PGP-DSBA

BUSINESS REPORT
PREDICTIVE MODELING

Dhruv Dosad

Contents

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.

Dataset for Problem 1: compactiv.xlsx

DATA DICTIONARY:

Column	Description
lread	Reads (transfers per second) between system memory and user memory
lwrite	writes (transfers per second) between system memory and user memory
scall	Number of system calls of all types per second
sread	Number of system read calls per second .
swrite	Number of system write calls per second .
fork	Number of system fork calls per second.
exec	Number of system exec calls per second.
rchar	Number of characters transferred per second by system read calls
wchar	Number of characters transfreed per second by system write calls
pgout	Number of page 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, 5 point summary). Perform Univariate, Bivariate Analysis, Multivariate Analysis.

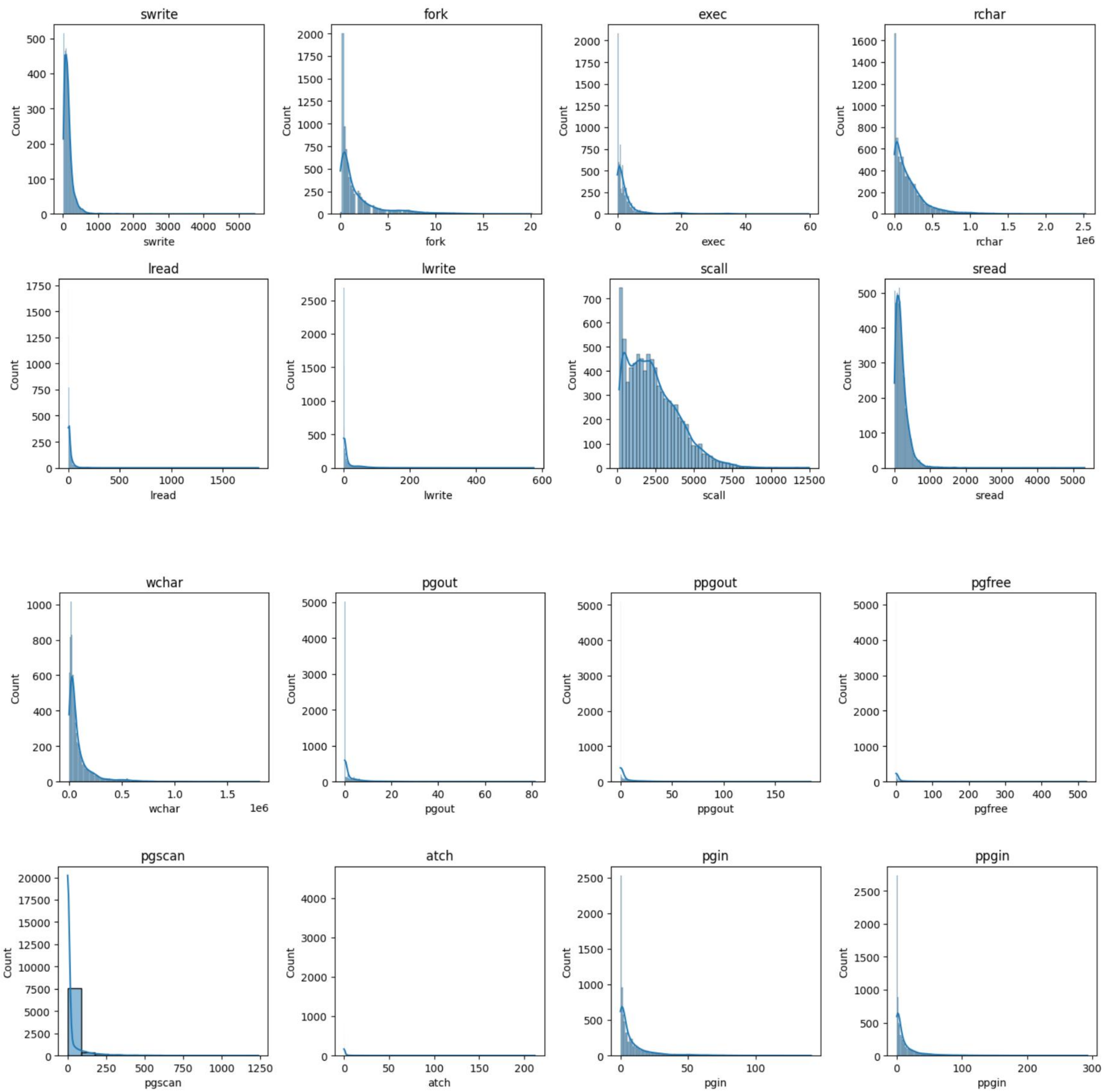
Performing the Exploratory Data Analysis (EDA)

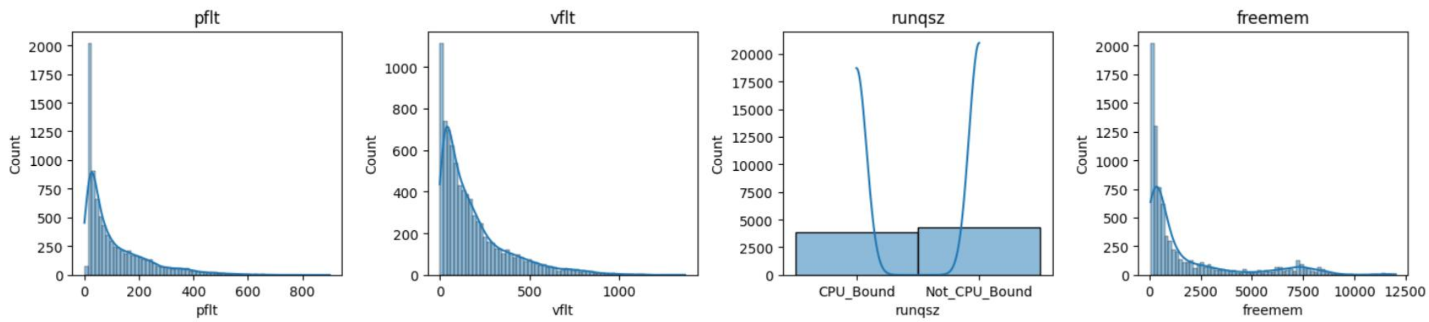
- Below are the key observations of the data set compactiv

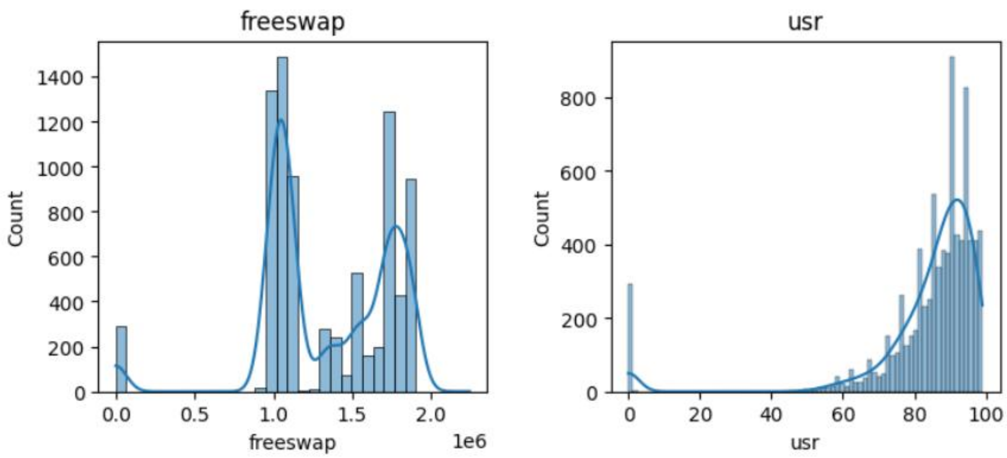
- The data has 8192 rows and 22 columns.
- There is 1 object type data types and rest are float & int data types
- First 5 values of the data set are as below:-

index	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem
0	1	0	2147	79	68	0.2	0.2	40671.0	53995.0	0.0	0.0	0.0	0.0	0.0	1.6	2.6	16.0	26.4	CPU_Bound	4670
1	0	0	170	18	21	0.2	0.2	448.0	8385.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	15.63	16.83	Not_CPU_Bound	7278
2	15	3	2162	159	119	2.0	2.4	NaN	31950.0	0.0	0.0	0.0	0.0	1.2	6.0	9.4	150.2	220.2	Not_CPU_Bound	702
3	0	0	160	12	16	0.2	0.2	NaN	8670.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	15.6	16.8	Not_CPU_Bound	7248
4	5	1	330	39	38	0.4	0.4	NaN	12185.0	0.0	0.0	0.0	0.0	0.0	1.0	1.2	37.8	47.6	Not_CPU_Bound	633

Univariate Analysis



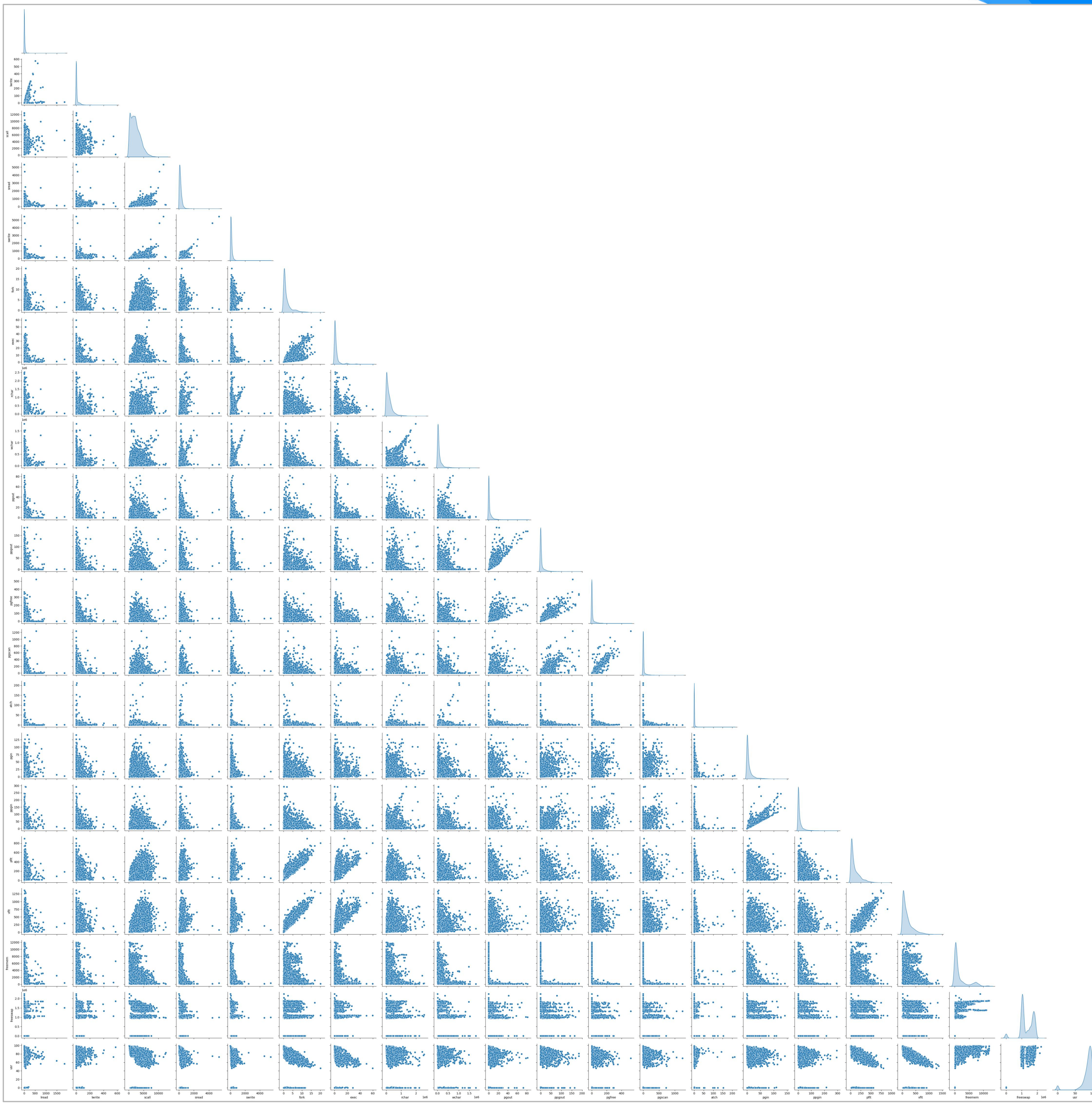


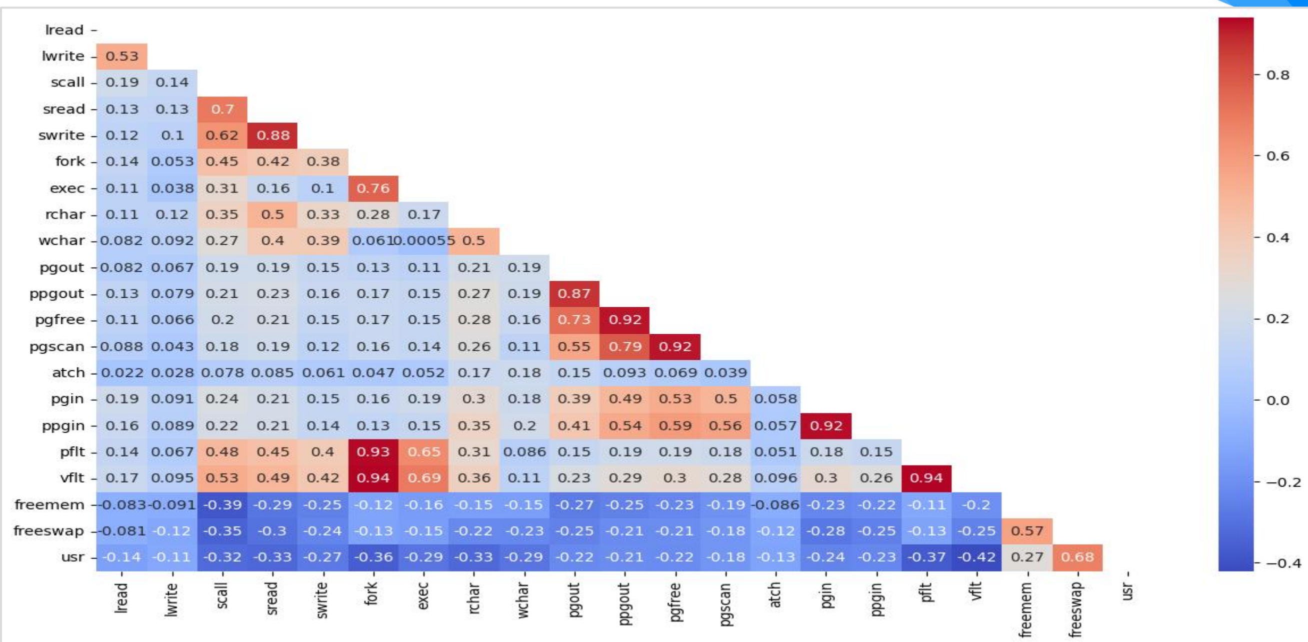


- The CPU runs in user mode 80% - 99% of the times or it stays idle
- The transfer for read and write is very quick.
- The System read-write rate is under 5% which means this is also quick.

Bivariate & Multivariate Analysis

- Both the page fault variables – pflt & vflt are highly correlated with the fork variable.
- Number of page out requests per second is also highly correlated to the number of pages, paged out per second variable.
- The same can be seen in heatmap below

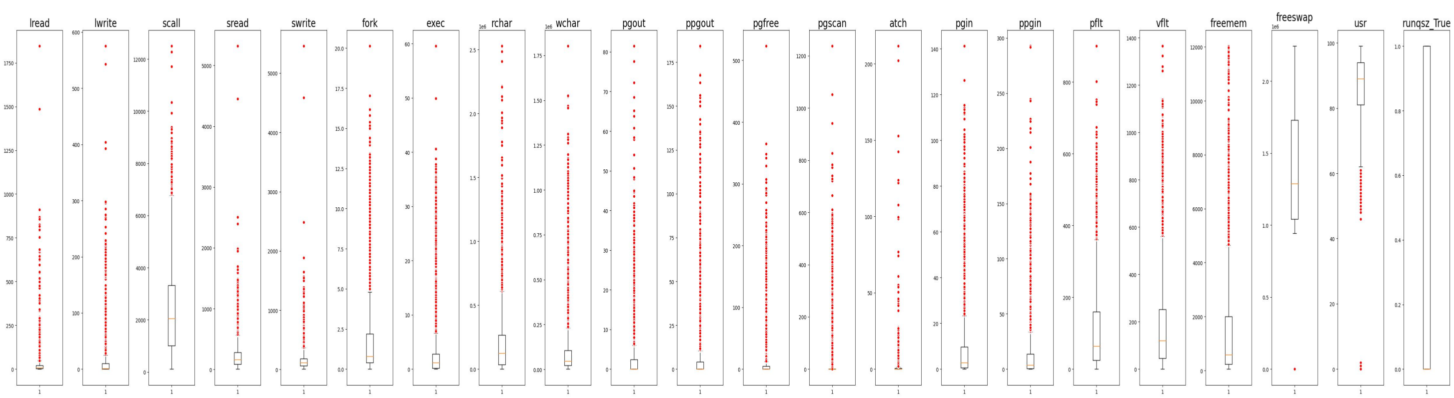

lread
lwrite
scall
sread
swrite
fork
exec
rchar
wchar
pgout
ppgout
pgfree
pgscan
atch
pgin
ppgin
pflt
vflt
freemem
freeswap
usr



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 are not necessarily required here as these do not have any signicant output due tp presence of 0s or inf.

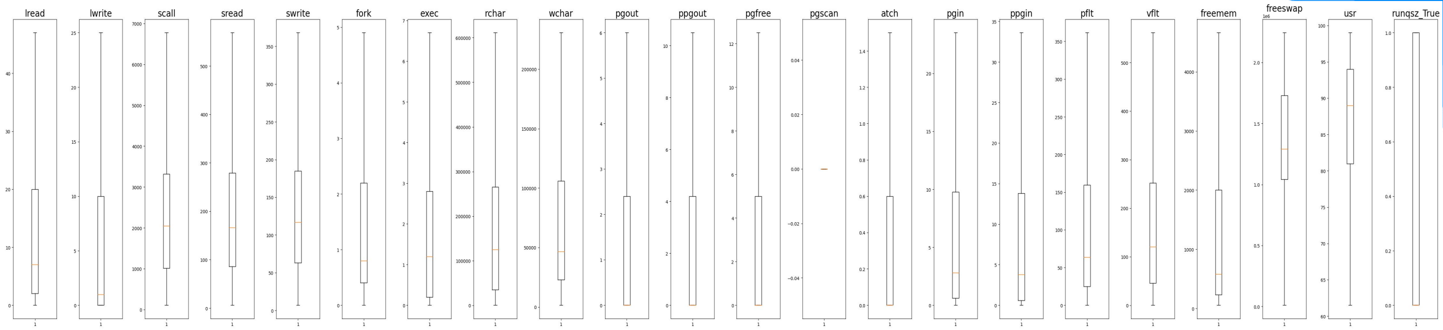
Presence of Outlier using the Boxplot.



- Form the boxplots, we obseve that there is presence of outliers in all the variables.
- Majority of the variables are highly skewed

Outliers after treatment

- We treated the outliers by adjusting them to the lower and upper values using the IQR.



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.

- **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 True and 0 as False and this is how the extended variable’s data looks

```
#Data After encoding
df.info()
```

```
<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   float64
 1   lwrite       8192 non-null   float64
 2   scall        8192 non-null   float64
 3   sread        8192 non-null   float64
 4   swrite       8192 non-null   float64
 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  freemem      8192 non-null   float64
 19  freeswap     8192 non-null   float64
 20  usr          8192 non-null   float64
 21  runqsz_True  8192 non-null   uint8  
dtypes: float64(21), uint8(1)
memory usage: 1.3 MB
```

Train – Test split & Model Building

- The data set is split into training and testing data in the ratio of 70:30.

```
# Split X and y into training and test set in 70:30 ratio
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30 , random_state=1)
```

- The Linear Regression model is built and fitted into the Training dataset.
- The coefficients of all the variables are calculated, and it clearly shows that features like ‘runqsz_CPU_Bound’,’pgout’ will directly impact the value of the target variable if all the other variables are 0.

- Similarly, is the case for the variables with negative coefficients.

The coefficient for lread is -0.06348150618196245

The coefficient for lwrite is 0.04816128709127112

The coefficient for scall is -0.0006638280111675074

The coefficient for sread is 0.00030825210315167515

The coefficient for swrite is -0.005421822297643799

The coefficient for fork is 0.029312727249365546

The coefficient for exec is -0.32116648389885805

The coefficient for rchar is -5.1668417594745746e-06

The coefficient for wchar is -5.402875235427529e-06

The coefficient for pgout is -0.36881906387335767

The coefficient for ppgout is -0.07659768212738409

The coefficient for pgfree is 0.08448414470559423

The coefficient for pgscan is -4.440892098500626e-16

The coefficient for atch is 0.6275741574813001

The coefficient for pgin is 0.01998790767863925

The coefficient for ppgin is -0.06733383975701812

The coefficient for pflt is -0.033602829377515235

The coefficient for vflt is -0.005463668798519861

The coefficient for freemem is -0.00045846718794751725

The coefficient for freeswap is 8.831840263033575e-06

The coefficient for runqsz_True is -1.6152978488249097

Model Performance

Sklearn method:-

- To check the model's performance, we calculate the Rsquare values or the Coefficient of Determinants for both Train and test data

Rsquare and RMSE for Training data.

- **Rsquare for Train data: 0.796108610127457**
- **RMSE for Train data: 4.419536092979902**
- This is a good value. This shows that almost 72% of the variance of the training dataset was captured by the model.

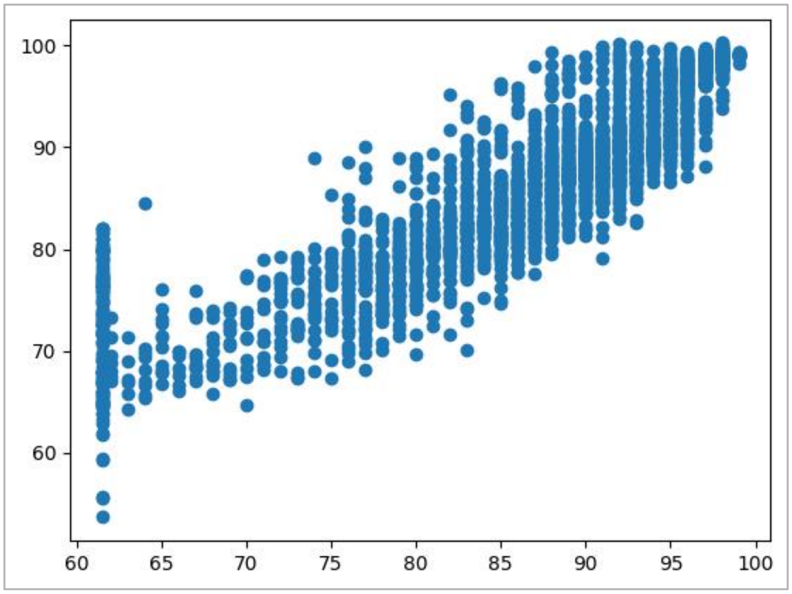
Rsquare and RMSE for test data.

- **Rsquare for Test data: 0.7677318597936156**
- **RMSE for Test data: 4.652295704192616**
- This is also a good value. This shows that almost 70% of the variance of the testing dataset was captured by the model.
- The model seems to be neither overfitting nor under-fitting, therefore this is a good model to go with.

Statsmodel method:-

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.796
Model:                            OLS   Adj. R-squared:                  0.795
Method:                 Least Squares   F-statistic:                     1115.
Date:                Sat, 29 Apr 2023   Prob (F-statistic):               0.00
Time:                        11:07:13   Log-Likelihood:                -16657.
No. Observations:                5734   AIC:                         3.336e+04
Df Residuals:                    5713   BIC:                         3.350e+04
Df Model:                          20
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
const          85.7370      0.296    289.444      0.000      85.156      86.318
lread          -0.0635      0.009     -7.071      0.000      -0.081      -0.046
lwrite          0.0482      0.013      3.671      0.000       0.022       0.074
scall          -0.0007   6.28e-05    -10.566      0.000      -0.001      -0.001
sread           0.0003      0.001      0.305      0.760      -0.002       0.002
swrite         -0.0054      0.001     -3.777      0.000      -0.008      -0.003
fork            0.0293      0.132      0.222      0.824      -0.229       0.288
exec           -0.3212      0.052     -6.220      0.000      -0.422      -0.220
rchar       -5.167e-06   4.88e-07    -10.598      0.000   -6.12e-06   -4.21e-06
wchar       -5.403e-06   1.03e-06     -5.232      0.000   -7.43e-06   -3.38e-06
pgout          -0.3688      0.090     -4.098      0.000      -0.545      -0.192
ppgout         -0.0766      0.079     -0.973      0.330      -0.231       0.078
pgfree          0.0845      0.048      1.769      0.077      -0.009       0.178
pgscan       1.568e-16    5.6e-17      2.800      0.005    4.71e-17    2.67e-16
atch            0.6276      0.143      4.394      0.000       0.348       0.908
pgin            0.0200      0.028      0.703      0.482      -0.036       0.076
ppgin          -0.0673      0.020     -3.415      0.001      -0.106      -0.029
pflt           -0.0336      0.002    -16.957      0.000      -0.037      -0.030
vflt           -0.0055      0.001     -3.830      0.000      -0.008      -0.003
freemem        -0.0005   5.07e-05     -9.038      0.000      -0.001      -0.000
freeswap     8.832e-06    1.9e-07     46.472      0.000    8.46e-06     9.2e-06
runqsz_True    -1.6153      0.126    -12.819      0.000      -1.862      -1.368
==============================================================================
Omnibus:                     1103.645   Durbin-Watson:                   2.016
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2372.553
Skew:                          -1.119   Prob(JB):                         0.00
Kurtosis:                       5.219   Cond. No.                     4.61e+22
==============================================================================
```

Predicted y values vs the actual y values for the test dataset



- From the above scatterplot, we can see that the actual and the predicted values are close enough, except for a few. This shows that the model performed good as per the data

Linear Regression equation from the final model

usr = (85.74) * const + (-0.06) * lread + (0.05) * lwrite + (-0.0) * scall + (0.0) * sread + (-0.01) * swrite + (0.03) * fork + (-0.32) * exec + (-0.0) * rchar + (-0.0) * wchar + (-0.37) * pgout + (-0.08) * ppgout + (0.08) * pgfree + (0.0) * pgscan + (0.63) * atch + (0.02) * pgin + (-0.07) * ppgin + (-0.03) * pflt + (-0.01) * vflt + (-0.0) * freemem + (0.0) * freeswap + (-1.62) * runqsz_True +

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.

The following are the observations for the above model:

1. **CPUs** have **two** operating modes, **kernel mode (system mode) and user mode**, developed to prevent applications from constantly crashing computers. Kernel mode processes have full access to the hardware.
2. When **U_Bound increases by 1 unit**, **usr increases by 0.234 units**, holding all other predictors constant. The **usr value** is **influenced by both positive and negative coefficients**. Positive coefficients lead to an increase in usr, while negative coefficients lead to a decrease.
3. The **most impactful variable** on 'usr' is **'runqsz_Not_CPU_Bound'**. Factors such as **pflt** (page faults due to protection errors) and **scall** (system calls per second) lead to a **decrease in time spent in user mode**, while an **increase in 'freemem'** (available memory pages) **leads to an increase** in **time spent** in user mode.
4. Columns such as 'pflt_square', 'freeswap', 'wchar', 'rchar', and 'freeswap_square' have **minimal impact** on usr. As these values increase, the time spent in user mode decreases.

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.

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=verlow, 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.

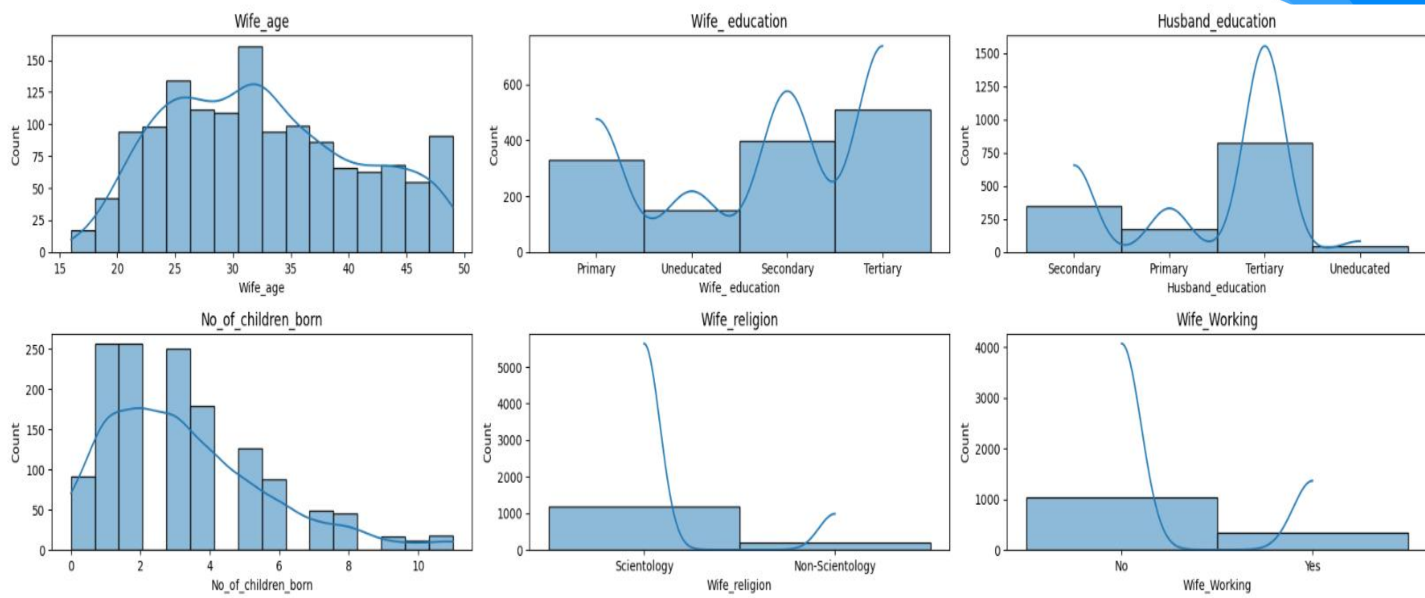
Performing the Exploratory Data Analysis (EDA)

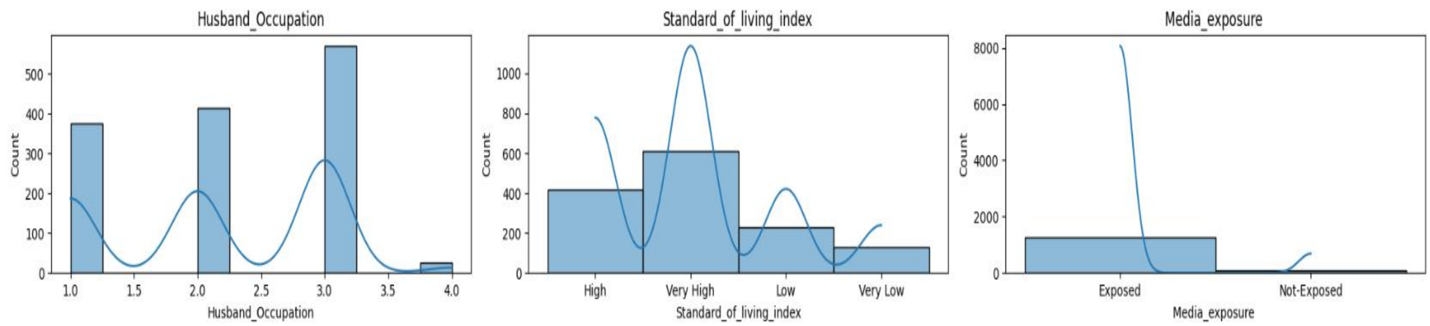
- Below are the key observations of the data set along with Univariate, Bivariate & Multivariate analysis

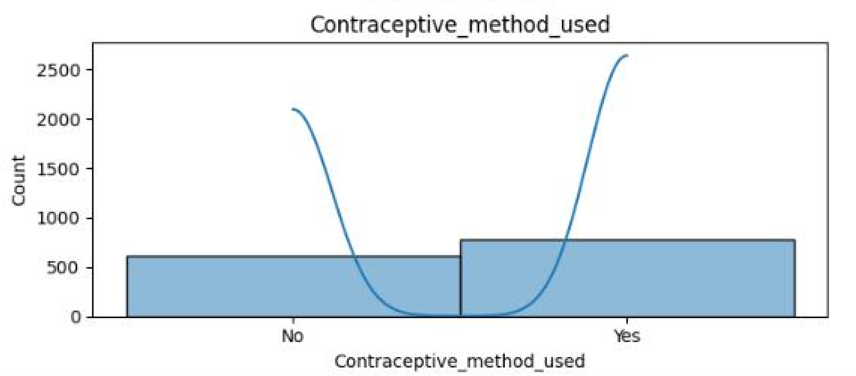
- The data has **1473** rows and **10** columns.
- There are 7 object type data types, 1 Int & 2 float data types
- First 5 values of the data set are as below:-

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

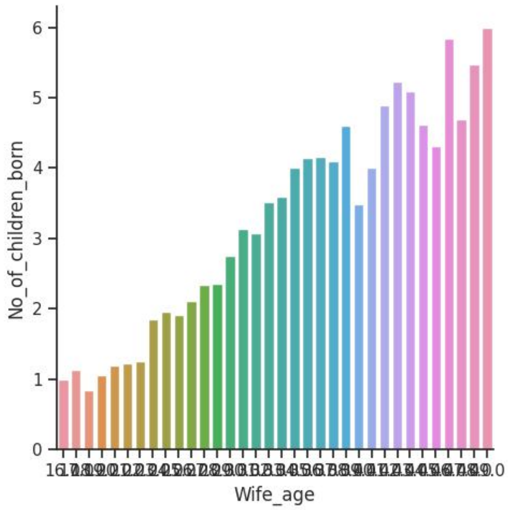
- **Missing** values in **'Wife_age**' and **'No_of_children_born'** treated using median imputation.
- **85 duplicate** rows present in the dataset.
- **Majority of women follow Scientology** and **are not working**.
- **Tertiary education** is the **most common** level for **both husbands and wives.**
- **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.

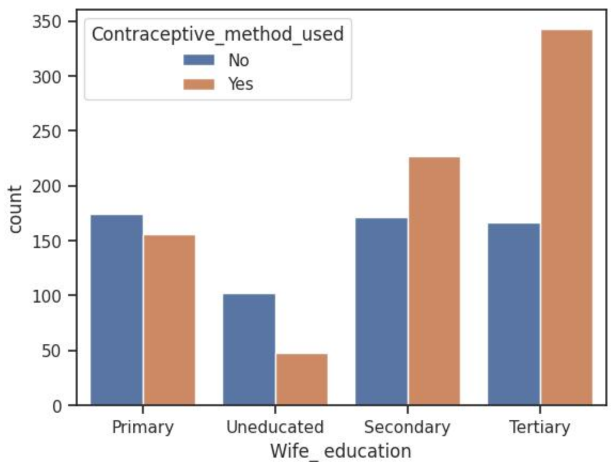


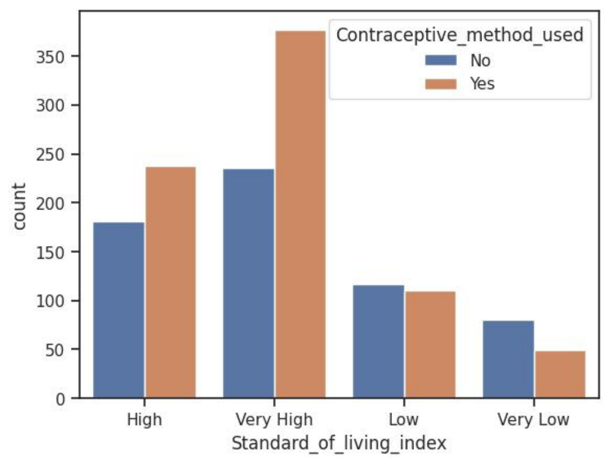




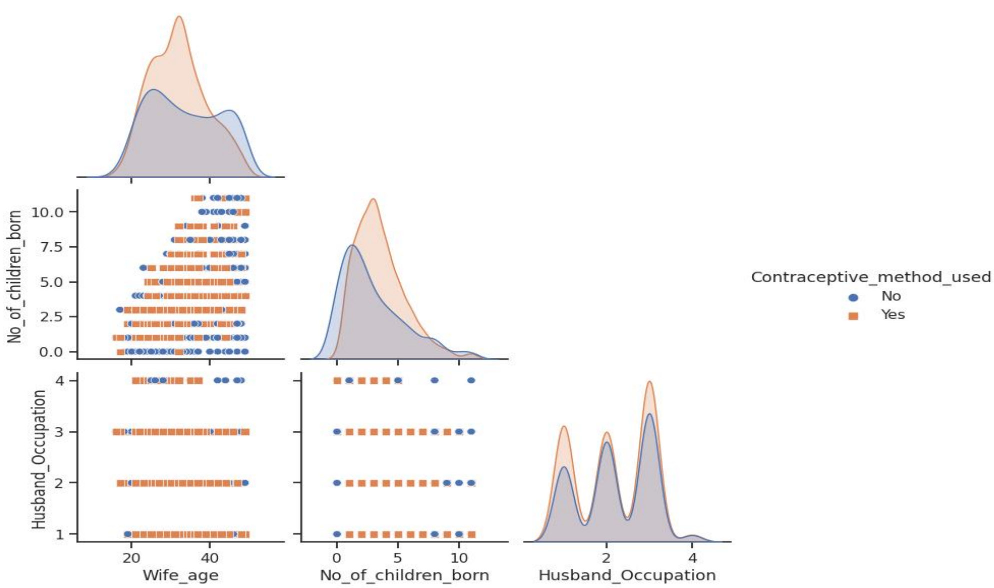
Bivariate Analysis

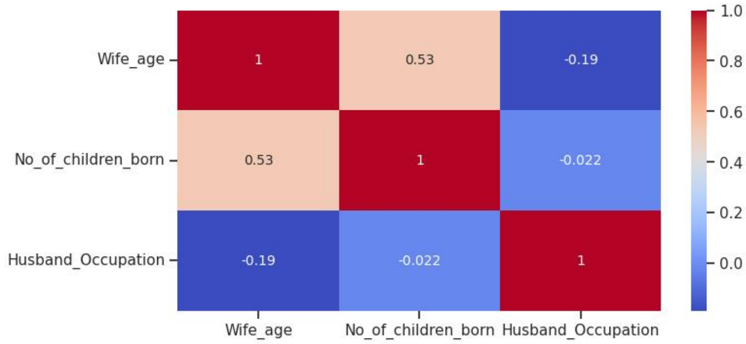






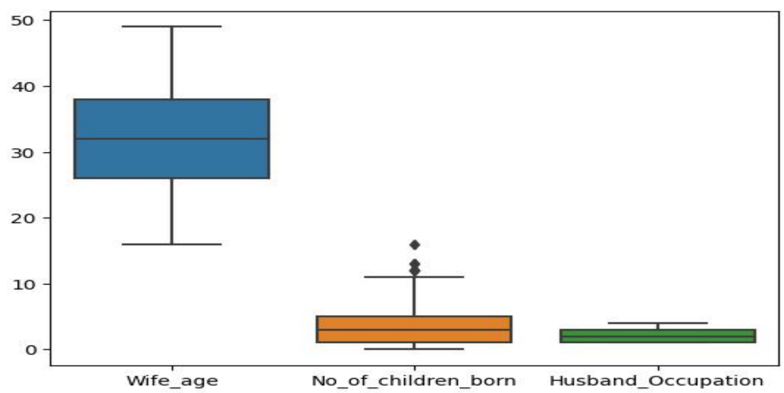
Multivariate Analysis





- 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

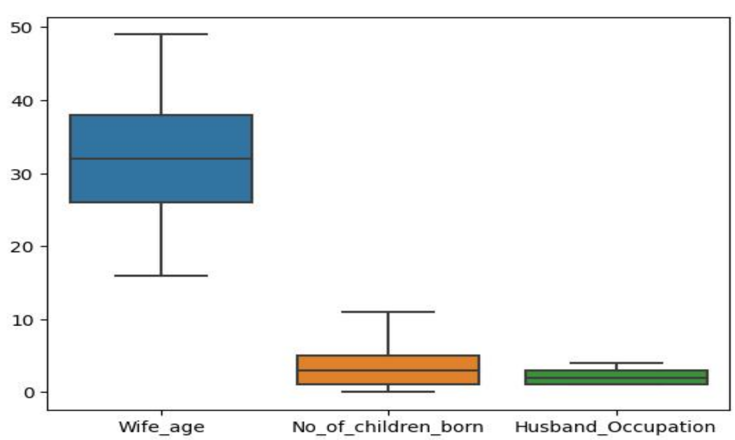
Presence of Outlier using the Boxplot.



- Form the boxplots, we observe that there is presence of outliers in 'No_of_children_born' variable.

Outliers after treatment

- We treated the outliers by adjusting them to the lower and upper values using the IQR.



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.

Encoding the data:

- The data has string & categorical type variables, we will need to encode them.
- "No" & "Yes" in the target variable is replaced by 0 and 1 respectively.
- Ordinal numbers are given to the values in variables Wife_ education, Husband_education & Standard_of_living_index
- After this dummy encoding is used to encode the data for the rest of the columns. The dataset looks like below:

	Wife_age	No_of_children_born	Contraceptive_method_used	Wife_education_1	Wife_education_2	Wife_education_3	Husband_education_1	Husband_education_2	Husband_education_3	Wife_religion_1	Wife_
0	24.0	3.0	0	0	0	0	1	0	0	1	
1	45.0	10.0	0	0	0	1	1	0	0	1	
2	43.0	7.0	0	0	0	0	1	0	0	1	
3	42.0	9.0	0	1	0	0	0	0	0	1	
4	36.0	8.0	0	1	0	0	1	0	0	1	

Train-Test split

- We will split the entire data set into a ratio of 70:30 into Training dataset and Testing dataset

```
# Split X and y into training and test set in 70:30 ratio
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.30 , random_state=1)
```

- Making 3 models using Decision Tree Classifier , Logistic Regression and LDA and comparing the Accuracy to find the best model

Train and Test Accuracy details to see that there is no huge Over/Under fitting

```
[35] dtc = DecisionTreeClassifier()
     lda= LinearDiscriminantAnalysis()
     lor= LogisticRegression()


     models=[dtc,lda,lor]

     accuracy_train=[]
     accuracy_test=[]


     for i in models:  # Computation of RMSE and R2 values
         i.fit(X_train,y_train)
         accuracy_train.append(accuracy_score(y_train,i.predict(X_train)))
         accuracy_test.append(accuracy_score(y_test,i.predict(X_test)))

     print(pd.DataFrame({'Train Accuracy': accuracy_train,'Test Accuracy': accuracy_test},
                 index=['Decision Tree Classifier','LDA','Logistic Regression']))

                               Train Accuracy  Test Accuracy
     Decision Tree Classifier        0.983522       0.597122
     LDA                             0.684861       0.630695
     Logistic Regression             0.682801       0.630695
```

Looks like Decision Tree Classifier, is under-fitting because train accuracy > test accuracy ., Let's Grid Search to get the best parameters or prune the tree

```
dtc = DecisionTreeClassifier(criterion='gini', max_depth=20, min_samples_leaf=3, min_samples_split=30)
#Using best parameters in above
lda= LinearDiscriminantAnalysis()
lor= LogisticRegression()


models=[dtc,lda,lor]

accuracy_train=[]
accuracy_test=[]


for i in models:  # Computation of RMSE and R2 values
    i.fit(X_train,y_train)
    accuracy_train.append(accuracy_score(y_train,i.predict(X_train)))
    accuracy_test.append(accuracy_score(y_test,i.predict(X_test)))

print(pd.DataFrame({'Train Accuracy': accuracy_train,'Test Accuracy': accuracy_test},
            index=['Decision Tree Classifier','LDA','Logistic Regression']))
```

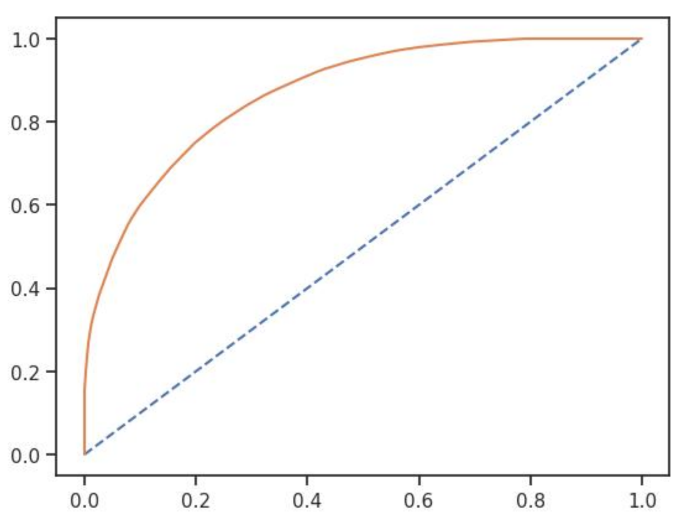
```
                          Train Accuracy  Test Accuracy
Decision Tree Classifier        0.782698       0.649880
LDA                             0.684861       0.630695
Logistic Regression             0.682801       0.630695
```

Clearly now the underfitting of the model in Decison tree is reduced and Decison tree classifier results in the best accuracy score thus this model will be selected for classification

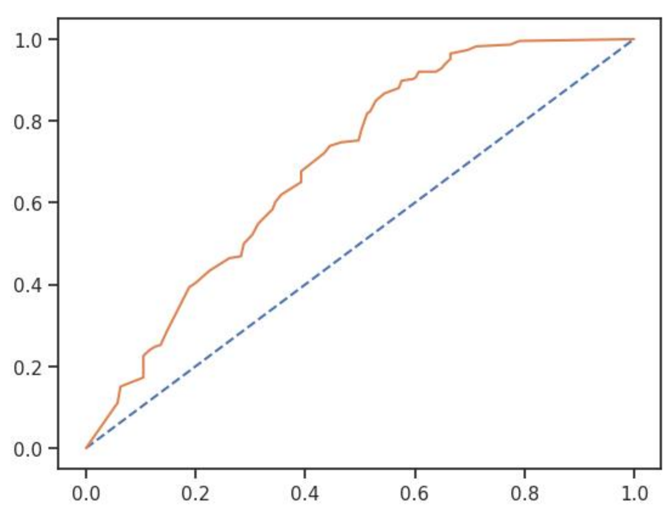
2.3 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model Final Model: Compare Both the models and write inference which model is best/optimized.

<u>ROC AUC Curve values for best model indicates that there is high level of seperatibility among the classes of the target variable</u>

AUC Curve for Train set



ROC Curve for Test set



Train Classification report

	precision	recall	f1-score	support
0	0.77	0.71	0.74	423
1	0.79	0.84	0.81	548
accuracy			0.78	971
macro avg	0.78	0.77	0.78	971
weighted avg	0.78	0.78	0.78	971

Test Classification report

	precision	recall	f1-score	support
0	0.66	0.50	0.57	191
1	0.65	0.78	0.71	226
accuracy			0.65	417
macro avg	0.65	0.64	0.64	417
weighted avg	0.65	0.65	0.64	417

Overall accuracy of the model – 65 % of total predictions are correct

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

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

Inferences:

- **Wife's education** and the **number of children born** significantly **influence the use of contraceptive** methods, as indicated by both the Logistic Regression and CART models.
- **Husband's education** also **plays an important role** in **determining the use of contraceptives**, as it influences the wife's decision-making process.

Recommendations:

- **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.