Comp-activ – Linear-Regression

Name :

Apoorv Purohit

Padmini Subramanian

Preyal Deep Chhabra

Rathi Sadhasivan

Renuka Prasad GM

Sukrit Kalia

PGP-DSBA Offline

August - 23

Date: 24/12/2023

Comp-activ.: Linear Regression – Business Report

Contents

List of Tables

List of Figures

Comp-activ.: Linear Regression – Business Report

Executive Summary

The comp-activ database is a collection of 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.
The requirement here is to formulate a linear equation to build a model to predict 'usr' (Portion of time (%) that cpu runs in user mode) and to analyse how each attribute affects the system to be in 'usr' mode using a list of system attributes.

Introduction

Assignment is to deep understanding of dataset and perform exploratory data analysis. Explore datasets with linear regression to validate, a data scientist built a linear equation model to predict the "usr" variable (Portion of time (%) that CPU runs in user mode) and how other system attributes are influencing the target variable. The dataset consists of 22 columns having numerical and categorical data and 8192 rows. Analyse different features of numerical data's present in dataset and how this data interrelationship with other numerical variables and which variables will helps to predict comp-activ using supervised linear regression approach. Dataset will explore more on summary statistics, null values, anomalies present in numerical variable, train and test the data under 70/30 combination, encode the data for numerical attributes to find the accuracy of the model and data visualization across numerical and categorical subjects. Assumptions are linear, multicollinearity, ViF, homoscedasticity and normality. Generate a Rsquare, RMSE & Adj Rsquare to give more insight on accurate prediction of comp-activ dataset. Do compare the prediction values between scikit learn and OLS method of linear regression.

Data Description

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 transferred 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 CPU runs in user mode

Sample of the dataset:

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	freeswap	usr
0	1	0	2147	79	68	0.2	0.2	40671.0	53995.0	0.0	0.0	0.0	0.0	0.0	1.6	2.6	16.00	26.40	CPU_Bound	4670	1730946	95
1	0	0	170	18	21	0.2	0.2	448.0	8385.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	15.63	16.83	Not_CPU_Bound	7278	1869002	97
2	15	3	2162	159	119	2.0	2.4	NaN	31950.0	0.0	0.0	0.0	0.0	1.2	6.0	9.4	150.20	220.20	Not_CPU_Bound	702	1021237	87
3	0	0	160	12	16	0.2	0.2	NaN	8670.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	15.60	16.80	Not_CPU_Bound	7248	1863704	98
4	5	1	330	39	38	0.4	0.4	NaN	12185.0	0.0	0.0	0.0	0.0	0.0	1.0	1.2	37.80	47.60	Not_CPU_Bound	633	1760253	90

Table 1. Dataset Sample

Data has 22 variables with more numerical variables (13 are of float64 type, 8 are of int64 type and 1 is of type object) in comp-activ data and other attributes influences more towards the prediction of linear model

Exploratory Data Analysis

Let's check types of variables present in data frame

```
 #   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
```

Total of 8192 rows and 22 columns in the dataset.

Data Visualization - Describe:

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
lread	8192.0	NaN	NaN	NaN	19.559692	53.353799	0.0	2.0	7.0	20.0	1845.0
lwrite	8192.0	NaN	NaN	NaN	13.106201	29.891726	0.0	0.0	1.0	10.0	575.0
scall	8192.0	NaN	NaN	NaN	2306.318237	1633.617322	109.0	1012.0	2051.5	3317.25	12493.0
sread	8192.0	NaN	NaN	NaN	210.47998	198.980146	6.0	86.0	166.0	279.0	5318.0
swrite	8192.0	NaN	NaN	NaN	150.058228	160.47898	7.0	63.0	117.0	185.0	5456.0
fork	8192.0	NaN	NaN	NaN	1.884554	2.479493	0.0	0.4	0.8	2.2	20.12
exec	8192.0	NaN	NaN	NaN	2.791998	5.212456	0.0	0.2	1.2	2.8	59.56
rchar	8088.0	NaN	NaN	NaN	197385.728363	239837.493526	278.0	34091.5	125473.5	267828.75	2526649.0
wchar	8177.0	NaN	NaN	NaN	95902.992785	140841.707911	1498.0	22916.0	46619.0	106101.0	1801623.0
pgout	8192.0	NaN	NaN	NaN	2.285317	5.307038	0.0	0.0	0.0	2.4	81.44
ppgout	8192.0	NaN	NaN	NaN	5.977229	15.21459	0.0	0.0	0.0	4.2	184.2
pgfree	8192.0	NaN	NaN	NaN	11.919712	32.36352	0.0	0.0	0.0	5.0	523.0
pgscan	8192.0	NaN	NaN	NaN	21.526849	71.14134	0.0	0.0	0.0	0.0	1237.0
atch	8192.0	NaN	NaN	NaN	1.127505	5.708347	0.0	0.0	0.0	0.6	211.58
pgin	8192.0	NaN	NaN	NaN	8.27796	13.874978	0.0	0.6	2.8	9.765	141.2
ppgin	8192.0	NaN	NaN	NaN	12.388586	22.281318	0.0	0.6	3.8	13.8	292.61
pflt	8192.0	NaN	NaN	NaN	109.793799	114.419221	0.0	25.0	63.8	159.6	899.8
vflt	8192.0	NaN	NaN	NaN	185.315796	191.000603	0.2	45.4	120.4	251.8	1365.0
runqsz	8192	2	Not_CPU_Bound	4331	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freemem	8192.0	NaN	NaN	NaN	1763.456299	2482.104511	55.0	231.0	579.0	2002.25	12027.0
freeswap	8192.0	NaN	NaN	NaN	1328125.959839	422019.426957	2.0	1042623.5	1289289.5	1730379.5	2243187.0
usr	8192.0	NaN	NaN	NaN	83.968872	18.401905	0.0	81.0	89.0	94.0	99.0

Fig 1. Dataset describe

Q1: 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

Univariate analysis for all the attributes presents in the model as shown in the fig

Insights from the univariant analysis:

- lread varies from 0 to 1845 s. Average value is 19.55 s.
- lread varies from 0 to 1845 s. Average value is 19.55 s.
- lwrite varies from 0 to 575. Average value is 13.10 s.
- scall varies from 109 to 12493. Average value is 2306.32 s.
- sread varies from 6 to 5318. Average value is 210.48 s.
- swrite varies from 7 to 5456. Average value is 150.06 s.
- fork varies from 0 to 20.12. Average value is 1.89 s.
- exec varies from 0 to 59.56. Average value is 2.79 s.
- rchar varies from 278 to 2526649. Average value is 197385.73 s.
- wchar varies from 1498 to 1801623. Average value is 95902.99 s.
- pgout varies from 0 to 81.44. Average value is 2.29 s.
- ppgout varies from 0 to 184.2. Average value is 5.98 s.
- pgfree varies from 0 to 523. Average value is 11.92 s.
- pgscan varies from 0 to 1237. Average value is 21.53 s.
- atch varies from 0 to 211.58. Average value is 1.13 s.
- pgin varies from 0 to 141.2. Average value is 8.28 s.
- ppgin varies from 0 to 292.61. Average value is 12.39 s.
- pflt varies from 0 to 899.8. Average value is 109.80.
- vflt varies from 0.2 to 1365. Average value is 185.32.
- runqsz takes one of the 2 values namely - CPU_Bound, Not_CPU_Bound.
- freemem varies from 55 to 12027. Average is 1763.46.
- freeswap varies from 2 to 2243187. Average is 1328125.96.
- usr varies from 0 to 99. Average is 83.97 %.

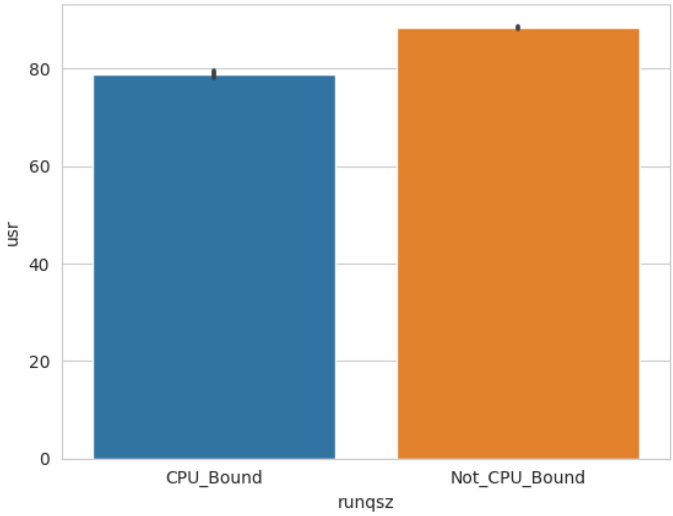


Fig 2. Categorical variable plot

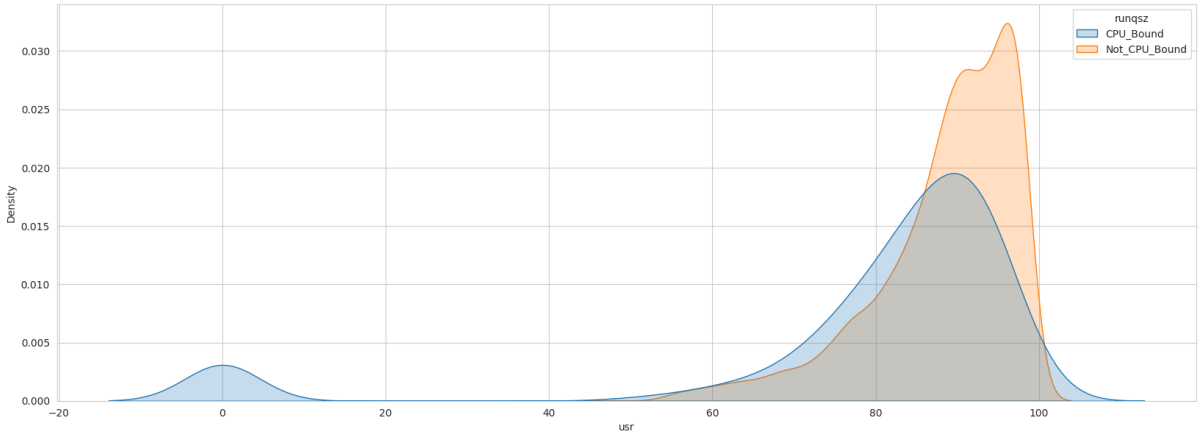


Fig 3. usr density for categorical variable

When user mode is running between 80 to 90% of time, it is majorly a non-CPU bound operation.
This variable might not be helpful in predicting the target.

Bivariate analysis for all the attributes presents in the model as shown in the fig

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	freemem	freeswap	usr
lread	1.000000	0.533737	0.191377	0.132881	0.119953	0.140284	0.110965	0.107973	0.081571	0.082463	0.130590	0.114438	0.087783	0.021563	0.189799	0.161345	0.137463	0.165539	-0.083214	-0.081293	-0.141394
lwrite	0.533737	1.000000	0.143404	0.128403	0.101524	0.052511	0.038237	0.115121	0.091718	0.067013	0.079485	0.065692	0.042608	0.028310	0.091068	0.089011	0.067024	0.094965	-0.091133	-0.116478	-0.111213
scall	0.191377	0.143404	1.000000	0.696887	0.619984	0.446766	0.308999	0.351413	0.274092	0.194529	0.208400	0.199778	0.177908	0.077969	0.241628	0.219070	0.481781	0.531760	-0.387520	-0.350629	-0.323188
sread	0.132881	0.128403	0.696887	1.000000	0.881069	0.416721	0.164084	0.502397	0.401784	0.193679	0.225906	0.212911	0.194272	0.085468	0.207012	0.210225	0.452020	0.491045	-0.286437	-0.302036	-0.332160
swrite	0.119953	0.101524	0.619984	0.881069	1.000000	0.376876	0.103643	0.331386	0.394230	0.151371	0.159442	0.145458	0.120180	0.061373	0.147000	0.144278	0.396580	0.416571	-0.248574	-0.237062	-0.272252
fork	0.140284	0.052511	0.446766	0.416721	0.376876	1.000000	0.763974	0.281229	0.060790	0.130192	0.166872	0.168082	0.160839	0.047194	0.163468	0.132181	0.931040	0.939348	-0.123357	-0.130442	-0.363277
exec	0.110965	0.038237	0.308999	0.164084	0.103643	0.763974	1.000000	0.169189	0.000547	0.111465	0.149334	0.146163	0.144855	0.052307	0.186099	0.149911	0.645239	0.691754	-0.158565	-0.153347	-0.288526
rchar	0.107973	0.115121	0.351413	0.502397	0.331386	0.281229	0.169189	1.000000	0.503531	0.211268	0.269160	0.277786	0.259446	0.171532	0.299784	0.347224	0.313246	0.363799	-0.149485	-0.220608	-0.329737
wchar	0.081571	0.091718	0.274092	0.401784	0.394230	0.060790	0.000547	0.503531	1.000000	0.192436	0.188876	0.159229	0.113638	0.181408	0.178290	0.200880	0.086128	0.111082	-0.149060	-0.226044	-0.288974
pgout	0.082463	0.067013	0.194529	0.193679	0.151371	0.130192	0.111465	0.211268	0.192436	1.000000	0.872445	0.730381	0.553916	0.147759	0.385648	0.414865	0.151285	0.229129	-0.269687	-0.245378	-0.221877
ppgout	0.130590	0.079485	0.208400	0.225906	0.159442	0.166872	0.149334	0.269160	0.188876	0.872445	1.000000	0.917790	0.785256	0.093336	0.488261	0.542392	0.185941	0.285708	-0.247554	-0.213791	-0.212295
pgfree	0.114438	0.065692	0.199778	0.212911	0.145458	0.168082	0.146163	0.277786	0.159229	0.730381	0.917790	1.000000	0.915217	0.069290	0.532834	0.593396	0.190468	0.301851	-0.234195	-0.210184	-0.216278
pgscan	0.087783	0.042608	0.177908	0.194272	0.120180	0.160839	0.144855	0.259446	0.113638	0.553916	0.785256	0.915217	1.000000	0.038693	0.496826	0.564991	0.179157	0.283031	-0.193580	-0.178119	-0.181488
atch	0.021563	0.028310	0.077969	0.085468	0.061373	0.047194	0.052307	0.171532	0.181408	0.147759	0.093336	0.069290	0.038693	1.000000	0.057639	0.057373	0.050594	0.095504	-0.086029	-0.121668	-0.125074
pgin	0.189799	0.091068	0.241628	0.207012	0.147000	0.163468	0.186099	0.299784	0.178290	0.385648	0.488261	0.532834	0.496826	0.057639	1.000000	0.923621	0.175644	0.303013	-0.230779	-0.278927	-0.241720
ppgin	0.161345	0.089011	0.219070	0.210225	0.144278	0.132181	0.149911	0.347224	0.200880	0.414865	0.542392	0.593396	0.564991	0.057373	0.923621	1.000000	0.149962	0.263352	-0.215318	-0.253941	-0.233682
pflt	0.137463	0.067024	0.481781	0.452020	0.396580	0.931040	0.645239	0.313246	0.086128	0.151285	0.185941	0.190468	0.179157	0.050594	0.175644	0.149962	1.000000	0.935370	-0.112774	-0.130609	-0.372495
vflt	0.165539	0.094965	0.531760	0.491045	0.416571	0.939348	0.691754	0.363799	0.111082	0.229129	0.285708	0.301851	0.283031	0.095504	0.303013	0.263352	0.935370	1.000000	-0.201790	-0.245384	-0.420685
freemem	-0.083214	-0.091133	-0.387520	-0.286437	-0.248574	-0.123357	-0.158565	-0.149485	-0.149060	-0.269687	-0.247554	-0.234195	-0.193580	-0.086029	-0.230779	-0.215318	-0.112774	-0.201790	1.000000	0.572632	0.270308
freeswap	-0.081293	-0.116478	-0.350629	-0.302036	-0.237062	-0.130442	-0.153347	-0.220608	-0.226044	-0.245378	-0.213791	-0.210184	-0.178119	-0.121668	-0.278927	-0.253941	-0.130609	-0.245384	0.572632	1.000000	0.678526
usr	-0.141394	-0.111213	-0.323188	-0.332160	-0.272252	-0.363277	-0.288526	-0.329737	-0.288974	-0.221877	-0.212295	-0.216278	-0.181488	-0.125074	-0.241720	-0.233682	-0.372495	-0.420685	0.270308	0.678526	1.000000

Fig 4. Pair plot for numerical variables

After filling the missing values in the model. The above image describes the stats values of the numerical values

Data visualization of heat map for all numerical values and very tricky to identify the strong correlation across the variables. After soting out the values for >0.7 as shown in the fig

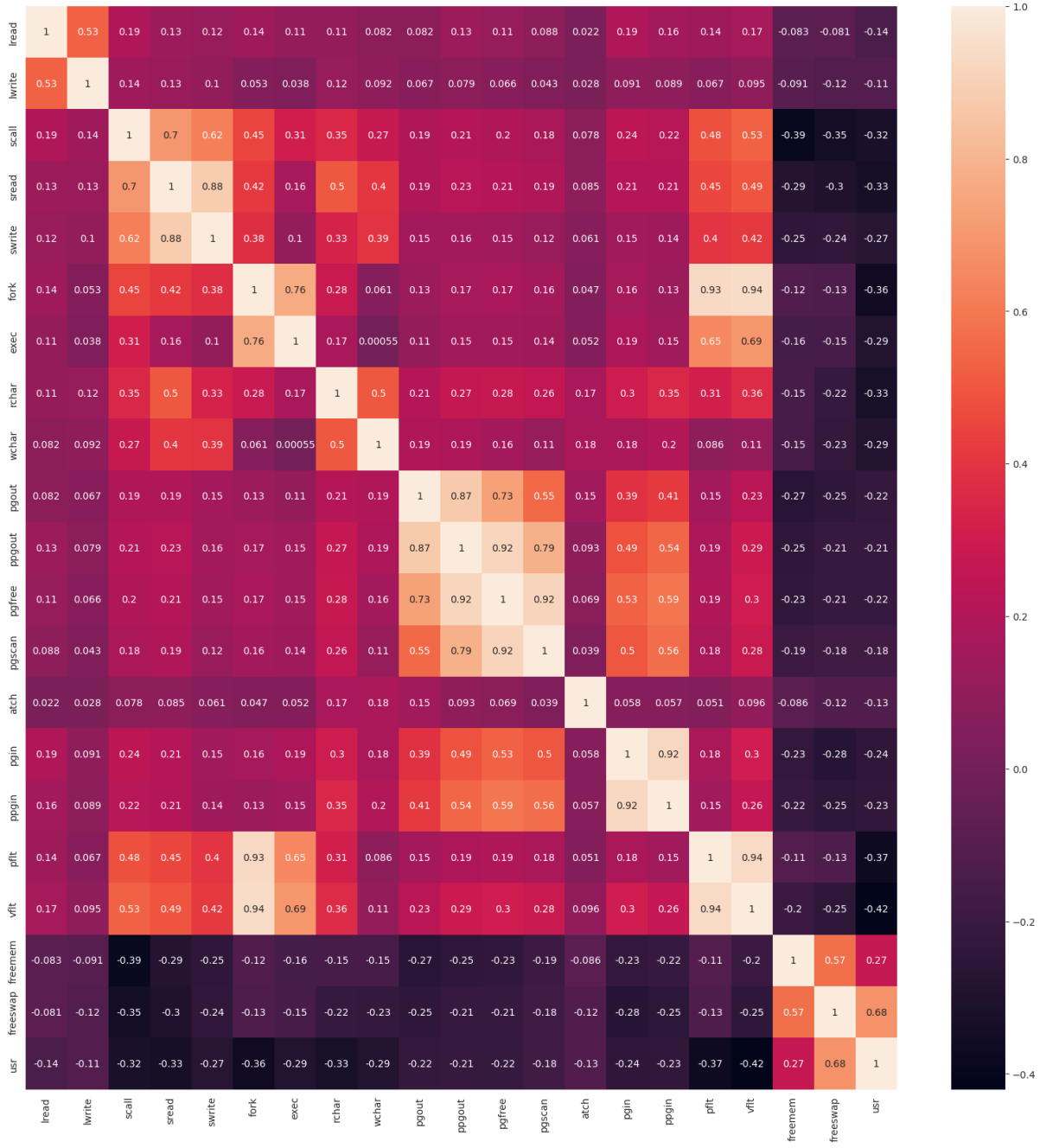


Fig 5. Heat map for numerical variables

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	freemem	freeswap	usr
lread	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
lwrite	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
scall	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
sread	NaN	NaN	NaN	1.000000	0.881069	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
swrite	NaN	NaN	NaN	0.881069	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
fork	NaN	NaN	NaN	NaN	NaN	1.000000	0.763974	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.93104	0.939348	NaN	NaN	NaN
exec	NaN	NaN	NaN	NaN	NaN	0.763974	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
rchar	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
wchar	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
pgout	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.000000	0.872445	0.730381	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
ppgout	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.872445	1.000000	0.917790	0.785256	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
pgfree	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.730381	0.917790	1.000000	0.915217	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
pgscan	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.785256	0.915217	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
atch	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
pgin	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.000000	0.923621	NaN	NaN	NaN	NaN	NaN
ppgin	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.923621	1.000000	NaN	NaN	NaN	NaN	NaN
pflt	NaN	NaN	NaN	NaN	NaN	0.931040	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.00000	0.935370	NaN	NaN	NaN
vflt	NaN	NaN	NaN	NaN	NaN	0.939348	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.93537	1.000000	NaN	NaN	NaN
freemem	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN
freeswap	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN
usr	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0

Fig 6. Pair plot for strong corelation variables

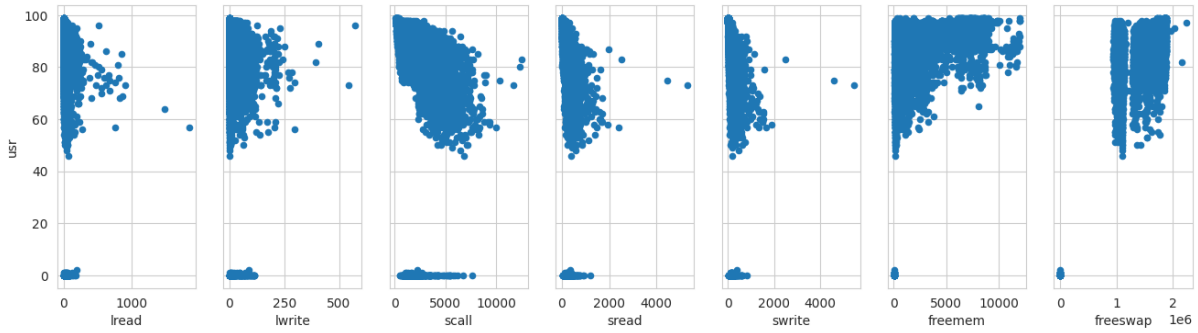


Fig 7. Strong corelation variables plot

Strong correlation variables as shown in the above image.

```
lread and usr corr is-        0.14139389688467285
lwrite and usr corr is-       0.11121341485022547
scall and usr corr is-        0.3231884096869675
sread and usr corr is-        0.33215995802851617
swrite and usr corr is-       0.2722518116362138
freemem and usr corr is-      0.27030831190910853
freeswap and usr corr is-     0.6785262417399952
```

Negatively correlated independent variables as shown in the below image

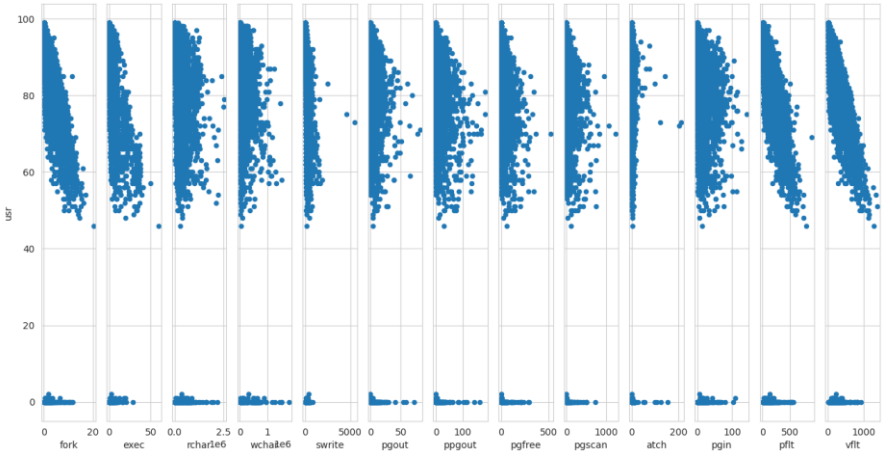
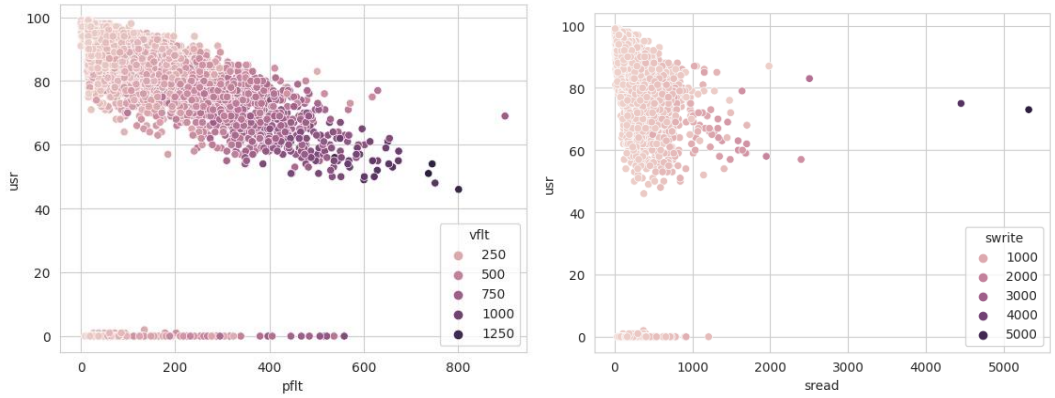


Fig 8. Negative corelation variables plot

```
fork and usr corr is          -0.3632768833634806
exec and usr corr is          -0.2885262168273979
rchar and usr corr is         -0.3297373412099098
wchar and usr corr is         -0.28897361772639185
pgout and usr corr is         -0.22187681320269698
ppgout and usr corr is        -0.21229458749761684
pgfree and usr corr is        -0.21627809168038353
pgscan and usr corr is        -0.18148800962245146
atch and usr corr is          -0.1250742154056606
pgin and usr corr is          -0.24171963028135585
ppgin and usr corr is         -0.23368239114707406
pflt and usr corr is          -0.37249475603039295
vflt and usr corr is          -0.420685309741213
```

Multi- variate Analysis:



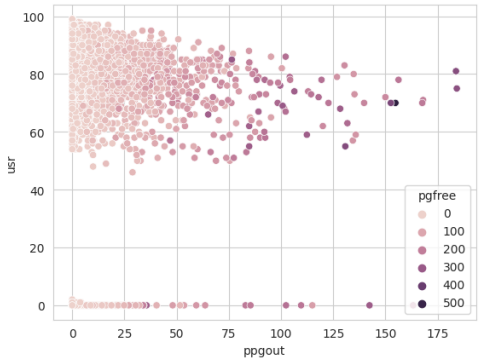


Fig 9. Multi-variate – Variables with "usr" plots

Outcome of the multivariate analysis

ppgout - Number of pages, paged out per second
pgfree - Number of pages per second placed on the free list.
Graphical represents that max ppgout and pgfree happens between 60 to 100% of the time - cpu runs in user mode.
For few data, ppgout and pgfree reported, when the usr is 0%.

Q2]. 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.

Check for missing/null values in the dataset

```
lread         0
lwrite        0
scall         0
sread         0
swrite        0
fork          0
exec          0
rchar       104
wchar        15
pgout         0
ppgout        0
pgfree        0
pgscan        0
atch          0
pgin          0
ppgin         0
pflt          0
vflt          0
runqsz        0
freemem       0
freeswap      0
usr           0
dtype: int64
```

```
lread       0
lwrite      0
scall       0
sread       0
swrite      0
fork        0
exec        0
rchar       0
wchar       0
pgout       0
ppgout      0
pgfree      0
pgscan      0
atch        0
pgin        0
ppgin       0
pflt        0
vflt        0
runqsz      0
freemem     0
freeswap    0
usr         0
dtype: int64
```

From the above data observed that rchar and wchar attributes having missing value in the model. For both attributes fill with median values.

Outliers & duplicates: Outliers are considered for validate the model, otherwise it may lose the generalization. Most of the independent variables left skewed. There are no duplicates present in the model.

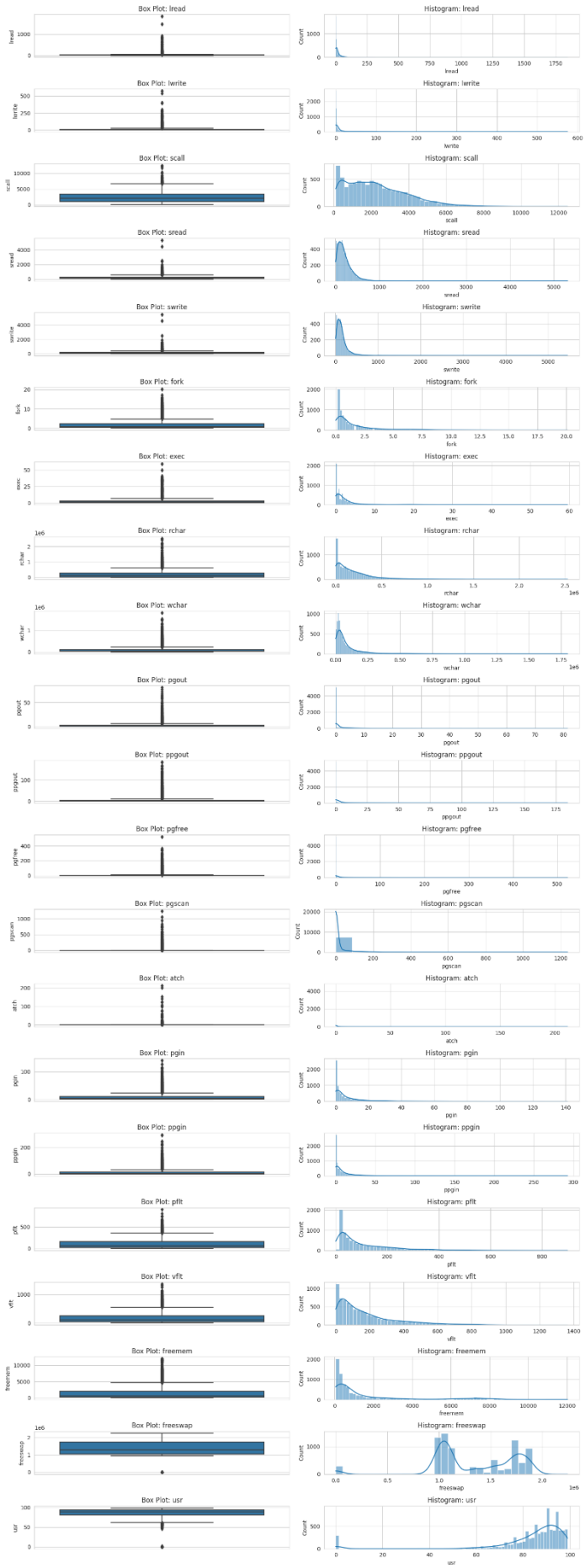

Box Plot: lread
Histogram: lread
Box Plot: lwrite
Histogram: lwrite
Box Plot: scall
Histogram: scall
Box Plot: sread
Histogram: sread
Box Plot: swrite
Histogram: swrite
Box Plot: fork
Histogram: fork
Box Plot: exec
Histogram: exec
Box Plot: rchar
Histogram: rchar
Box Plot: wchar
Histogram: wchar
Box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

Fig 10. Outliers for numerical variables

To check for 0 values in the various columns.

```
False    7517
True      675
Name: lread, dtype: int64
```

```
False    5508
True     2684
Name: lwrite, dtype: int64
```

```
False    8171
True       21
Name: fork, dtype: int64
```

```
False    8171
True       21
Name: exec, dtype: int64
```

```
True     4878
False    3314
Name: pgout, dtype: int64
```

```
True     4878
False    3314
Name: ppgout, dtype: int64
```

```
True     4869
False    3323
Name: pgfree, dtype: int64
```

O values present in the model means that system is idle and acceptable.

Q3]. 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.

Encode the data (having string values) for Modelling. Here, we have a categorical variable - runqsz, that needs to be encoded. We do that by using the pd.get_dummies() function.

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	runqsz	freemem	freeswap	usr
0	1	0	2147	79	68	0.2	0.2	40671.0	53995.0	0.0	0.0	0.0	0.0	0.0	1.6	2.6	16.00	26.40	CPU_Bound	4670	1730946	95
1	0	0	170	18	21	0.2	0.2	448.0	8385.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	15.63	16.83	Not_CPU_Bound	7278	1869002	97
2	15	3	2162	159	119	2.0	2.4	125473.5	31950.0	0.0	0.0	0.0	0.0	1.2	6.0	9.4	150.20	220.20	Not_CPU_Bound	702	1021237	87
3	0	0	160	12	16	0.2	0.2	125473.5	8670.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	15.60	16.80	Not_CPU_Bound	7248	1863704	98
4	5	1	330	39	38	0.4	0.4	125473.5	12185.0	0.0	0.0	0.0	0.0	0.0	1.0	1.2	37.80	47.60	Not_CPU_Bound	633	1760253	90

	lread	lwrite	scall	sread	swrite	fork	exec	rchar	wchar	pgout	ppgout	pgfree	pgscan	atch	pgin	ppgin	pflt	vflt	freemem	freeswap	usr	runqsz_Not_CPU_Bound
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	16.00	26.40	4670	1730946	95	0
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	7278	1869002	97	1
2	15	3	2162	159	119	2.0	2.4	125473.5	31950.0	0.0	0.0	0.0	0.0	1.2	6.0	9.4	150.20	220.20	702	1021237	87	1
3	0	0	160	12	16	0.2	0.2	125473.5	8670.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	15.60	16.80	7248	1863704	98	1
4	5	1	330	39	38	0.4	0.4	125473.5	12185.0	0.0	0.0	0.0	0.0	0.0	1.0	1.2	37.80	47.60	633	1760253	90	1

Split the model into train and test data with combination of 70:30 respectively. Apply Linear regression using scikit learn. Co-efficient values of each dependent variables

```
array([[ 0.00000000e+00, -2.00907136e-02,  7.54223130e-03,
         9.48347297e-04,  7.76397886e-04, -1.37048518e-03,
        -1.95902796e+00, -7.77074449e-03, -3.58025820e-06,
        -8.73552125e-06, -1.92444878e-01,  1.46693589e-01,
        -1.05131055e-01,  1.60555409e-02, -5.22497285e-02,
         5.77837657e-02, -3.80869667e-02, -3.95719448e-02,
         2.24758172e-02, -1.77175474e-03,  3.44969262e-05,
         8.25313251e+00]])
```

RMSE, MAE and R2 are calculated for train and test conditions using scikit learn.

RMSE for train	11.23107203968622	RMSE for test	10.62349610899093
MAE for train	8.22919677892451	MAE for test	7.881026960699974
R2 for train	0.6483203363848282	R2 for test	0.6126182156395615

Fit Linear Model using OLS - finding significance of all the features

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.648
Model:	OLS	Adj. R-squared:	0.647
Method:	Least Squares	F-statistic:	501.4
Date:	Fri, 22 Dec 2023	Prob (F-statistic):	0.00
Time:	06:48:04	Log-Likelihood:	-22005.
No. Observations:	5734	AIC:	4.405e+04
Df Residuals:	5712	BIC:	4.420e+04
Df Model:	21		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	40.9187	0.745	54.948	0.000	39.459	42.379
lread	-0.0201	0.003	-6.311	0.000	-0.026	-0.014
lwrite	0.0075	0.006	1.270	0.204	-0.004	0.019
scall	0.0009	0.000	6.399	0.000	0.001	0.001
sread	0.0008	0.002	0.401	0.688	-0.003	0.005
swrite	-0.0014	0.002	-0.564	0.573	-0.006	0.003
fork	-1.9590	0.262	-7.487	0.000	-2.472	-1.446
exec	-0.0078	0.053	-0.147	0.883	-0.112	0.096
rchar	-3.58e-06	9.04e-07	-3.962	0.000	-5.35e-06	-1.81e-06
wchar	-8.736e-06	1.42e-06	-6.131	0.000	-1.15e-05	-5.94e-06
pgout	-0.1924	0.063	-3.041	0.002	-0.316	-0.068
ppgout	0.1467	0.037	4.004	0.000	0.075	0.219
pgfree	-0.1051	0.019	-5.551	0.000	-0.142	-0.068
pgscan	0.0161	0.006	2.843	0.004	0.005	0.027
atch	-0.0522	0.024	-2.175	0.030	-0.099	-0.005
pgin	0.0578	0.029	1.972	0.049	0.000	0.115
ppgin	-0.0381	0.019	-2.019	0.044	-0.075	-0.001
pflt	-0.0396	0.004	-9.150	0.000	-0.048	-0.031
vflt	0.0225	0.003	6.544	0.000	0.016	0.029
freemem	-0.0018	7.85e-05	-22.579	0.000	-0.002	-0.002
freeswap	3.45e-05	4.58e-07	75.394	0.000	3.36e-05	3.54e-05
runqsz_Not_CPU_Bound	8.2531	0.315	26.229	0.000	7.636	8.870

Omnibus:	1100.363	Durbin-Watson:	2.016
Prob(Omnibus):	0.000	Jarque-Bera (JB):	2515.169
Skew:	-1.090	Prob(JB):	0.00
Kurtosis:	5.403	Cond. No.	7.14e+06

Table 2. OLS Regression Results for given model dataset

Interpretation of R-squared

The R-squared value tells us that our model can explain 64.8% of the variance in the training set.

Assumptions for the linear regression model

1.Check for multicollinearity using VIF

```
VIF values:

const                  25.112715
lread                   1.433961
lwrite                  1.393485
scall                   2.644729
sread                   5.622068
swrite                  4.858065
fork                   19.204162
exec                    3.493582
rchar                   2.027391
wchar                   1.713692
pgout                   5.034523
ppgout                 13.649128
pgfree                 16.833355
pgscan                  7.250087
atch                    1.115113
pgin                    7.468946
ppgin                   8.090578
pflt                   11.178061
vflt                   19.594407
freemem                 1.696487
freeswap                1.728523
runqsz_Not_CPU_Bound    1.117571
dtype: float64
```

The VIF values indicate that the features sread fork pgout ppgout pgfree pgscan pgin ppgin pflt vflt swrite are correlated with one or more independent features.
Multicollinearity affects only the specific independent variables that are correlated.
Therefore, trust the p-values of the specific variables.
To treat multicollinearity, we will have to drop one or more of the correlated features.
Will drop the variable that has least impact on adjusted R-squared of the model. This process continues until VIF <2(TOTALLY 17 iterations done to attain the value) and observe the effect on our predictive model.

```
VIF values:

const                  23.247591
lread                   1.414675
lwrite                  1.376778
scall                   1.697874
exec                    1.799440
rchar                   1.636976
wchar                   1.457106
pgout                   1.575059
pgscan                  1.744268
atch                    1.080401
pgin                    1.561030
pflt                    2.179444
freemem                 1.680253
freeswap                1.614479
runqsz_Not_CPU_Bound    1.109176
dtype: float64
```

Final OLS regression results as shown in the below image

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.641
Model:	OLS	Adj. R-squared:	0.641
Method:	Least Squares	F-statistic:	1024.
Date:	Fri, 22 Dec 2023	Prob (F-statistic):	0.00
Time:	06:48:10	Log-Likelihood:	-22060.
No. Observations:	5734	AIC:	4.414e+04
Df Residuals:	5723	BIC:	4.422e+04
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	42.5801	0.709	60.041	0.000	41.190	43.970
lread	-0.0170	0.003	-6.219	0.000	-0.022	-0.012
scall	0.0011	0.000	9.303	0.000	0.001	0.001
exec	-0.1472	0.038	-3.859	0.000	-0.222	-0.072
rchar	-3.114e-06	7.96e-07	-3.915	0.000	-4.67e-06	-1.55e-06
wchar	-9.939e-06	1.31e-06	-7.582	0.000	-1.25e-05	-7.37e-06
pgout	-0.1274	0.030	-4.180	0.000	-0.187	-0.068
pflt	-0.0427	0.002	-22.212	0.000	-0.046	-0.039
freemem	-0.0018	7.87e-05	-22.421	0.000	-0.002	-0.002
freeswap	3.37e-05	4.41e-07	76.344	0.000	3.28e-05	3.46e-05
runqsz_Not_CPU_Bound	8.2238	0.316	26.014	0.000	7.604	8.844

Omnibus:	1156.491	Durbin-Watson:	2.020
Prob(Omnibus):	0.000	Jarque-Bera (JB):	2711.188
Skew:	-1.132	Prob(JB):	0.00
Kurtosis:	5.495	Cond. No.	6.75e+06

Table 3. OLS Regression Results for VIF <2

Now, removed all the p-values > 0.05. 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 0.647 to 0.641).
This shows that these variables did not have much predictive power.

Assumptions of Linear Regression.

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

Check for below assumptions of Linear Regression,

1. Linearity
2. Independence
3. Homoscedasticity
4. Normality of error terms
5. No strong Multicollinearity

	Actual Values	Fitted Values	Residuals
0	95	100.855575	-5.855575
1	58	51.774323	6.225677
2	92	88.750448	3.249552
3	87	105.293648	-18.293648
4	88	88.642581	-0.642581

Test for linearity and independence

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

How to check linearity? Make a plot of fitted values vs residuals. If they don't follow any pattern (the curve is a straight line), then assume that model is linear otherwise model is showing signs of non-linearity.

How to fix if this assumption is not followed? Try with different transformations.

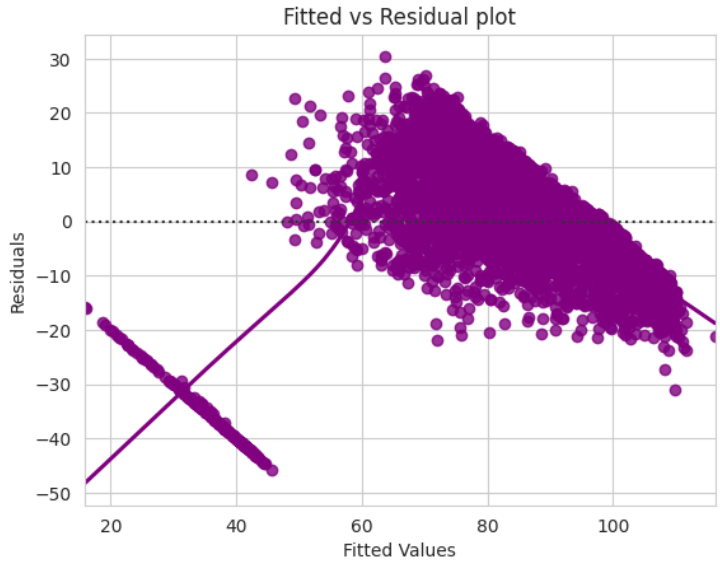


Fig 11. Fitted vs Residual plot

observe a pattern in the residual vs fitted values, hence will try to transform the continuous variables in the data.

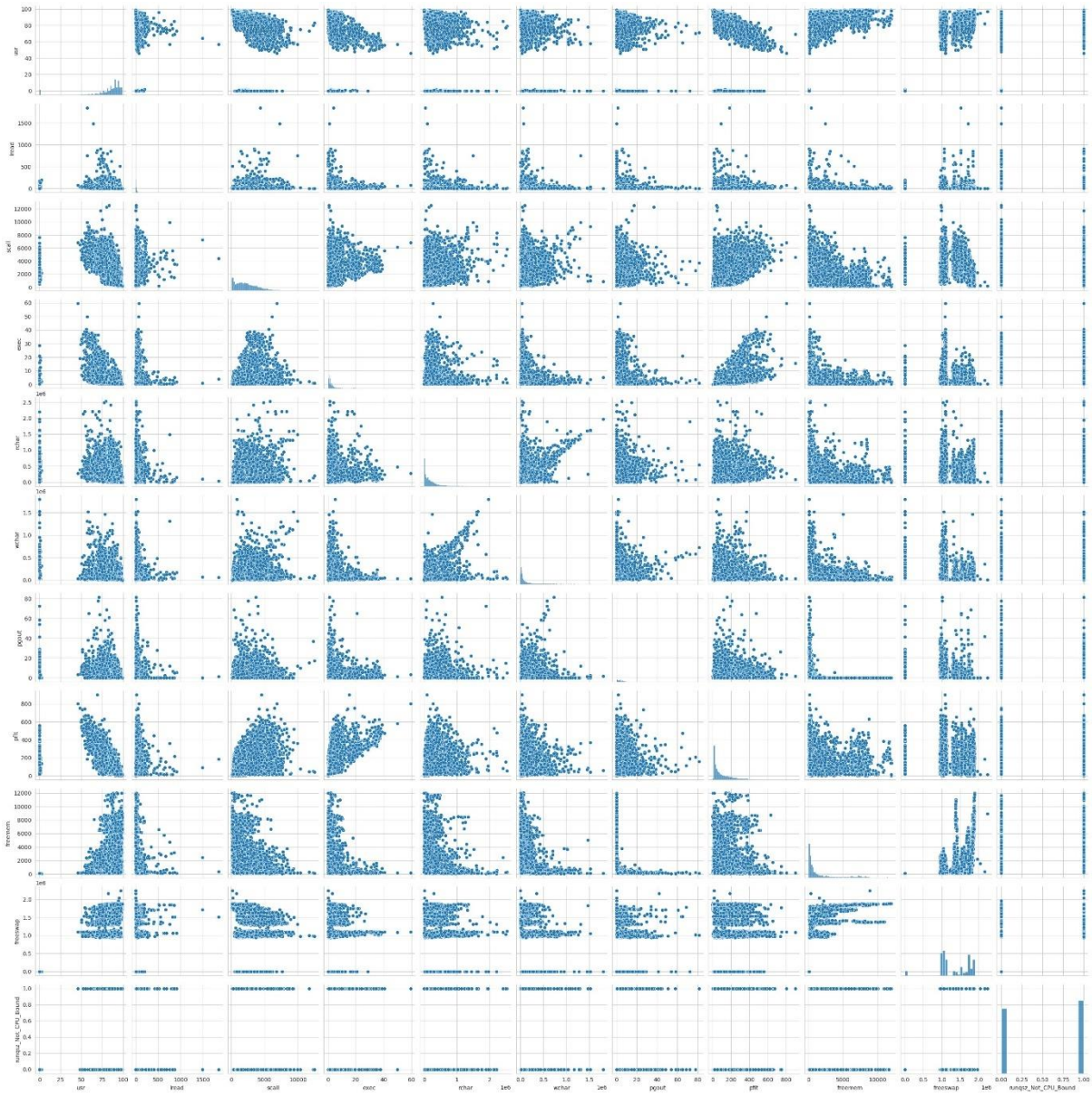


Fig 12. Pair plot - Distribution of variable in training set

Transformation y data can be fixed:

Initially iterated with small values but not influencing in the model performance. Training data of y will multiply with power of 5.

After transformation OLS regression results as shown in the below image

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.791
Model:	OLS	Adj. R-squared:	0.790
Method:	Least Squares	F-statistic:	2163.
Date:	Fri, 22 Dec 2023	Prob (F-statistic):	0.00
Time:	07:14:47	Log-Likelihood:	-1.2751e+05
No. Observations:	5734	AIC:	2.551e+05
Df Residuals:	5723	BIC:	2.551e+05
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	4.301e+09	6.88e+07	62.466	0.000	4.17e+09	4.44e+09
lread	-3.046e+06	2.66e+05	-11.472	0.000	-3.57e+06	-2.53e+06
scall	-2.913e+05	1.16e+04	-25.089	0.000	-3.14e+05	-2.69e+05
exec	-3.041e+07	3.7e+06	-8.211	0.000	-3.77e+07	-2.32e+07
rchar	-879.3866	77.237	-11.386	0.000	-1030.800	-727.973
wchar	-1291.9625	127.268	-10.151	0.000	-1541.456	-1042.469
pgout	-3.78e+07	2.96e+06	-12.777	0.000	-4.36e+07	-3.2e+07
pflt	-8.77e+06	1.87e+05	-47.015	0.000	-9.14e+06	-8.4e+06
freemem	-2.841e+04	7644.471	-3.717	0.000	-4.34e+04	-1.34e+04
freeswap	2192.5030	42.847	51.170	0.000	2108.506	2276.500
runqsz_Not_CPU_Bound	5.395e+08	3.07e+07	17.579	0.000	4.79e+08	6e+08

Omnibus:	318.467	Durbin-Watson:	1.960
Prob(Omnibus):	0.000	Jarque-Bera (JB):	422.005
Skew:	-0.526	Prob(JB):	2.31e-92
Kurtosis:	3.813	Cond. No.	6.75e+06

Fig 13. Pair plot of distribution of variable in training set

Adj. R-squared has increased from 64.1 to 79.1

Reiterate the fitted and residuals

	Actual Values	Fitted Values	Residuals
0	7737809375	7.178093e+09	5.597160e+08
1	656356768	-1.107059e+09	1.763416e+09
2	6590815232	6.518566e+09	7.224886e+07
3	4984209207	7.395780e+09	-2.411571e+09
4	5277319168	6.575776e+09	-1.298456e+09

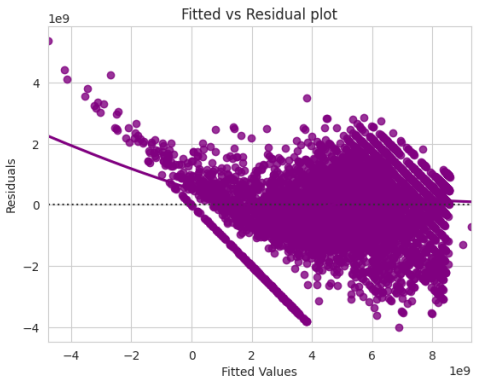


Fig 13. Fitted Vs Residual after Transformation

Test for normality

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

What does non-normality indicate?
It suggests that there are a few unusual data points which must be studied closely to make a better model.

How to check the Normality?
It can be checked via QQ Plot - residuals following normal distribution will make a straight-line plot, otherwise another test to check for normality is Shapiro-Wilk test.

How to Make residuals normal?
Apply transformations like log, exponential, arcsinh, etc as per our data.

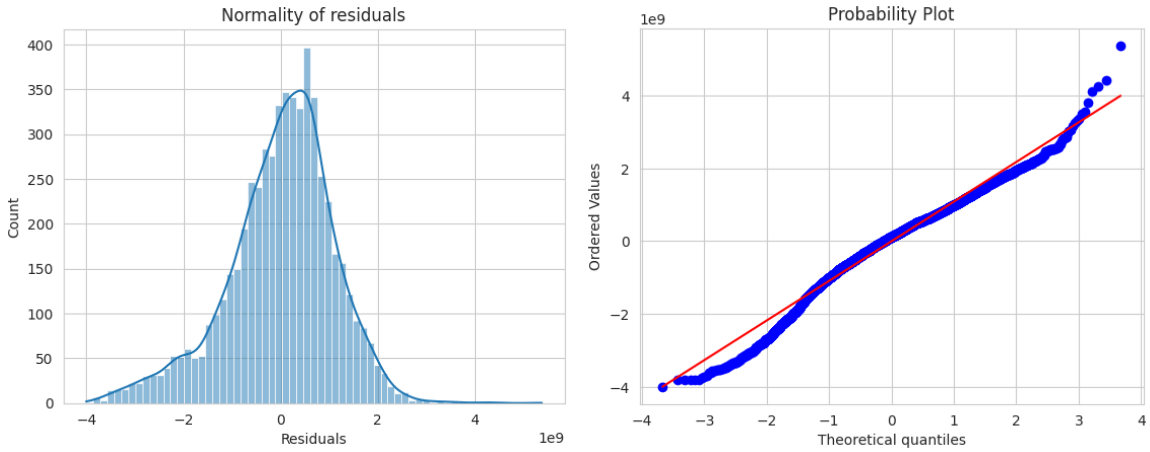


Fig 13. Normal distribution and QQ plot

Many points are lying on the straight line in QQ plot

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

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

Since p-value < 0.05, the residuals are not normal according to Shapiro test. Might be willing to accept this distribution as close to being normal.

Test for homoscedasticity

[('F statistic', 0.9969808068069038), ('p-value', 0.5321955741682427)]

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

All the assumptions of linear regression are now satisfied. Let's check the summary of our final model.

OLS Regression Results

Dep. Variable:	usr	R-squared:	0.791
Model:	OLS	Adj. R-squared:	0.790
Method:	Least Squares	F-statistic:	2163.
Date:	Fri, 22 Dec 2023	Prob (F-statistic):	0.00
Time:	07:17:28	Log-Likelihood:	-1.2751e+05
No. Observations:	5734	AIC:	2.551e+05
Df Residuals:	5723	BIC:	2.551e+05
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	4.301e+09	6.88e+07	62.466	0.000	4.17e+09	4.44e+09
lread	-3.046e+06	2.66e+05	-11.472	0.000	-3.57e+06	-2.53e+06
scall	-2.913e+05	1.16e+04	-25.089	0.000	-3.14e+05	-2.69e+05
exec	-3.041e+07	3.7e+06	-8.211	0.000	-3.77e+07	-2.32e+07
rchar	-879.3866	77.237	-11.386	0.000	-1030.800	-727.973
wchar	-1291.9625	127.268	-10.151	0.000	-1541.456	-1042.469
pgout	-3.78e+07	2.96e+06	-12.777	0.000	-4.36e+07	-3.2e+07
pflt	-8.77e+06	1.87e+05	-47.015	0.000	-9.14e+06	-8.4e+06
freemem	-2.841e+04	7644.471	-3.717	0.000	-4.34e+04	-1.34e+04
freeswap	2192.5030	42.847	51.170	0.000	2108.506	2276.500
runqsz_Not_CPU_Bound	5.395e+08	3.07e+07	17.579	0.000	4.79e+08	6e+08

Omnibus:	318.467	Durbin-Watson:	1.960
Prob(Omnibus):	0.000	Jarque-Bera (JB):	422.005
Skew:	-0.526	Prob(JB):	2.31e-92
Kurtosis:	3.813	Cond. No.	6.75e+06

Table 4. OLS Regression after 18th model

Observations R-squared of the model is 0.791 and adjusted R-squared is 0.790, which shows that the model is able to explain ~79% variance in the data. This is good enough.

Prediction of the model

```
RMSE for train OLS :  1100864151.77311      RMSE for test OLS :  1065854574.2752588
MAE for train OLS  :  844951691.0657396     MAE for test OLS  :  815522254.6165146
R2 for train OLS   :  0.7907720025504168    R2 for test OLS   :  0.8026337457755653
```

RMSE on the train and test sets are comparable. So, model is not suffering from overfitting. R square for train and test are comparable. R square of resulting of 80.2% for the test condition.

Coefficient of all attributes as shown in the below image

```
const                  4.300634e+09
lread                 -3.046103e+06
scall                 -2.912822e+05
exec                  -3.041464e+07
rchar                 -8.793866e+02
wchar                 -1.291962e+03
pgout                 -3.780059e+07
pflt                  -8.769882e+06
freemem               -2.841261e+04
freeswap               2.192503e+03
runqsz_Not_CPU_Bound   5.395169e+08
dtype: float64
```

The equation of linear regression model as shown below

```
usr = 4300634224.578651 + -3046103.4387196465 * ( lread ) +  -
291282.2177427411 * ( scall ) +  -30414643.722644746 * ( exec ) +  -
879.3865900024477 * ( rchar ) +  -1291.9624959003754 * ( wchar ) +  -
37800588.771157466 * ( pgout ) +  -8769882.36471764 * ( pflt ) +  -
28412.607429342264 * ( freemem ) +  2192.5030092386014 * ( freeswap ) +
539516947.605248 * ( runqsz_Not_CPU_Bound )
```

Conclusion & Recommendation

Model undergoes all the checks as follow Linearity, Independence, Homoscedasticity, Normality of error terms, No Multicollinearity.

Observations R-squared of the model is 0.791 and adjusted R-squared is 0.790, which shows that the model is able to explain ~79% variance in the data. It's not require to validate further.

The multicollinearity between the dependent variables have been removed and only such variables, whose VIF (Variance Inflation Factor) < 2 have been considered to check the performance of model using linear regression.

18 models have been built and used for validation

The RMSE on the train and test sets are comparable. Therefore, the model is not suffering from overfitting.

The R square values for train and test are comparable. R square increased to 80.2%

Scale of RMSE and MAE depends on the target variable, so scaled the target variable by a power of 5 i.e expected to get higher values of RMSE and MAE.

Hence, "ols_res18" is good for prediction model and inference purposes.

THE END