Business Report

Regression Problem Statement

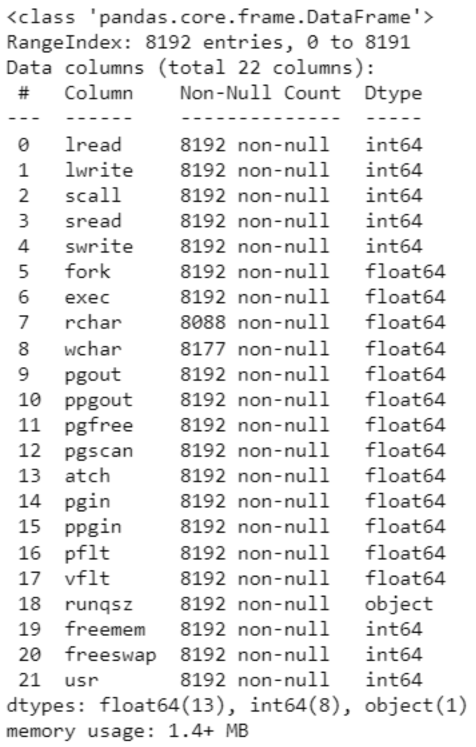
Ayush Sharma

Table of Contents

A. **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.**

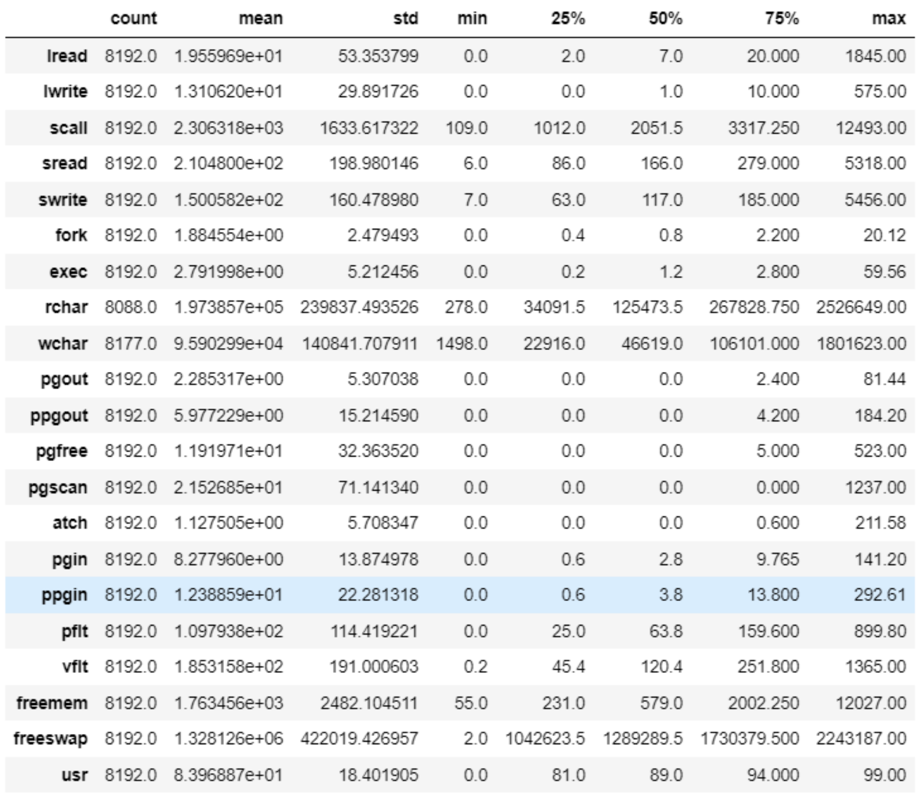
Ans:

Data Summary

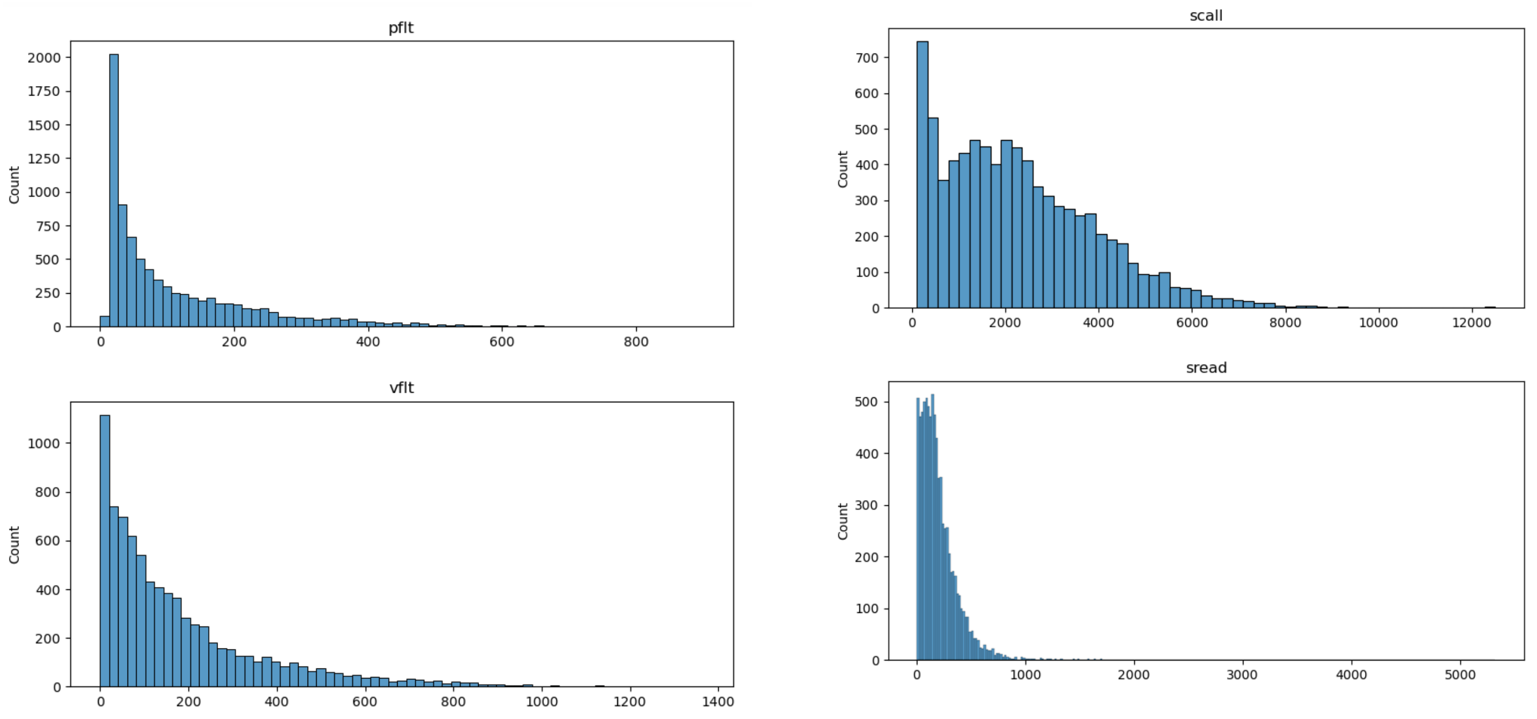


```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8192 entries, 0 to 8191
Data columns (total 22 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   lread     8192 non-null   int64  
 1   lwrite    8192 non-null   int64  
 2   scall     8192 non-null   int64  
 3   sread     8192 non-null   int64  
 4   swrite    8192 non-null   int64  
 5   fork      8192 non-null   float64
 6   exec      8192 non-null   float64
 7   rchar     8088 non-null   float64
 8   wchar     8177 non-null   float64
 9   pgout     8192 non-null   float64
 10  ppgout    8192 non-null   float64
 11  pgfree    8192 non-null   float64
 12  pgscan    8192 non-null   float64
 13  atch      8192 non-null   float64
 14  pgin      8192 non-null   float64
 15  ppgin     8192 non-null   float64
 16  pflt      8192 non-null   float64
 17  vflt      8192 non-null   float64
 18  runqsz    8192 non-null   object 
 19  freemem   8192 non-null   int64  
 20  freeswap  8192 non-null   int64  
 21  usr       8192 non-null   int64  
dtypes: float64(13), int64(8), object(1)
memory usage: 1.4+ MB
```

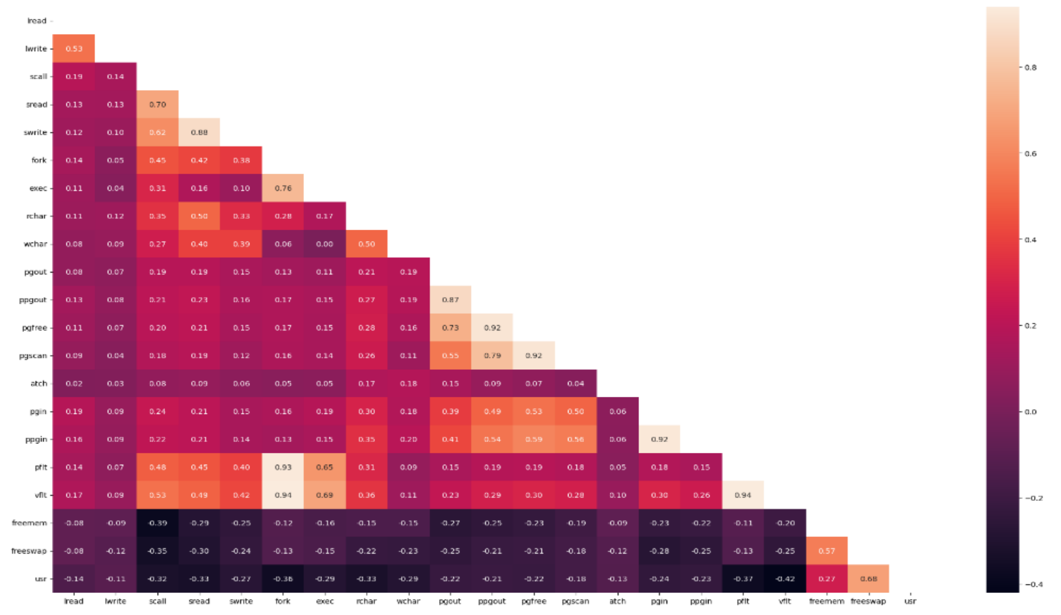
Data Info



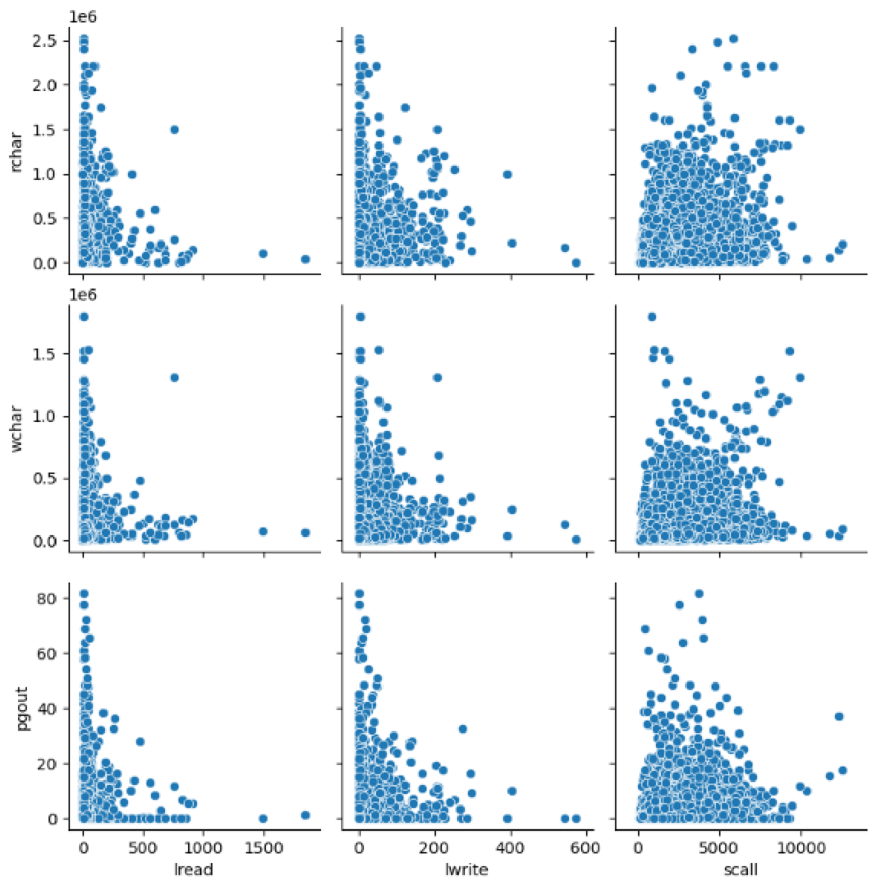
	count	mean	std	min	25%	50%	75%	max
lread	8192.0	1.955969e+01	53.353799	0.0	2.0	7.0	20.000	1845.00
lwrite	8192.0	1.310620e+01	29.891726	0.0	0.0	1.0	10.000	575.00
scall	8192.0	2.306318e+03	1633.617322	109.0	1012.0	2051.5	3317.250	12493.00
sread	8192.0	2.104800e+02	198.980146	6.0	86.0	166.0	279.000	5318.00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



pflt
scall
vflt
sread
Count



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



Inferences:

1. Univariate Analysis:

- There are a total of 22 columns and 8192 rows in the dataset.
- There are 21 numeric variables and 1 categorical variable in the dataset.
- The histograms for approximately all of the independent variables are right-skewed.
- Only the variable *'freeswap'* has a different distribution.
- Due to similar trends in the data, multicollinearity between the variables can possess a challenge which in turn can effect our linear model
- The dependent variable usr or portion of time (%) ranges from 0 to 99% with its mean and median values at 84% and 89% respectively.
- The process run queue size is the only categorical variable in the dataset with 2 categories of systems namely CPU bound and Non-CPU bound. Both categories have almost equal distribution throughout the dataset.

2. Bivariate Analysis-There are considerable number of features that are highly correlated:

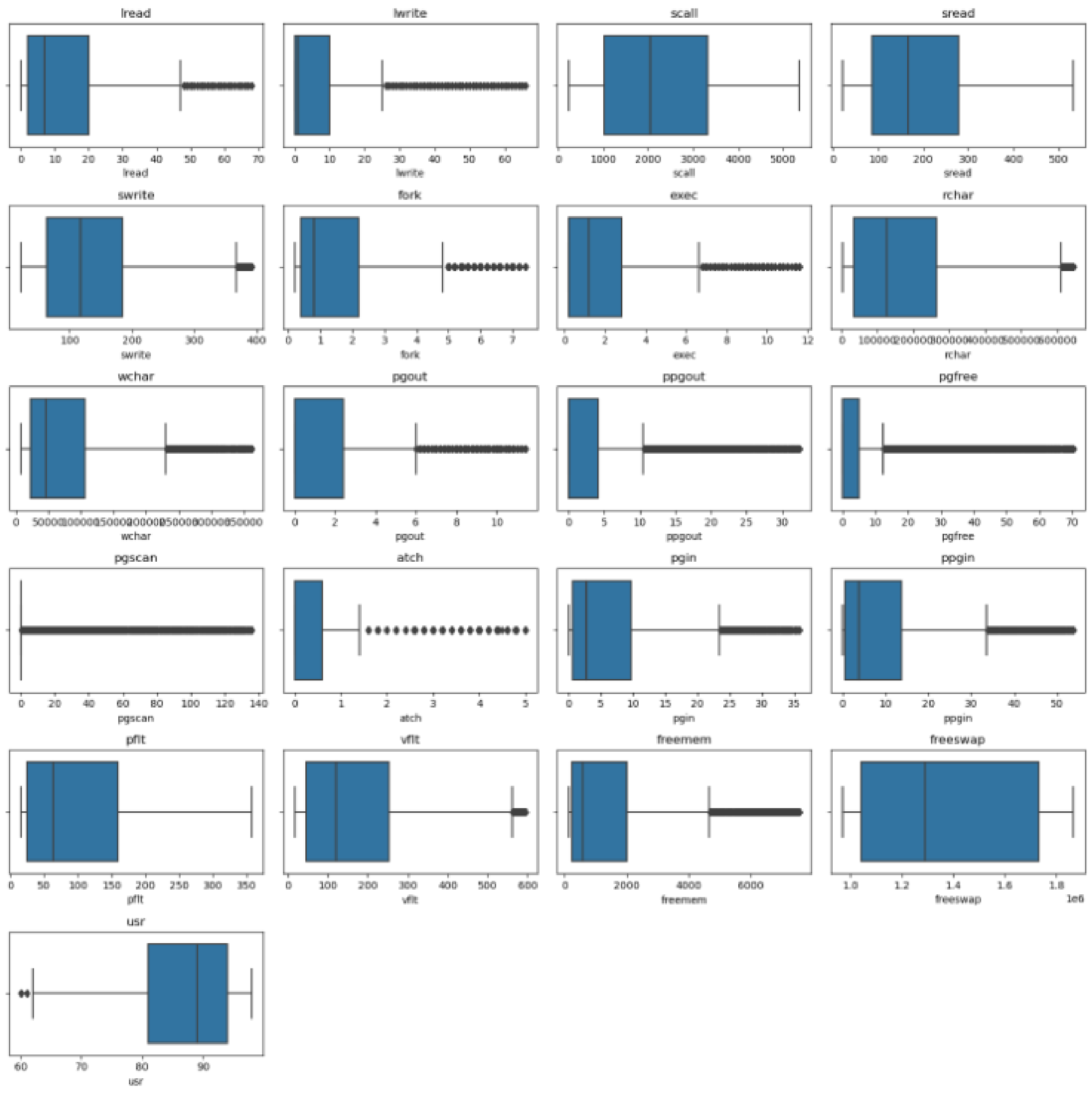
- *'lwrite'* shows high correlation with *'lread'* (0.85)

- *'scall'* shows high correlation with *'swrite'* (0.74) and *'sread'* (0.77)
- *'sread'* shows high correlation with *'swrite'* (0.88)
- *'fork'* shows very high correlation with *'pfit'* (0.94) and *'vfit'* (0.93) and high correlation with *'exec'*
- *'exec'* shows high correlation with *'pfit'* (0.73) and *'vfit'* (0.74)
- *'pgout'* shows very high correlation with *'ppgout'* (0.92), *'pgfree'* (0.82) and average correlation with *'pgscan'* (0.69)
- *'ppgout'* shows very high correlation with *'pgfree'* (0.94) and *'pgscan'* (0.85)
- *'ppgout'* shows very high correlation with *'pgfree'* (0.94) and *'pgscan'* (0.85)
- *'pgfree'* shows extremely high correlation with *'pgscan'* (0.95)
- *'pgin'* shows extremely high correlation with *'ppgin'* (0.96)
- *'pfit'* shows very high correlation with *'ppgin'* (0.93)
- *'pfit'* shows high correlation with *'vfit'* (0.93)

B. **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.**

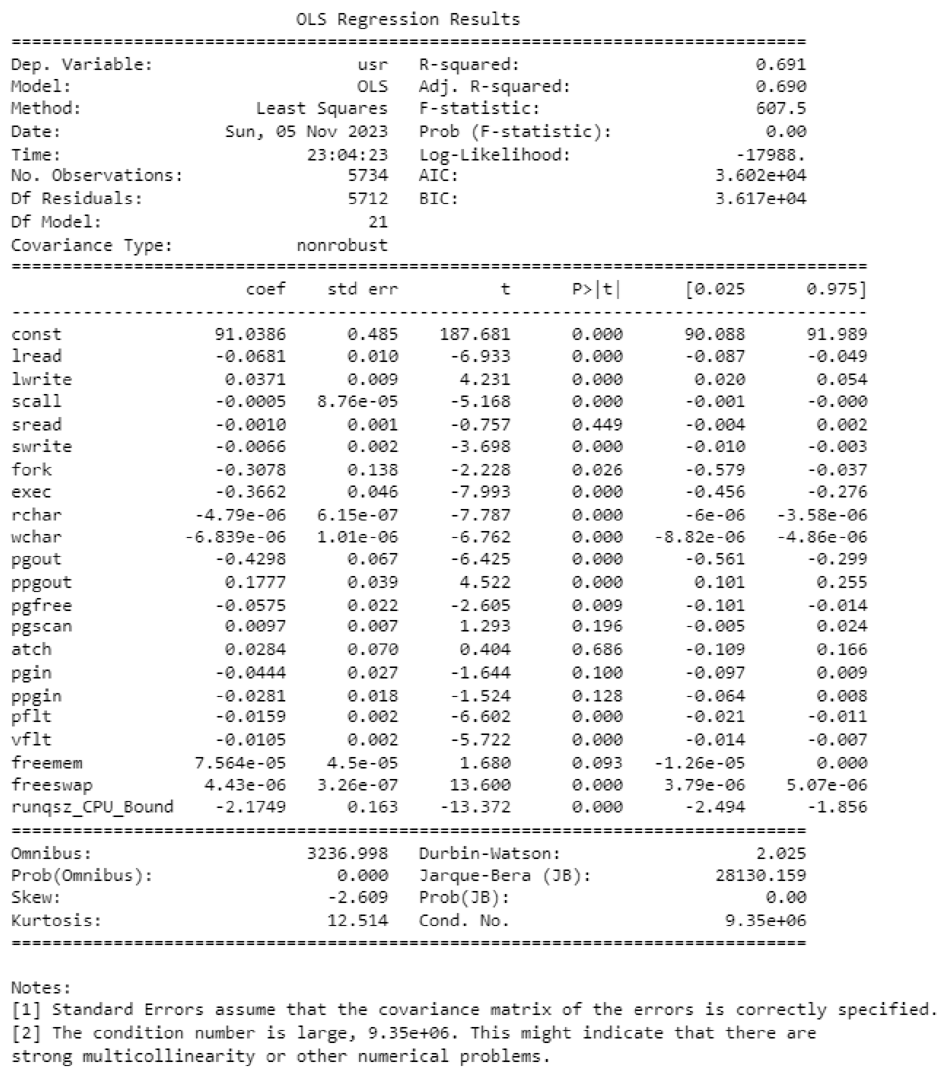
Ans:

Visualizing the Treated Outliers using Boxplots



C. **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.**

Ans:



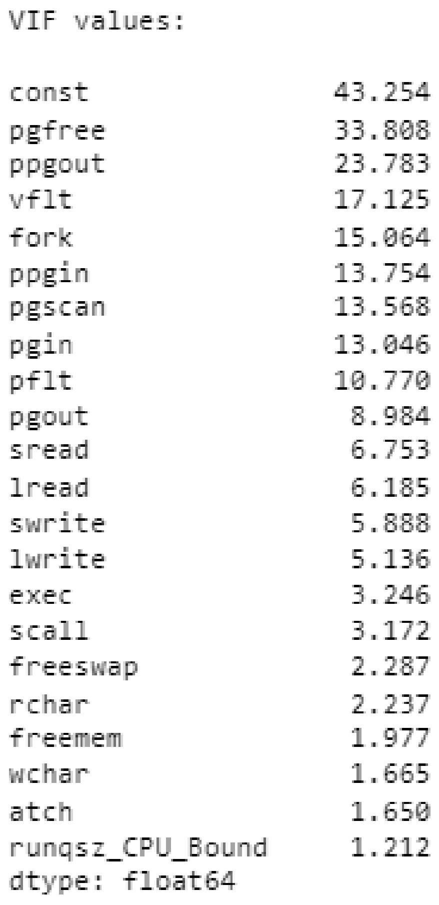
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    usr   R-squared:                       0.691
Model:                            OLS   Adj. R-squared:                  0.690
Method:                 Least Squares   F-statistic:                     607.5
Date:                Sun, 05 Nov 2023   Prob (F-statistic):               0.00
Time:                        23:04:23   Log-Likelihood:                -17988.
No. Observations:                5734   AIC:                         3.602e+04
Df Residuals:                    5712   BIC:                         3.617e+04
Df Model:                          21
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const               91.0386      0.485    187.681      0.000      90.088      91.989
lread               -0.0681      0.010     -6.933      0.000      -0.087      -0.049
lwrite               0.0371      0.009      4.231      0.000       0.020       0.054
scall               -0.0005   8.76e-05     -5.168      0.000      -0.001      -0.000
sread               -0.0010      0.001     -0.757      0.449      -0.004       0.002
swrite              -0.0066      0.002     -3.698      0.000      -0.010      -0.003
fork                -0.3078      0.138     -2.228      0.026      -0.579      -0.037
exec                -0.3662      0.046     -7.993      0.000      -0.456      -0.276
rchar             -4.79e-06   6.15e-07     -7.787      0.000      -6e-06   -3.58e-06
wchar            -6.839e-06   1.01e-06     -6.762      0.000   -8.82e-06   -4.86e-06
pgout               -0.4298      0.067     -6.425      0.000      -0.561      -0.299
ppgout               0.1777      0.039      4.522      0.000       0.101       0.255
pgfree              -0.0575      0.022     -2.605      0.009      -0.101      -0.014
pgscan               0.0097      0.007      1.293      0.196      -0.005       0.024
atch                 0.0284      0.070      0.404      0.686      -0.109       0.166
pgin                -0.0444      0.027     -1.644      0.100      -0.097       0.009
ppgin               -0.0281      0.018     -1.524      0.128      -0.064       0.008
pflt                -0.0159      0.002     -6.602      0.000      -0.021      -0.011
vflt                -0.0105      0.002     -5.722      0.000      -0.014      -0.007
freemem           7.564e-05    4.5e-05      1.680      0.093   -1.26e-05       0.000
freeswap           4.43e-06   3.26e-07     13.600      0.000    3.79e-06    5.07e-06
runqsz_CPU_Bound    -2.1749      0.163    -13.372      0.000      -2.494      -1.856
==============================================================================
Omnibus:                     3236.998   Durbin-Watson:                   2.025
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            28130.159
Skew:                          -2.609   Prob(JB):                         0.00
Kurtosis:                      12.514   Cond. No.                     9.35e+06
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 9.35e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Interpretation of R-squared and p-values:

- The R-squared value tells us that our model can explain 69.1% of the variance in the training set.
- The p-values (P>|t|) are mostly low for all the predictors however, due to the presence of high multicollinearity in our data, the p-values are bound to be different.
- Hence, we need to ensure that there is no multicollinearity in order to interpret the p-values.

Checking for Multicollinearity using VIF:

- It was observed earlier that there was significant correlation existing amongst some of the variables in the data.
- Multicollinearity must be dealt with in order to ensure the correct p-values of the predictors which in turn would help increase our model's overall efficiency.
- We will detect the multicollinearity among the variables using the <b> Variance Inflation Factor (VIF)



```
VIF values:

const             43.254
pgfree            33.808
ppgout            23.783
vflt              17.125
fork              15.064
ppgin             13.754
pgscan            13.568
pgin              13.046
pflt              10.770
pgout              8.984
sread              6.753
lread              6.185
swrite             5.888
lwrite             5.136
exec               3.246
scall              3.172
freeswap           2.287
rchar              2.237
freemem            1.977
wchar              1.665
atch               1.650
runqsz_CPU_Bound   1.212
dtype: float64
```

-It can be clearly indicated from the high VIF values that many features in our dataset are correlated with each other.

-We can trust the p-values of the predictors with low VIF values in the range of 5-10 as the multicollinearity only affects the correlated variables.

-To treat multicollinearity, we will have to drop one or more of the correlated features.

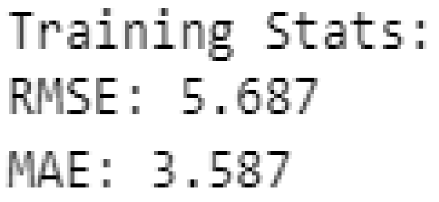
-We will drop the variable that has the least impact on the adjusted R-squared of the model and at the same time has the highest VIF value as compared to the other variables.

D. 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.

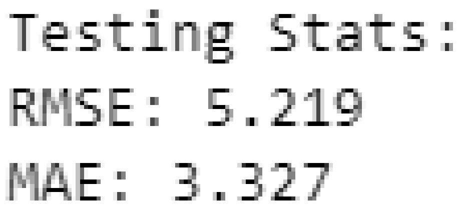
Ans: After dropping the features causing strong multicollinearity and the statistically insignificant ones, our model performance hasn't dropped sharply (adj. R-squared has dropped from 0.690 to 0.678). This shows that these variables did not have much predictive power.

The linear regression equation as per the model is as follows:

usr = 89.96449453310615 + -0.018567740634552986 * (lwrite) + -0.000873338591281 7161 * (scall) + -0.41358157303340654 * (exec) + -5.508513736091345e-06 * (rchar) + -7.302704743464582e-06 * (wchar) + -0.22569610873524631 * (pgout) + -0.103 42733892364857 * (pgin) + -0.039640267509977414 * (pflt) + 5.3752393057612255e -06 * (freeswap) + -2.307279944031199 * (runqsz_CPU_Bound)



```
Training Stats:
RMSE: 5.687
MAE: 3.587
```



```
Testing Stats:
RMSE: 5.219
MAE: 3.327
```

- We can observe that RMSE on the train and test sets are comparable. So, our model is not suffering from overfitting.
- MAE indicates that our current model is able to predict mpg within a mean error of 3.32 units on the test data.
- Hence, we can conclude that our model is good for prediction as well as inference purposes.