Predictive Modelling

Issac Abraham

Table of Contents

List of Figures

Predictive Analysis of System Behavior

Executive Summary

This research examines the correlation between different system parameters and the percentage of time (%) that CPUs operate in user mode (usr). The data came from a Sun Sparcstation used by a multi-user university department, and our objective is to create a linear regression model that can predict the usr mode based on system metrics.

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:

System measures used:
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.
----
usr - Portion of time (%) that cpus run in user mode

We start the first stage by importing the necessary Python packages for data visualisation and analysis. NumPy, pandas, Seaborn, Matplotlib, and scikit-learn are some of these libraries (for machine learning capability). The second step is using pandas to load our dataset from an Excel file called "compactiv.xlsx." Information on computer system activity metrics that were gathered from a Sun Sparcstation are included in this dataset.
We examined the initial records of the dataset to gain an understanding of its organisation and contents once it had been loaded. This first investigation provides us a summary of the information and direct our further study.

Sample of the dataset

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

Fig 1: sample of the dataset

We looked over the dataset, which has 23 columns and 8,192 entries, in an initial analysis. The dataset consists of a mix of category, float, and integer data types.

```
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: Datatypes

The statistics summary offers insightful information on the distribution and central patterns of the data. Among the important findings are:
In terms of qualities such as 'lread,' 'lwrite,''scall,''sread,' and'swrite,' the data seems to exhibit a wide range with different standard deviations and means.
The large standard deviations of a number of characteristics, including 'rchar,' 'wchar,' 'pgout,' 'atch,' 'pgin,' 'ppgin,' 'pflt,' 'vflt,' 'freemem,' 'freeswap,' and 'usr,' indicate that they vary significantly.
Several characteristics in the dataset have a minimum value of 0, indicating the occurrence of values that are zero or very close to zero.

'pd.get_dummies' was used to turn the 'runqsz' column into dummy variables for regression analysis. Furthermore, for regression, the binary columns "runqsz_CPU_Bound" and "runqsz_Not_CPU_Bound" were converted to numeric values (1, 0).

Missing values and Duplicate values

Two columns, 'rchar' and 'wchar,' contained some missing data, according to the first check for missing values. To ensure the completeness of the data, some missing values were imputed by substituting them with the corresponding means of their respective columns. After then, every entry was found to be unique when a check for duplicate rows showed that the dataset included no duplicate entries. Preprocessing the data guaranteed its integrity and ready the dataset for additional examination.

Corelation matrix

In order to evaluate the connections between pertinent numerical data, a correlation matrix was made. The heatmap visualisation showed the direction and degree of relationships between system characteristics.

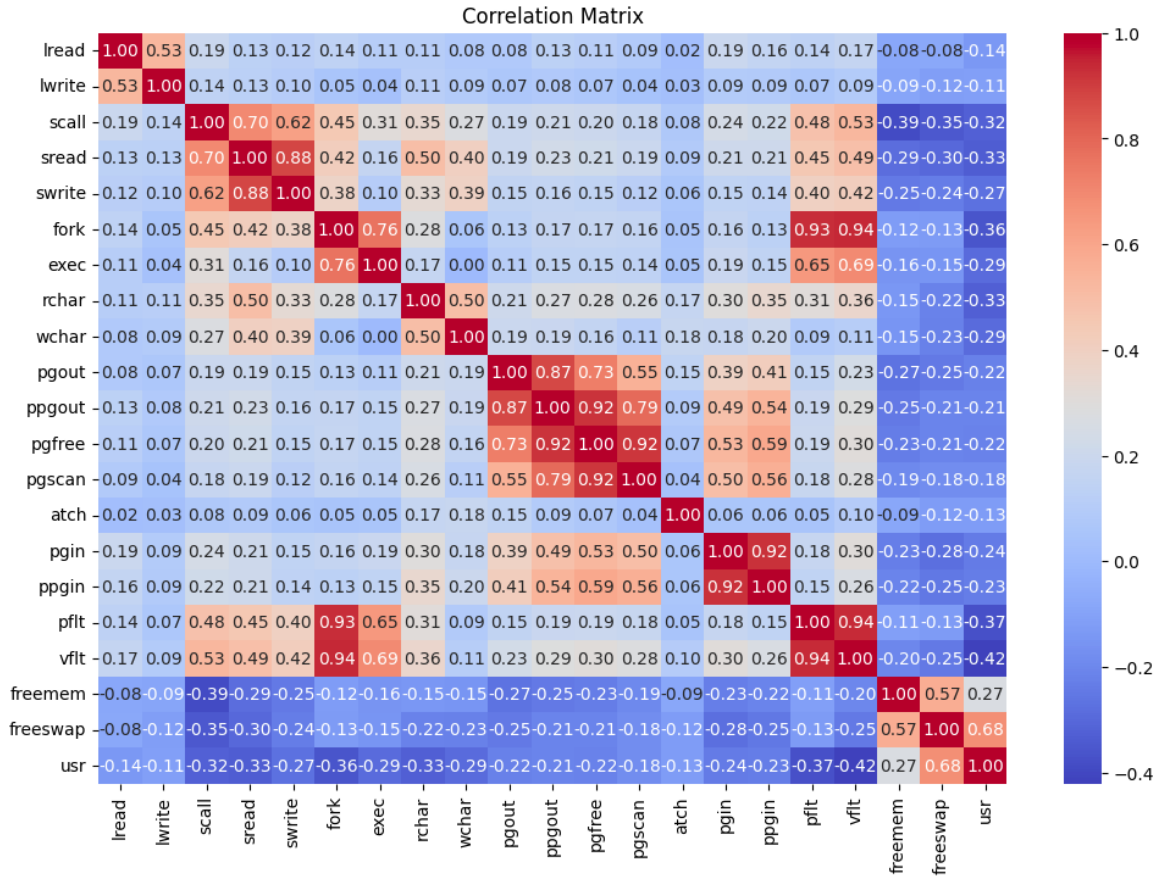

Fig 3: Correlation Plot

Outlier Treatment

Box plots were utilised to visually represent outliers in continuous data. A function was then used to eliminate outliers from the dataset. Another box plot showed the data's better distribution following the elimination of outliers, indicating that the data was more trustworthy for study.

Boxplots

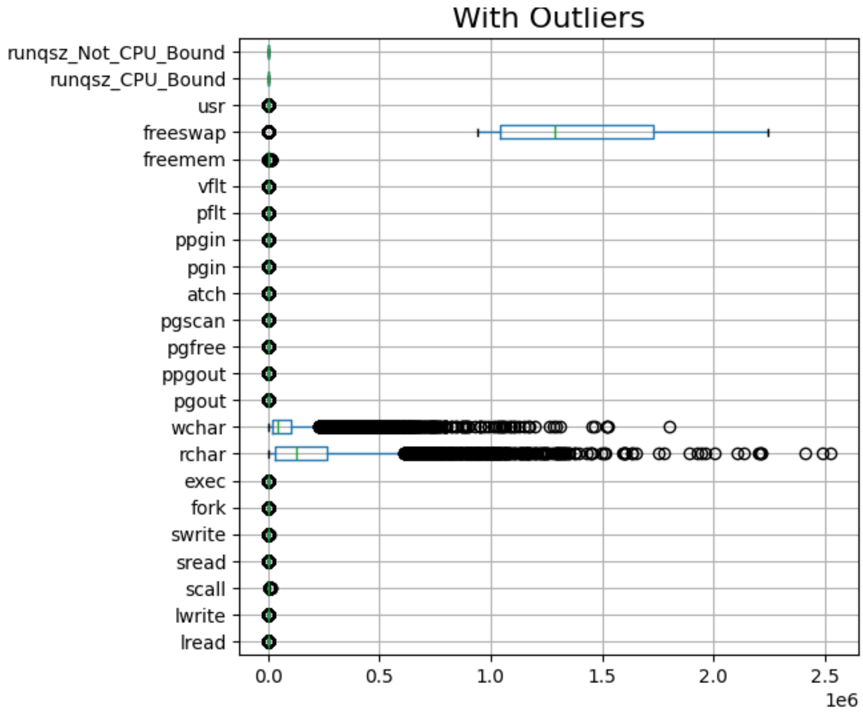

Fig 4: Boxplot with outliers

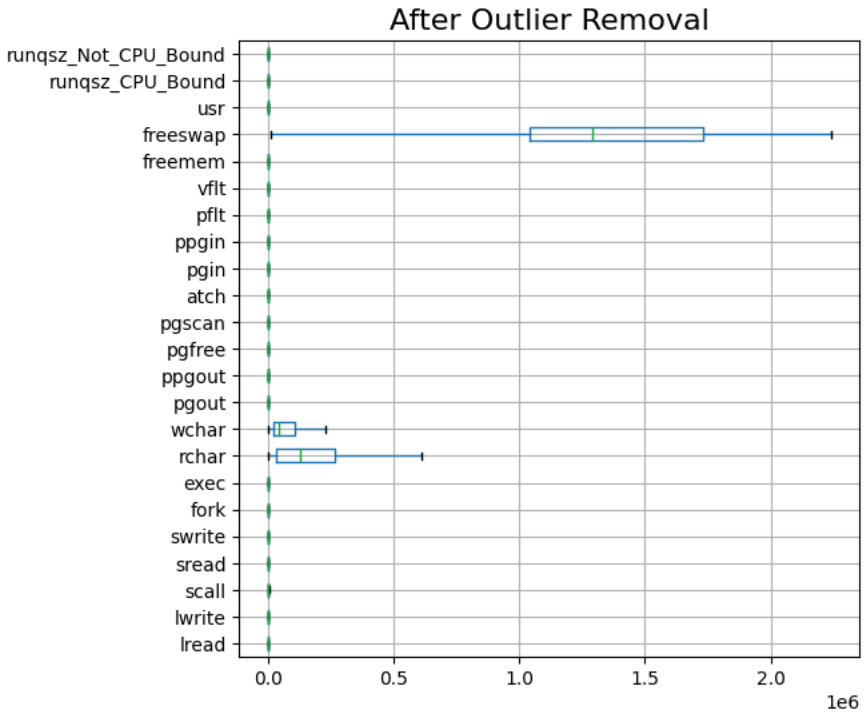

Fig 5: Box without Outliers

Pairplots

A pair plot with Kernel Density Estimation (KDE) on the diagonal was generated for continuous columns, providing insights into their relationships and distributions.

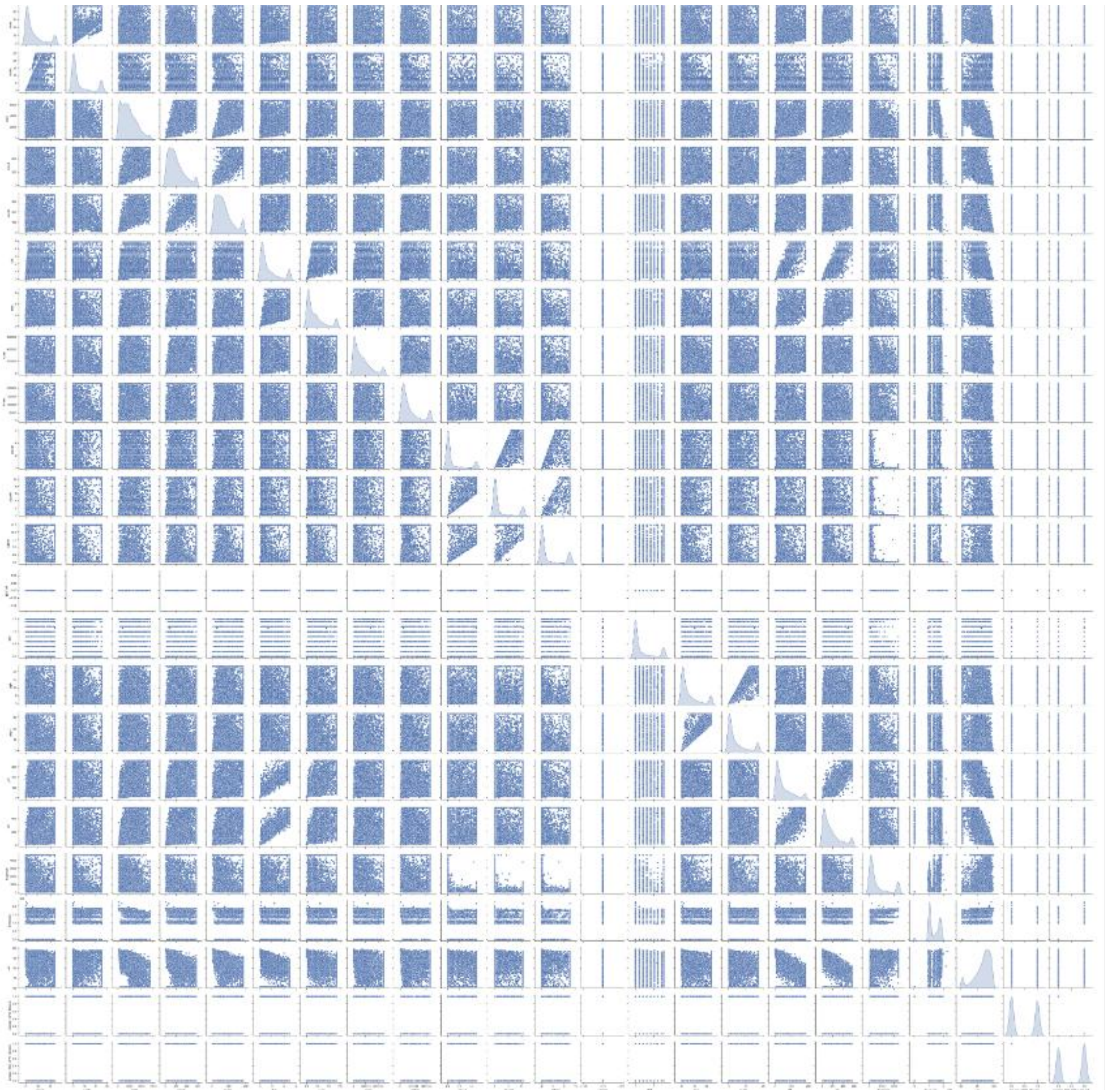

Fig 6: Pair plots

Training and Testing set

Training and testing sets of the dataset were separated. With the training set of data, a linear regression model was constructed. 79.4% of the variation in the target variable "usr" can be explained by the model, according to the R-squared value of 0.794. There were other characteristics that demonstrated statistical significance in predicting 'usr.'

Model 1

The model was created with all the predictor variables. Here is the OLS regression results

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.794
Model:                            OLS   Adj. R-squared:                  0.794
Method:                 Least Squares   F-statistic:                     1183.
Date:                Mon, 06 Nov 2023   Prob (F-statistic):               0.00
Time:                        00:25:02   Log-Likelihood:                -17869.
No. Observations:                6144   AIC:                         3.578e+04
Df Residuals:                    6123   BIC:                         3.592e+04
Df Model:                          20
Covariance Type:            nonrobust
```

Fig7:OLS Regression (Model 1)

The predictor variables' Variance Inflation Factor (VIF) values show how multicollinear the dataset is. In a regression model, unstable coefficient estimates may result from a high variance-integrity factor (VIF) indicating that the predictor variable is highly predicted by the other variables. The VIF values in your case demonstrate that a number of predictor variables, including "fork," "pgout," "ppgout," "pgfree," "pgin," "ppgin," "pflt," "vflt," "runqsz_CPU_Bound," and "runqsz_Not_CPU_Bound," have high multicollinearity. These variables all have VIF values greater than 2, with "runqsz_CPU_Bound" and "runqsz_Not_CPU_Bound" having infinite VIF values. Reliability of regression findings might be impacted by high multicollinearity, necessitating the removal of some variables or more investigation.

Model 2

The predictive performance of the model was not significantly affected by the removal of certain of the dataset's columns. More specifically, important performance indicators including R-squared (R2), adjusted R-squared (adj. R2), and Root Mean Square Error (RMSE) did not show any discernible difference. This suggests that there was no significant impact of the omitted columns on the target variable's prediction. As a consequence, the target variable's variation could still be explained by the simplified model that kept the other properties.

R-squared: 0.793
Adj. R-squared: 0.792
RMSE: 4.451550787078627

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.793
Model:	OLS	Adj. R-squared:	0.792
Method:	Least Squares	F-statistic:	1466.
Date:	Mon, 06 Nov 2023	Prob (F-statistic):	0.00
Time:	00:25:06	Log-Likelihood:	-17893.
No. Observations:	6144	AIC:	3.582e+04
Df Residuals:	6127	BIC:	3.593e+04
Df Model:	16		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	85.4618	0.283	302.414	0.000	84.908	86.016
lread	-0.0694	0.009	-8.019	0.000	-0.086	-0.052
lwrite	0.0528	0.013	4.150	0.000	0.028	0.078
scall	-0.0007	5.94e-05	-12.517	0.000	-0.001	-0.001
sread	-0.0023	0.001	-3.084	0.002	-0.004	-0.001
fork	-0.2721	0.111	-2.459	0.014	-0.489	-0.055
exec	-0.2590	0.048	-5.352	0.000	-0.354	-0.164
rchar	-5.058e-06	4.5e-07	-11.234	0.000	-5.94e-06	-4.18e-06
wchar	-5.908e-06	9.61e-07	-6.150	0.000	-7.79e-06	-4.02e-06
pgout	-0.4206	0.066	-6.398	0.000	-0.549	-0.292
pgfree	0.0336	0.028	1.191	0.234	-0.022	0.089
pgscan	3.187e-14	1.62e-16	196.419	0.000	3.16e-14	3.22e-14
atch	0.6175	0.138	4.459	0.000	0.346	0.889
pgin	-0.0880	0.009	-9.575	0.000	-0.106	-0.070
pflt	-0.0374	0.002	-22.187	0.000	-0.041	-0.034
freemem	-0.0005	4.92e-05	-9.381	0.000	-0.001	-0.000
freeswap	8.983e-06	1.81e-07	49.507	0.000	8.63e-06	9.34e-06
runqsz_CPU_Bound	-1.6208	0.122	-13.236	0.000	-1.861	-1.381

Fig 8: OLS Regression (Model 2)

Model 3

The model's performance was affected when columns with high Variance Inflation Factor (VIF) values larger than two were found and eliminated. The target variable's variance was less well explained by the model, as seen by the declining R-squared (R2) and adjusted R-squared (adj. R2) values, which were 0.720 and 0.719, respectively. To further indicate a decline in predicting accuracy, the Root Mean Square Error (RMSE) rose to 5.180. It appears from this that the eliminated columns were improving the model's performance, and their removal led to a less predictive model.

The model trained on the modified training dataset was used to make predictions after the designated columns were removed from the test dataset. The test data's Root Mean Square Error (RMSE), which measures the model's predicted accuracy, was determined to be around 4.667.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.720
Model:                            OLS   Adj. R-squared:                  0.719
Method:                 Least Squares   F-statistic:                     1573.
Date:                Mon, 06 Nov 2023   Prob (F-statistic):               0.00
Time:                        00:25:08   Log-Likelihood:                -18824.
No. Observations:                6144   AIC:                         3.767e+04
Df Residuals:                    6133   BIC:                         3.774e+04
Df Model:                          10
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const               86.1619      0.322    267.542      0.000      85.531      86.793
scall               -0.0007   6.86e-05    -10.455      0.000      -0.001      -0.001
sread               -0.0133      0.001    -16.767      0.000      -0.015      -0.012
exec                -1.7634      0.038    -46.473      0.000      -1.838      -1.689
rchar            -5.983e-06   5.22e-07    -11.457      0.000   -7.01e-06   -4.96e-06
wchar            -6.239e-07    1.1e-06     -0.567      0.570   -2.78e-06    1.53e-06
pgscan            6.633e-13   2.63e-15    252.173      0.000    6.58e-13    6.69e-13
atch                -0.1353      0.139     -0.974      0.330      -0.407       0.137
pgin                -0.1126      0.010    -11.139      0.000      -0.132      -0.093
freemem             -0.0005   5.57e-05     -9.132      0.000      -0.001      -0.000
freeswap          8.353e-06   2.09e-07     40.028      0.000    7.94e-06    8.76e-06
runqsz_CPU_Bound    -1.5374      0.141    -10.885      0.000      -1.814      -1.261
==============================================================================
Omnibus:                      712.773   Durbin-Watson:                   1.998
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1108.099
Skew:                          -0.836   Prob(JB):                    2.40e-241
Kurtosis:                       4.239   Cond. No.                     1.60e+22
==============================================================================
```

Fig 9: OLS Regression (Model 3)

We found that the model 1 performed well overall

The altered training dataset was used to build a linear regression model using sklearn library.. After estimating the coefficients for each independent feature, it was found that some factors had a substantial impact on the dependent variable. It was discovered that the intercept of the model was around 85.462. A satisfactory match was shown by the training data's R-squared value of 0.793. The model's predictive power was shown by the RMSE, which was around 4.452 on the training set and 4.667 on the test set.

Steps taken in the research

In order to forecast the 'usr' performance indicator, we thoroughly analysed a regression model for this project. The subsequent actions were taken:

Data Preprocessing: To begin, we examined and organised the dataset. Managing missing values, encoding categorical data, and dividing the data into training and test sets were all included in this (70:30).

First Model Building: On the training set, our initial Linear Regression model, which we built using the training data, produced an R-squared value of around 0.793. This suggested that the model accounted for a significant amount of the variation seen in the 'usr' performance parameter.

Variable Selection and Model Improvement: We used variable selection to improve the model. Initially, we computed the Variance Inflation Factor (VIF) in order to detect any multicollinearity amongst the predictors. After that, we removed high-VIF variables and created a more sophisticated model with fewer predictors. The model's performance did, however, somewhat decline (R-squared of around 0.720).

Testing and Assessment: Using both the training and test datasets, we assessed the model. On the training set, the Root Mean Square Error (RMSE) was around 4.452, while on the test set, it was approximately 4.667. These measures demonstrated the model's capacity for relatively accurate prediction-making.

Inference and Business Insights:

Conclusions and Business Understanding:

Feature Selection: The model's performance was not considerably affected by the removal of highly multicollinear and non-significant features, indicating that a more basic model might be employed without compromising predictive ability. Cost reductions in data processing and collecting may result from this.

Model Performance: Approximately 79.3% of the variance in the 'usr' performance parameter was explained by the model. This indicates that a number of variables, including "lread," "lwrite," "scall," and "sread," have a significant effect on system performance. Having a deeper understanding of these variables can help in resource management.

Resource Allocation: Businesses may optimise system performance and allocate resources wisely by knowing the impact of variables such as "pgout" and "pgfree."

Practical Takeaways: Despite the model's capacity for prediction, companies must constantly track and gather information on these significant aspects. They may then change and modify in real time to enhance system performance.

Cost Reduction: Several factors had minimal effect on the 'usr' performance indicator, according to the investigation. This can help organisations cut expenses and streamline their data collecting activities.

Conclusion:

In conclusion, the research shows how a linear regression model and data analysis may give organisations useful information. It emphasises how crucial it is to comprehend key variables, allocate resources optimally, and make data-driven decisions in order to improve system performance and cut expenses.

Predictive Modeling for Contraceptive Method Use

Executive Summary:

We report on the findings of our predictive modelling research to better understand and forecast the usage of contraceptive methods among Indonesian married women. To address this significant public health issue, we implemented three distinct machine learning algorithms: Logistic Regression, Linear Discriminant Analysis (LDA), and Classification and Regression Trees (CART).

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.

Dataset for Problem 2: Contraceptive_method_dataset.xlsx

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

During the project's first phase, we imported NumPy, Pandas, Matplotlib, Seaborn, and Scikit-learn, among other key libraries for data analysis and machine learning. Next, we imported the 'Contraceptive Method Dataset' from an Excel spreadsheet that includes information on 1,473 Indonesian married women. Many factors are included in this dataset, including the wife's age, educational attainment, husband's education, number of children, and more. Predictive modelling will utilise these factors to estimate the utilisation of contraceptive methods.

Sample dataset

	0	1	2	3	4
Wife_age	24.0	45.0	43.0	42.0	36.0
Wife_education	0.0	3.0	0.0	1.0	1.0
Husband_education	1.0	1.0	1.0	0.0	1.0
No_of_children_born	3.0	10.0	7.0	9.0	8.0
Wife_religion	1.0	1.0	1.0	1.0	1.0
Wife_Working	0.0	0.0	0.0	0.0	0.0
Husband_Occupation	2.0	3.0	3.0	3.0	3.0
Standard_of_living_index	0.0	2.0	2.0	0.0	1.0
Media_exposure	0.0	0.0	0.0	0.0	0.0
Contraceptive_method_used	0.0	0.0	0.0	0.0	0.0

Fig10: Sample dataset

There are 10 columns and 1,473 rows in the dataset. It has a variety of data kinds, with some missing numbers in the columns labelled "Wife age" and "No of children born." There are several categorical characteristics that need to be preprocessed before modelling. At first, there were 80 duplicate rows in the dataset, making 1,473 rows and 10 columns altogether. The dataset was pared down to 1,393 rows and 10 columns after duplicates were eliminated, improving data integrity.

The unique counts of categorical features in the dataset are as follows:

- Wife_education: Tertiary (515), Secondary (398), Primary (330), Uneducated (150).
- Husband_education: Tertiary (827), Secondary (347), Primary (175), Uneducated (44).
- Wife_religion: Scientology (1186), Non-Scientology (207).
- Wife_Working: No (1043), Yes (350).
- Standard_of_living_index: Very High (618), High (419), Low (227), Very Low (129).
- Media_exposure: Exposed (1284), Not-Exposed (109).
- Contraceptive_method_used: Yes (779), No (614).

These unique counts provide insight into the distribution of categorical variables in the dataset.

Boxplots

The summary statistics of the dataset show the distribution and central tendency of the continuous variables. To visualise the data distribution and find any outliers, box plots were made for the variables "Wife age," "No of children born," and "Husband occupation" using a 1.5 IQR threshold.

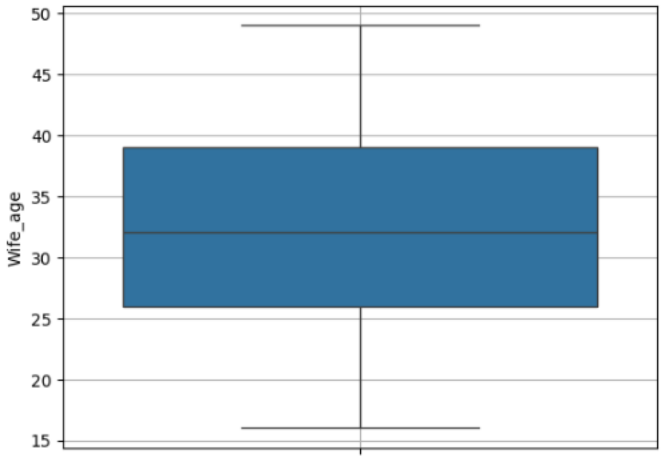

Fig 11: Boxplot Wife Age

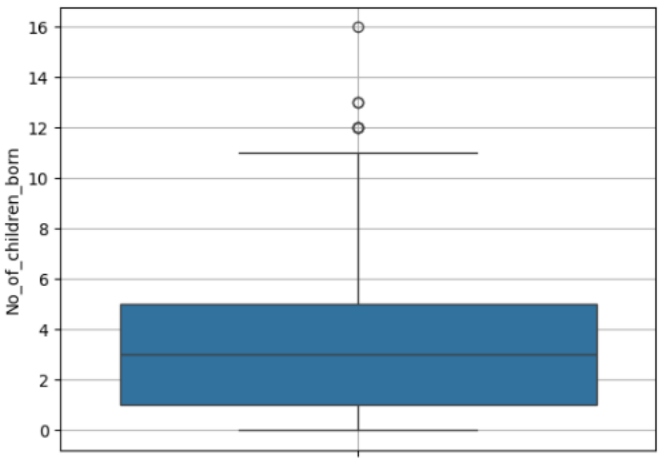

Fig 12: Boxplot No of children born

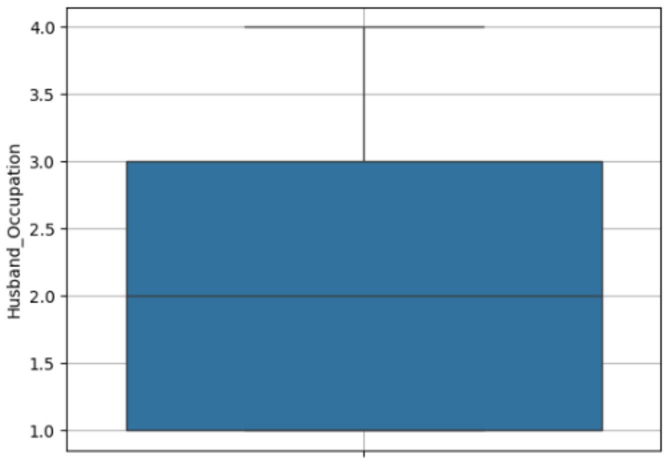

Fig 13 Boxplot Husband Occupation

Outlier Removal

For the 'No_of_children_born' variable, an outlier removal function was used. A moderate association between the wife's age and the number of children born was shown by the correlation matrix between 'Wife age' and 'No of children born', which had a positive correlation of around 0.54 after excluding outliers.

Imputing Missing values

The `SimpleImputer` function from the scikit-learn package was utilised to impute the missing values in 'Wife_age' and 'No_of_children_born,' with the strategy set to'median.' By using the median of the corresponding columns to fill in the missing values, this technique successfully ensured the correctness and completeness of the data for subsequent modelling and analysis.

Also, Column names 'Wife_education' and 'Media_exposure' were renamed to remove extra spaces for consistency in the dataset.

Label Encoder

Categorical variables must be converted into a numerical representation using Label Encoder in order for machine learning techniques to work. Although algorithms normally require numerical inputs, this modification makes it possible for them to function well with categorical data. In our investigation, categorical characteristics were encoded using Label Encoder to get the data ready for modelling. To guarantee that the categorical features are correctly translated and that the data is prepared for further machine learning analysis, the dataset was simultaneously divided into training and testing sets in a 70:30 ratio.

Pairplots

Pairplots are an effective method for visualising connections between many variables in data. To help with the comprehension of correlations and dependencies, they provide a grid of scatterplots for numerical data. More thorough exploratory data analysis is made possible by the addition of categorical colour differences. The present study utilised a Seaborn pairplot to visually represent variable relationships and emphasise specific data points based on the 'Contraceptive_method_used.'

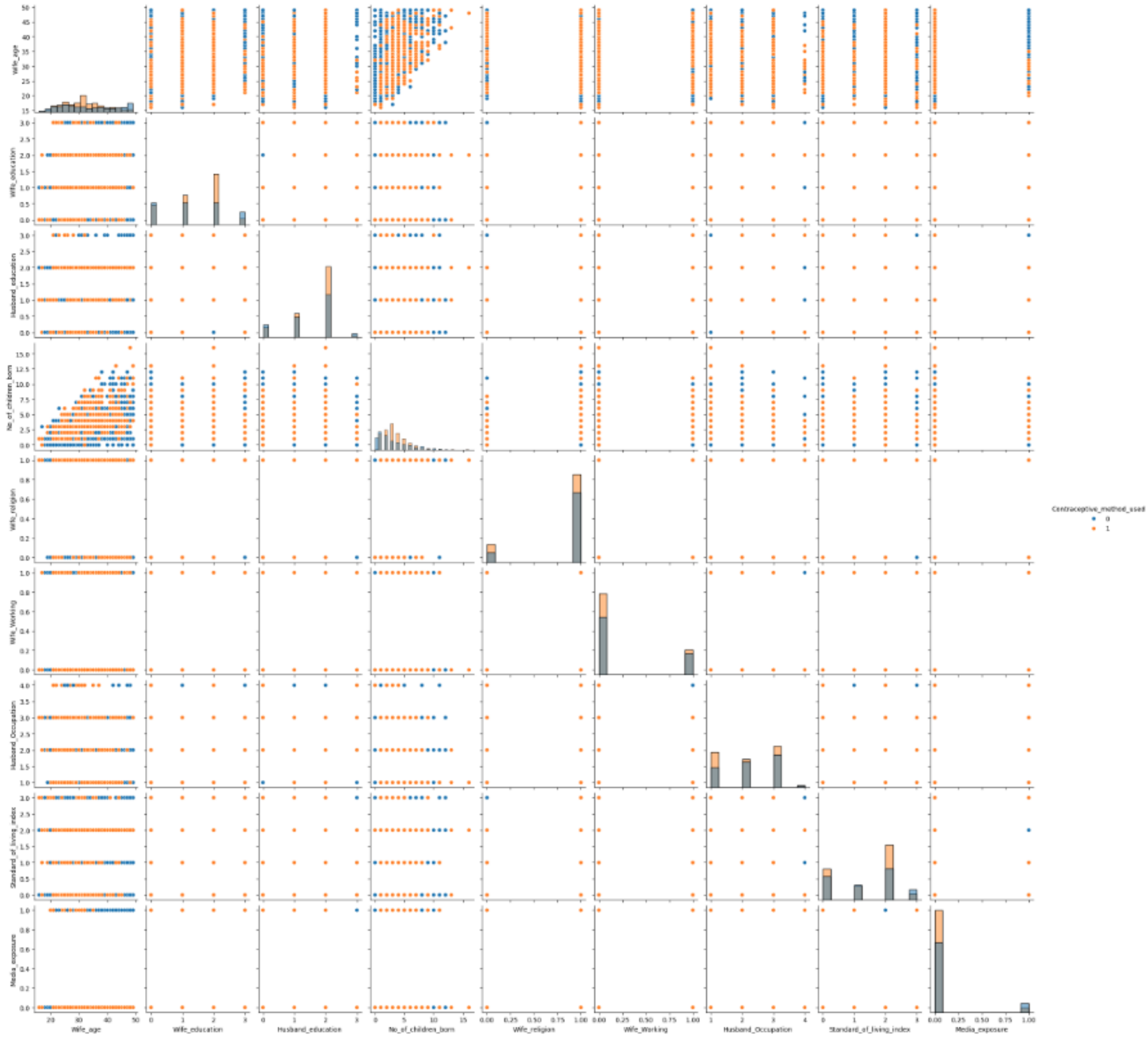

Fig 14: Pairplots

Training and Assessing Models to Predict Contraceptive Methods

The first stage in our attempt to predict contraceptive method choices was to divide the information into subgroups for testing and training. As we were getting the data ready for model development, we made sure that our results could be replicated using a 70:30 split ratio and a reliable random seed.

Based on the various characteristics in the dataset, three different machine learning models were developed, trained, and prepared to predict the use of contraceptive methods:

Logistic Regression Model: Trained to estimate contraceptive method preferences, our logistic regression model is specifically designed for binary classification problems.

LDA (Linear Discriminant Analysis) Model: The LDA model is a good fit for classification issues because it uses dimensionality reduction to improve class separability.

Classification and regression trees (CART): CART model divides data according to feature values in order to provide a prediction model for choosing a contraceptive technique. It does this by using a decision tree-based methodology.

The basis for these models was mostly constructed using the training dataset. We can now assess their prediction ability by determining how well they anticipate preferences for contraceptive methods using datasets for both training and testing. We evaluate the models and determine which performs better at the job of contraceptive method prediction by evaluating accuracy, confusion matrices, ROC curves, and ROC AUC scores.

Accuracy, ROC curves, Precision, Recall

We've put in place a function that allows us to thoroughly assess how well our machine learning models are working. Key performance parameters, including as accuracy, precision, recall, and ROC curves, are computed and shown by this function. We may methodically evaluate and compare these crucial measures by giving parameters such as a model, input data, and the name of the dataset. This assessment approach is essential to our analysis and decision-making since it enables us to assess the efficacy of our models in a well-informed manner.

Evaluation Logistic regression model

Key indicators were used to evaluate the Logistic Regression model's performance. It obtained an accuracy of 0.65, precision of 0.66, and recall of 0.83 on the training dataset. The testing dataset produced findings with an accuracy of 0.67, precision of 0.66, and recall of 0.86, indicating reliable and consistent performance.

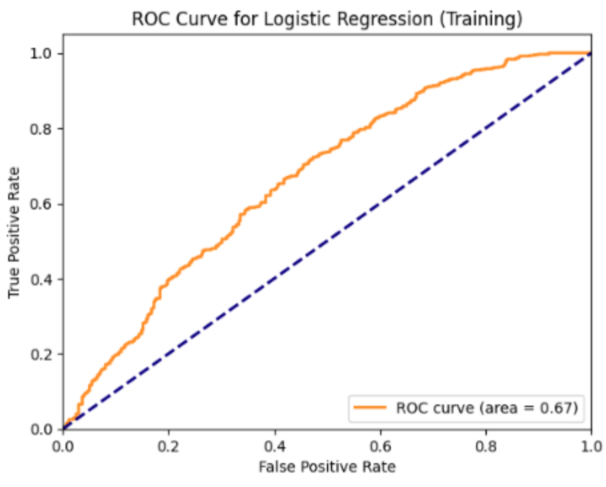

Fig 15: ROC Curve for Logistic Regression(Training)

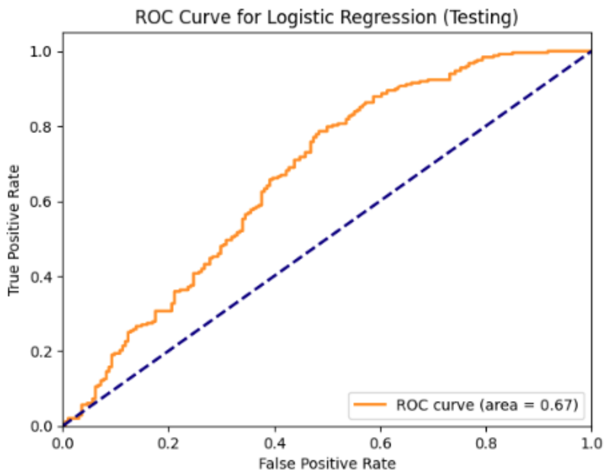

Fig 16 ROC Curve for Logistic Regression(Testing)

Evaluating Linear Discriminant Analysis (LDA model)

On both the training and testing datasets, performance measures for the Linear Discriminant Analysis (LDA) model were evaluated. The LDA model showed precision values of 0.66 and 0.65, recall values of 0.84 and 0.87, and accuracy of 0.65 and 0.67 for training and testing, respectively.

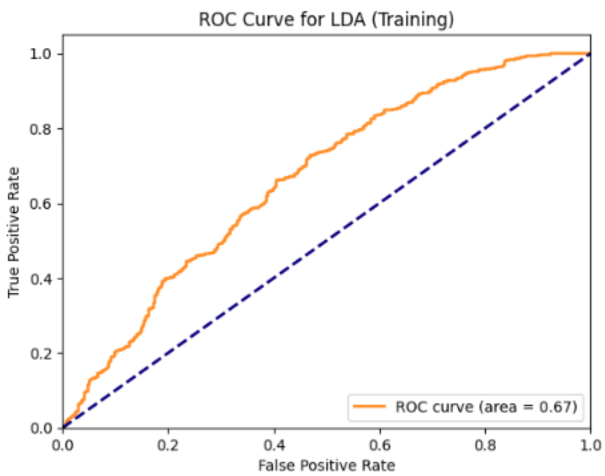

Fig 17 ROC curve for LDA (Training)

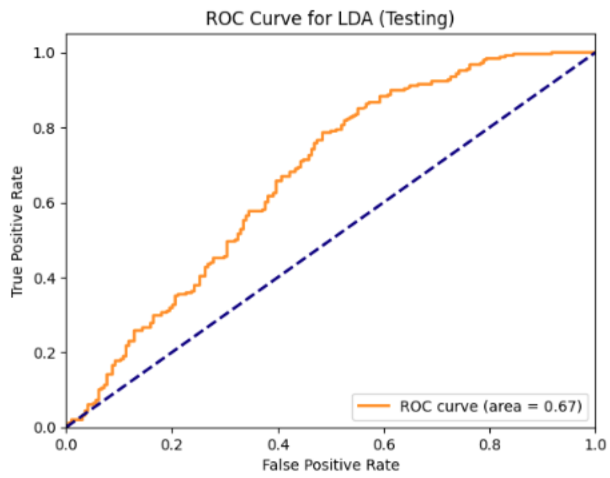

Fig 18 ROC curve for LDA(Testing)

Evaluating Classification and Regression Trees (CART) Model

The Classification and Regression Trees (CART) model underwent performance evaluation on both training and testing datasets. It exhibited impressive accuracy of 0.98 and 0.66 on the respective datasets. The confusion matrix for training displayed strong results, with precision and recall scores of 0.99 and 0.97. However, the testing dataset showed slightly lower accuracy, with a precision score of 0.71 and a recall of 0.65.

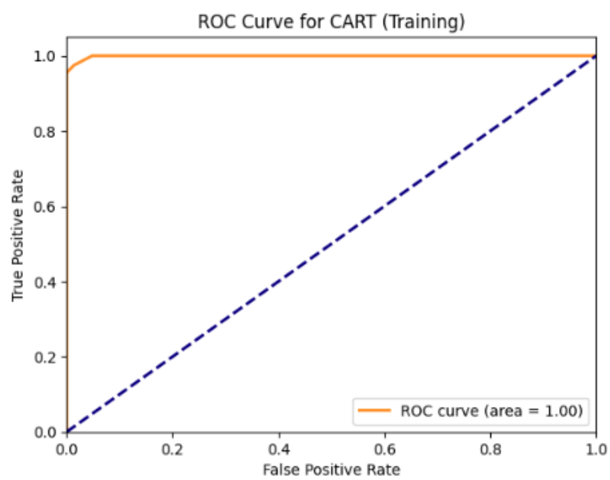

Fig 19 ROC curve for CART(Training)

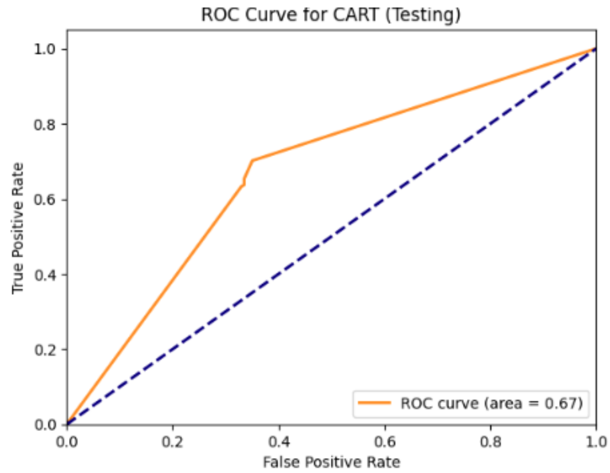

Fig 20 ROC curve for CART(Testing)

The CART model outperformed the other two models with respect to accuracy (0.98), precision (0.99), and recall (0.97) on the training dataset.
With an accuracy of 0.66, precision of 0.71, and recall of 0.65, the CART model's performance declined when evaluated on the testing dataset.
The LDA model performed consistently on both training and testing datasets, achieving an accuracy of 0.67. On the testing dataset, it also demonstrated balanced precision (0.66) and greater recall (0.87).
On the testing dataset, the Logistic Regression model produced competitive results with an accuracy of 0.67, balanced precision (0.66), and good recall (0.86).

Summary of Case study:

Data Preparation: After loading the dataset and removing any duplicates, missing values were imputed using the median value.

Data Encoding: To prepare categorical characteristics for machine learning models, LabelEncoder was used to encode them.

Data Split: To train and assess machine learning models, the dataset was divided into testing and training sets in a 70:30 ratio.

Model Training: Using the training data, models for logistic regression, LDA, and CART were created and trained.

Model Evaluation: On both training and testing datasets, the models were assessed using ROC curves, ROC AUC scores, accuracy, precision, recall, confusion matrices, and ROC curves.

Conclusion: Based on their performance, the top-performing models were determined, and conclusions were made. Family planning and healthcare recommendations were given.

Insights

The business implications of these models may be enormous for family planning organisations, legislators, and healthcare professionals. These models can be used to identify potential users of contraceptives so that specific services, information, and support can be provided to this population. This may result in better healthcare results and more efficient family planning techniques.

Ensuring data quality and minimising the influence of outliers on the models are crucial. Overfitting and poor model generalisation can result from outliers.

Conclusion:

This project demonstrates how machine learning can be utilized for predicting contraceptive method usage.

END OF REPORT