Volume 10, Issue 7, July– 2025 International Journal of Innovative Science and Research Technology
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/25jul1607

Automated Data Collection and Predictive Budget Analysis for Government Fleet Scheme Management

Emmanuel Turatsinze[1]; Placide Mukwende[2]; Dr. Fidele Hategekimana[3];
Dr. Pacifique Nizeyimana[4]

[1]Ministry of Finance and Economic Planning
[1,2,3,4]Adventist University of Central Africa, Kigali, Rwanda

Publication Date: 2025/08/01

Abstract: Government Fleet Scheme Management in Rwanda has historically relied on manual, paper-based processes, resulting in inefficiencies, lack of transparency, and forecasting inaccuracies. This work focuses on the development and evaluation of a web-based platform GFSMIS (Government Fleet Scheme Management Information System) designed to automate scheme workflows and incorporate predictive analytics using machine learning techniques. A linear regression model was trained on historical scheme data (2021/2022 to 2023/2024) to predict annual budget from for beneficiaries and a random forest model was trained on historical data from 2021/2022 to 2025/2026 to predict annual budget for institutions. GFSMIS integrates user role management, digital signature verification, and real-time data visualization through dashboards. The linear regression predictive model achieved a Mean Absolute Error (MAE) of 183,217,212 FRW and a Relative Error of 1.21% for beneficiary schemes, demonstrating high accuracy and the random forest predictive model achieved a MAE of 428,794,619 FRW with a relative error of 2.13% for institution schemes, demonstrating high accuracy as well. The system supports fiscal planning, transparency, and automation for national-level decision-making.

Keywords: *MINECOFIN, Government Fleet Scheme, GFSMIS, Beneficiary, FSR, FSM, CBM, MoS and Commissioner of Customs, Budget Forecasting, ML, Linear Regression, Random Forest, Predictive Analytics, Workflow Automation, Rwanda, Public, Digitization, MVC, Java, Weka Library, Spring Core, PostgreSQL.*

How to Cite: Emmanuel Turatsinze; Placide Mukwende; Dr. Fidele Hategekimana; Dr. Pacifique Nizeyimana (2025), Automated Data Collection and Predictive Budget Analysis for Government Fleet Scheme Management. *International Journal of Innovative Science and Research Technology*, 10(7), 2526-2531.
https://doi.org/10.38124/ijisrt/25jul1607

I. INTRODUCTION

Government Fleet Schemes in Rwanda provide tax exemptions for public servants and institutions acquiring official vehicles or other assets. Traditionally, this process was entirely manual, involving physical document submissions and offline approval workflows. This system posed challenges in operational efficiency, auditability, and budget predictions.

➢ *This Study Introduces GFSMIS, a System Designed to:*

- Automate beneficiary and institutional scheme request workflows
- Integrate a linear regression-based model to predict annual budget for beneficiaries
- Integrate a random forest model to predict annul budget for institutions
- Provide dashboards and predictive reports for informed decision-making

II. LITERATURE REVIEW

Recent advancements in financial machine learning (ML) have transformed traditional methods of asset management and fiscal planning. López de Prado (2018) illustrates how ML, when applied with scientific rigor, enhances predictive accuracy, resource optimization, and process efficiency principles central to this study on automated budget prediction for Rwanda's Government Fleet Scheme. His emphasis on practical implementation, feature engineering, and model validation directly informs the use of Linear Regression and Random Forest models in this research.

Building on this foundation, Kelly and Xiu (2023) survey state-of-the-art ML applications in financial markets, demonstrating superior performance in predictive modeling, feature selection, and handling high-dimensional data. Their findings affirm the methodological approach of this study, particularly the deployment of Random Forest for complex

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/25jul1607

institutional data and Linear Regression for beneficiary-level prediction to preserve interpretability.

Together, these works provide critical justification for model selection, support the use of standard evaluation metrics like Mean Absolute Error and Relative Error, and emphasize transparency an essential feature in public finance applications. The integration of ML with domain expertise, as advocated by both studies, reinforces this project's collaborative design with Rwanda's Ministry of Finance and Economic Planning (MINECOFIN).

III. METHODOLOGY

A design science research approach was used. System architecture followed a modular Model View Controller (MVC) pattern, developed in Java using Spring Core, ZK framework, and PostgreSQL. Predictive modeling was done using the Weka library, with training data derived from historical Excel records digitized and structured in GFSMIS.

- **Sample size**: 4,012 scheme requests (FY 2021/2022 to 2024/2025) for individuals.
- **Sample size**: 321,673 scheme requests (FY 2021/2022 to 2025/2026) for institutions.
- **Features**: Fiscal Year, Scheme Category (Level 1–5), Tax Contribution.
- **Features**: Fiscal Year, Scheme Entity Type (Public or Private) Tax Contribution.
- **Training/Test split**: 80/20.

Table 1 Beneficiary by Level

LEVEL OF BENEFICIARY	MAXIMUM CEILING
LEVEL 1	Beneficiary carry maximum ceiling of 65,000,000 FRW
LEVEL 2	Beneficiary carry maximum ceiling of 45,000,000 FRW
LEVEL 3	Beneficiary carry maximum ceiling of 25,000,000 FRW
LEVEL 4	Beneficiary carry maximum ceiling of 15,000,000 FRW
LEVEL 5	Beneficiary carry maximum ceiling of 3,000,000 FRW

Table 2 Institution by Type

INSTITUTION	TYPE
MINECOFIN carry out all tax exemptions for an importation asset at any cost.	PUBLIC
MINECOFIN carry out all tax exemptions for an importation asset at any cost.	PRIVATE

IV. SYSTEM IMPLEMENTATION

- *The GFSMIS System Includes:*

- **User Management**: Role-based access for Beneficiary, Fleet Scheme Register (FSR), Fleet Scheme Manager (FSM), Chief Budget Manager (CBM), Minister of State (MoS), and Commissioner of Customs.
- **Approval Workflow**: Digital clearance from submission to validation.
- **Digital Signature**: Rwanda Information Society Authority (RISA)-issued e-certificates for secure approvals.
- **Reporting & Dashboard**: ZK-based visual dashboards with real-time metrics.
- **Predictive Budget Model**: Embedded regression forecasting engine.

V. PREDICTIVE MODEL RESULTS

- *Beneficiary Model*

The integrated regression model provided reliable individual forecasts:

- **MAE**: 183,217,212 FRW
- **Relative Error**: 1.21%
- **Predicted Budget (FY 2025/2026)**: 15,177,380,628 FRW by selecting the fiscal years shown in figure 3.
- **Predicted Records**: 1,849

Visual tools (e.g., bar plots and tree maps) aided budget trend analysis. Results showed a strong correlation between beneficiary levels and budget allocations. See figures 1 to 3

- *Entity Model*

The integrated random forest model provided reliable for institution forecasts:

- **MAE**: 428,794,619 FRW
- **Relative Error**: 2.13%
- **Predicted Budget (FY 2026/2027)**: 55,920,803,058 FRW by selecting the fiscal years shown in figure 6.
- **Predicted Records:** 84,186

Visual tools (e.g., bar plots and tree maps) aided budget trend analysis. Results showed a strong correlation between institutions and budget allocations. See figures 4 to 6

VI. DISCUSSION AND IMPACT

The findings validate that integrating machine learning in fiscal tools enhances transparency and operational efficiency. GFSMIS allows decision-makers at MINECOFIN to visualize spending, allocate budgets accurately, and reduce approval delays.

- **For policymakers:** Better data-driven planning.
- **For IT teams:** A model for embedding ML into operational systems.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/25jul1607

➢ **For beneficiaries:** Improved transparency and service delivery.

VII. CONCLUSION

The GFSMIS represents a critical step in digitizing Rwanda's public service processes. It proves that machine learning, when embedded in well-designed information systems, can improve service delivery and budget planning.

ACKNOWLEDGMENT

The author extends sincere gratitude to Dr. Fidele HATEGEKIMANA, academic supervisor, and Dr. Pacifique NIZEYIMANA, Director of AUCA Gishushu Campus, for their valuable guidance and encouragement throughout this research. Special thanks are extended to the Ministry of Finance and Economic Planning (MINECOFIN), particularly to Mr. Placide MUKWENDE, the technical supervisor, and the Fleet Scheme Management Team under the Office of the Minister of State in Charge of the National Treasury, for their support and for providing access to the necessary datasets. Appreciation is also extended to the Adventist University of Central Africa and to fellow MSc students for their collaboration and inspiration throughout the study.

REFERENCES

[1]. López de Prado, Marcos and López de Prado, Marcos, Advances in Financial Machine Learning: Lecture 2/10 (seminar slides) (September 29, 2018). Available at SSRN: https://ssrn.com/abstract=3257415 or http://dx.doi.org/10.2139/ssrn.3257415
[2]. Marcos López de Prado, Frank J. Fabozzi. Crowdsourced Investment Research through Tournaments. Journal of Financial Data Science, Vol. 2, No. 1, 2020, https://jfds.pm-research.com/content/2/1/86
[3]. Ishiki et al., "Measurement and analysis of differential work hardening behavior," *Int. J. Mater. Form.*, 2023.
[4]. BFI Working Paper, "Machine Learning for Financial Forecasting," Becker Friedman Institute, 2023.
[5]. Corporate Finance Institute, "Regression Analysis," [Online]. Available: https://corporatefinanceinstitute.com/ and https://bfi.uchicago.edu/wp-content/uploads/2023/07/BFI_WP_2023-100.pdf
[6]. https://www.minecofin.gov.rw/1/publications/reports
[7]. https://www.researchgate.net/publication/375722976_Machine_Learning_for_Financial_Forecasting
[8]. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4501707
[9]. https://corporatefinanceinstitute.com/resources/data-science/regression-analysis/
[10]. https://interviewkickstart.com/blogs/articles/financial-forecasting-ml-predictive-analytics#Main
[11]. https://stefanini.com/en/insights/news/machine-learning-models-for-precise-predictive-analytics
[12]. https://www.pyquantnews.com/free-python-resources/machine-learning-for-predictive-financial-analysis
[13]. https://bfi.uchicago.edu/wp-content/uploads/2023/07/BFI_WP_2023-100.pdf

FIGURES

```
// Predict for each level separately
for (String level : allLevels) {
    double levelTaxPrediction = predictWithLevel(taxModel, trainData, targetYear, level);

    // Create and train count model properly
    LinearRegression countModel = new LinearRegression();
    countModel.setOptions(new String[]{"-S", "1"});

    // Apply weights to count data (same weighting scheme as tax data)
    double[] countWeights = new double[countData.numInstances()];
    for (int i = 0; i < countWeights.length; i++) {
        countWeights[i] = Math.pow(1.3, i);
        countData.instance(i).setWeight(countWeights[i]);
    }
    countModel.buildClassifier(countData);

    // Now make the prediction
    double levelCountPrediction = predictWithLevel(countModel, countData, targetYear, level);

    // Apply growth constraints per level
    double lastYearTax = getLastYearValueForLevel(yearlyLevelTaxTotals, level);
    levelTaxPrediction = constrainPrediction(levelTaxPrediction, lastYearTax, minGrowth: 0.0, maxGrowth: 0.10);

    int lastYearCount = (int) getLastYearValueForLevel(yearlyLevelRecordCounts, level);
    levelCountPrediction = constrainPrediction(levelCountPrediction, lastYearCount, minGrowth: 0.0, maxGrowth: 0.15);

    predictedTaxes += levelTaxPrediction;
    predictedCount += levelCountPrediction;
}
```

Fig 1 Beneficiary Forecasting Model

```
// Set results
forecastedFiscalTotal = BigDecimal.valueOf(predictedTaxes);

// Evaluate the model on test data
Evaluation eval = new Evaluation(trainData);
eval.evaluateModel(taxModel, testData);
// The mean absolute error
meanError = BigDecimal.valueOf(eval.meanAbsoluteError());

// prediction of the total requests
predictedRecordCount = (int) Math.round(predictedCount);

BigDecimal re = meanError.divide(forecastedFiscalTotal, scale: 4, RoundingMode.HALF_UP);

// The relative error computation
relativeError = re.multiply(BigDecimal.valueOf(100));
theError = relativeError.toString().substring(0,4) + "%";
```

Fig 2 Beneficiary Forecasting Output Source Codes

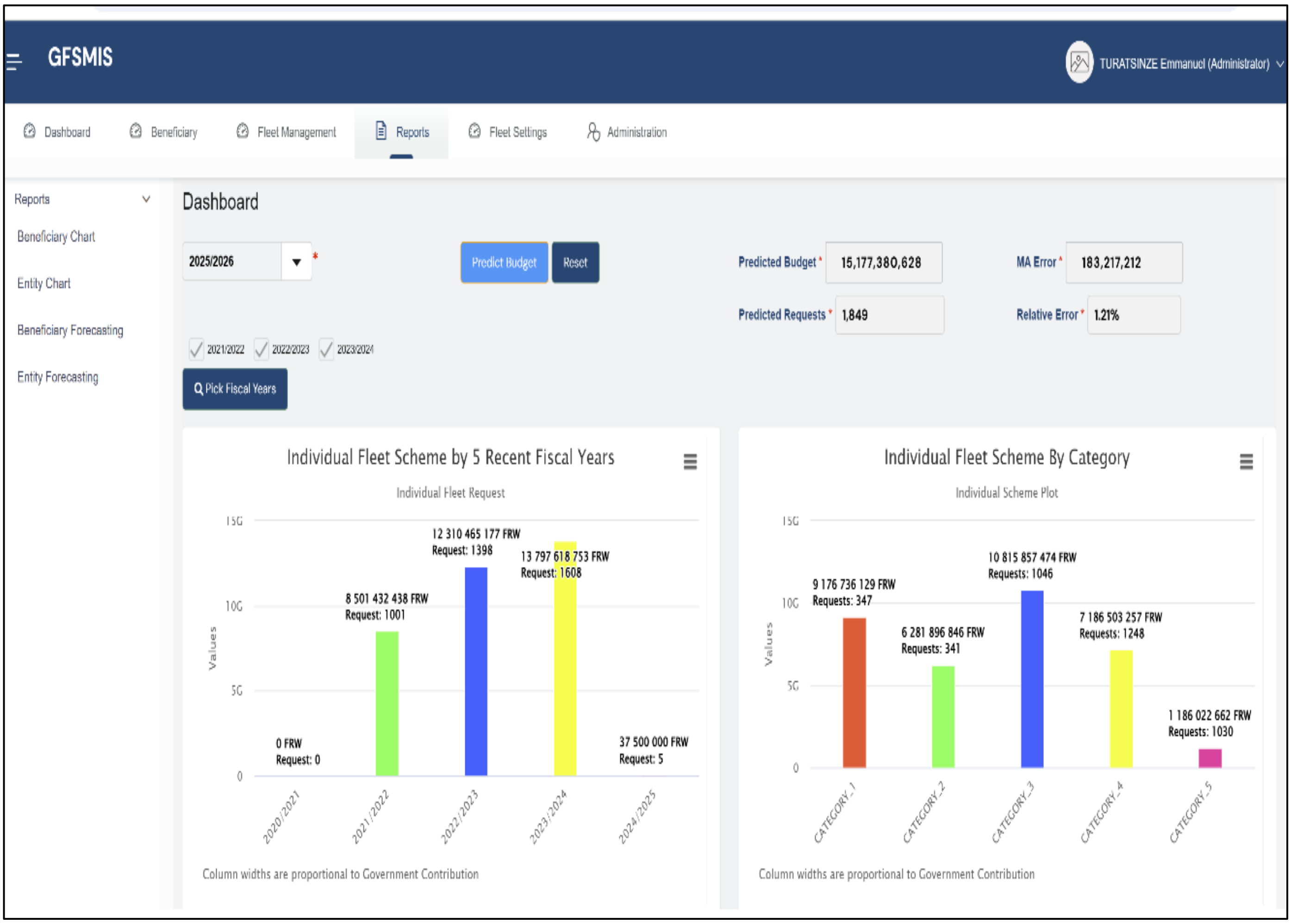

Fig 3 Beneficiary Forecasting Outputs

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/25jul1607

```
// Model for the budget prediction
RandomForest taxModel = new RandomForest();
taxModel.setNumIterations(100);
taxModel.buildClassifier(trainData);

// Model for the record prediction
RandomForest countModel = new RandomForest();
countModel.setNumIterations(100);
countModel.buildClassifier(countData);

int targetYear = Integer.parseInt(targetPeriod.getName().split( regex: "/")[0]);
double predictedTaxes = 0;
double predictedCount = 0;

Set<String> allEntityTypes = yearlyEntityTypeTaxTotals.values().stream() Stream<Map<...>>
        .flatMap(map -> map.keySet().stream()) Stream<String>
        .collect(Collectors.toSet());
```

Fig 4 Institution Forecasting Model

```
// Predict for each entity type separately
for (String entityType : allEntityTypes) {
    double entityTypeTaxPrediction = predictWithEntityType(taxModel, taxData, targetYear, entityType);
    double entityTypeCountPrediction = predictWithEntityType(countModel, countData, targetYear, entityType);

    double lastYearTax = getLastYearValueForEntityType(yearlyEntityTypeTaxTotals, entityType);
    entityTypeTaxPrediction = constrainPrediction(entityTypeTaxPrediction, lastYearTax,  minGrowth: -0.05,  maxGr

    int lastYearCount = (int) getLastYearValueForEntityType(yearlyEntityTypeRecordCounts, entityType);
    entityTypeCountPrediction = constrainPrediction(entityTypeCountPrediction, lastYearCount,  minGrowth: -0.05,

    // Make the prediction
    predictedTaxes += entityTypeTaxPrediction;
    predictedCount += entityTypeCountPrediction;
}
forecastedFiscalTotal = BigDecimal.valueOf(predictedTaxes);

// Evaluation of the model
Evaluation eval = new Evaluation(trainData);
eval.evaluateModel(taxModel, trainData);
meanError = BigDecimal.valueOf(eval.meanAbsoluteError());
predictedRecordCount = (int) Math.round(predictedCount);

BigDecimal re = meanError.divide(forecastedFiscalTotal,  scale: 4, RoundingMode.HALF_UP);
relativeError = re.multiply(BigDecimal.valueOf(100));
theError = relativeError.toString().substring(0, 4) + "%";
```

Fig 5 Institution Forecasting Output Source Codes

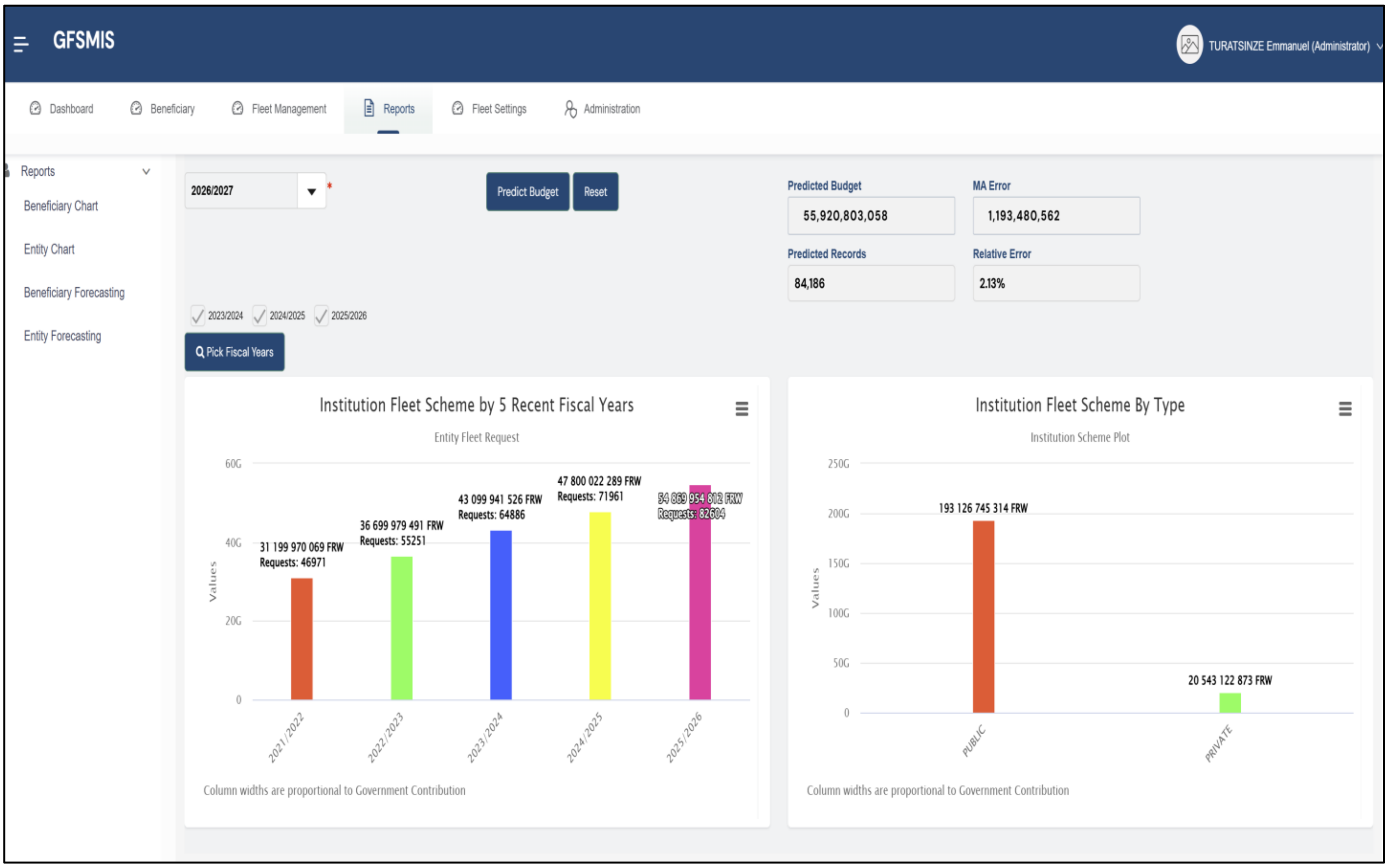

Fig 6 Institution Forecasting Output Results and Dashboard