared:          0.791
Method:                 Least Squares      F-statistic:          1357.
Date:                  Sun, 05 Mar 2023      Prob (F-statistic):    0.00
Time:                  11:28:43      Log-Likelihood:       -16702.
No. Observations:      5734      AIC:                  3.344e+04
Df Residuals:          5717      BIC:                  3.355e+04
Df Model:              16
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	84.9890	0.293	290.429	0.000	84.415	85.563
lread	-0.0674	0.009	-7.623	0.000	-0.085	-0.050
lwrite	0.0496	0.013	3.829	0.000	0.024	0.075
scall	-0.0007	6.3e-05	-11.218	0.000	-0.001	-0.001
sread	0.0023	0.001	2.261	0.024	0.000	0.004
swrite	-0.0074	0.001	-5.206	0.000	-0.010	-0.005
exec	-0.3101	0.048	-6.438	0.000	-0.405	-0.216
rchar	-5.214e-06	4.86e-07	-10.739	0.000	-6.17e-06	-4.26e-06
wchar	-4.386e-06	1.03e-06	-4.256	0.000	-6.41e-06	-2.37e-06
pgout	-0.5322	0.067	-7.923	0.000	-0.664	-0.401
pgfree	0.1030	0.029	3.552	0.000	0.046	0.160
atch	0.5437	0.143	3.803	0.000	0.263	0.824
ppgin	-0.0600	0.007	-8.998	0.000	-0.073	-0.047
pflt	-0.0408	0.001	-38.014	0.000	-0.043	-0.039
runqsz	-1.7107	0.127	-13.514	0.000	-1.959	-1.463
freemem	-0.0005	5.12e-05	-9.356	0.000	-0.001	-0.000
freeswap	9.328e-06	1.87e-07	49.855	0.000	8.96e-06	9.7e-06

```

=====
Omnibus:                921.359      Durbin-Watson:          2.020
Prob(Omnibus):          0.000      Jarque-Bera (JB):       1791.848
Skew:                   -0.988      Prob(JB):               0.00
Kurtosis:               4.895      Cond. No.                7.02e+06
=====

```



VIF values after dropping 'vflt':

```

const      24.671813
lread      5.184407
lwrite     4.205451
scall      2.926841
sread     6.520178
swrite     5.407892
exec       2.820231
rchar      2.067881
wchar      1.544589
pgout      6.312670
pgfree     6.068725
atch       1.833481
ppgin      1.581030
pflt       3.441366
runqsz     1.150426
freemem    1.969675
freeswap   1.772518
dtype: float64

```

Similar to these reiterations, I also dropped 'pgfree', 'sread', and 'lwrite'. Finally, the VIF values were low enough (lower than 5) for all variables to finalise the linear regression model, shown as below:

```

=====
                        OLS Regression Results
=====
Dep. Variable:          usr      R-squared:                0.790
Model:                  OLS      Adj. R-squared:           0.790
Method:                 Least Squares      F-statistic:           1659.
Date:                  Sun, 05 Mar 2023      Prob (F-statistic):       0.00
Time:                  11:28:44      Log-Likelihood:          -16719.
No. Observations:      5734      AIC:                    3.347e+04
Df Residuals:          5720      BIC:                    3.356e+04
Df Model:              13
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	85.1098	0.292	291.503	0.000	84.537	85.682
lread	-0.0384	0.004	-8.656	0.000	-0.047	-0.030
scall	-0.0007	6e-05	-11.335	0.000	-0.001	-0.001
swrite	-0.0052	0.001	-4.897	0.000	-0.007	-0.003
exec	-0.3179	0.048	-6.598	0.000	-0.412	-0.223
rchar	-4.734e-06	4.37e-07	-10.825	0.000	-5.59e-06	-3.88e-06
wchar	-4.411e-06	1.03e-06	-4.303	0.000	-6.42e-06	-2.4e-06
pgout	-0.3363	0.038	-8.808	0.000	-0.411	-0.261
atch	0.5511	0.143	3.848	0.000	0.270	0.832
ppgin	-0.0599	0.007	-9.197	0.000	-0.073	-0.047
pflt	-0.0414	0.001	-39.390	0.000	-0.043	-0.039
runqsz	-1.7475	0.127	-13.809	0.000	-1.996	-1.499
freemem	-0.0005	5.12e-05	-9.512	0.000	-0.001	-0.000
freeswap	9.315e-06	1.87e-07	49.901	0.000	8.95e-06	9.68e-06

```

=====
Omnibus:                937.245      Durbin-Watson:           2.020
Prob(Omnibus):          0.000      Jarque-Bera (JB):        1839.036
Skew:                   -1.000      Prob(JB):                0.00
Kurtosis:               4.923      Cond. No.                 6.98e+06
=====

```

VIF values after dropping 'lwrite':

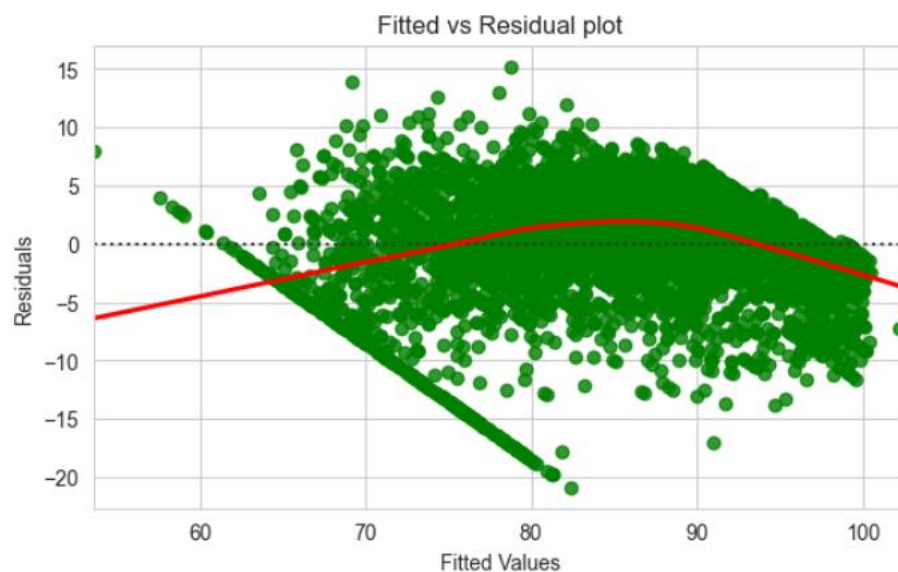
```
const      24.432718
lread      1.294422
scall      2.640944
swrite     2.939251
exec       2.806984
rchar      1.669087
wchar      1.520686
pgout      2.028511
atch       1.830347
ppgin      1.503077
pflt       3.284418
runqsz     1.143599
freemem    1.955042
freeswap   1.755178
dtype: float64
```

- After dropping the variables that were causing multi-collinearity, we have built the model. Note that it has not lost its R-squared value significantly.
- The former R-squared value was 0.793 and the final value is 0.790, which is a drop of 0.003 (insignificant).
- Therefore, model\_25 is the final model.
- Following features are included in the final model: 'const', 'lread', 'scall', 'swrite', 'exec', 'rchar', 'wchar', 'pgout', 'atch', 'ppgin', 'pflt', 'runqsz', 'freemem', 'freeswap'.

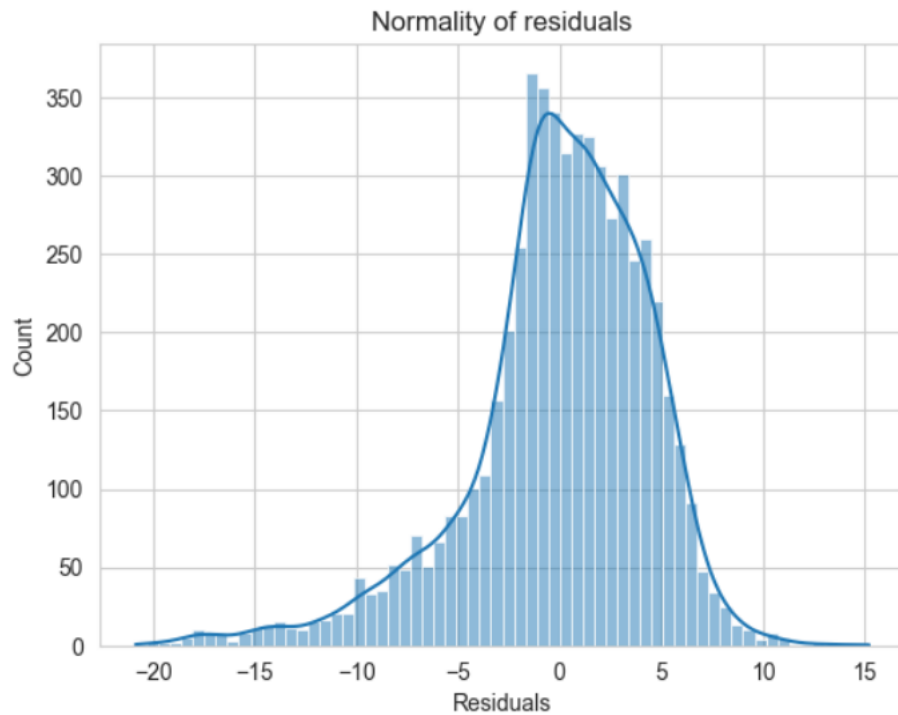
- ii. The predicted values are generated giving residuals against actual values as shown below:  
Note: This is only the first five rows of the data for reference.

	Actual Values	Fitted Values	Residuals
0	81.0	86.873553	-5.873553
1	93.0	88.050585	4.949415
2	64.0	64.330359	-0.330359
3	86.0	86.061522	-0.061522
4	94.0	98.473539	-4.473539

- iii. Residual plot:

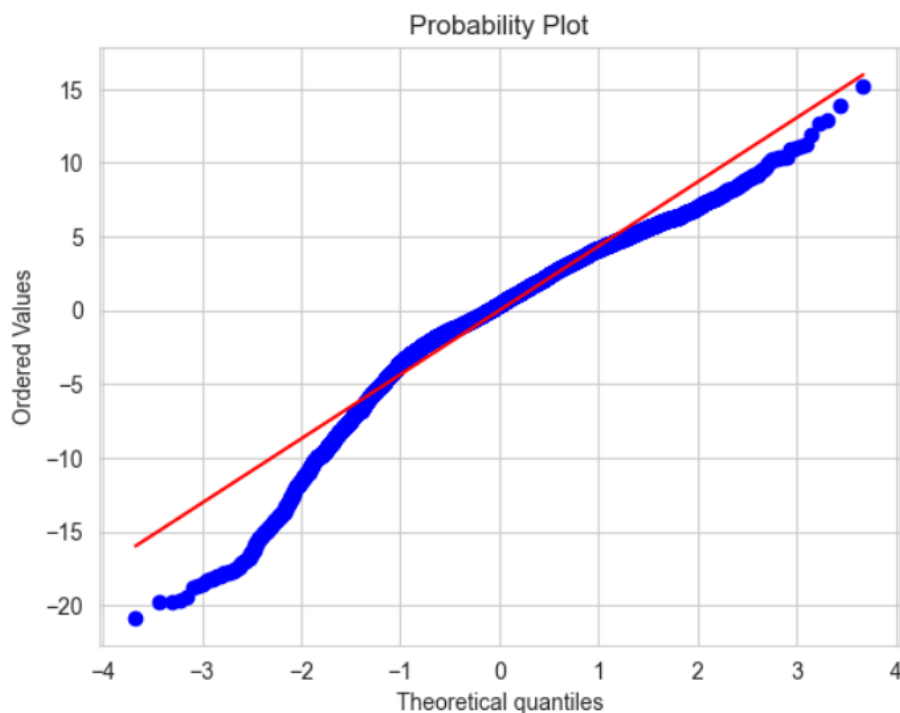


- iv. There is some non-linearity spotted in the residual plot. In the attempt to try and correct it, I looked at the pairplot of the dataset to check where the non-linearity might be coming from. 'freemem' seemed to bring in some non-linearity. However, on attempting to adjust it through freemem-squared transformation, the model did not hold good, as the VIF values spiked for multiple variables once again.
- v. Checking the normality assumption for residuals:



The distribution looks close to normal and can be accepted.

- vi. QQ plot for residuals:



- vii. Goldfeld–Quandt test is used to check for homoscedasticity, which is giving the p-value of 0.37, i.e., > 0.05. Hence, the test **fails to reject** the null hypothesis that the residuals are homoscedastic.
- viii. All assumptions of linear regression are roughly satisfied.

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

- i. Step 1: Created initial model  
 Step 2: Checked the VIF values for all variables along with the original model's R-squared and adj. R-squared values.  
 Step 3: Dropped variables with highest VIF value one by one and checked R-squared and adj. R-squared values of the model after dropping each variable.  
 Step 4: Finally, the model was left with features - 'const', 'lread', 'scall', 'swrite', 'exec', 'rchar', 'wchar', 'pgout', 'atch', 'ppgin', 'pflt', 'runqsz', 'freemem', 'freeswap', with all the VIF values under 5.  
 Step 5: The R-squared value of the final model was 0.790 as against 0.793 that we got in the beginning.  
 Step 6: Created dataframe with actual, predicted and residual values.  
 Step 7: Checked the linear regression assumptions like linearity, normality, homoscedasticity for the residuals.
- ii. Below is the final equation of linear regression that will help us understand the relationship between 'usr' and independent variables:
 
$$\begin{aligned} \text{usr} = & 85.0493905502606 + (-0.038355148624787956 * (\text{lread})) + (-0.0006796180899208 * (\text{scall})) + (-0.00516827790480312 * (\text{swrite})) + (-0.31791824161693133 * (\text{exec})) + (-4.734426806664307e-06 * (\text{rchar})) \\ & + (-4.410733409483615e-06 * (\text{wchar})) + (-0.3362796067741809 * (\text{pgout})) + 0.551091802085945 * (\text{atch}) + (-0.05990782621159879 * (\text{ppgin})) + (-0.04137305160578238 * (\text{pflt})) + (-1.7474768971194923 * (\text{runqsz})) + (-0.00048675572982739054 * (\text{freemem})) + 9.315381320207643e-06 * (\text{freeswap}) \end{aligned}$$
- iii. RMSE for train set = 4.467  
 RMSE for test set = 4.592
- iv. Mean absolute error for train set = 3.333  
 Mean absolute error for test set = 3.385
- v. The RMSE values for both, train and test data are close by, which indicates that the model is stable and not suffering from overfitting.
- vi. Business insights from the linear regression equation:
  - a. 'usr' is majorly dependent on 'runqsz', 'atch', 'pgout', 'exec' when compared to other features. According to this model, we can decrease the time that CPUs run in user mode by making machines depend less on CPU bound processes.
  - b. If we reduce the number of page attaches (satisfying a page fault by reclaiming a page in memory) per second (atch), we can lower the 'usr'.
  - c. Reducing the number of page out requests per second (pgout) and number of system exec calls per second (exec) will also have a positive impact on usr.

## Problem 2: Logistic Regression, LDA and CART

You are a statistician at the Republic of Indonesia Ministry of Health and you are provided with a data of 1473 females collected from a Contraceptive Prevalence Survey. The samples are married women who were either not pregnant or do not know if they were at the time of 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.

**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.**

- i. Head of the data (the first five rows): (first give rows for reference)

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

- ii. The contraceptive dataset has 1473 rows and 10 columns.
- iii. Null values: 'Wife\_age' has 71 null values while 'No\_of\_children\_born' has 21 null values.
- iv. 80 duplicates found and dropped from the dataset.
- v. The info on the original dataset is as follows: 2 *float64* datatype, 1 *int64* datatype, 7 *object* datatype variables.
- vi. The 5-point-summary (min, 25%, 50%, 75%, max) of each variable is given below through data description:

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Wife_age	1393.0	NaN	NaN	NaN	32.53051	8.088188	16.0	26.0	32.0	38.0	49.0
Wife_education	1393	4	Tertiary	515	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Husband_education	1393	4	Tertiary	827	NaN	NaN	NaN	NaN	NaN	NaN	NaN
No_of_children_born	1393.0	NaN	NaN	NaN	3.286432	2.381791	0.0	1.0	3.0	5.0	16.0
Wife_religion	1393	2	Scientology	1186	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Wife_Working	1393	2	No	1043	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Husband_Occupation	1393.0	4.0	3.0	570.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Standard_of_living_index	1393	4	Very High	618	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Media_exposure	1393	2	Exposed	1284	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Contraceptive_method_used	1393	2	Yes	779	NaN	NaN	NaN	NaN	NaN	NaN	NaN

- vii. The null values in 'Wife\_age' and 'No\_of\_children\_born' have been replaced with median of the respective features.
- viii. **Univariate Analysis:**

- Value counts of categorical variables are as follows:

```
Value counts for Wife_ education:
Tertiary      515
Secondary     398
Primary       330
Uneducated    150
Name: Wife_ education, dtype: int64
```

```
Value counts for Husband_education:
Tertiary      827
Secondary     347
Primary       175
Uneducated     44
Name: Husband_education, dtype: int64
```

```
Value counts for Wife_religion:
Scientology    1186
Non-Scientology 207
Name: Wife_religion, dtype: int64
```

```
Value counts for Wife_Working:
No      1043
Yes      350
Name: Wife_Working, dtype: int64
```

```
Value counts for Husband_Occupation:
3      570
2      415
1      381
4       27
Name: Husband_Occupation, dtype: int64
```

```
Value counts for Standard_of_living_index:
Very High    618
High         419
Low          227
Very Low     129
Name: Standard_of_living_index, dtype: int64
```

```
Value counts for Media_exposure :
Exposed      1284
Not-Exposed   109
Name: Media_exposure , dtype: int64
```

```
Value counts for Contraceptive_method_used:
Yes      779
No       614
Name: Contraceptive_method_used, dtype: int64
```

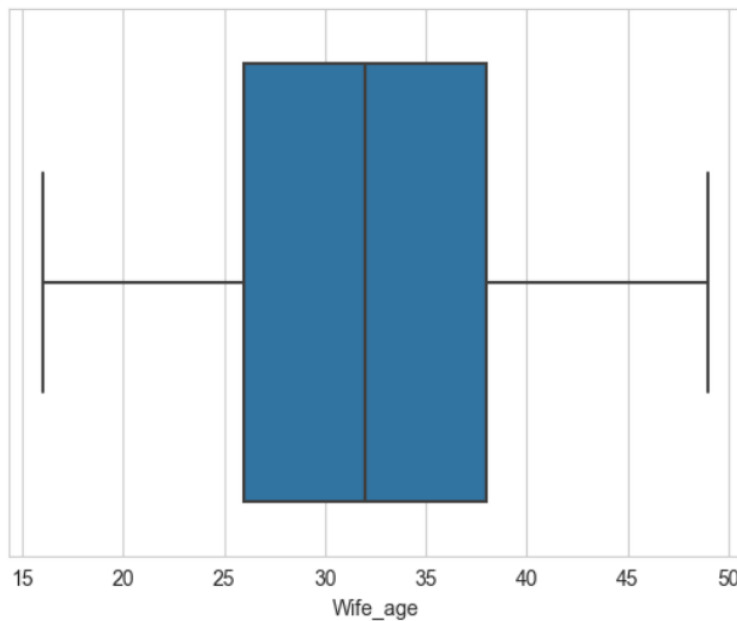
- Value counts for non-categorical, numeric variables:

```
Value counts for Wife_age:
[24. 45. 43. 42. 36. 19. 38. 21. 27. 44. 26. 48. 39. 37. 46. 40. 29. 31.
 33. 25. 28. 47. 32. 49. 34. 20. 22. 30. 23. 35. 41. 17. 18. 16.]
```

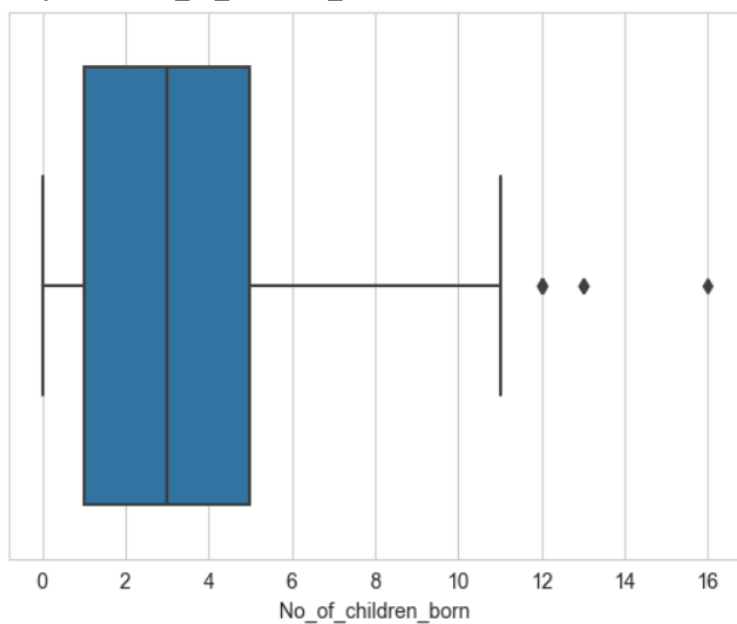
```
Value counts for No_of_children_born:
[ 3. 10.  7.  9.  8.  0.  6.  1.  2.  4.  5. 12. 11. 13. 16.]
```

- Checking outliers for the above continuous and discrete variables, i.e., Wife\_age and No\_of\_children\_born:

**Boxplot for Wife\_age**



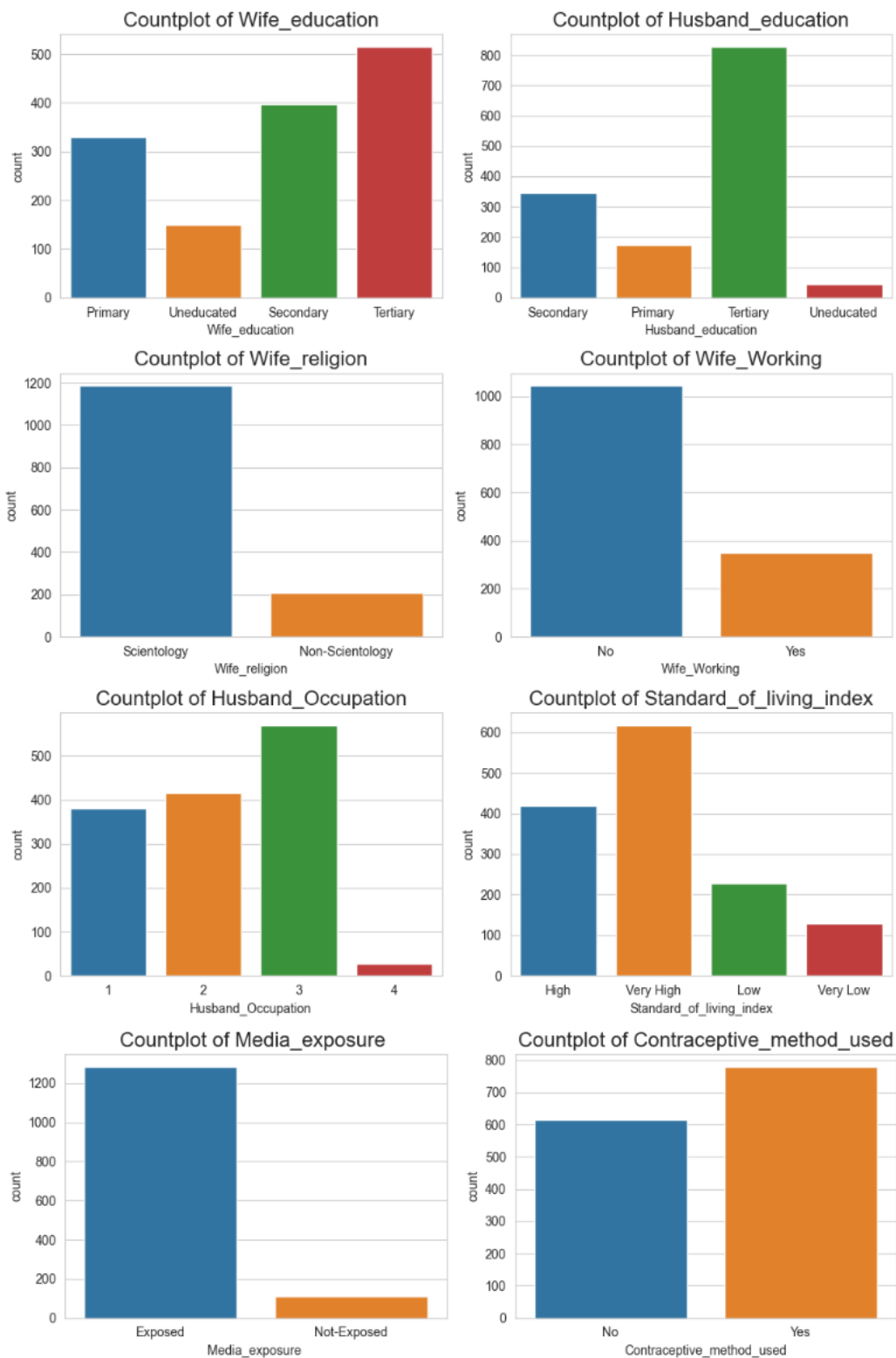
**Boxplot for No\_of\_children\_born**



- As shown above, Number\_of\_children\_born has outliers.



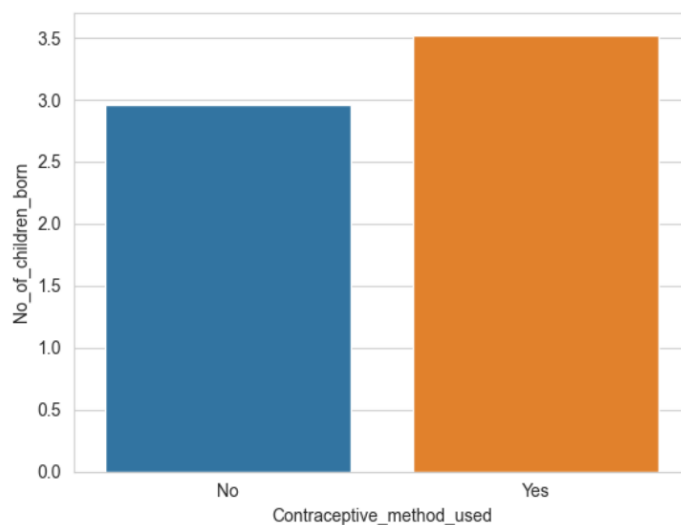
- Treating outliers with the IQR method where the outliers beyond the upper range are replaced with the upper range values using the formula:  $\text{upper range} = Q3 + (1.5 * IQR)$ .
- Countplots for individual features are shown below:



- Insights:
  - a. Majority (estimate: 64%) wives are educated at secondary and tertiary levels.
  - b. 59% husbands are educated to tertiary level while 24% are educated till secondary level. When compared to wives in tertiary level education, more husbands are in this tier.
  - c. According to the above countplots, majority wives are not working belong to scientology religion.
  - d. Most husbands are working in tier 3 while almost equal proportion of husbands work in 1 and 2 tiers. The tier 4 of occupation only has about 1.93% of husbands, which is the lowest.
  - e. When it comes to standard of living, majority records belong to 'very high' and 'high' categories.
  - f. 55.92% of the records use contraceptive while the rest 44% don't use contraceptive.

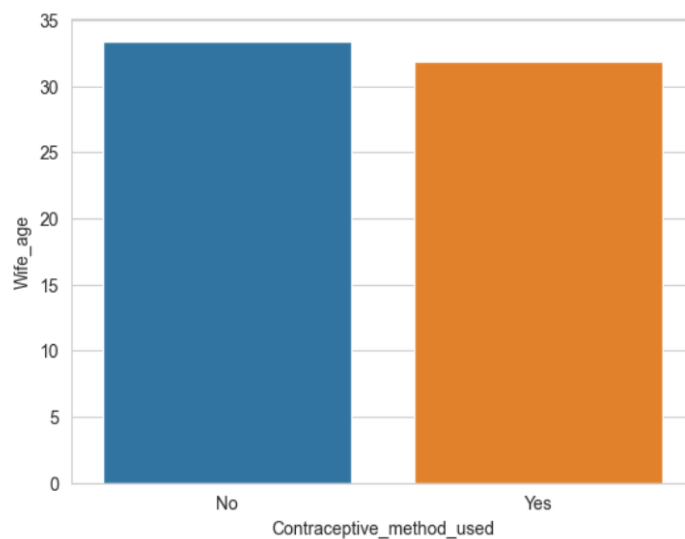
#### ix. Bivariate Analysis:

- When checking the relationship between No\_of\_children\_born with Contraceptive\_method\_used, we see that there is not a major gap between the two categories of Used and Not Used.

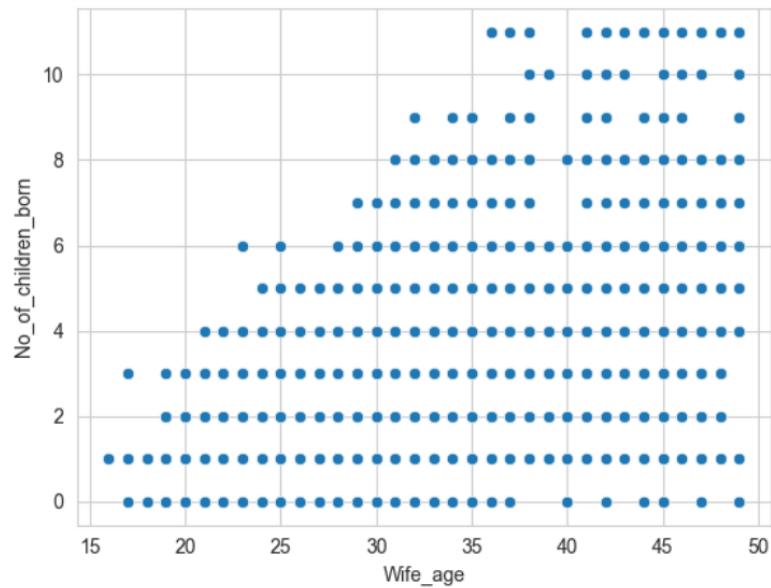


This could be an indication of ineffective contraceptives used by the records in the dataset.

- When trying to answer the question, “Do more younger women want to use contraceptives for birth control?”, there could not be found any pattern as shown below:

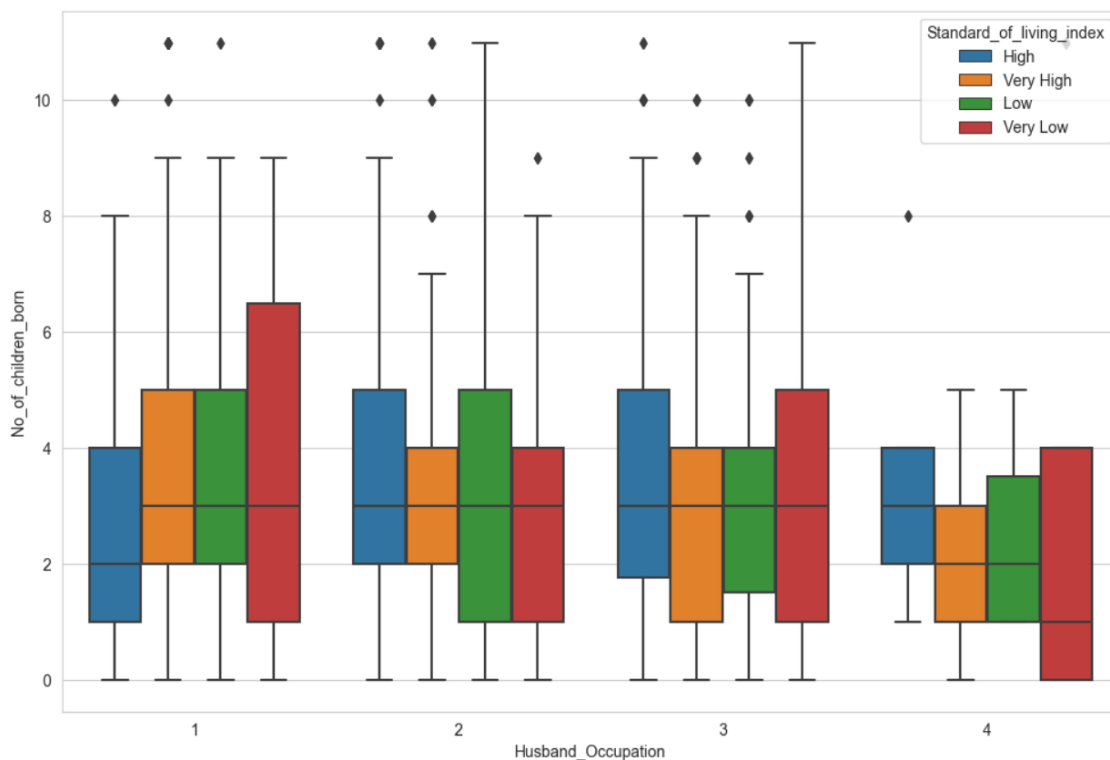


- Moreover, there was only a 0.53 correlation between Wife\_age and No\_of\_children\_born. Scatterplot for the two variables is shown below:

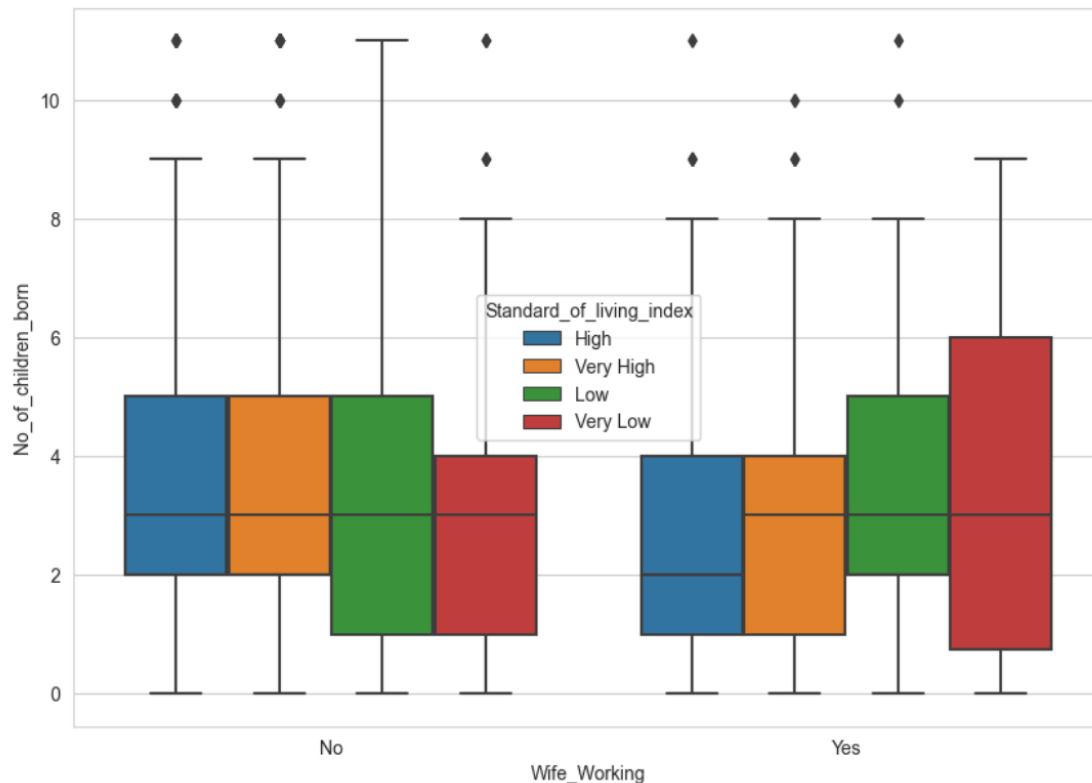


#### x. Multivariate Analysis:

- Number of children born in Husband\_Occupation 4 is the lowest.
- Below is the boxplot of No of children born for each Husband\_Occupation, with Standard of living as hue:



- In Occupation 2, records with 'Low' Standard of living have the highest number of children whereas in Occupation 3, records with 'Very Low' Standards of living have the highest number of children. This could indicate the inaccessibility of contraceptives and education in these standards of living and Occupation.
- Next, I checked if Wife\_Working has an impact on the number of children:



- There was no major difference found in the pattern of Working and Non-working wives with respect to the living standards. However, wives who are not working have higher number of children in all categories of living standards from 'Very High' to 'Very Low'.

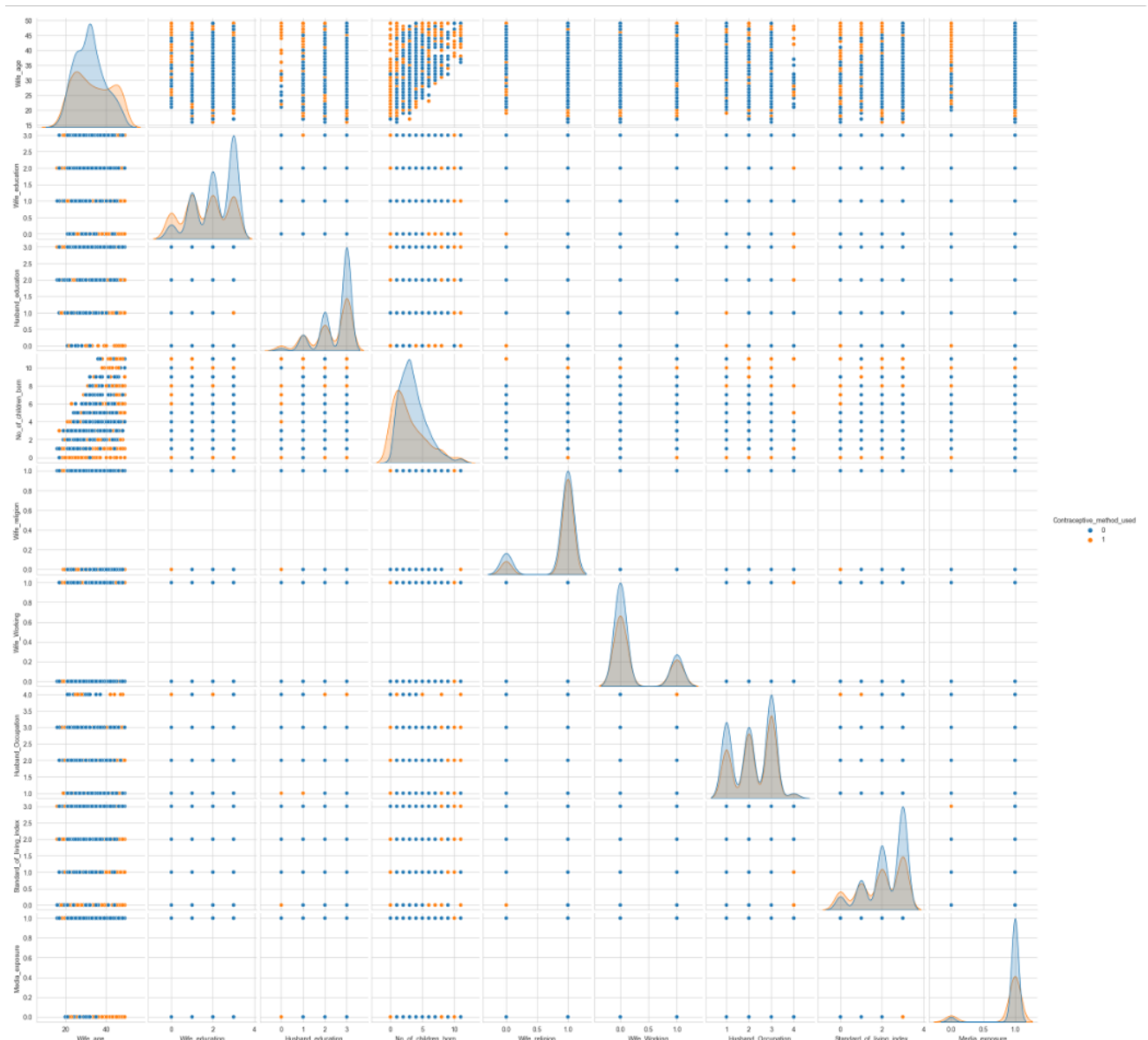
2.2 Do not scale the data. Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Logistic Regression and LDA (linear discriminant analysis) and CART.

## **Logistic Regression**

- To prepare the dataset for modelling, I have converted string values of categorical columns to int values, in an ordinal style.
- Criteria used to encode string data: In ordinal categories, 0 indicates the lowest value. For Binary independent variables, 0 indicates 'No' and 1 indicates 'Yes'. For Binary dependent variable, 0 represents 'Yes' and 1 represents 'No', since 'No' class is our class of interest for dependent variable.
- All variables converted to numerical values:

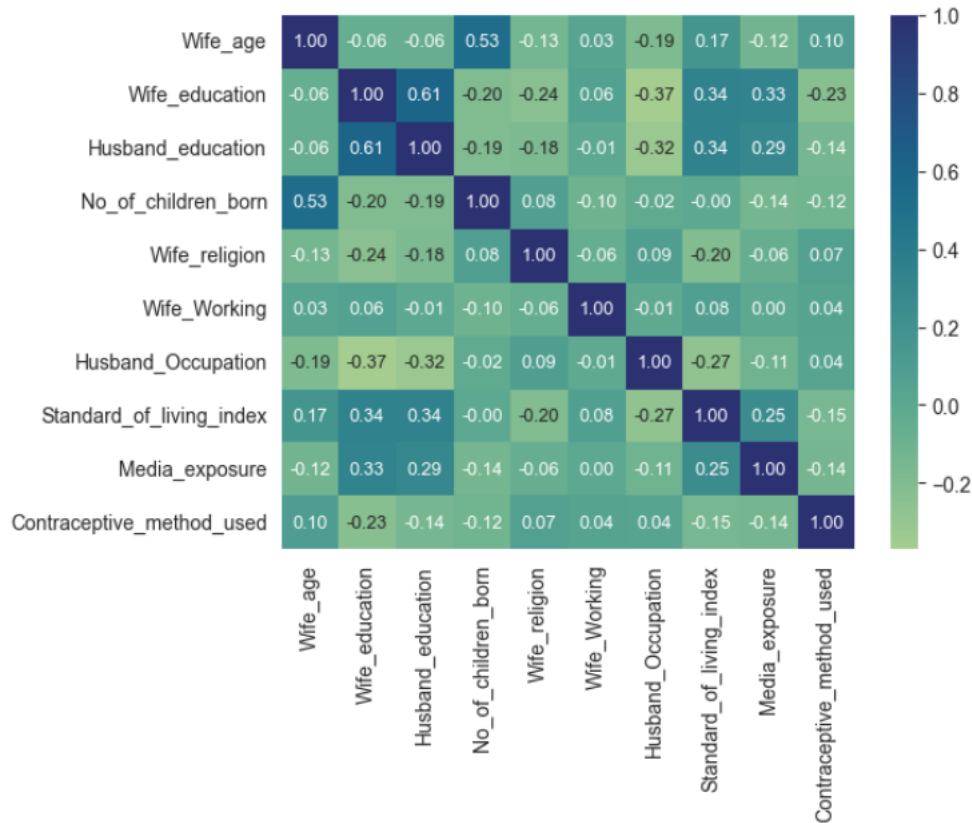
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1393 entries, 0 to 1472
Data columns (total 10 columns):
#   Column                                Non-Null Count  Dtype
---  ---                                -
0   Wife_age                             1393 non-null   float64
1   Wife_education                       1393 non-null   int64
2   Husband_education                   1393 non-null   int64
3   No_of_children_born                 1393 non-null   float64
4   Wife_religion                       1393 non-null   int64
5   Wife_Working                        1393 non-null   int64
6   Husband_Occupation                  1393 non-null   int64
7   Standard_of_living_index            1393 non-null   int64
8   Media_exposure                      1393 non-null   int64
9   Contraceptive_method_used           1393 non-null   int64
dtypes: float64(2), int64(8)
memory usage: 152.0 KB
```

iv. Pairplot with class variable 'Contraceptive\_method\_used' as hue:



The diagonals represent the overlap between the independent variables and the class variable. Most of the independent variables overlap between both the classes, which means that they are poor or weak predictors and do not split the binary class well.

v. Following is the heatmap indicating correlations amongst variables:



The independent variables do not seem to have multicollinearity. Hence, it is safe to go ahead with building a Logit model.

vi. LOGIT TRAIN DATA SET MODEL:

	precision	recall	f1-score	support
0	0.67	0.80	0.73	547
1	0.66	0.50	0.57	428
accuracy			0.66	975
macro avg	0.66	0.65	0.65	975
weighted avg	0.66	0.66	0.66	975

Accuracy score: 0.6646

LOGIT TEST DATA SET MODEL:

	precision	recall	f1-score	support
0	0.66	0.85	0.75	232
1	0.71	0.46	0.56	186
accuracy			0.68	418
macro avg	0.69	0.66	0.65	418
weighted avg	0.69	0.68	0.66	418

Accuracy score: 0.677

## **LDA**

- i. As mentioned in the question above, not scaling the data for LDA as pre-processing.
- ii. Checked independent variables for correlation. All independent variables have low correlation so it is safe to go ahead.
- iii. LDA TRAIN DATA SET MODEL:

	precision	recall	f1-score	support
0	0.67	0.81	0.73	547
1	0.66	0.48	0.56	428
accuracy			0.66	975
macro avg	0.66	0.64	0.64	975
weighted avg	0.66	0.66	0.65	975

Accuracy score: 0.6646

LDA TEST DATA SET MODEL:

	precision	recall	f1-score	support
0	0.65	0.86	0.74	232
1	0.71	0.42	0.53	186
accuracy			0.67	418
macro avg	0.68	0.64	0.63	418
weighted avg	0.68	0.67	0.65	418

Accuracy score: 0.665

## **CART**

- i. CART TRAIN DATA SET MODEL:

	precision	recall	f1-score	support
0	0.77	0.83	0.80	547
1	0.76	0.68	0.72	428
accuracy			0.76	975
macro avg	0.76	0.75	0.76	975
weighted avg	0.76	0.76	0.76	975

Accuracy score: 0.7641



## CART TEST DATA SET MODEL:

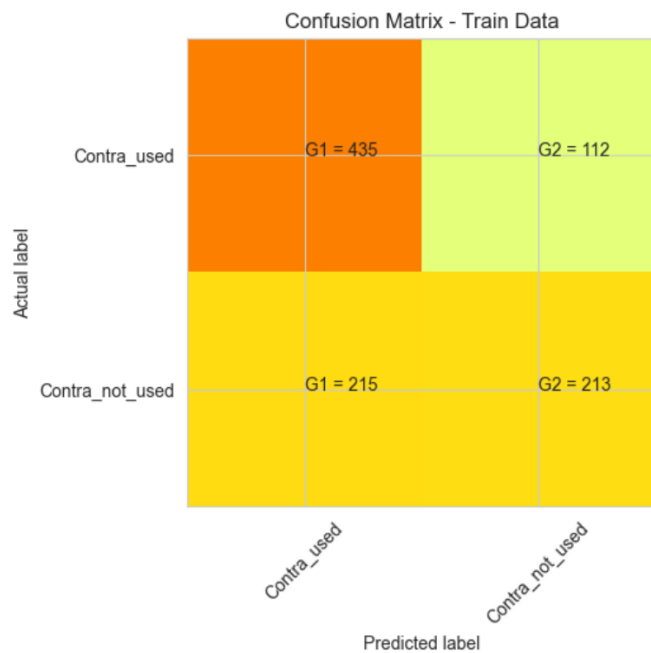
	precision	recall	f1-score	support
0	0.66	0.75	0.70	232
1	0.63	0.53	0.57	186
accuracy			0.65	418
macro avg	0.65	0.64	0.64	418
weighted avg	0.65	0.65	0.65	418

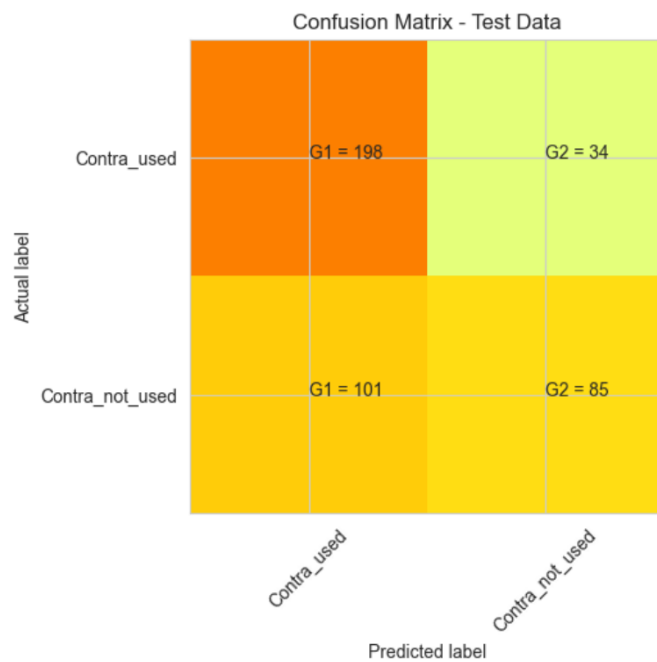
Accuracy score: 0.6507

- 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.

## Logistic Regression

i. Confusion matrix:



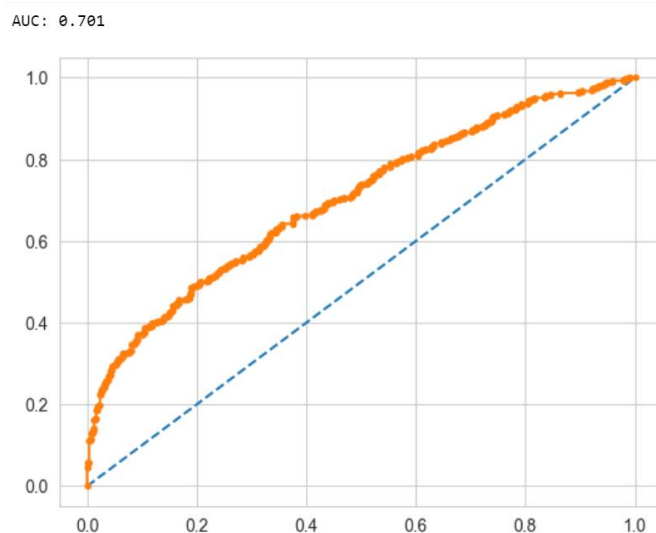


- The logit model has an accuracy score of 0.67 and precision score of 0.71 (for class 1), which is an acceptable score.

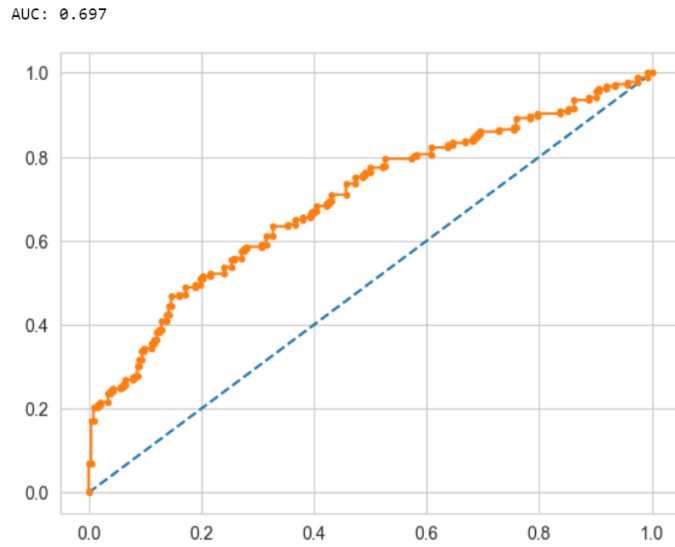
vii. In the above confusion matrix:

G1 = 198 is True Positive  
 G1 = 101 is False Positive  
 G2 = 85 is True Negative  
 G2 = 34 is False Negative

viii. ROC curve and AUC score for train set:



ROC curve and AUC score for test set:

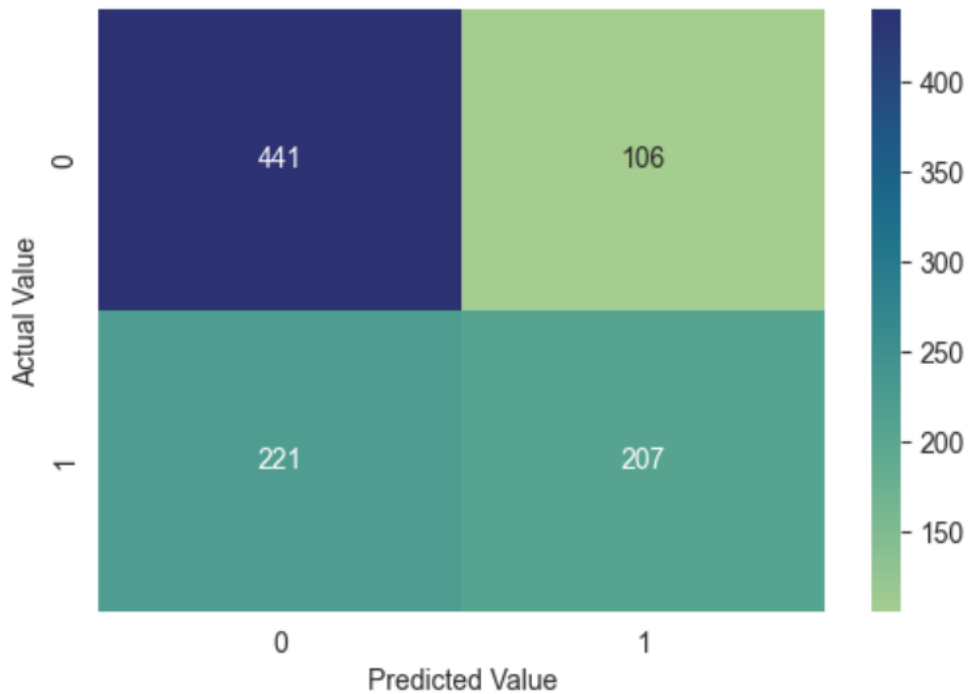


- ix. Equation:  $Y = 0.04305369 + X1 \cdot 0.07867066 + X2 \cdot (-0.44697918) + X3 \cdot (-0.14787693) + X4 \cdot (-0.27758321) + X5 \cdot 0.39656854 + X6 \cdot 0.16490436 + X7 \cdot (-0.13464495) + X8 \cdot (-0.19488841) + X9 \cdot (-0.46818363)$

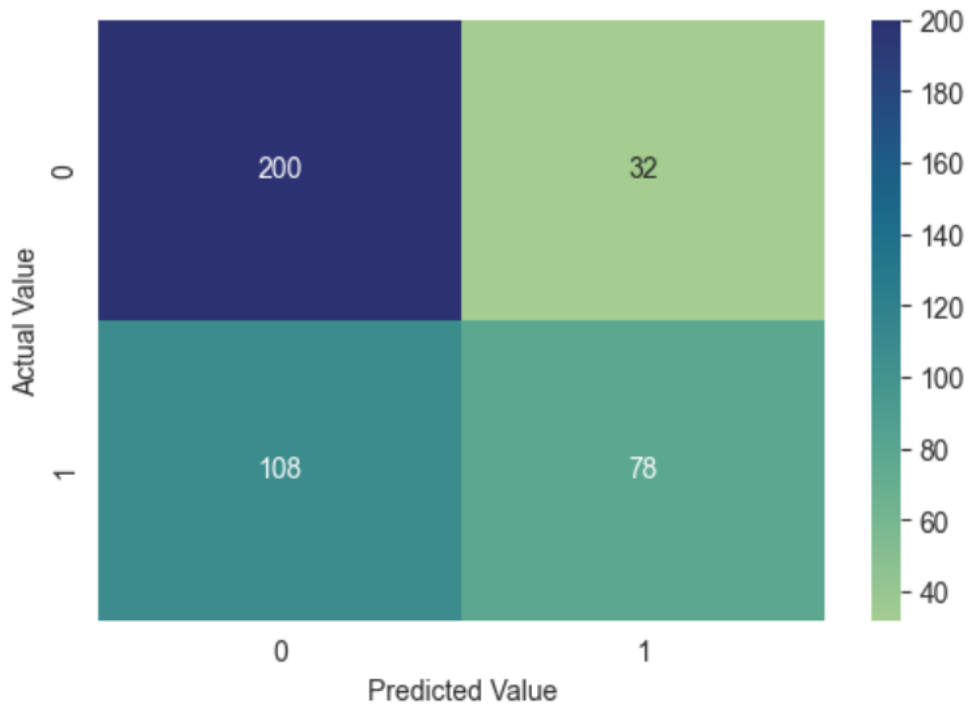
## LDA

- iv. Confusion matrix:

TRAIN SET:



TEST SET:



- The LDA model has predicted 110 rows as 1 (target class) and 308 rows as 0.

v. Equation:

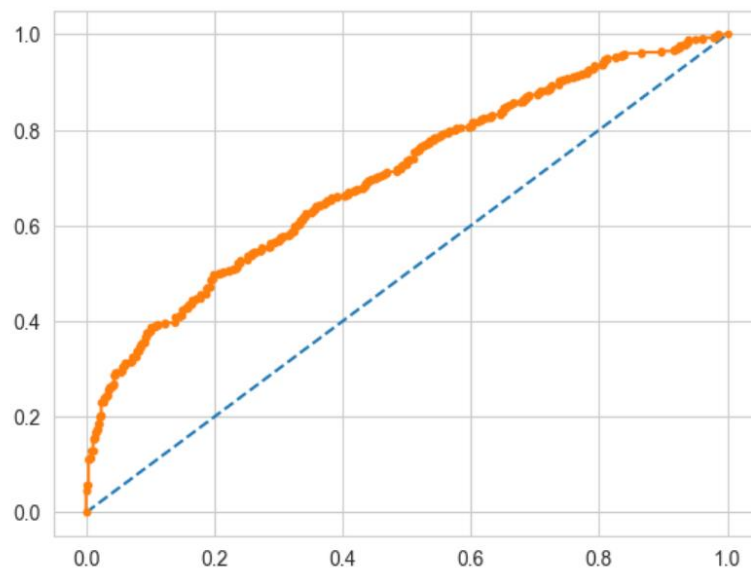
$$\text{LDF} = 0.091 + X1 \cdot 0.076 + X2 \cdot (-0.454) + X3 \cdot (-0.141) + X4 \cdot (-0.270) + X5 \cdot 0.406 + X6 \cdot (0.169) + X7 \cdot (-0.142) + X8 \cdot (-0.202) + X9 \cdot (-0.480)$$

vi. X9 and X2 have the most discriminating power towards the class variable, since its coefficient has the highest magnitude.

vii. The least discriminating power is coming from X1 since it has the lowest magnitude.

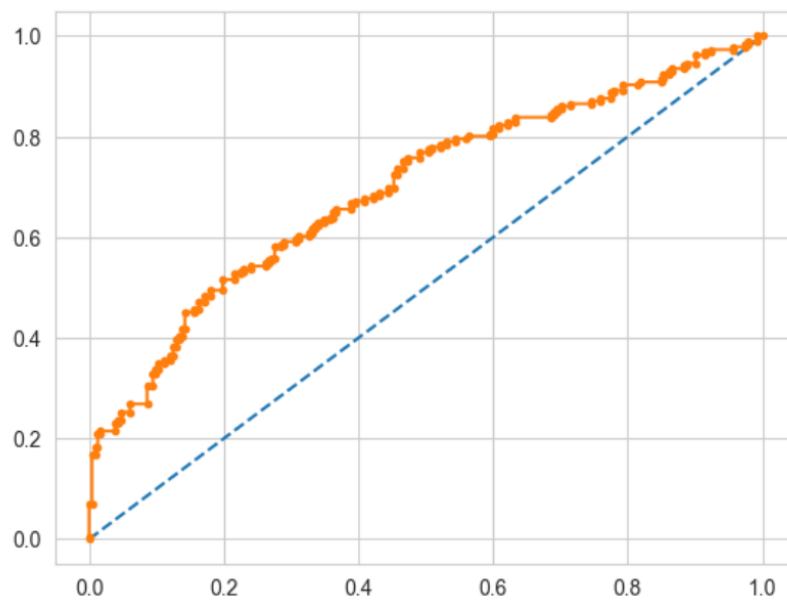
viii. ROC Curve and AUC score for train set:

AUC: 0.701



ROC Curve and AUC score for test set:

AUC: 0.697



**CART**

ii. Gini importance of all the features:

	Gini_Imp
Wife_age	0.302709
Wife_education	0.103852
Husband_education	0.076279
No_of_children_born	0.237810
Wife_religion	0.040490
Wife_Working	0.052719
Husband_Occupation	0.089974
Standard_of_living_index	0.078466
Media_exposure	0.017701

iii. Above is the Gini importance for each feature. This parameter tells us how much each feature has been used in splitting the dependent variable. The above values are normalized.

iv. After creating the decision tree, pruning was done to regularize the model. The following parameters were used for pruning:

Max depth = 9

Min sample leaf = 10

Min sample split = 30

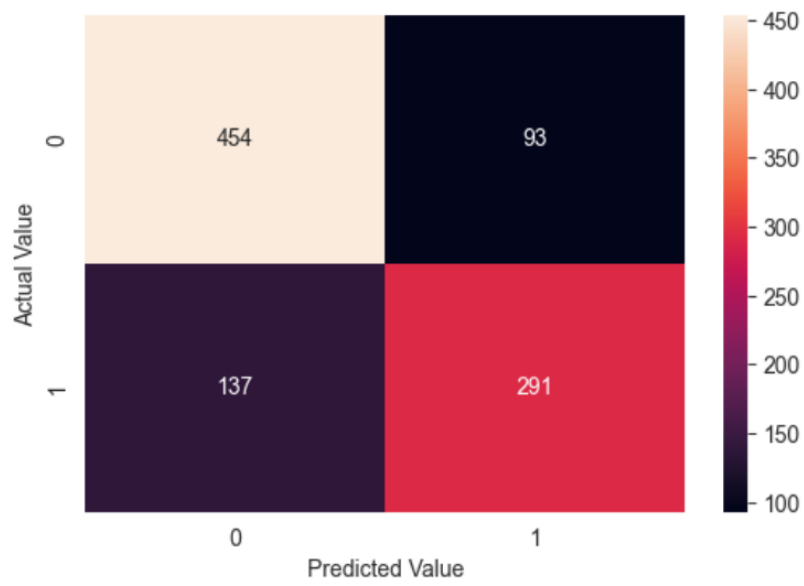
v. New Gini importance table:

	Gini_Imp
Wife_age	0.323388
Wife_education	0.154638
Husband_education	0.029856
No_of_children_born	0.396749
Wife_religion	0.007844
Wife_Working	0.002985
Husband_Occupation	0.039359
Standard_of_living_index	0.045180
Media_exposure	0.000000

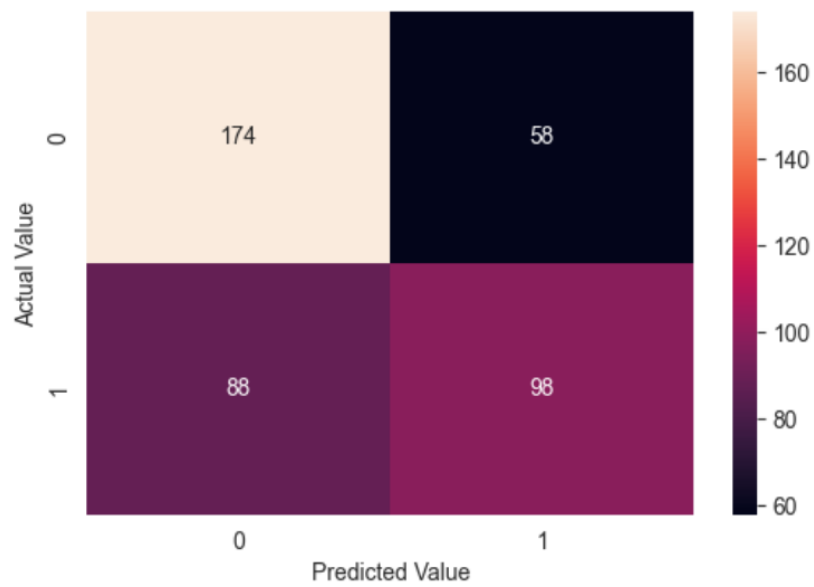
- The feature importance of No\_of\_children\_born, Wife\_age, and Wife\_education are the highest.
- If the gini importance of any variable is 0, it suggests that this particular variable was not used in splitting the class variable, hence, it can be dropped.
- Above, we can see that Media\_exposure feature was not used in classification.
- Wife\_working and Wife\_religion do not seem to have much effect on 'contraceptive\_method\_used' as well.

vi. Confusion matrix:

TRAIN DATA:

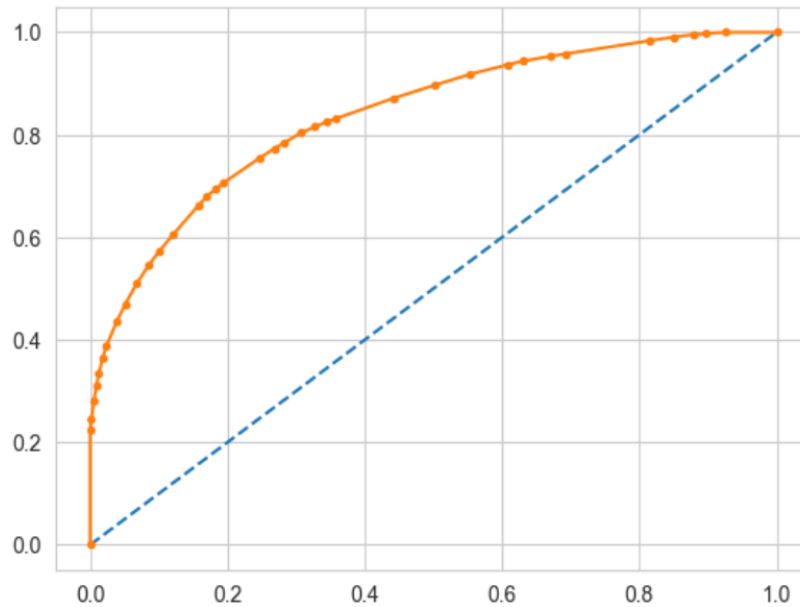


TEST DATA:



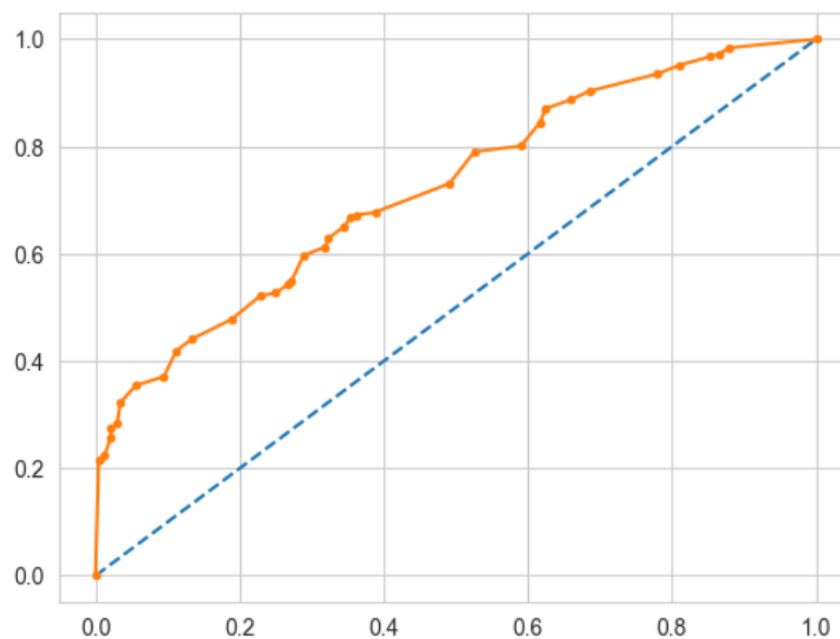
vii. ROC Curve and AUC Score for train set:

AUC: 0.838



ROC Curve and AUC Score for test set:

AUC: 0.722



## **MODEL COMPARISON AND INSIGHTS:**

- i. The model for LOGIT and LDA on training data has the same accuracy score of 0.0.6646. However, LOGIT performed better on test data with the accuracy score of 0.677 against that of CART, i.e., 0.665. One point to be noted is that there was a slight increase the accuracy score of LDA models from train to test set.
- ii. The CART train data set accuracy score was 0.764 but dropped to 0.65 for the test data set, which is the biggest drop of all three models.



- iii. The test data precision score for class 1 for both LOGIT and LDA is 0.71, while for CART it is 0.63.
- iv. The highest AUC score is given by the CART model, i.e., 0.722 (for test data).
- v. On a relative scale, the best of the three models is Logistic Regression model.
- vi. Logit was able to make 283 correct predictions, LDA made 278 correct predictions, while CART made 272 correct predictions. I would go with the Logistic Regression model to make predictions.

## 2.4 Inference: Basis on these predictions, what are the insights and recommendations.

Please explain and summarise the various steps performed in this project. There should be proper business interpretation and actionable insights present.

- i. The Logit model gave:
  - 198 True Positives
  - 101 False Positives
  - 85 True Negatives
  - 34 is False Negatives
- ii. X9, followed by X2 and X3 have the most impact on the dependent variable and hence they have the most power to distinguish between the two classes. X9 and X2 have an inverse effect on Y, while X3 has a direct impact on Y.
- iii. Steps involved in building the above models:
  - a. Step 1: Split the data into X\_train, X\_test, y\_train, y\_test.
  - b. Step 2: Create empty model and fit it into the training set.
  - c. Step 3: Get model coefficients to check the weight of each variable on the dependent variable.
  - d. Step 4: Check the intercept of the model.
  - e. Step 5: Check the AUC score and ROC curve along with the confusion matrix to see how well the discriminating features have predicted the Y variable.