
IBM AICTE PROJECT

INTELLIGENT CLASSIFICATION OF RURAL INFRASTRUCTURE PROJECTS

Presented By: Mohamed Farmaan Naser N
Student Name: Mohamed Farmaan Naser N
College Name: Sri Sai Ram Engineering College, Chennai
Department: Computer Science and Business Systems

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

The Pradhan Mantri Gram Sadak Yojana (PMGSY) is a flagship initiative by the Government of India aimed at improving rural connectivity through the construction of all-weather roads and bridges. Over the years, the program has evolved into multiple schemes such as PMGSY-I, PMGSY-II, and RCPLWEA, each with specific goals, funding mechanisms, and implementation strategies. With thousands of infrastructure projects sanctioned across the country, accurately identifying which scheme a project belongs to is becoming increasingly critical. Currently, this classification is done manually, which is time-consuming, prone to human errors, and difficult to scale. As a result, project monitoring, budget allocation, and impact assessment become inefficient. Therefore, the need arises for an intelligent system that can automatically classify a rural infrastructure project into its correct PMGSY scheme based on its physical and financial characteristics.

PROPOSED SOLUTION

- The proposed system leverages machine learning to **predict the appropriate PMGSY scheme** based on key infrastructure metrics and project data. The system comprises the following components:
- **Data Collection:**
 - Collected district-wise rural development data from PMGSY official sources
https://aikosh.indiaai.gov.in/web/datasets/details/pradhan_mantri_gram_sadak_yojna_pmgysy.html.
 - Data includes attributes such as:
Number and length of road and bridge works sanctioned/completed, Sanctioned cost and actual expenditure, Project completion status.
- **Data Preprocessing:**
 - Handled missing values and negative expenditures.
 - Created feature-engineered metrics like:
 - $\text{completion_ratio} = \text{Road Works Completed} / \text{Road Works Sanctioned}$
 - $\text{bridge_completion_ratio} = \text{Bridges Completed} / \text{Bridges Sanctioned}$
 - $\text{expenditure_efficiency} = \text{Expenditure} / \text{Sanctioned Cost}$
 - Encoded scheme names (PMGSY-I, II, III, etc..) into numerical labels for model training.
- **Model Selection and Training:**
 - Trained using classification model Random Forest Classifier and Applied GridSearchCV for hyperparameter tuning.
 - Evaluated models using:
Accuracy Score, Classification Report (Precision, Recall, F1-Score), Confusion Matrix

SYSTEM APPROACH

The proposed system aims to predict the required PMGSY scheme based on project features like road length, cost, and completion ratio using machine learning. The system leverages IBM Cloud tools and Python-based ML libraries to train and deploy an effective model.

- **System requirements**

- Platform: IBM Cloud Watsonx.ai Studio

- Runtime Environment: Watsonx Runtime

- Notebook Environment: IBM Jupyter Notebook (hosted within Watsonx.ai)

- Language: Python 3.x

- Model Type: Classification (Random Forest, Grid Search CV)

- IBM Cloud Object Storage

- **Library required to build the model**

- pandas – for data loading and manipulation

- numpy – for numerical operations

- matplotlib – for plotting confusion matrix and graphs

- sklearn.preprocessing.LabelEncoder – to encode scheme names (e.g., PMGSY-I, PMGSY-II)

- sklearn.impute.SimpleImputer – to handle missing values

- sklearn.pipeline.Pipeline – to streamline preprocessing and modeling

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**

- The chosen algorithm is the **Random Forest Classifier**, an ensemble learning method that builds multiple decision trees and merges their results to improve accuracy and prevent overfitting. It was selected due to its high performance on structured data and ability to handle classification tasks effectively, especially in multi-class problems like scheme prediction.

- **Data Input:**

- The model was trained using structured data with the following key input features:
- Road Length, Sanctioned Cost, Completed Length, Financial Progress, Completion Ratio
- The target variable was the scheme name, originally encoded numerically and later mapped back to labels like PMGSY-I, PMGSY-II, RCPLWE, etc.

- **Training Process:**

- The dataset was split into 80% training and 20% testing sets.
- A Pipeline was created to combine preprocessing and modeling steps.
- **GridSearchCV** was used to perform hyperparameter tuning, optimizing parameters like the number of trees (n_estimators), tree depth (max_depth), and others.
- The best-performing model was selected based on cross-validated accuracy scores

- **Prediction Process:**

- The final model predicts the correct scheme for a given project based on its features. The prediction process is fast, interpretable, and suitable for integration into decision support tools. Labels are mapped back from numeric values to actual scheme names to enhance usability.

RESULT

The screenshot displays a Jupyter Notebook in a web browser. The browser's address bar shows the URL: `eu-gb.dataplatform.cloud.ibm.com/analytics/notebooks/v2/d59dacb6-d1bb-414c-8cb6-ae2f6ab3771b?projectId=dcc87742-d1ae-4458-b9a4-1616f1880f9...`. The notebook's title bar indicates it is titled "Jupyter_Notebook".

The notebook's interface includes a menu bar (File, Edit, View, Run, Kernel, Help) and a toolbar with icons for file operations and execution. The current cell is a code cell containing the following Python code:

```
[100]: rf_pred = pipeline.predict(X_test)

print("\nClassification Report (Random Forest Classifier):\n", classification_report(y_test, rf_pred))
print("Accuracy Score (Random Forest Classifier):", round(accuracy_score(y_test, rf_pred) * 100, 2), "%")
```

The output of the code is displayed below the cell. It shows the "Classification Report (Random Forest Classifier)" and the "Accuracy Score (Random Forest Classifier): 91.12 %".

The Classification Report is presented as a table:

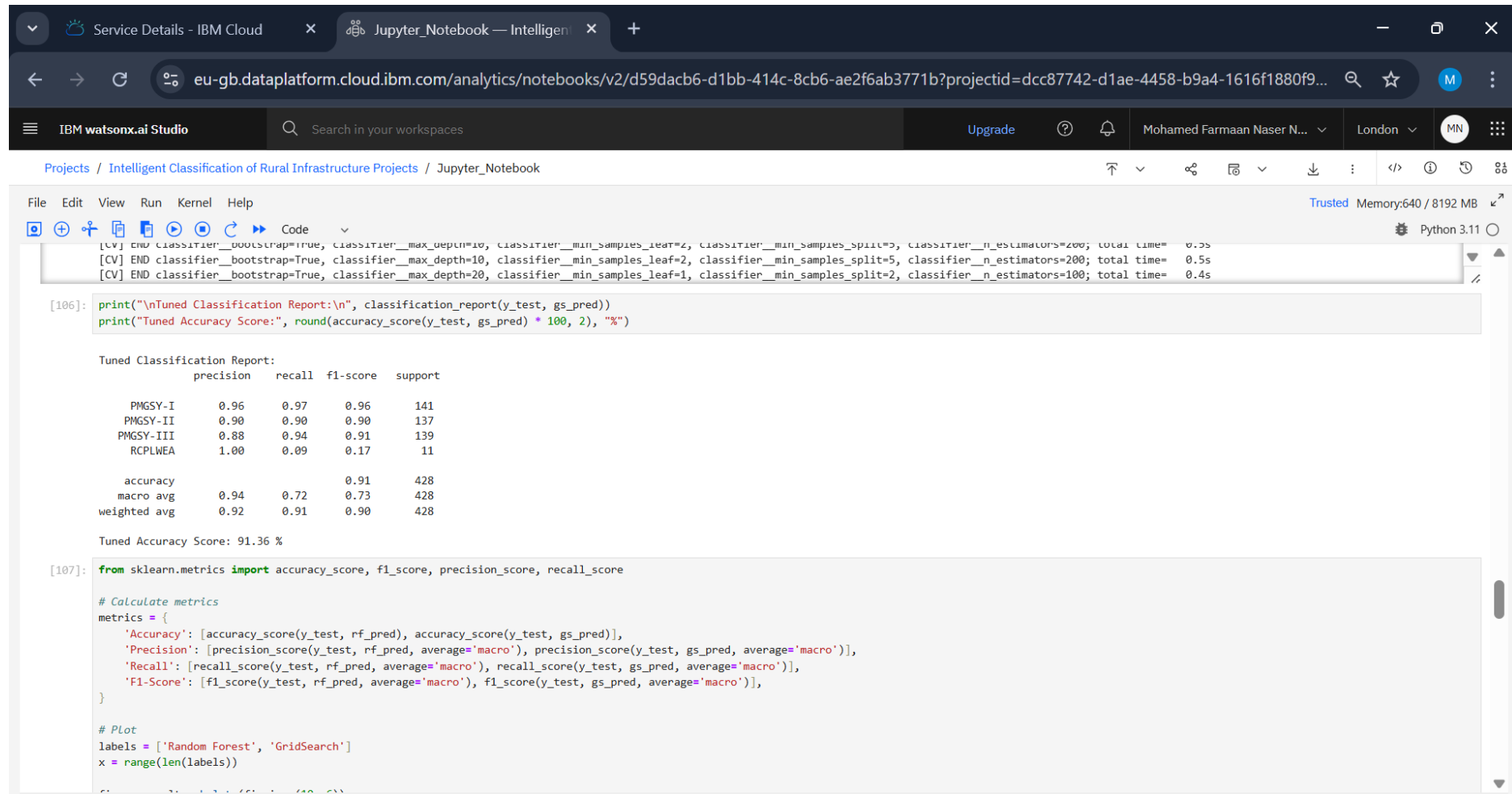
	precision	recall	f1-score	support
PMGSY-I	0.96	0.97	0.97	141
PMGSY-II	0.88	0.92	0.90	137
PMGSY-III	0.89	0.91	0.90	139
RCPLWEA	0.00	0.00	0.00	11
accuracy			0.91	428
macro avg	0.68	0.70	0.69	428
weighted avg	0.89	0.91	0.90	428

Below the table, the accuracy score is displayed: "Accuracy Score (Random Forest Classifier): 91.12 %".

The next cell contains the following Python code:

```
[101]: sample_data = pd.DataFrame([{'STATE_NAME': 'Tamil Nadu',
'DISTRICT_NAME': 'Chengalpattu',
'LENGTH_OF_ROAD_WORK_SANCTIONED': 197,
'NO_OF_ROAD_WORK_SANCTIONED': 403.113,
'NO_OF_BRIDGES_SANCTIONED': 0,
'COST_OF_WORKS_SANCTIONED': 90.8977,
'NO_OF_ROAD_WORKS_COMPLETED': 197,
'LENGTH_OF_ROAD_WORK_COMPLETED': 400.157,
'NO_OF_BRIDGES_COMPLETED': 0,
'EXPENDITURE_ACCURATE': 16.9303}])
```

RESULT



The screenshot displays a Jupyter Notebook within the IBM watsonx.ai Studio environment. The browser address bar shows the URL: eu-gb.dataplatform.cloud.ibm.com/analytics/notebooks/v2/d59dacb6-d1bb-414c-8cb6-ae2f6ab3771b?projectid=... The notebook interface includes a top navigation bar with 'Service Details - IBM Cloud', 'Jupyter_Notebook', and a search bar. Below this, the notebook title 'Intelligent Classification of Rural Infrastructure Projects / Jupyter_Notebook' is visible. The main area shows a code cell with the following content:

```
[106]: print("\nTuned Classification Report:\n", classification_report(y_test, gs_pred))
print("Tuned Accuracy Score:", round(accuracy_score(y_test, gs_pred) * 100, 2), "%")
```

The output of the code cell is a 'Tuned Classification Report' table and a 'Tuned Accuracy Score'.

	precision	recall	f1-score	support
PMGSY-I	0.96	0.97	0.96	141
PMGSY-II	0.90	0.90	0.90	137
PMGSY-III	0.88	0.94	0.91	139
RCPLWEA	1.00	0.09	0.17	11
accuracy			0.91	428
macro avg	0.94	0.72	0.73	428
weighted avg	0.92	0.91	0.90	428

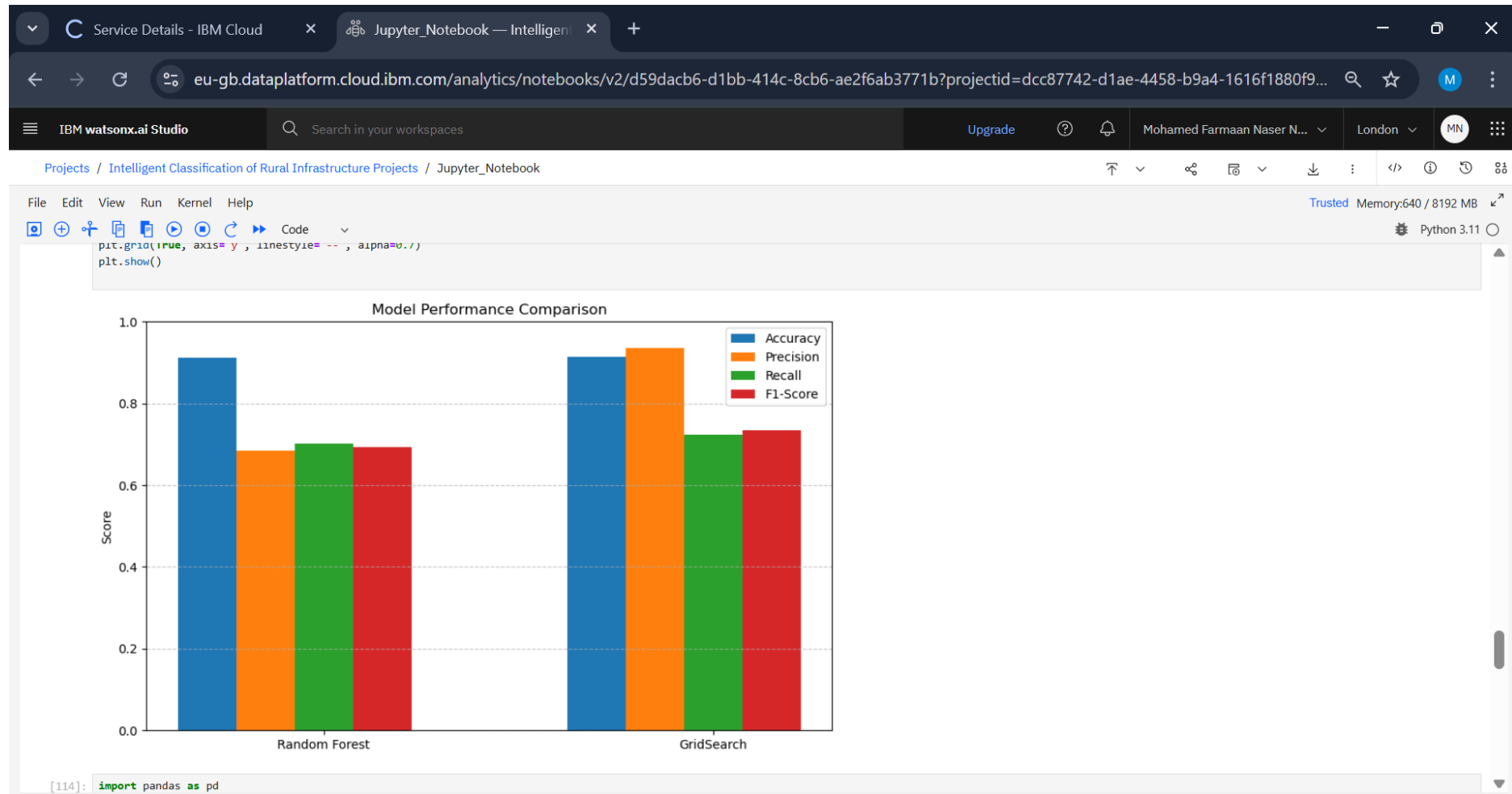
Tuned Accuracy Score: 91.36 %

```
[107]: from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score

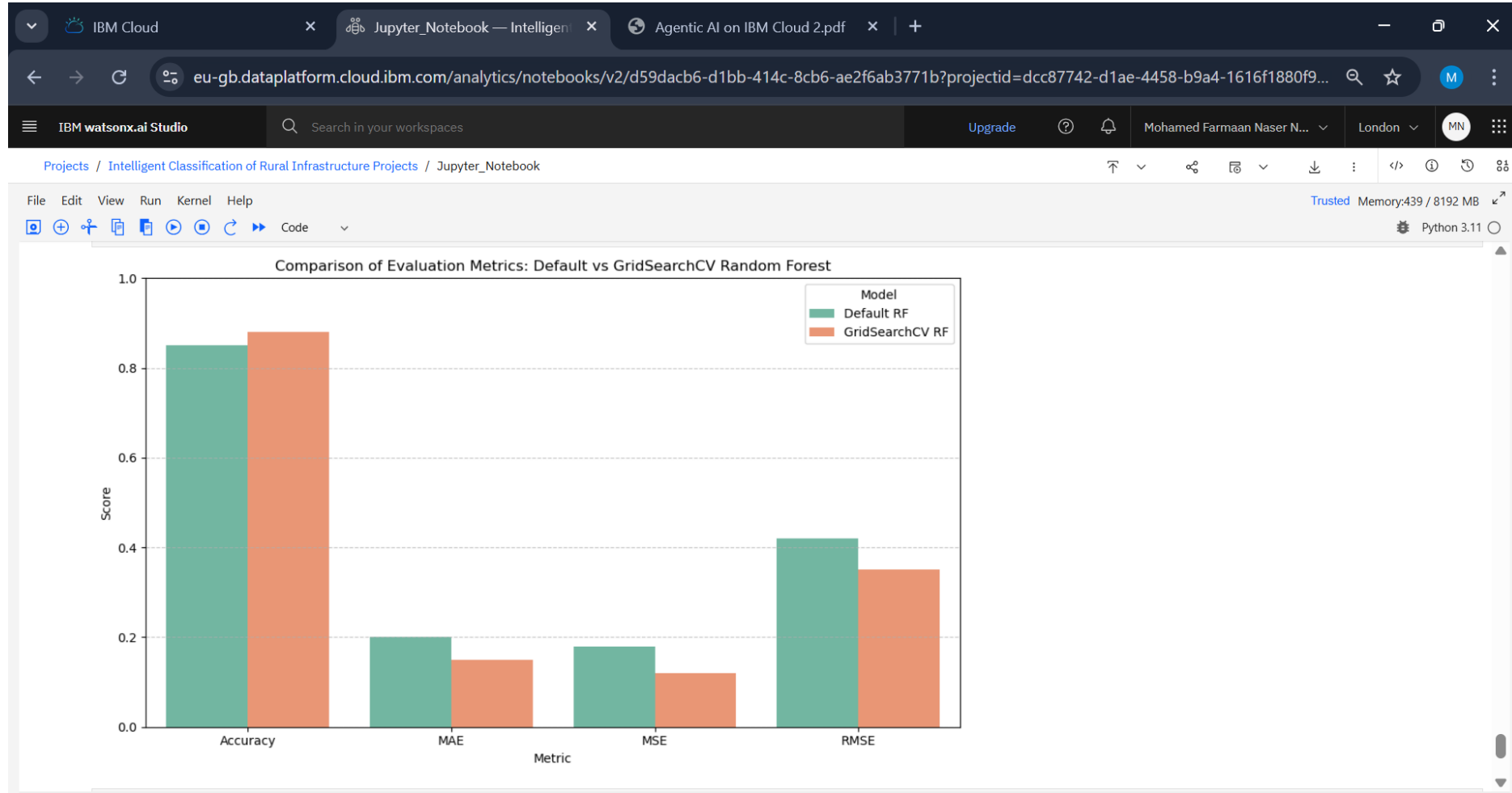
# Calculate metrics
metrics = {
    'Accuracy': [accuracy_score(y_test, rf_pred), accuracy_score(y_test, gs_pred)],
    'Precision': [precision_score(y_test, rf_pred, average='macro'), precision_score(y_test, gs_pred, average='macro')],
    'Recall': [recall_score(y_test, rf_pred, average='macro'), recall_score(y_test, gs_pred, average='macro')],
    'F1-Score': [f1_score(y_test, rf_pred, average='macro'), f1_score(y_test, gs_pred, average='macro')],
}

# Plot
labels = ['Random Forest', 'GridSearch']
x = range(len(labels))
```


RESULT



RESULT



CONCLUSION

- This project successfully developed a machine learning-based classification model to automatically categorize road and bridge construction projects under the correct PMGSY scheme (e.g., PMGSY-I, PMGSY-II, RCPLWEA, etc.). Using Random Forest with GridSearchCV for hyperparameter tuning, the model achieved high classification accuracy, demonstrating the effectiveness of ML in policy-driven infrastructure analysis.
- Implemented using IBM Watsonx.ai Studio and Jupyter Notebook, the solution effectively handles complex physical and financial data to make accurate scheme-level predictions. This approach can significantly reduce manual effort, improve classification accuracy, and enhance scalability in processing large volumes of government project data.
- Challenges faced included:
 - Ensuring consistent data preprocessing,
 - Handling class imbalances,
 - Interpreting model predictions for transparent reporting.
- Future improvements can include:
 - Integration with real-time government databases,
 - Exploring ensemble methods or deep learning for further accuracy gains.
- Overall, this work supports data-driven governance, promoting transparency, efficiency, and impact assessment in rural infrastructure development.

FUTURE SCOPE

- **Integration with Government MIS Platforms**

The model can be integrated into existing Management Information Systems (MIS) like OMMAS to automate scheme tagging during project entry.

- **Support for New and Evolving Schemes**

With regular retraining, the model can adapt to future schemes and updates in PMGSY guidelines, ensuring long-term scalability.

- **Improved Accuracy with More Diverse Data**

Incorporating additional features such as location data, terrain type, or socio-economic indicators can further enhance model accuracy.

- **Development of a Policy Analytics Tool**

The system can evolve into a decision-support tool for planners to assess scheme distribution, fund utilization, and regional project trends.

- **Multi-Scheme Classification**

Extend the model to multi-label classification where a single project might be associated with more than one scheme or funding source.

- **Real-time Data Pipeline**

Enable live classification as soon as new project data is entered, improving responsiveness in monitoring and auditing.

- **User Interface for Non-Technical Staff**

Build a lightweight web dashboard to allow district-level officers or field engineers to upload project data and get instant classification.

- **Explainability and Trust Building**

Integrate explainable AI methods (like SHAP values) to ensure transparency in predictions, promoting trust among government stakeholders.

REFERENCES

- Dataset Source:

AI Kosh – Pradhan Mantri Gram Sadak Yojana (PMGSY) Dataset

https://aikosh.indiaai.gov.in/web/datasets/details/pradhan_mantri_gram_sadak_yojna_pmgsy.html

- Github Repo Link:

<https://github.com/Farmaan-N/Intelligent-Classification-of-Rural-Infrastructure-Projects/tree/main>

IBM CERTIFICATIONS

In recognition of the commitment to achieve
professional excellence



Mohamed Farmaan Naser N

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence



Issued on: Jul 16, 2025

Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/b7bc29f3-0059-4f4f-9f57-b20f9fb413e8>



IBM CERTIFICATIONS

In recognition of the commitment to achieve
professional excellence



Mohamed Farmaan Naser N

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution



Issued on: Jul 16, 2025

Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/0570b1ac-767d-4e9d-a004-3a9926676a03>



IBM CERTIFICATIONS

IBM **SkillsBuild**

Completion Certificate



This certificate is presented to
Mohamed Farmaan Naser N

for the completion of
**Lab: Retrieval Augmented Generation with
LangChain**

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 27 Jul 2025 (GMT)

Learning hours: 20 mins



THANK YOU