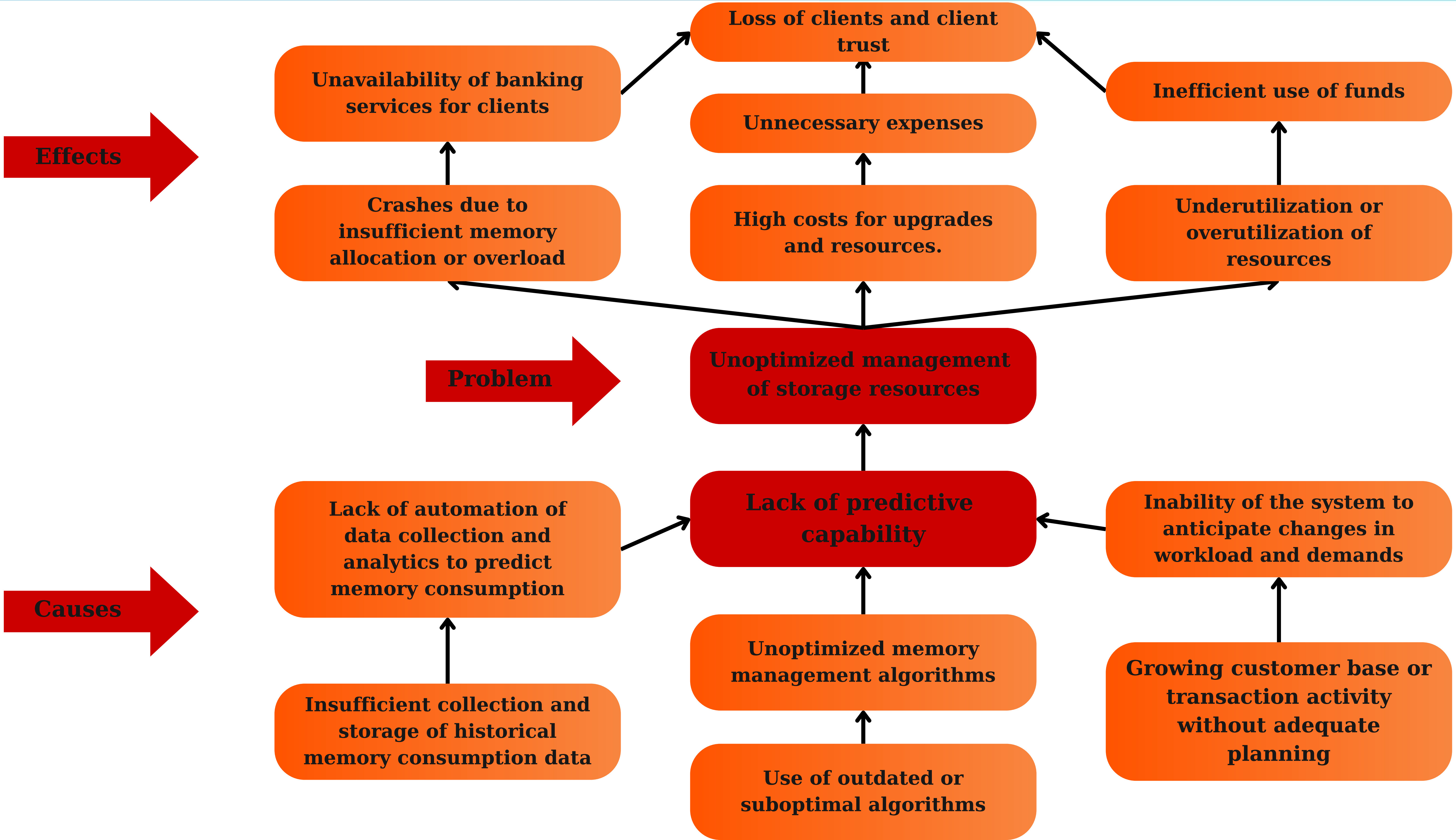
UNIVERSITATEA TEHNICĂ A MOLDOVEI

Banking Systems: Forecasting of Data Storage

It’s all about size

Deliu Maria MI-213

Problem


Effects
Problem
Causes
Loss of clients and client trust
Unavailability of banking services for clients
Unnecessary expenses
Inefficient use of funds
Crashes due to insufficient memory allocation or overload
High costs for upgrades and resources.
Underutilization or overutilization of resources
Unoptimized management of storage resources
Lack of predictive capability
Lack of automation of data collection and analytics to predict memory consumption
Inability of the system to anticipate changes in workload and demands
Unoptimized memory management algorithms
Insufficient collection and storage of historical memory consumption data
Growing customer base or transaction activity without adequate planning
Use of outdated or suboptimal algorithms

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

Data

FCIM

History

~~ID - observation id~~

~~DATE - observation date~~

~~SYSTEM id - system id~~

SIZE - system size in GB

LOAD_TPD - number of transactions

ACCOUNTS_ALL - number of accounts

ACCOUNTS_ACTIVE - number of active accounts

Non_kept_size - data sent to warehouse

Backup_size - size of system backup

LongOps_min - time of long operations in minutes

Kept_size - size of system after sending data to retention

Backup_Efficiency - how much of original data is backed up

Systems

~~ID - observation id~~

~~NAME - system name~~

Stage - system production stage

~~Description - system description~~

Type - system type

Size, Gb - current size in GB

Data Keep, years - number of years that data is kept online

Backup retention, month - retention of backups in months

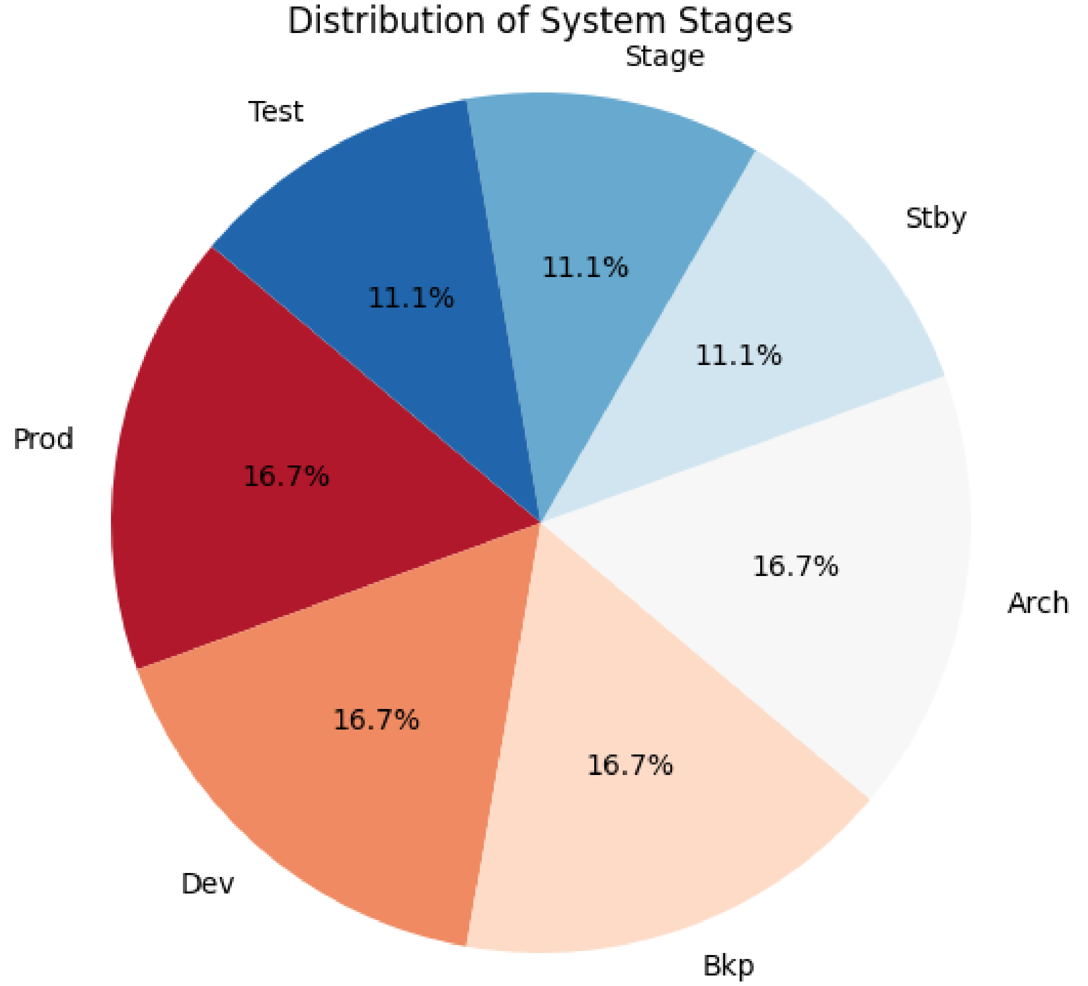
Depreciation period, years - when data becomes outdated

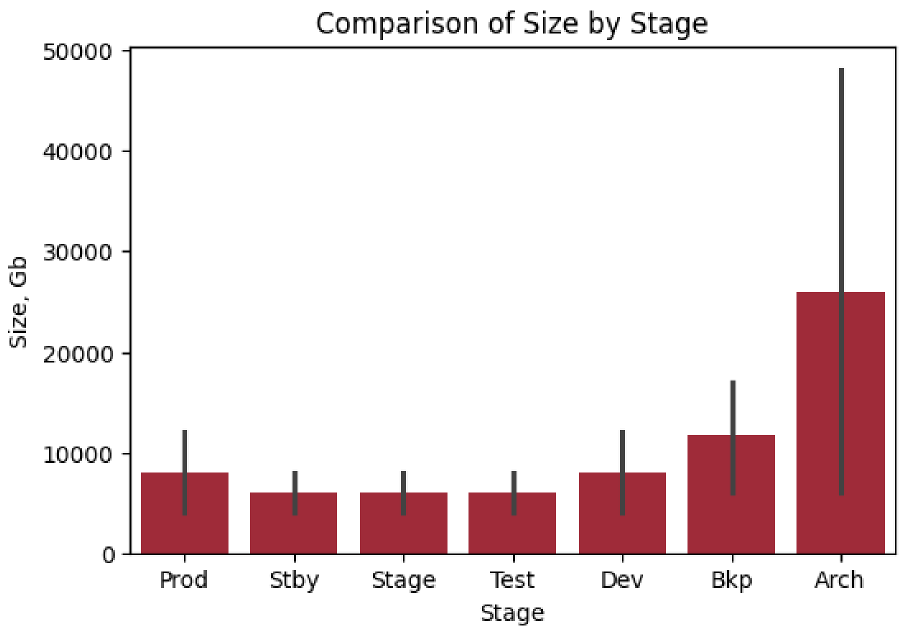
Data retention, years - retention of data in years

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

FCIM

Distribution





1. **Dev** - **Initial Developement stage**
2. **Test** - **Testing stage**
3. **Stage** - **Production imitation (Staging)**
4. **Prod** - **Production**
5. **Stby** - **Standby, backup enviroment in case of failures**
6. **Bkp** - **Backup**
7. **Arch** - **Archives**

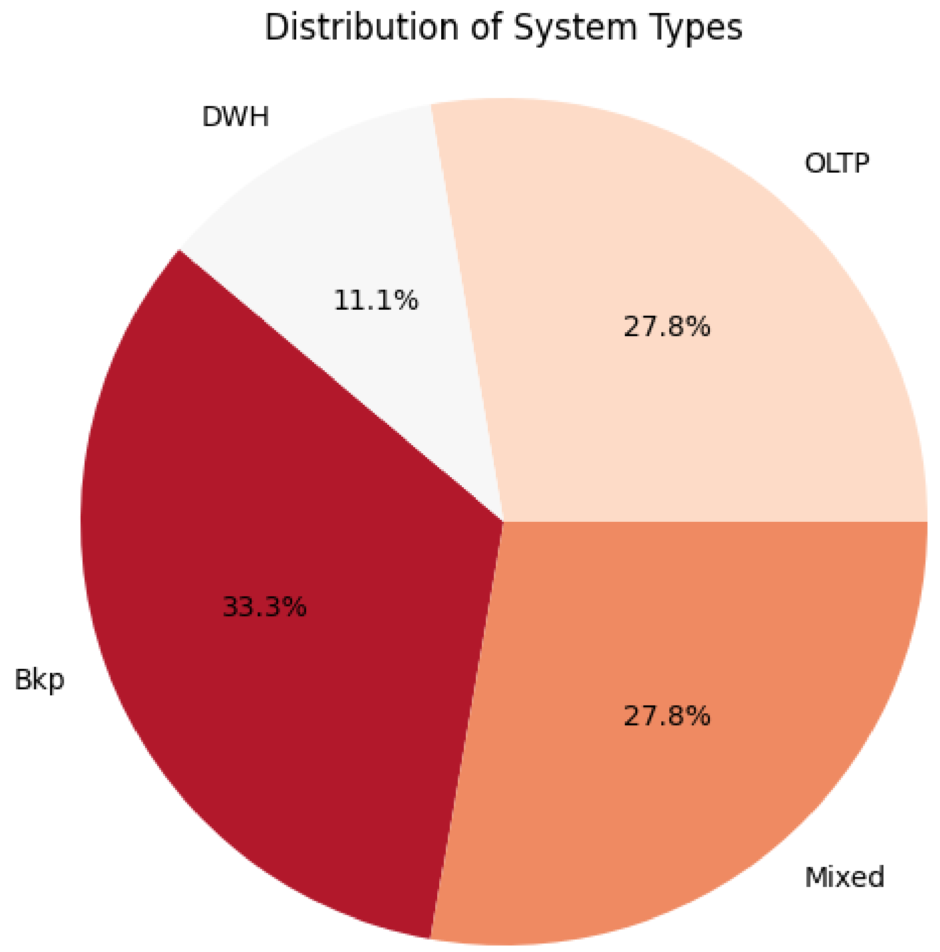
Size: The archives and backups take a lot of space with testing and staging taking the least space

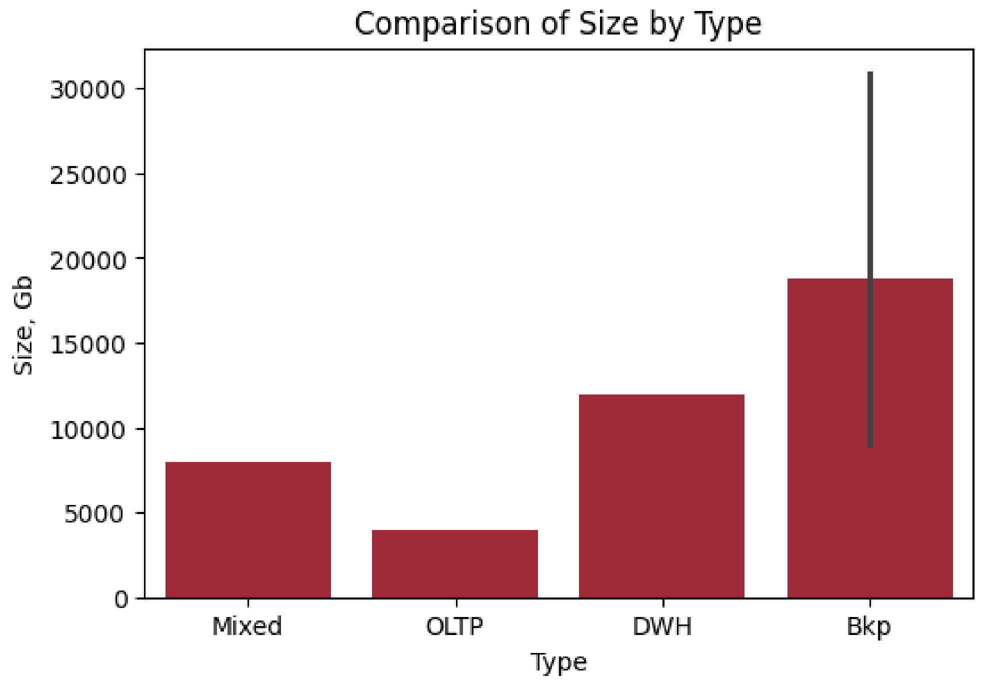
OLTP - **Online Transaction Processing**

DWH - **Data Warehouse**

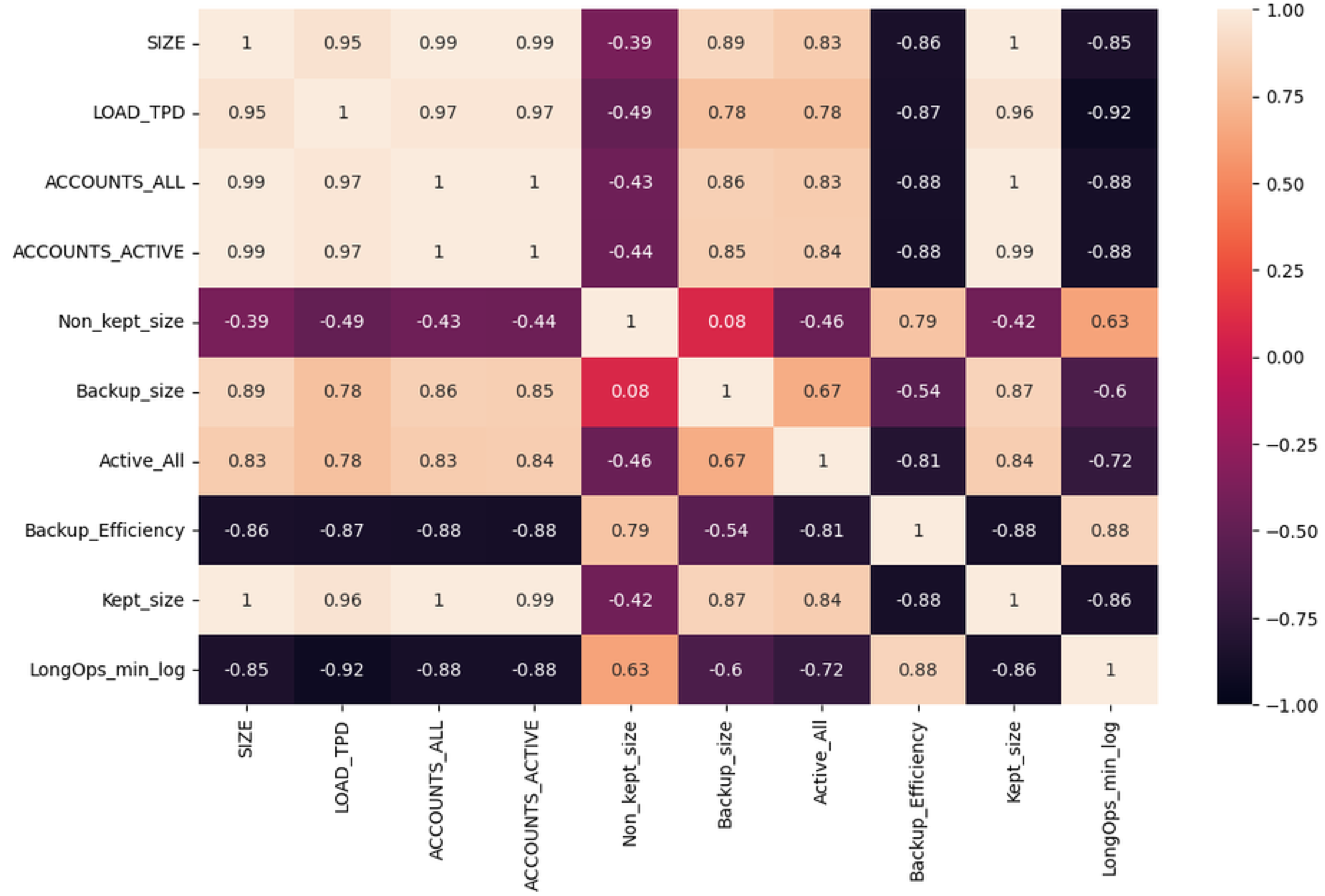
Bkp - **Backup**

Mixed - **Both OLTP and DWH**



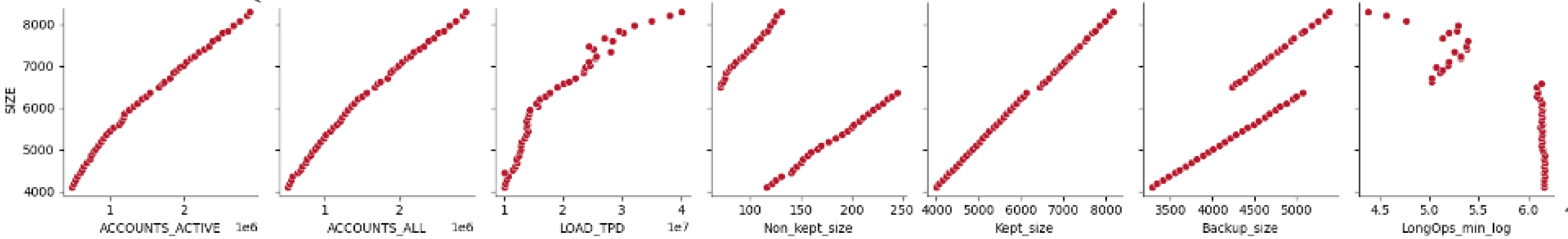


Correlation



	SIZE	LOAD_TPD	ACCOUNTS_ALL	ACCOUNTS_ACTIVE	Non_kept_size	Backup_size	Active_All	Backup_Efficiency	Kept_size	LongOps_min_log
SIZE	1	0.95	0.99	0.99	-0.39	0.89	0.83	-0.86	1	-0.85
LOAD_TPD	0.95	1	0.97	0.97	-0.49	0.78	0.78	-0.87	0.96	-0.92
ACCOUNTS_ALL	0.99	0.97	1	1	-0.43	0.86	0.83	-0.88	1	-0.88
ACCOUNTS_ACTIVE	0.99	0.97	1	1	-0.44	0.85	0.84	-0.88	0.99	-0.88
Non_kept_size	-0.39	-0.49	-0.43	-0.44	1	0.08	-0.46	0.79	-0.42	0.63
Backup_size	0.89	0.78	0.86	0.85	0.08	1	0.67	-0.54	0.87	-0.6
Active_All	0.83	0.78	0.83	0.84	-0.46	0.67	1	-0.81	0.84	-0.72
Backup_Efficiency	-0.86	-0.87	-0.88	-0.88	0.79	-0.54	-0.81	1	-0.88	0.88
Kept_size	1	0.96	1	0.99	-0.42	0.87	0.84	-0.88	1	-0.86
LongOps_min_log	-0.85	-0.92	-0.88	-0.88	0.63	-0.6	-0.72	0.88	-0.86	1

Most of factors have high correlation rate with size, with highest being accounts and load



EDA

Conclusions

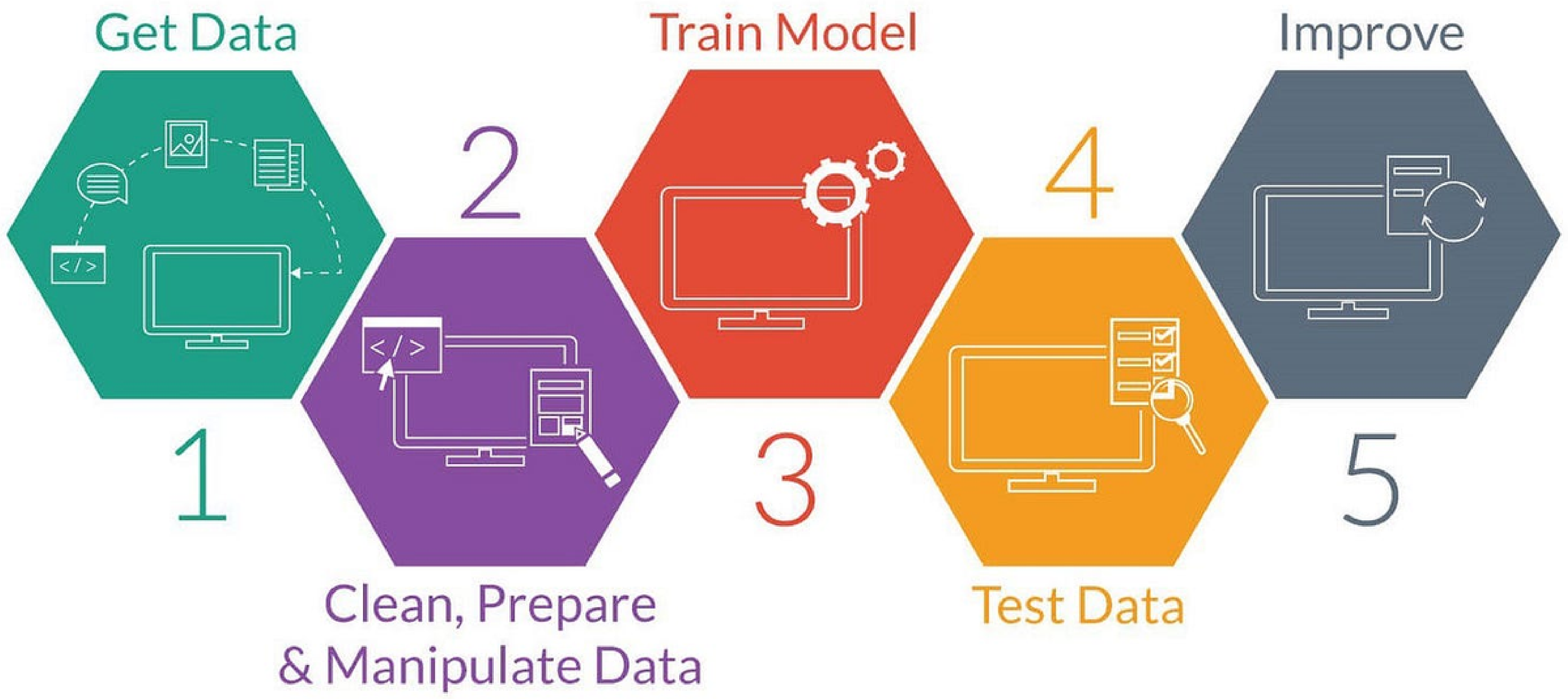
EDA helps understanding the dataset better, the attributes and their potential impact: ex. Size Variation across System Stages and Types

EDA helps identify patterns and relationships:
The correlation heatmap provided insights into the interdependence between variables, with Accounts and load playing a big role.

Problem: Lack of forecasting ability in banking systems to predict needed memory resources

Workflow

Selecting the right machine learning model for size prediction in banking systems necessitates a balance between accuracy, interpretability, and scalability. Considering factors such as data complexity, feature importance, and computational efficiency aids in identifying the optimal model



Pre-processing data

Original data - if all numerical data is in simial scale, we may use the data as is

Min-Max scaling - For every feature, the minimum value of that feature gets transformed into a 0, the maximum value gets transformed into a 1, and every other value gets transformed into a decimal between 0 and 1.

Z-score (Standardization) - A z-score, or standard score, is used for standardizing scores on the same scale by dividing a score's deviation by the standard deviation in a data set.

One-hot encodig - One hot encoding is a technique that we use to represent categorical variables as numerical values in a machine learning model.

	Stage	Type	Size, Gb	Data Keep, years	Backup retention, month	Depreciation period, years	Data retention, years
0	Prod	Mixed	8000	2	1	5	7
1	Stby	Mixed	8000	2	0	0	0
2	Stage	Mixed	8000	2	0	0	0
3	Test	Mixed	8000	2	0	0	0
4	Dev	Mixed	8000	2	0	0	0

	Size, Gb	Stage_Arch	Stage_Bkp	Stage_Dev	Stage_Prod	Stage_Stage	Stage_Stby	Stage_Test	Type_Bkp	Type_DWH
0	8000	False	False	False	True	False	False	False	False	False
1	8000	False	False	False	False	False	True	False	False	False
2	8000	False	False	False	False	True	False	False	False	False
3	8000	False	False	False	False	False	False	True	False	False
4	8000	False	False	True	False	False	False	False	False	False

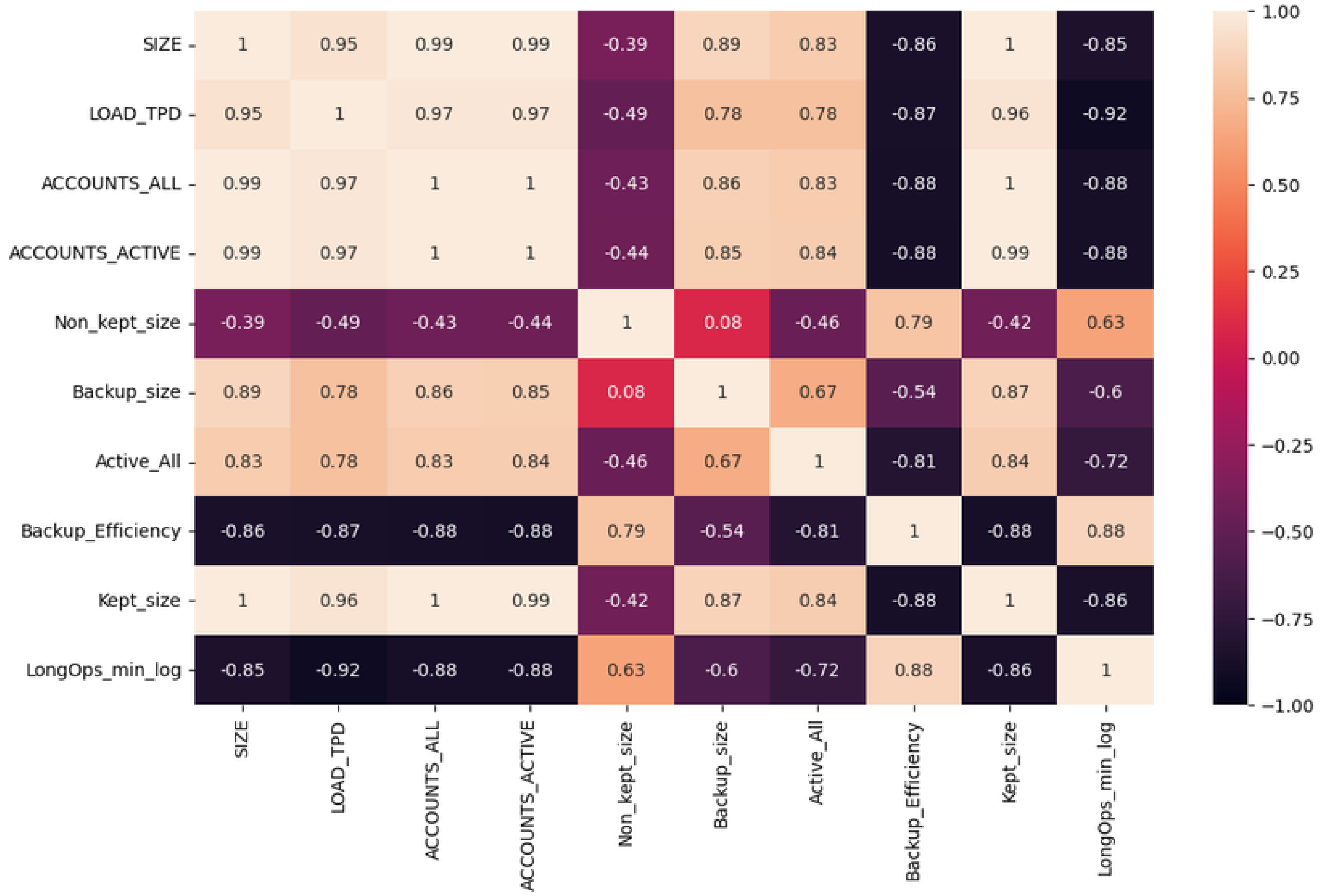
Feature Selection

Filetring Methods

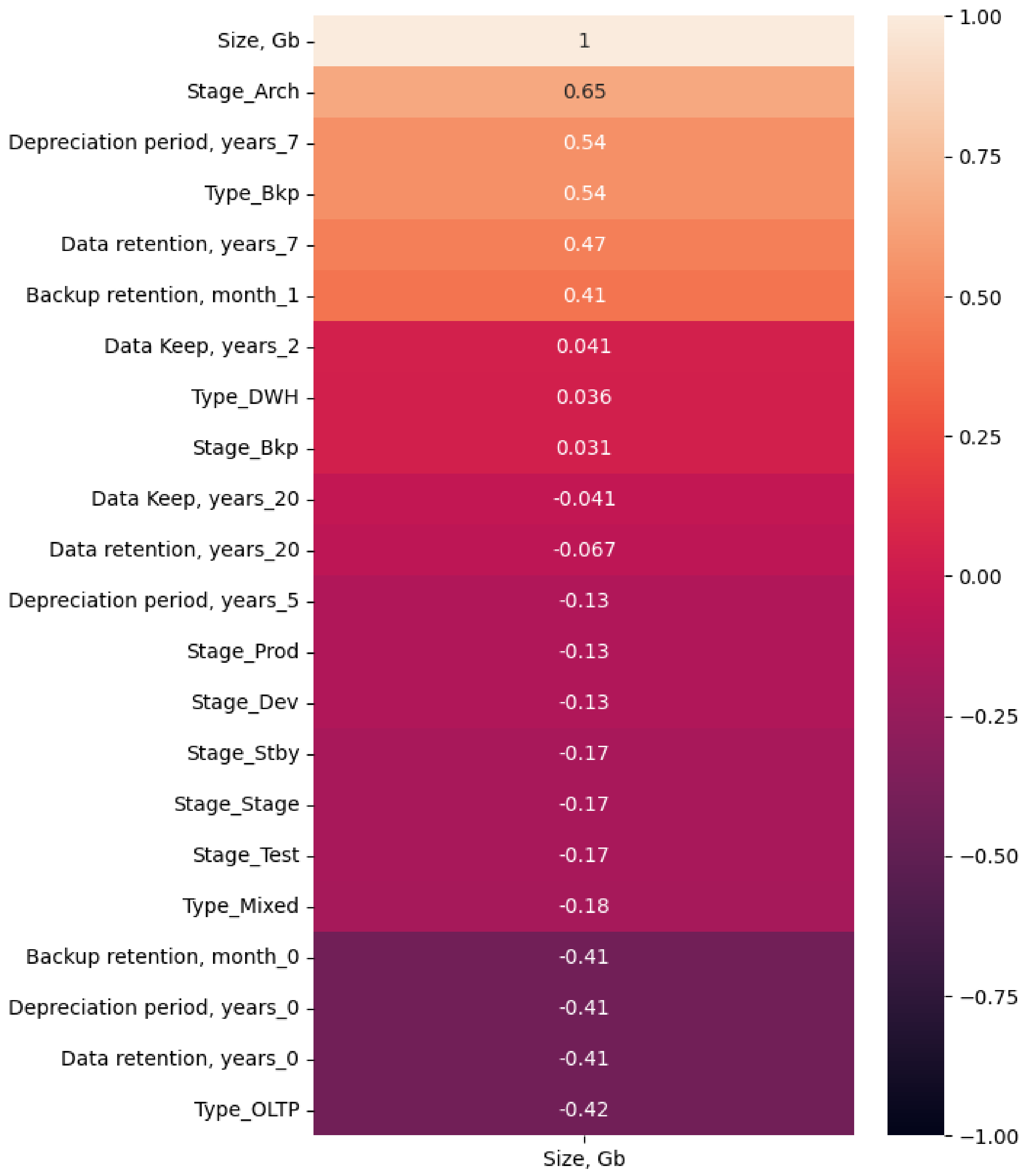
Filtering methods - these methods rank features based on certain criteria and select the most relevant ones.

Correlation Matrix - The correlation matrix helps in identifying features that are highly correlated with the target variable or with each other.

```
SIZE                 1.000000
Kept_size            0.999233
ACCOUNTS_ALL         0.994394
ACCOUNTS_ACTIVE      0.993276
LOAD_TPD             0.950841
Backup_size          0.888219
Active_All           0.834463
Non_kept_size       -0.386100
Backup_Efficiency   -0.863575
LongOps_min         -0.868219
```



We can use a correlation to examine the relationship between a numerical response variable (Y) and one-hot encoded categorical predictor variables (X). However, instead of correlation, other techniques can be used to explore these relationships



FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

Feature Selection

Filetring Methods

FCIM

Mutual Information

measures the dependency between two variables. In the context of feature selection, it measures the dependency between each feature and the target variable.

	Feature	Mutual_Information
8	Kept_size	2.507539
1	ACCOUNTS_ALL	2.442205
2	ACCOUNTS_ACTIVE	2.374539
3	Non_kept_size	2.206586
0	LOAD_TPD	1.773067
4	Backup_size	1.634102
6	Active_All	1.013875
5	LongOps_min	0.907988
7	Backup_Efficiency	0.677698

	Feature	Mutual_Information
10	Type_OLTP	0.570966
9	Type_Mixed	0.409793
7	Type_Bkp	0.310595
12	Data Keep, years_20	0.215579
17	Depreciation period, years_7	0.178916
13	Backup retention, month_0	0.125105
15	Depreciation period, years_0	0.119153
18	Data retention, years_0	0.115846
14	Backup retention, month_1	0.113862
3	Stage_Prod	0.110844
8	Type_DWH	0.091077
0	Stage_Arch	0.068194
1	Stage_Bkp	0.056263
4	Stage_Stage	0.044917
11	Data Keep, years_2	0.041137
19	Data retention, years_7	0.026873
20	Data retention, years_20	0.020820
6	Stage_Test	0.000000
5	Stage_Stby	0.000000
16	Depreciation period, years_5	0.000000
2	Stage_Dev	0.000000

SelectKBest

selects the K most informative features based on statistical tests like ANOVA, chi-squared, or mutual information

```
Selected Features:
Index(['LOAD_TPD', 'ACCOUNTS_ALL', 'ACCOUNTS_ACTIVE', 'Backup_size',
       'Kept_size'],
      dtype='object')
```

```
Index(['Stage_Arch', 'Type_Bkp', 'Type_OLTP', 'Depreciation period, years_7',
       'Data retention, years_7'],
      dtype='object')
```

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

Feature Selection

Wrapper and Tree-based Methods

FCIM

Wrapper Methods

These methods select subsets of features based on the performance of a specific machine learning algorithm.

Recursive Feature Elimination (RFE)

It works by recursively fitting the model and eliminating the least significant features based on their importance ranking

```
Index(['ACCOUNTS_ALL', 'Non_kept_size', 'Backup_size',
'Backup_Efficiency',
       'Kept_size'],
      dtype='object')
```

```
Index(['Stage_Arch', 'Type_DWH', 'Type_Mixed', 'Type_OLTP',
       'Data Keep, years_2'],
      dtype='object')
```

Random Forest Feature Importance (Tree-based Method)

Random Forests can measure feature importance by analyzing how much each feature contributes to decreasing impurity (Gini/entropy) in decision trees within the forest.

	Feature	Importance
1	ACCOUNTS_ALL	0.269816
0	LOAD_TPD	0.229871
2	ACCOUNTS_ACTIVE	0.179556
8	Kept_size	0.152755
6	Active_All	0.044308
7	Backup_Efficiency	0.036282
3	Non_kept_size	0.033193
5	LongOps_min	0.030932
4	Backup_size	0.023287

	Feature	Importance
0	Stage_Arch	0.550354
10	Type_OLTP	0.176526
19	Data retention, years_7	0.066221
7	Type_Bkp	0.043870
1	Stage_Bkp	0.039255
8	Type_DWH	0.031555
17	Depreciation period, years_7	0.029509
9	Type_Mixed	0.027740
11	Data Keep, years_2	0.010710
3	Stage_Prod	0.005689
20	Data retention, years_20	0.005473
16	Depreciation period, years_5	0.005208
13	Backup retention, month_0	0.003724
12	Data Keep, years_20	0.003005
14	Backup retention, month_1	0.000576
15	Depreciation period, years_0	0.000408
18	Data retention, years_0	0.000178
6	Stage_Test	0.000000
5	Stage_Stby	0.000000
4	Stage_Stage	0.000000
2	Stage_Dev	0.000000

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

Feature Selection

Embedded Methods

FCIM

Lasso, Ridge, Elastic Net

These are regularization techniques used in linear models. They introduce penalties to the model's coefficients during training by shrinking or eliminating the coefficients of less important features.

Lasso (L1 regularization) penalizes the absolute size of coefficients, effectively performing feature selection by shrinking some coefficients to zero.

Ridge (L2 regularization) penalizes the squared size of coefficients, limiting their overall size, but rarely setting them to zero.

Elastic Net combines both L1 and L2 regularization. It works well when you have a large number of features and/or some of them are correlated.

Lasso

```
Mean RMSE: 24.451901252388588
Standard Deviation of RMSE: 9.480103508120399

Dropped Features:
6           Active_All
7    Backup_Efficiency
Name: Feature, dtype: object

Kept Features:
0           LOAD_TPD
1       ACCOUNTS_ALL
2    ACCOUNTS_ACTIVE
3      Non_kept_size
4        Backup_size
5        LongOps_min
8          Kept_size
```

Ridge

```
Mean RMSE: 0.016868618076120584
Standard Deviation of RMSE: 0.007866987256212363

Ridge Coefficients:
             Feature   Coefficient
8          Kept_size  0.9994456656
3      Non_kept_size  0.9949121455
4        Backup_size  0.0008910016
5        LongOps_min  0.0000423301
1       ACCOUNTS_ALL  0.0000003031
0           LOAD_TPD -0.0000000016
2    ACCOUNTS_ACTIVE -0.0000000605
6         Active_All -0.0001913706
7  Backup_Efficiency -0.0003334751

Ridge Intercept:
-0.249002596920036
```

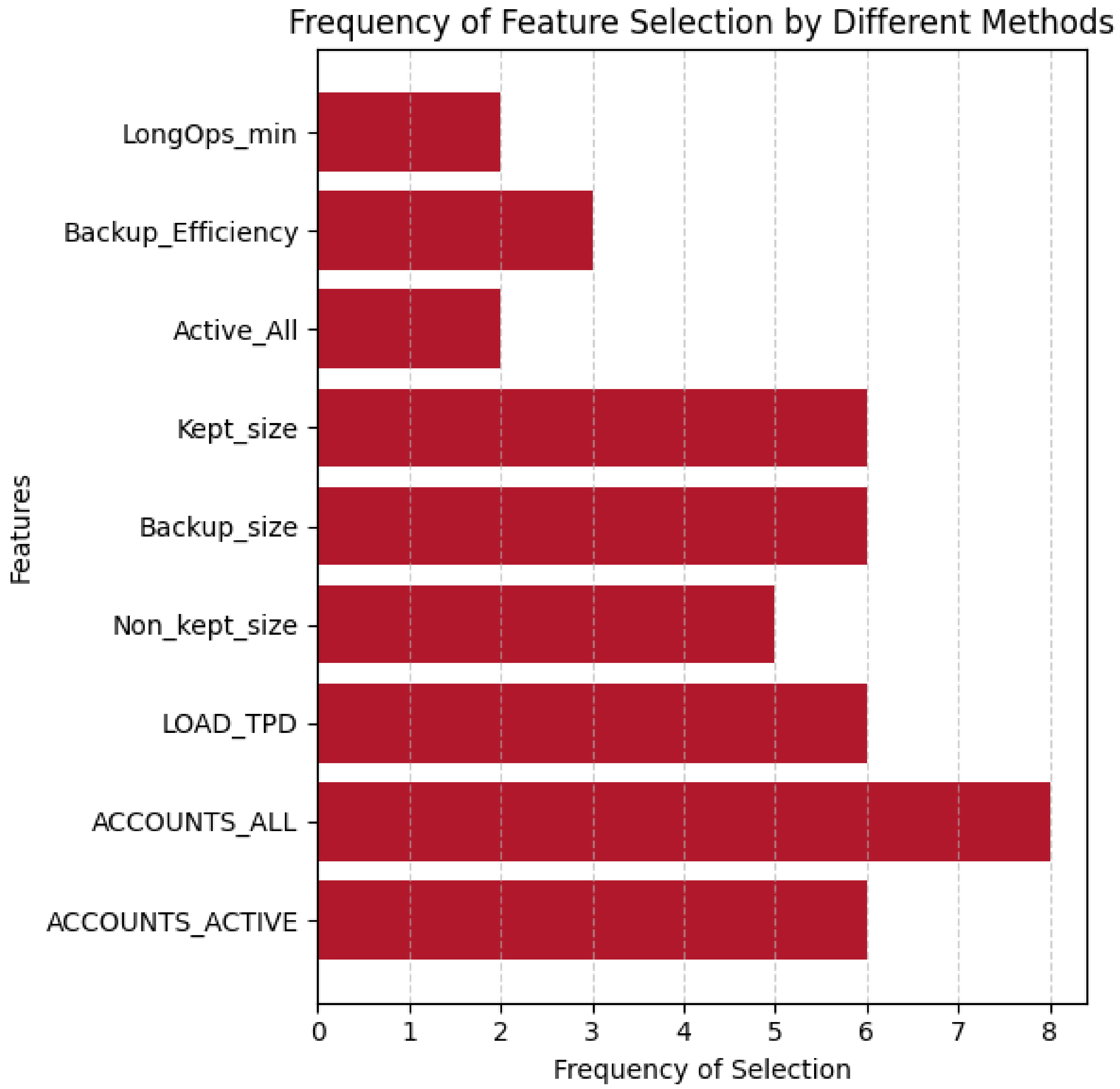
Elastic Net

```
Mean RMSE: 124.00440267745985
Standard Deviation of RMSE: 25.968418499514115

Elastic Net Alpha (Regularization Parameter):
16580600.000000002
Elastic Net L1 Ratio (Mixing Parameter): 0.5
Elastic Net Coefficients:
             Feature   Coefficient
1       ACCOUNTS_ALL  0.0014149449
2    ACCOUNTS_ACTIVE  0.0003410720
3      Non_kept_size  0.0000000000
4        Backup_size  0.0000000000
5        LongOps_min  0.0000000000
6         Active_All  0.0000000000
7  Backup_Efficiency -0.0000000000
8          Kept_size  0.0000000000
0           LOAD_TPD -0.0000053243
Elastic Net Intercept: 3598.2722311385273
```

Feature Selection

Conclusions



It's better to focus on accounts (all and active), on load, and backup or kept size

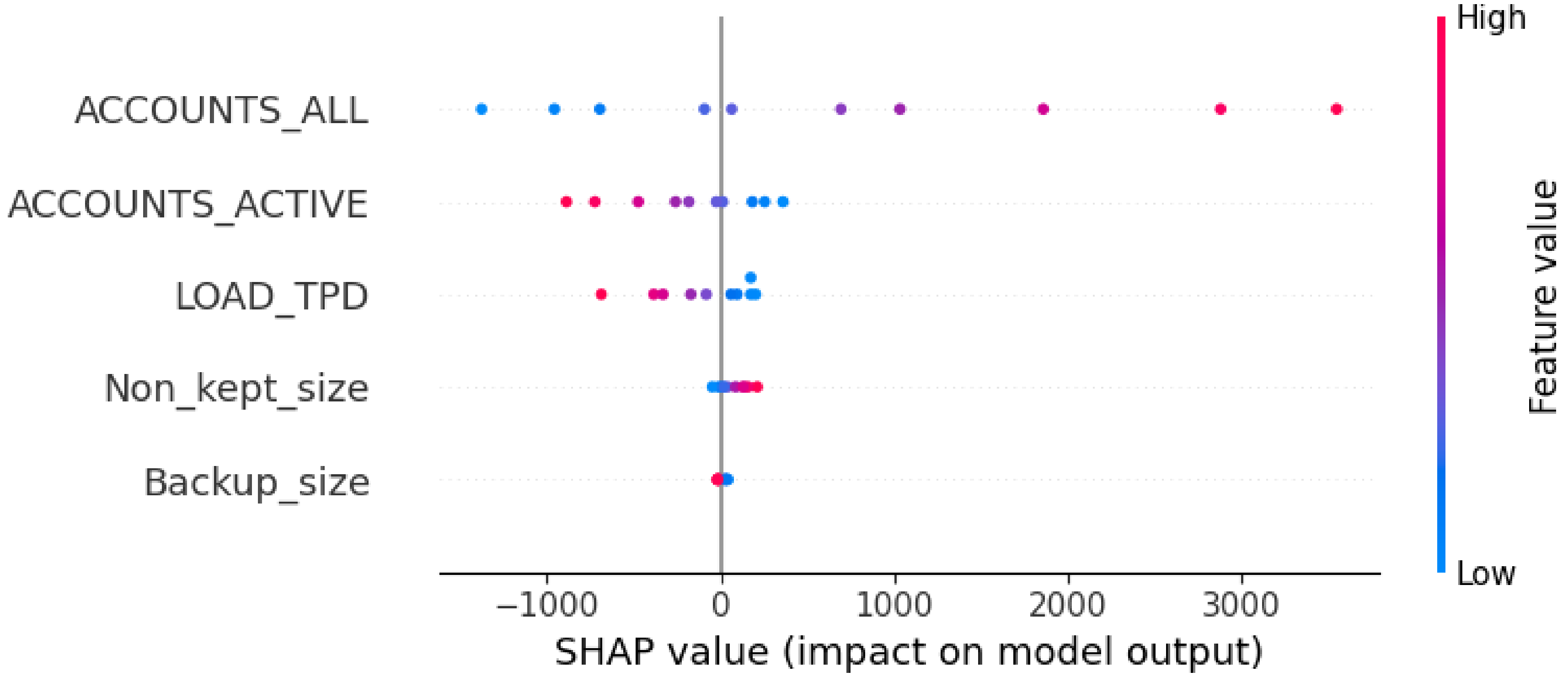
The general data about systems (types, stages) may not be appropriate for predictions

Models should be tested on both selected and all features for comparison

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

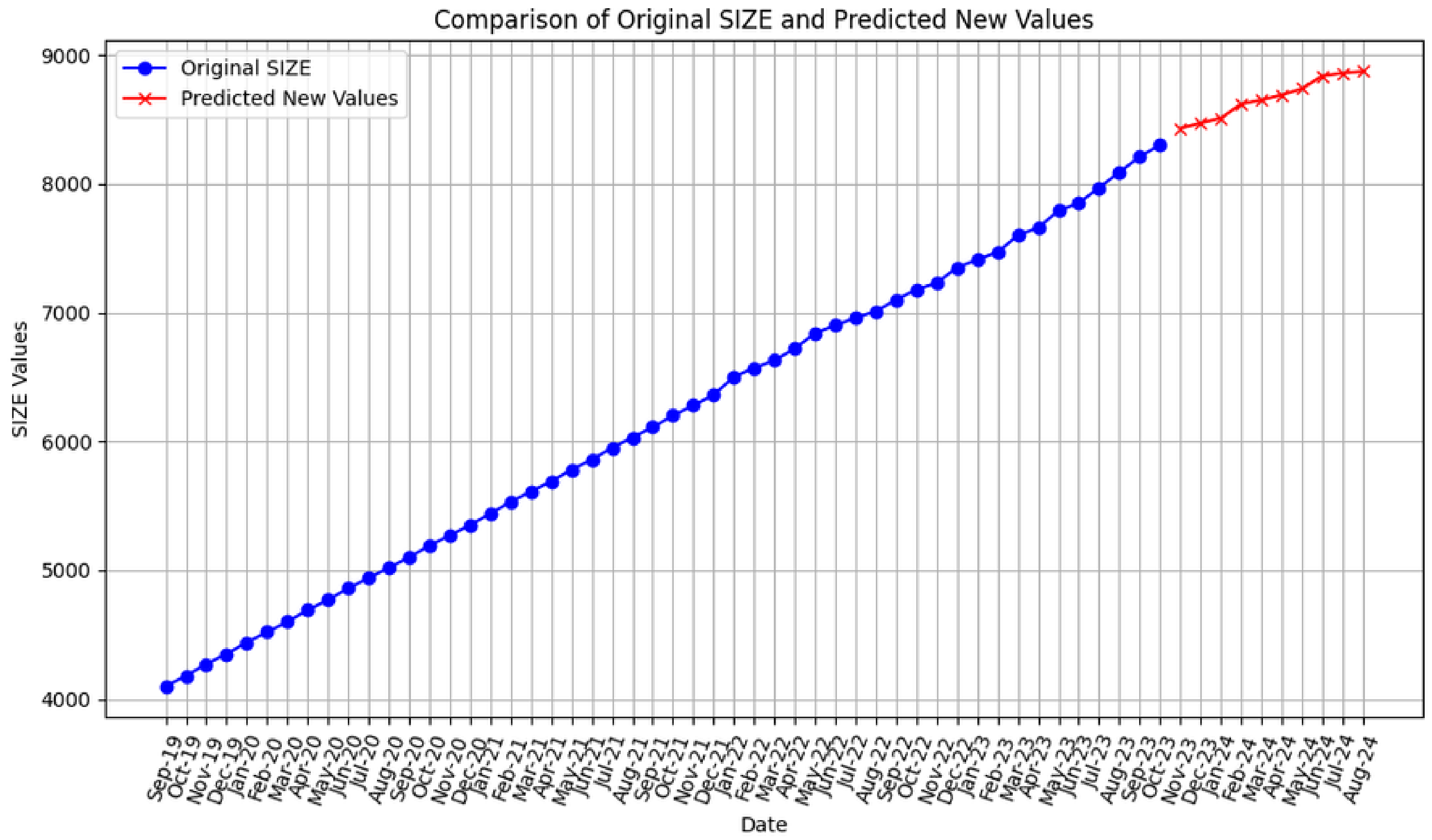
Multiple Linear Regression (MLR)

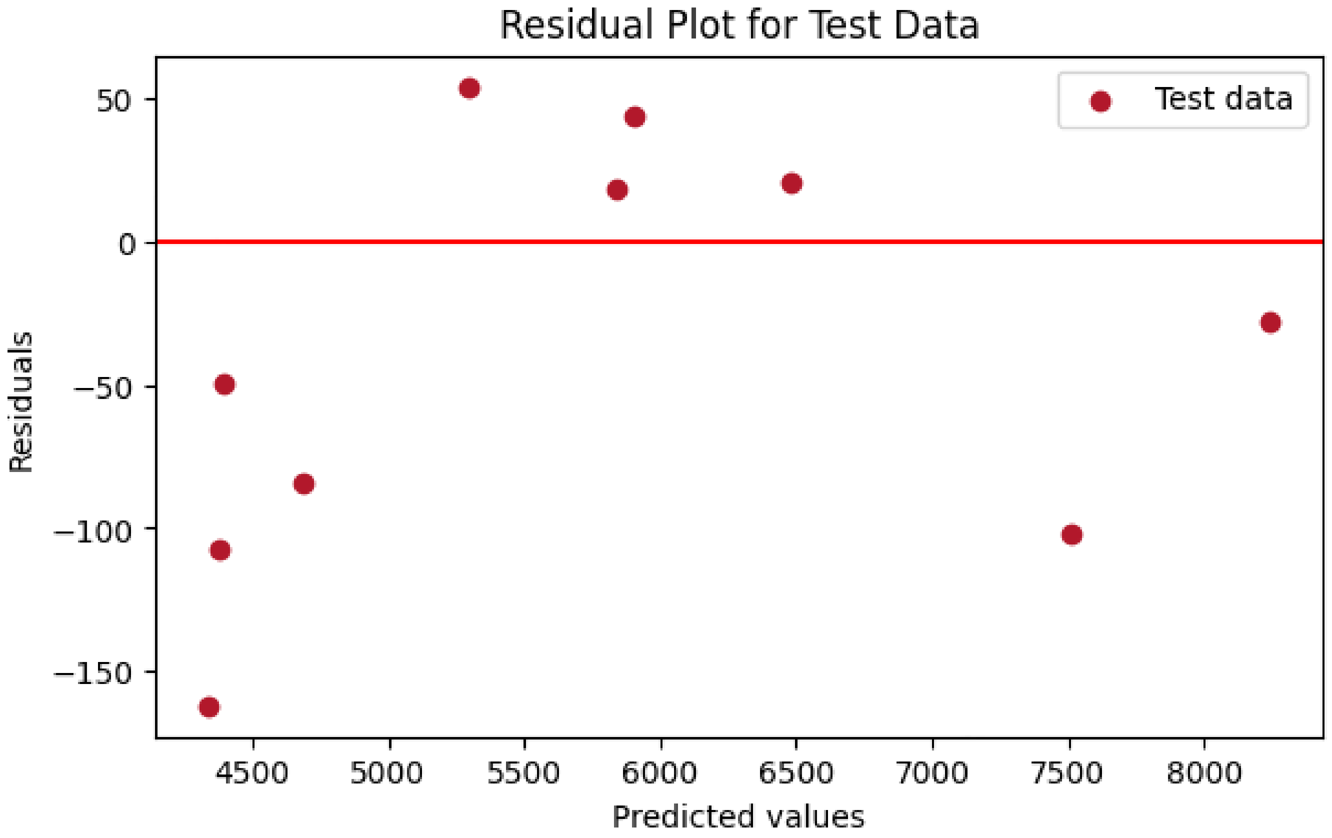
FCIM

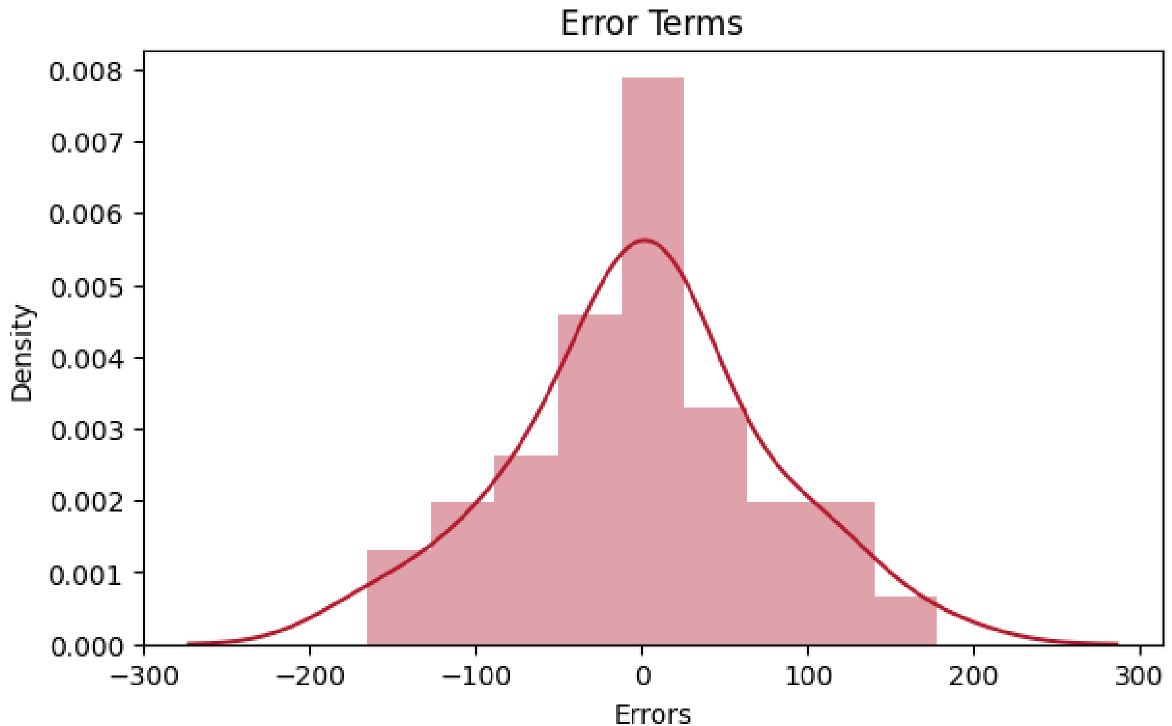


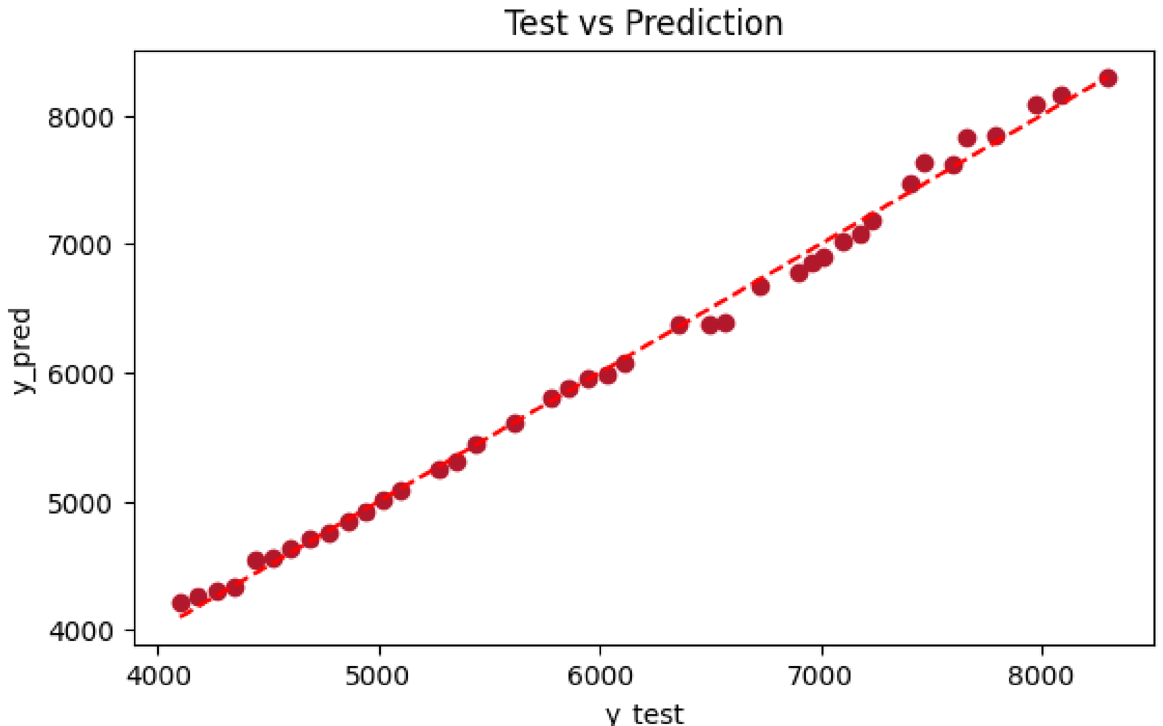
Root Mean Squared Error (RMSE): 80.042

Rsquare-Score: 0.990





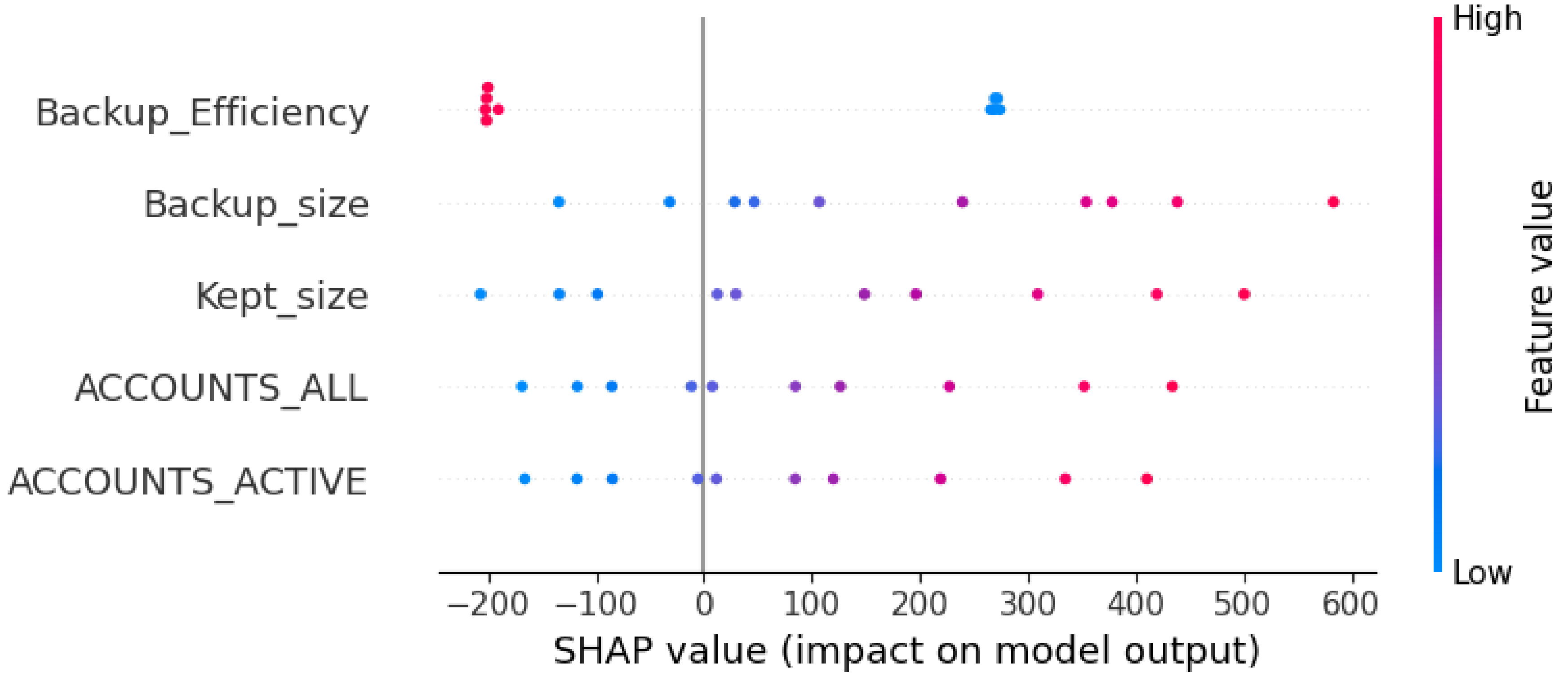




FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

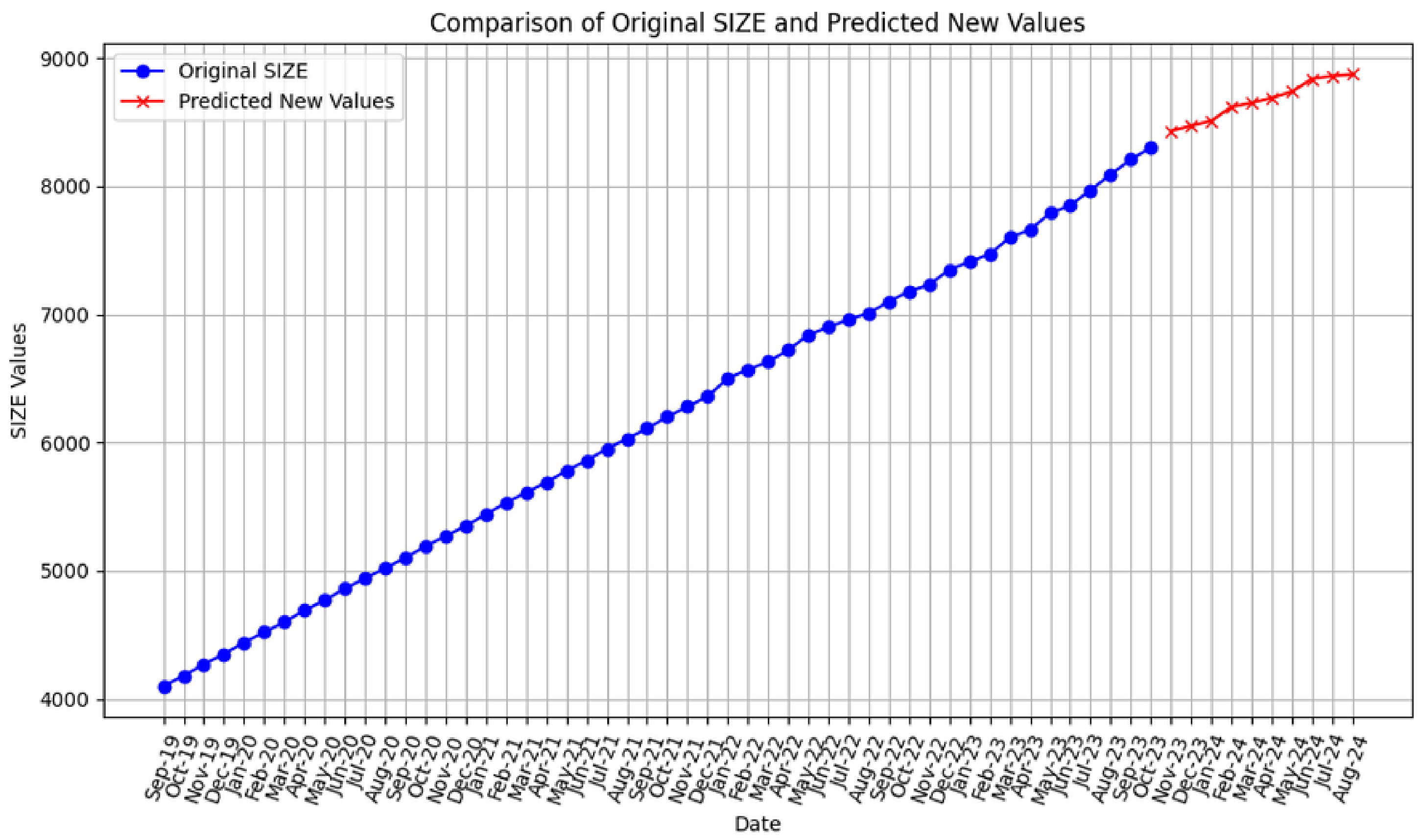
Ridge Regression Model

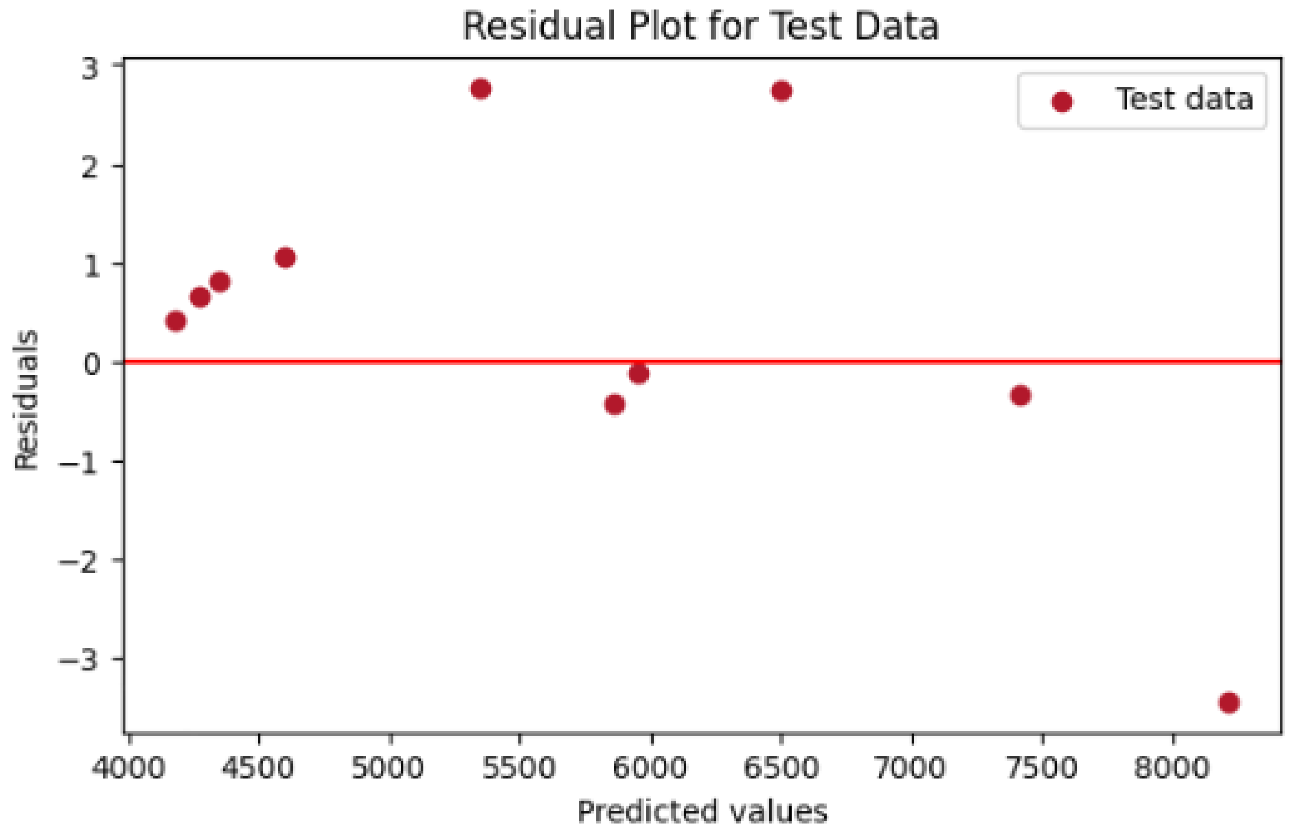
FCIM

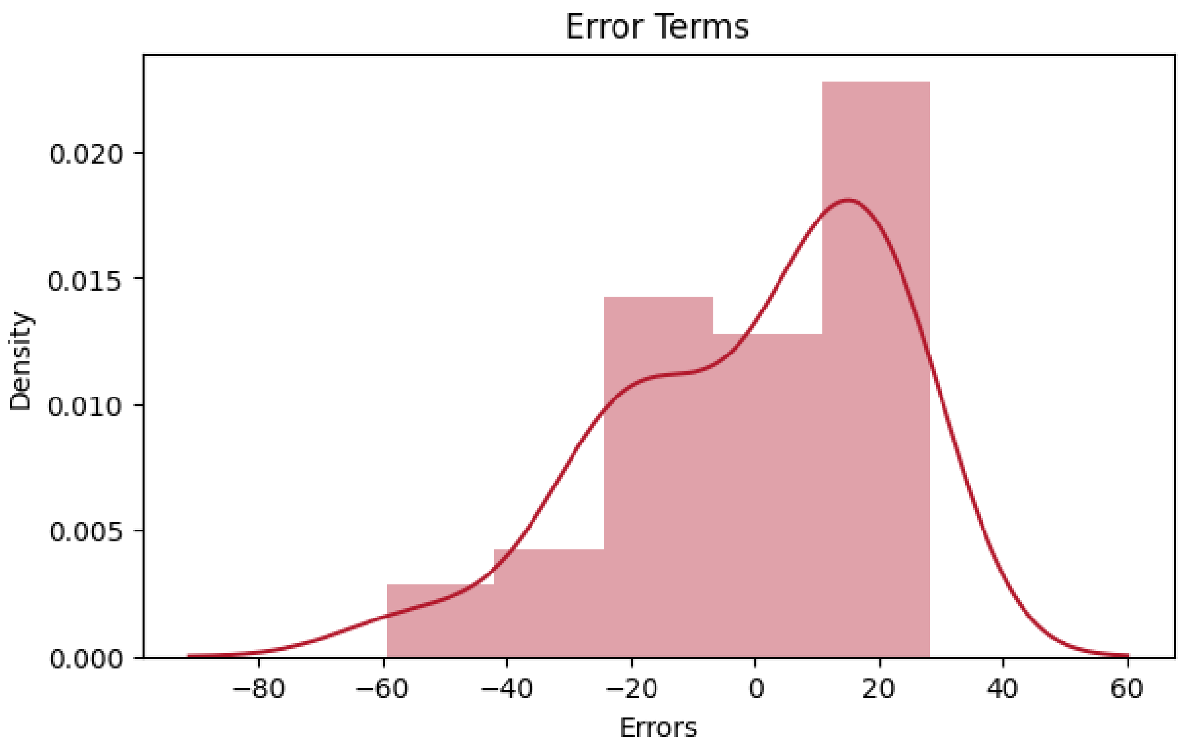


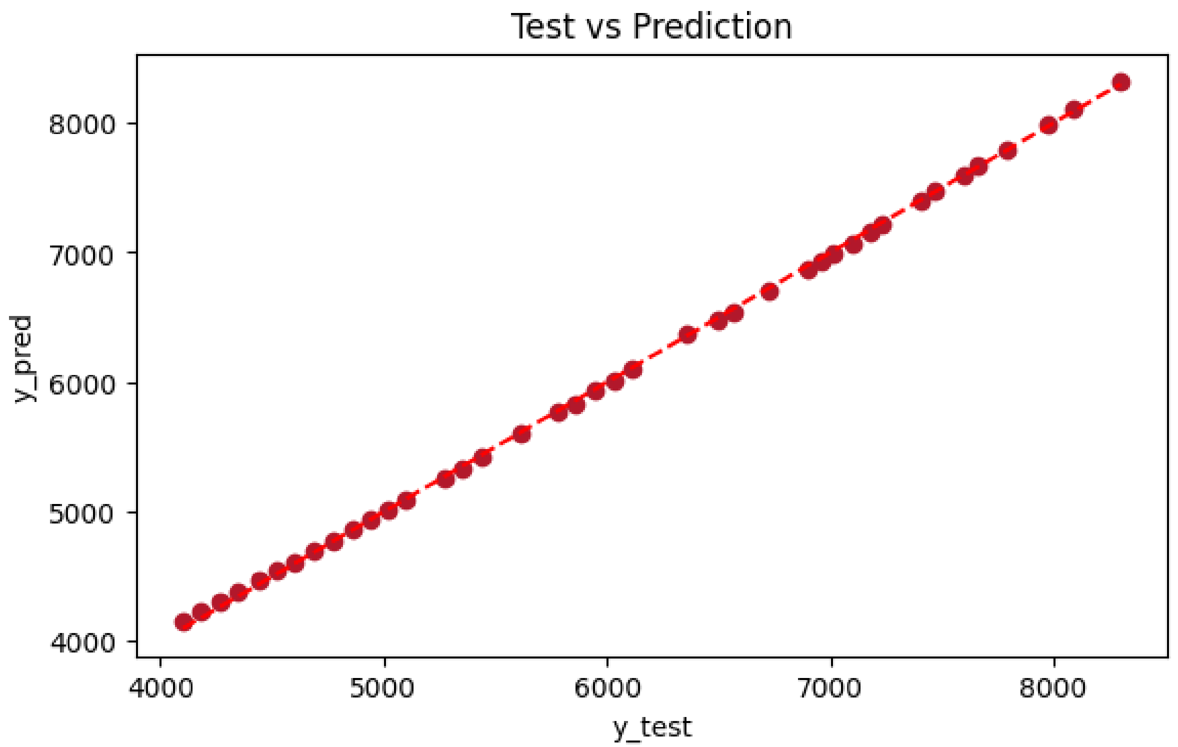
Root Mean Squared Error (RMSE): 1.724

Rsquare-Score: 0.9995





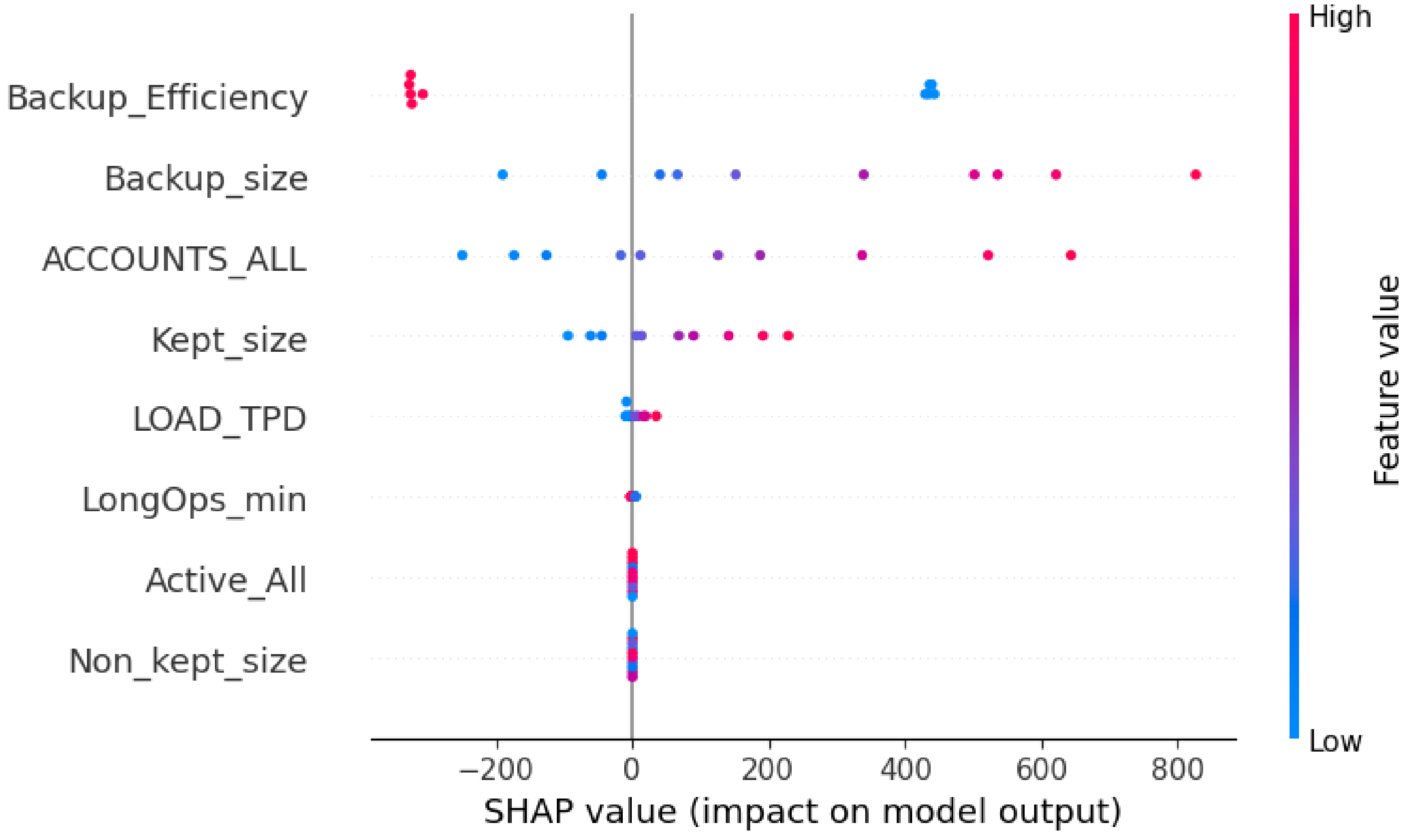




FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

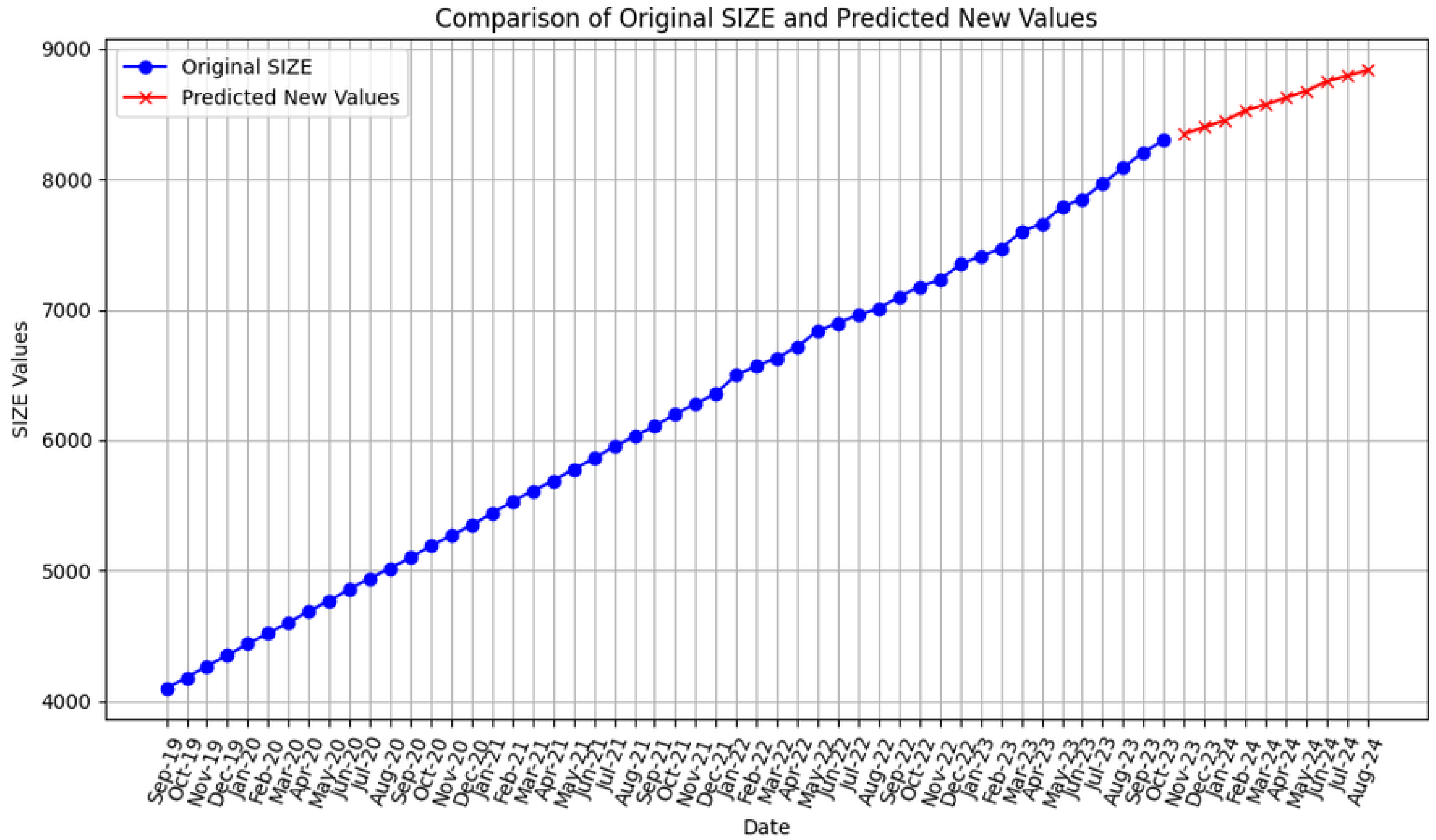
Lasso Regression Model

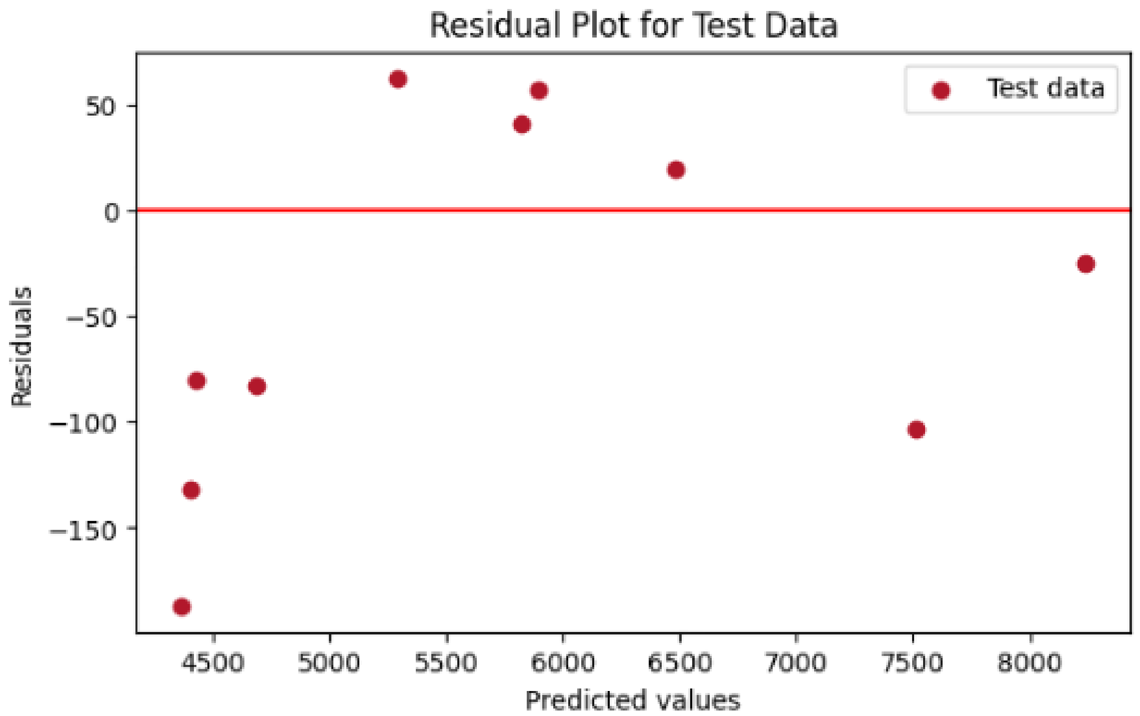
FCIM

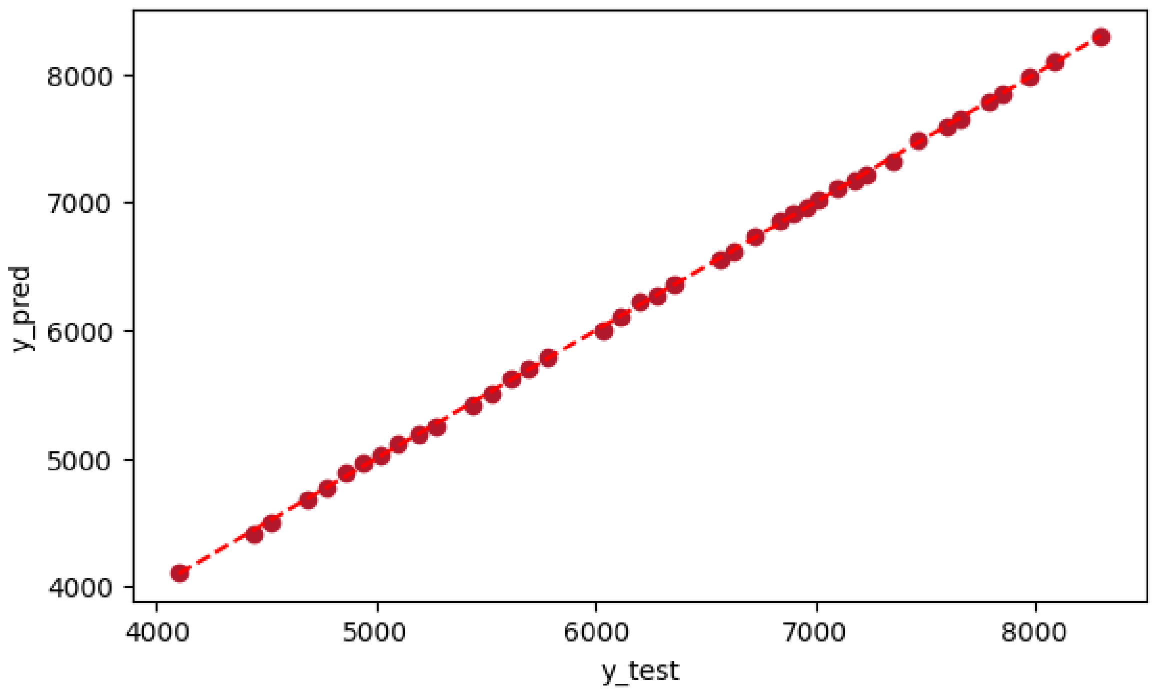


Root Mean Squared Error (RMSE): 92.745

Rsquare-Score: 0.99986



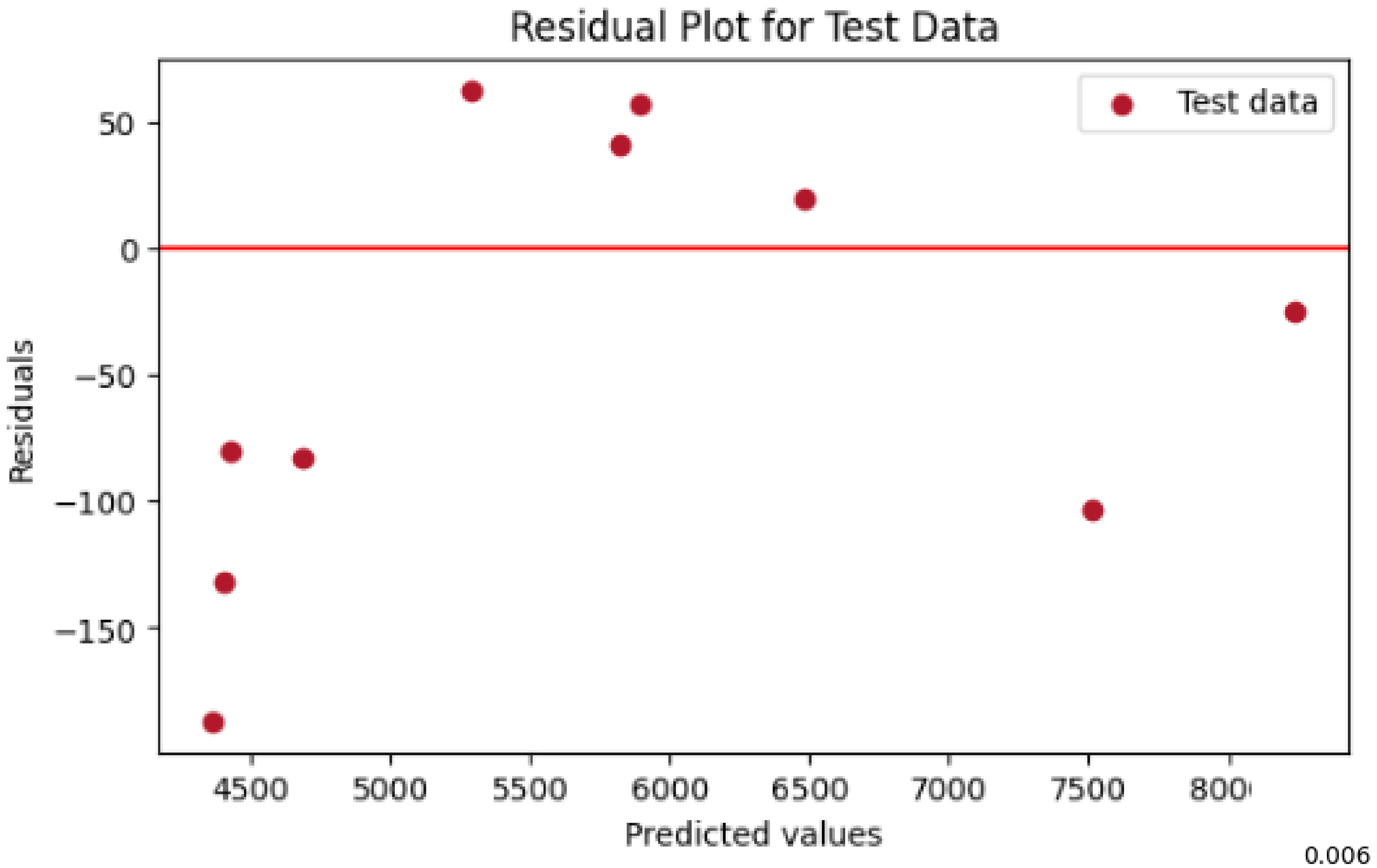




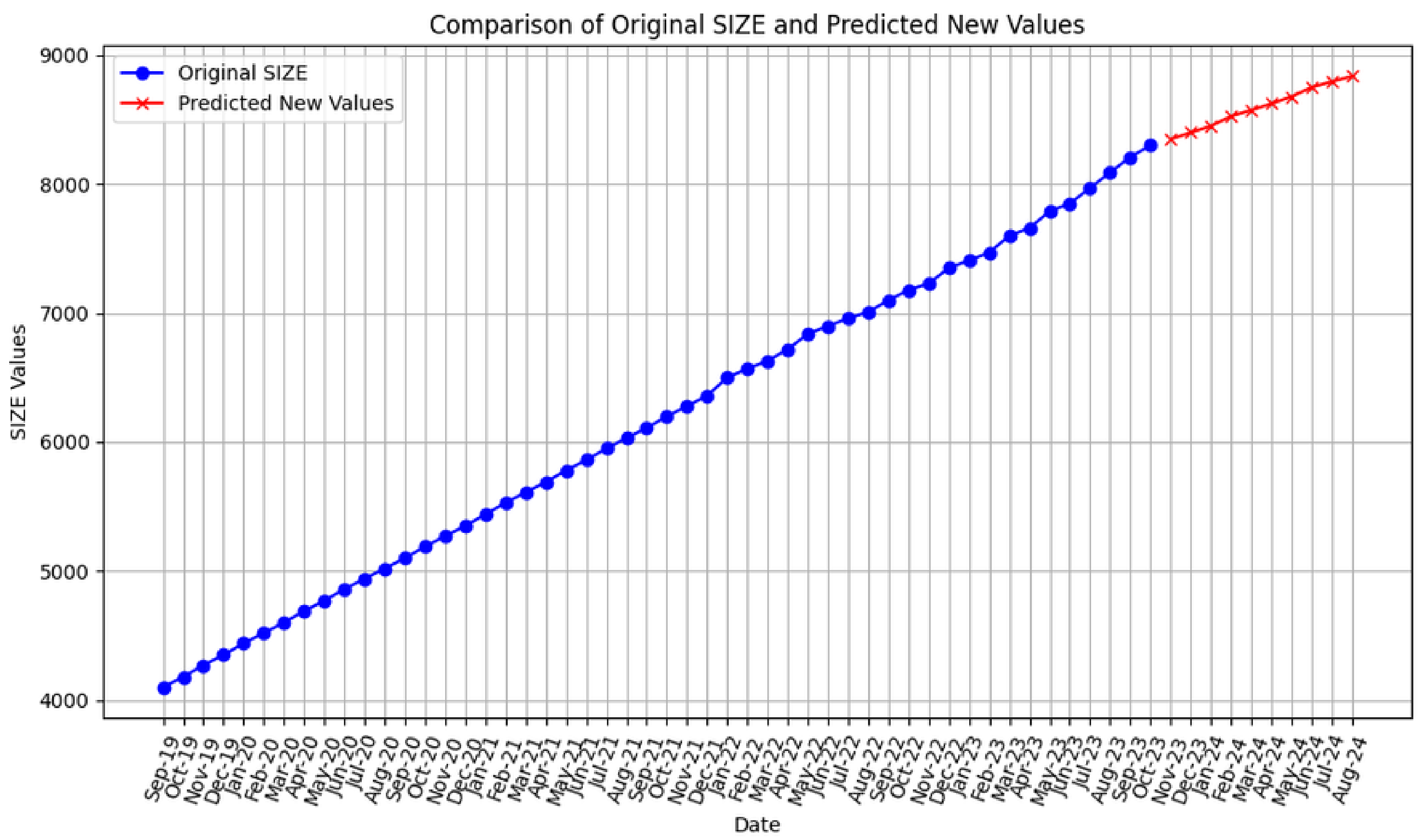
FACULTATEA
CALCULATOARE, INFORMATICĂ
ȘI MICROELECTRONICĂ

Elasting-Net Regression Model

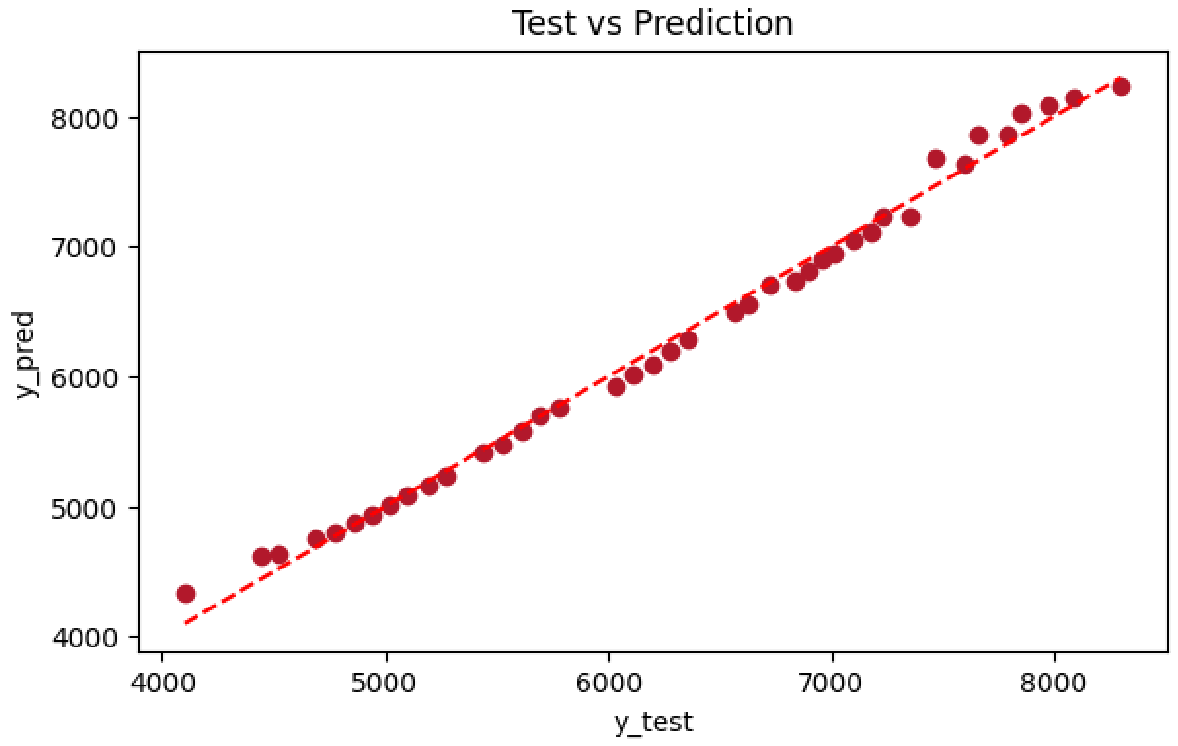
FCIM



Rsquare-Score: 0.99507



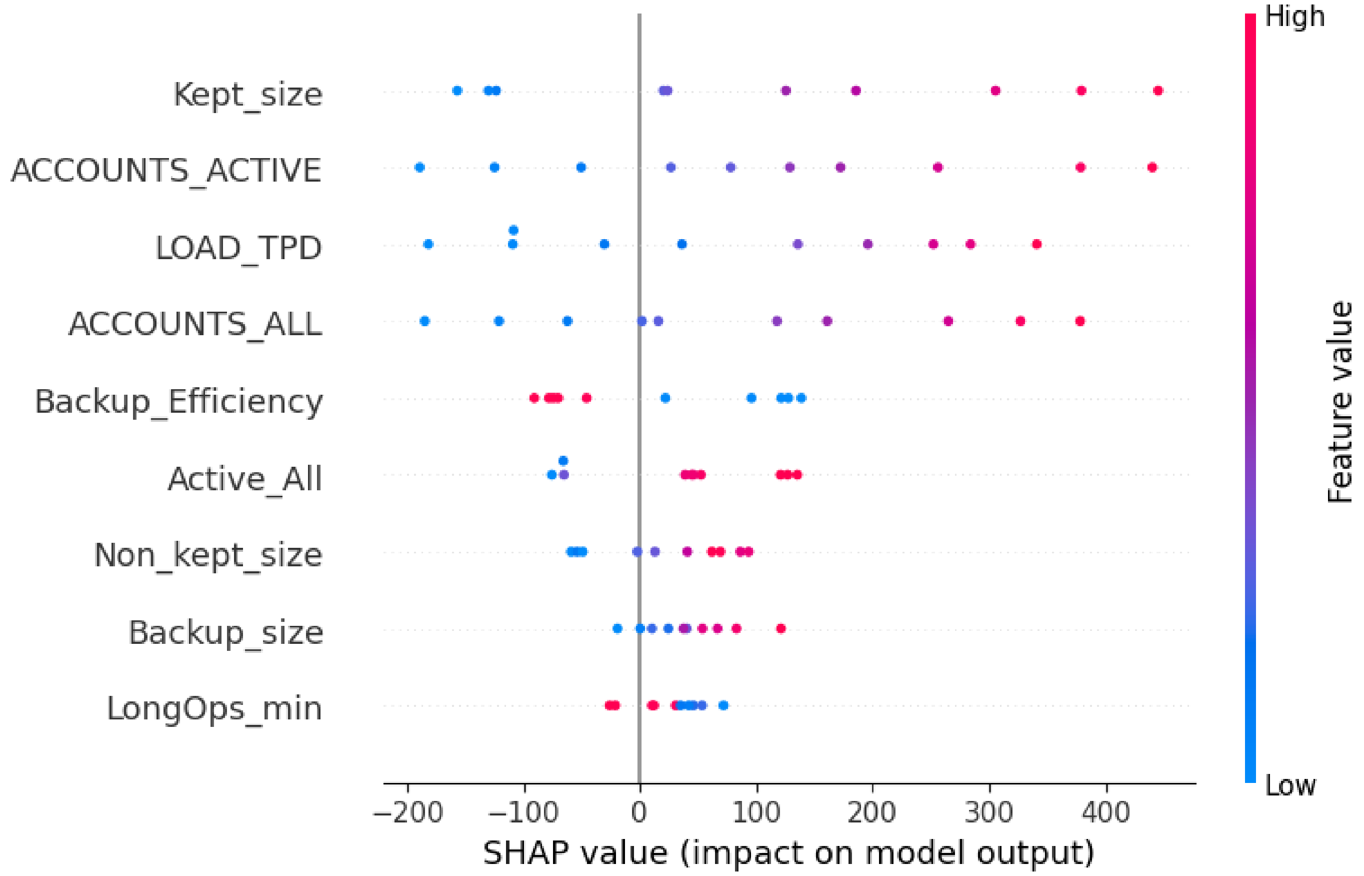
Root Mean Squared Error (RMSE): 92.745



FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

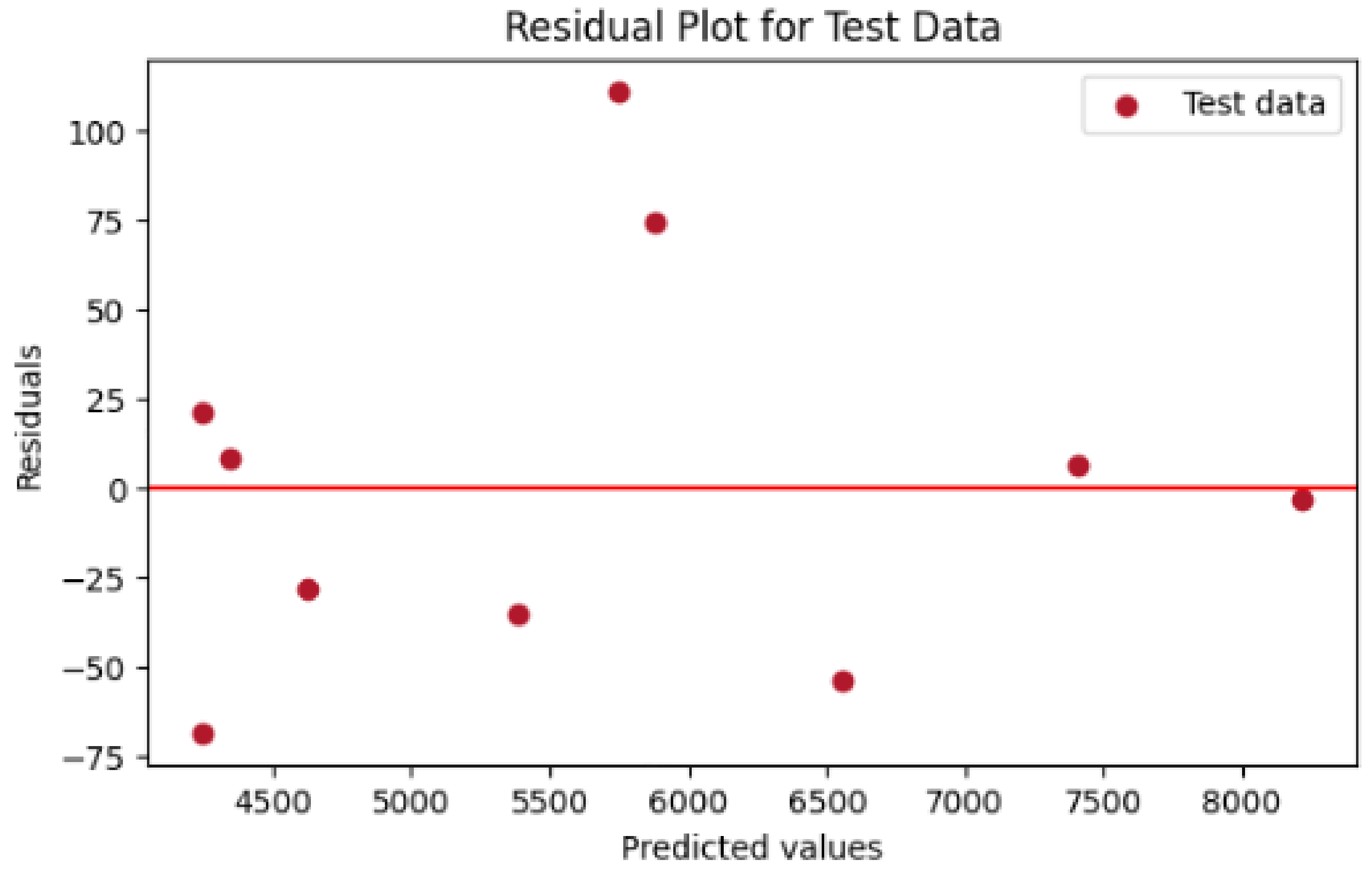
Random Forest Regression Model

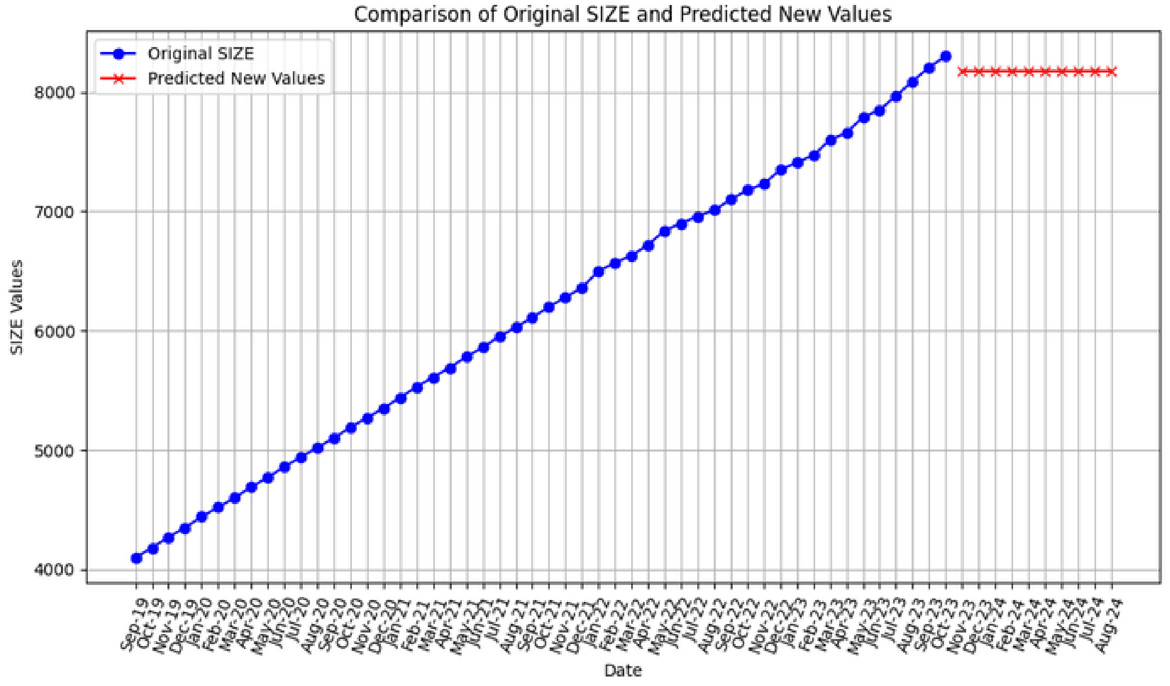
FCIM

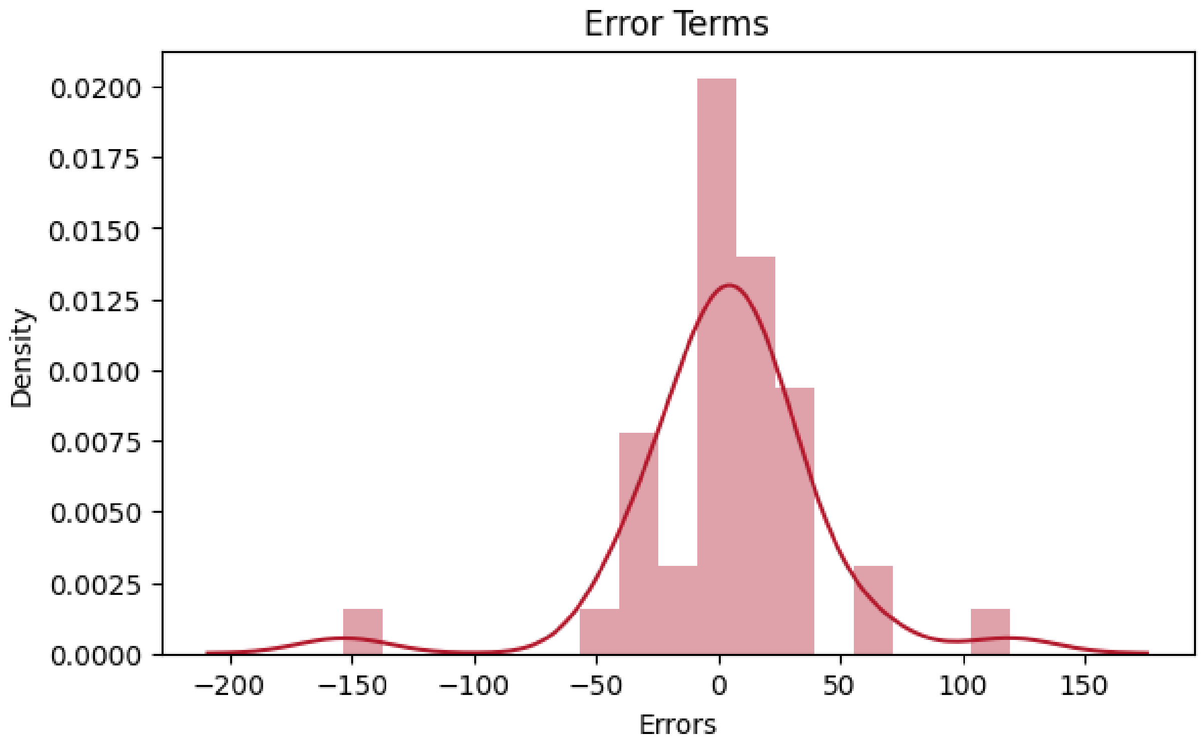


Root Mean Squared Error (RMSE): 53.079

Rsquare-Score: 0.99858





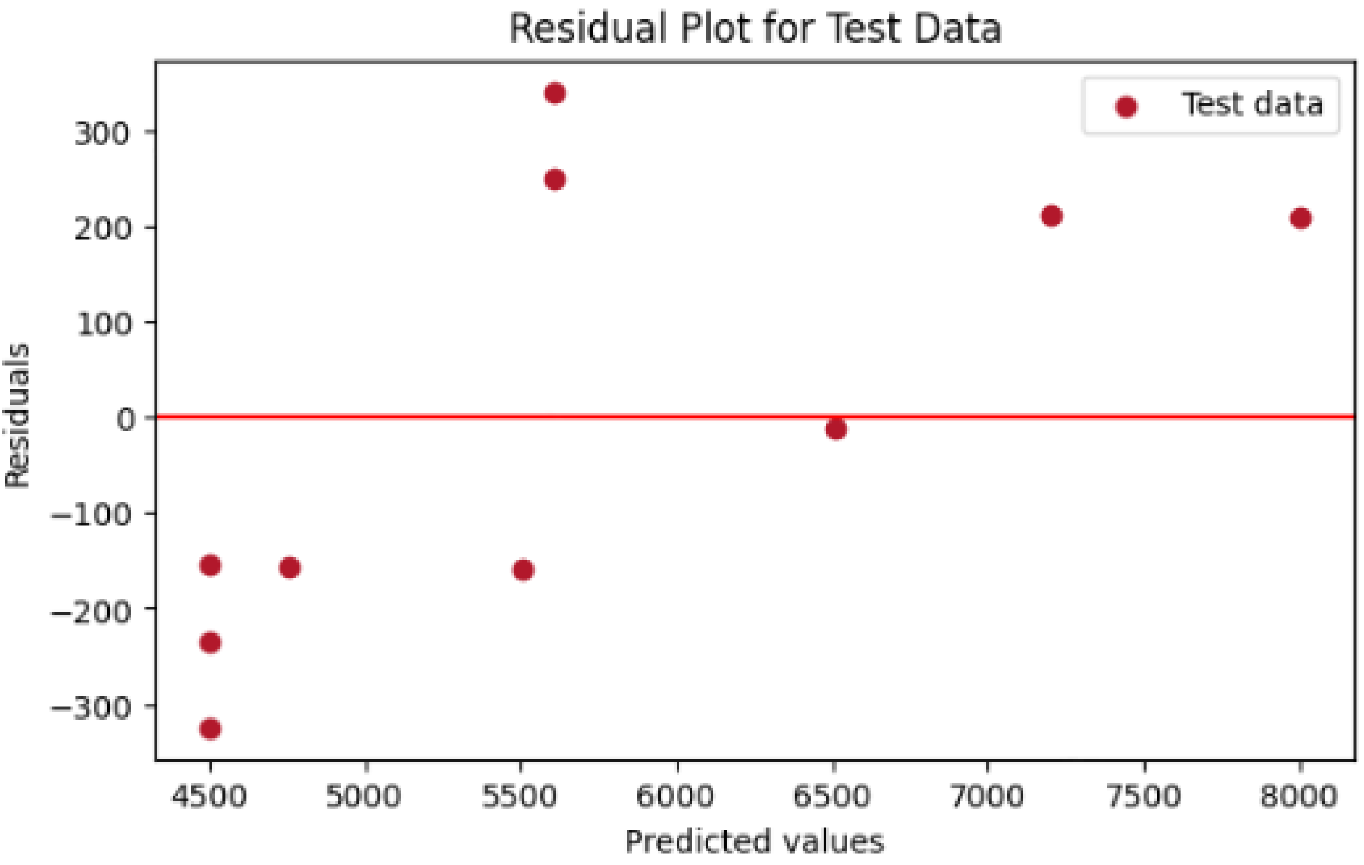


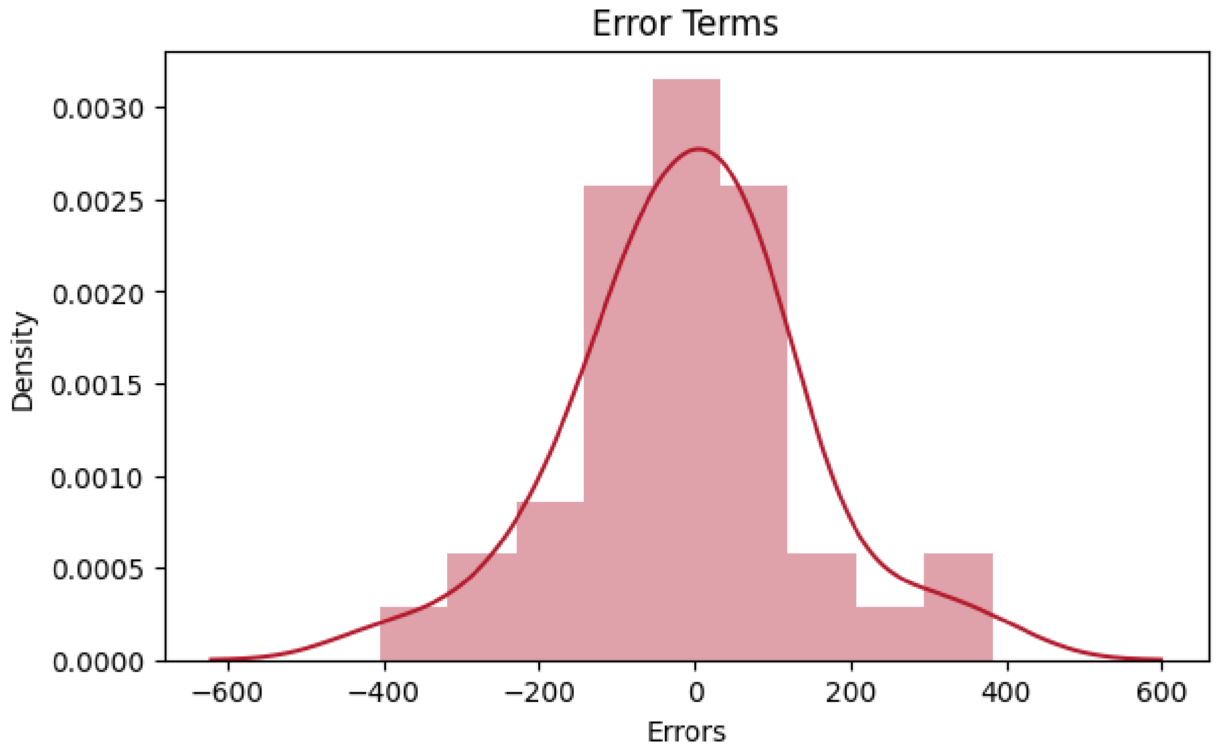
FACULTATEA
CALCULATOARE, INFORMATICĂ
ȘI MICROELECTRONICĂ

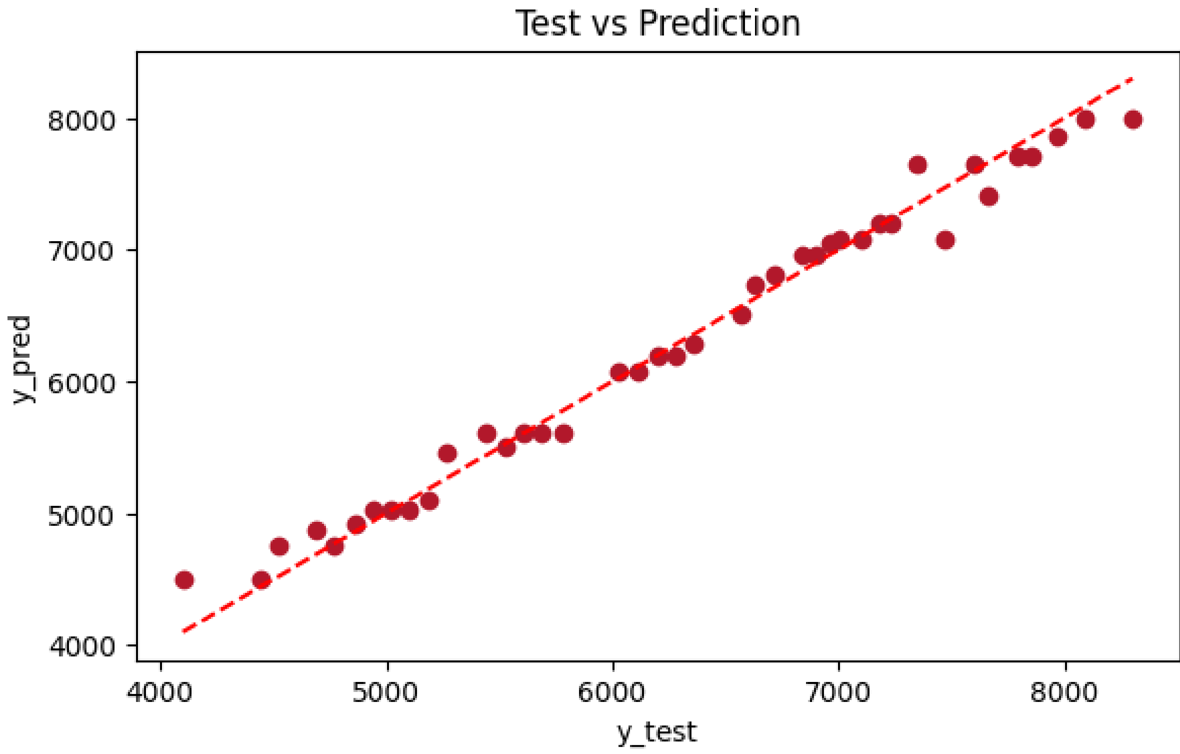
K Neighbours Regressor Model

from K-Nearest Neighbors (KNN) family

FCIM



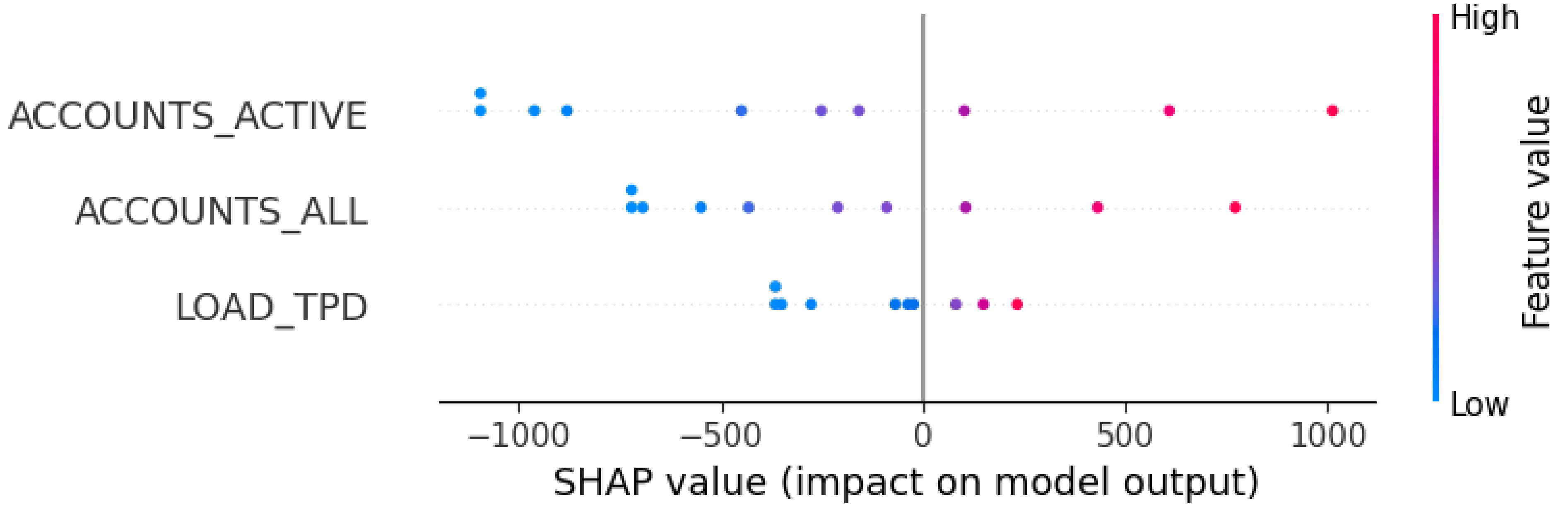


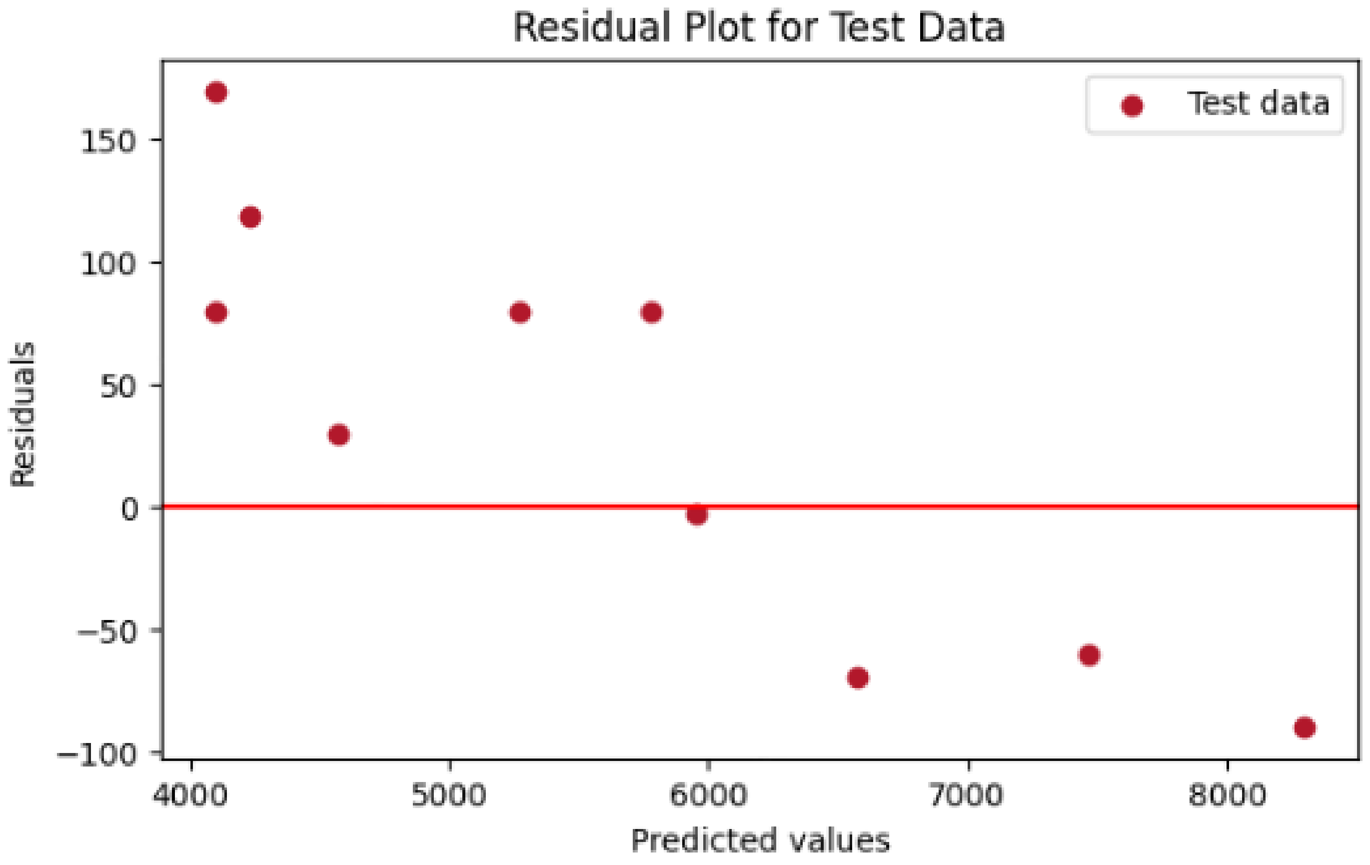


Root Mean Squared Error (RMSE): 223.61

Rsquare-Score: 0.97134

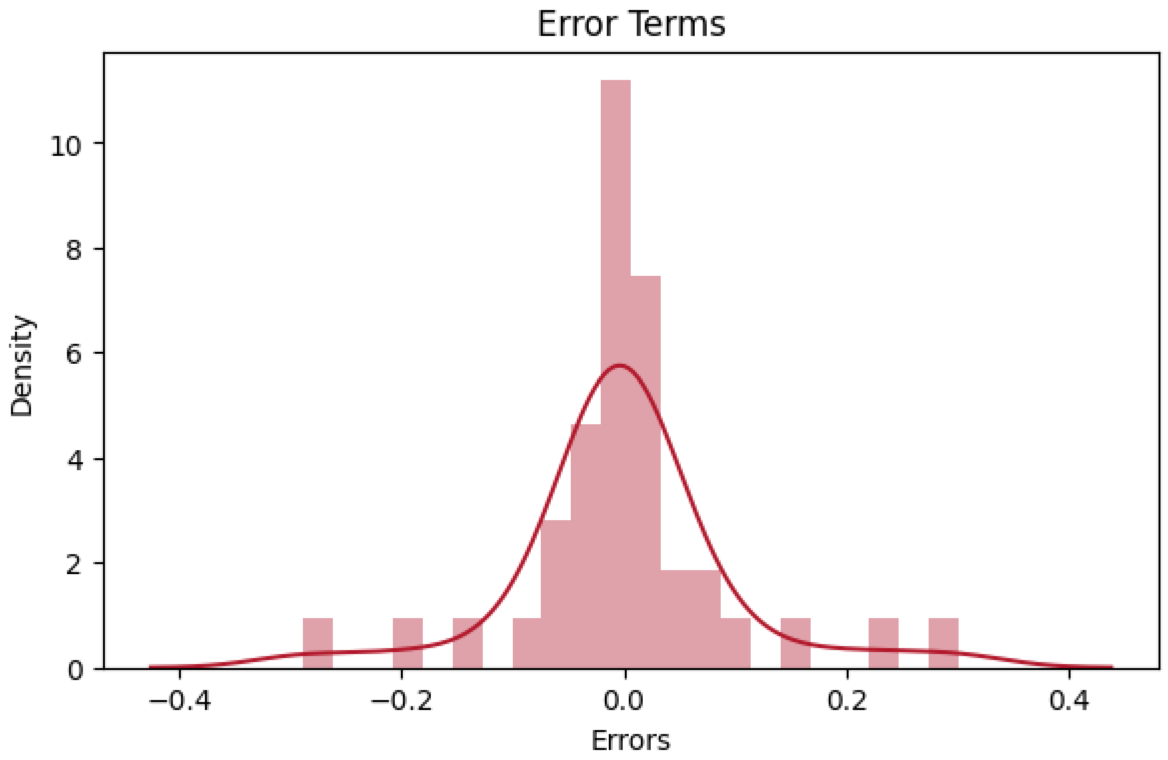
Gradient Boosting Regressor



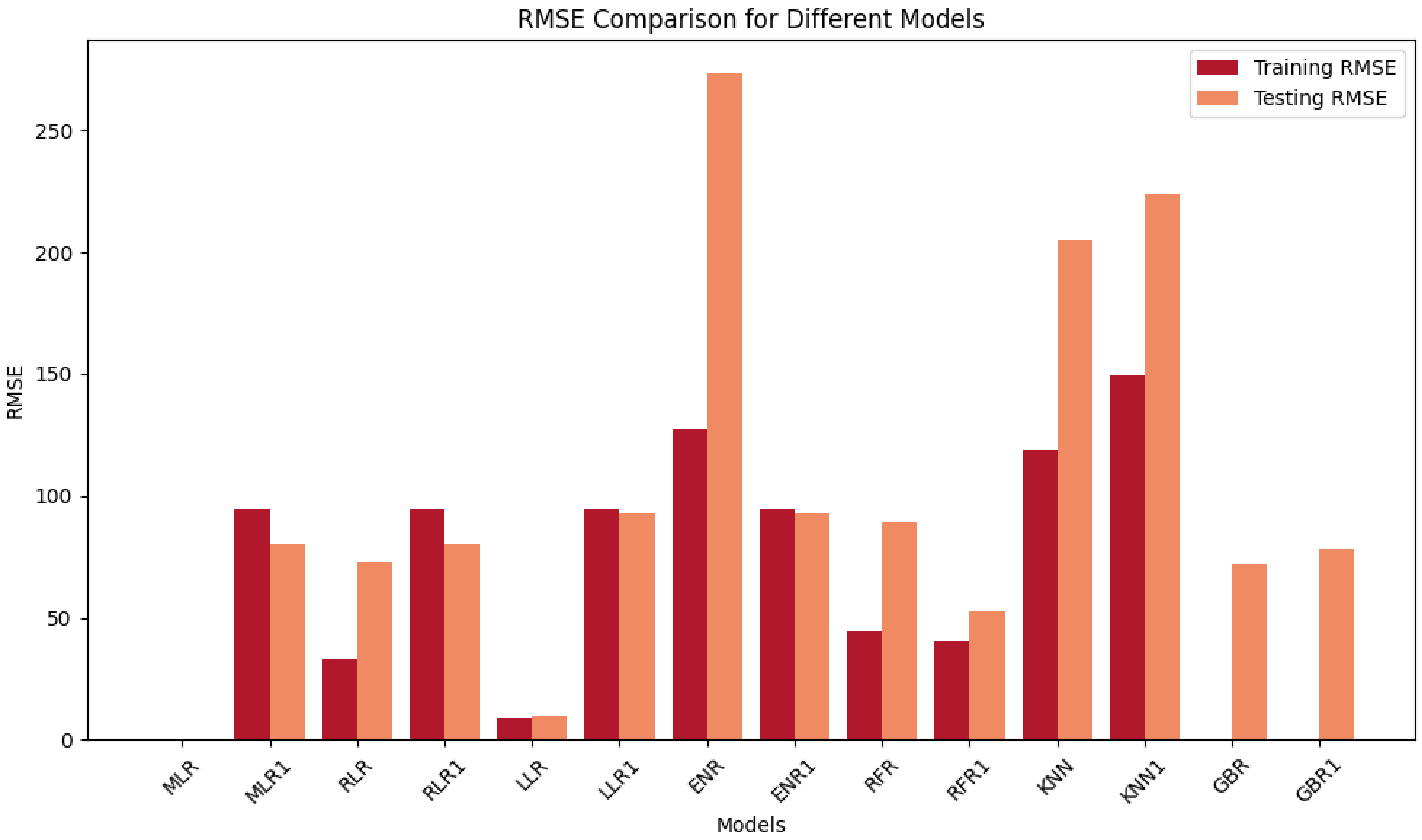


Root Mean Squared Error (RMSE): 0.088

Rsquare-Score: 0.996

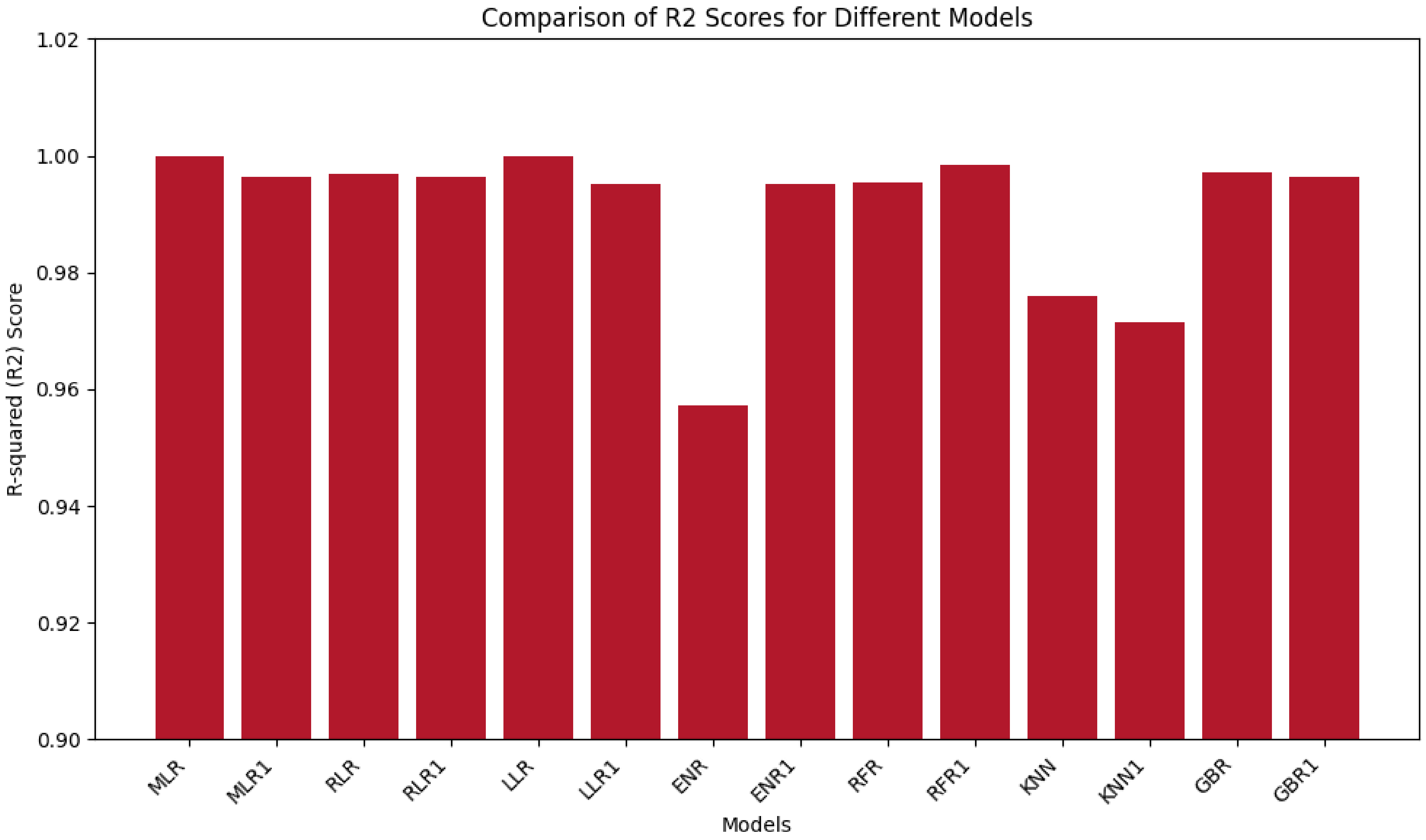


Metrics comparison



Best Models according to low RMSE: Ridge, Lasso, Random Forest and Multiple Linear Regression

Best Models according to high Rsquare score: Ridge, Lasso, Multiple Linear Regression and Random Forest



Conclusions

Based on model comparison, according to sum of factors like (Rsquare, RMSE, residual destribution and prediction on new data), best models are Multiple Linear Regression, Lasso and Ridge