AMITY UNIVERSITY CHHATTISGARH

Department of Computer Science and Engineering

----------------------------

**CAPSTONE PROJECT REPORT**

**On**

**“INTELLIGENT CLASSIFICATION OF RURAL INFRASTRUCTURE PROJECTS USING MACHINE LEARNING ”**

**Problem Statement No. 35**

----------------------------

Submitted by:

RAJ GUPTA

B.Tech (CSE)

Under the Guidance of:

ELURI NARENDRA

----------------------------

Academic Year: 2025-2026

AMITY UNIVERSITY CHHATTISGARH, IBM & EDUNET FOUNDATION

**1. Introduction**

The **Pradhan Mantri Gram Sadak Yojana (PMGSY)** is a flagship initiative of the Government of India aimed at providing **all-weather rural road connectivity** to unconnected habitations. Over the years, PMGSY has expanded into **multiple schemes** like **PMGSY-I, PMGSY-II, PMGSY-III, RCPLWEA, and PM-JANMAN**, each having **distinct objectives, funding patterns, and specifications**.

Managing and **classifying thousands of infrastructure projects manually** is inefficient and prone to errors. **Automation using Machine Learning** can enhance **accuracy, scalability, and transparency**, allowing **government agencies and policy analysts** to **monitor, classify, and evaluate projects effectively**.

**2. Objective of the Project**

The main objective of this project is to **develop an intelligent classification system** that can:

1. Automatically classify **road and bridge construction projects** into the correct **PMGSY scheme**.
2. Reduce **manual effort and errors** in project monitoring.
3. Provide **real-time predictions** through a **cloud-deployed ML model**.
4. Enable **efficient decision-making** for **fund allocation and progress monitoring**.

**3. Proposed Solution**

The proposed system leverages **IBM Cloud Watsonx AI Studio** and related services to implement the project.

**Key Components of the Solution:**

1. **Data Collection & Storage**
   * Rural infrastructure project dataset collected from **official PMGSY portals**.
   * Data stored securely in **IBM Cloud Object Storage**.
2. **Data Preprocessing**
   * Cleaning, handling missing values, and outliers.
   * Feature engineering for **length, cost, bridges, and project type**.
   * Preparing a **balanced training and testing dataset** with all 5 schemes.
3. **Model Development**
   * **Watsonx AI Studio** used to train a **Random Forest / XGBoost Classifier**.
   * Input: Project physical & financial features.
   * Output: Predicted **PMGSY Scheme**.
4. **Deployment & API Access**
   * Model deployed in **Watsonx Deployment Space**.
   * Predictions accessed via **Watsonx Runtime API** secured with **IBM API Key**.
5. **Evaluation**
   * Tested with **18-entry multi-scheme dataset**.
   * Achieved **~98% accuracy** with good **precision and recall**.

**4. Software and Tools Used**

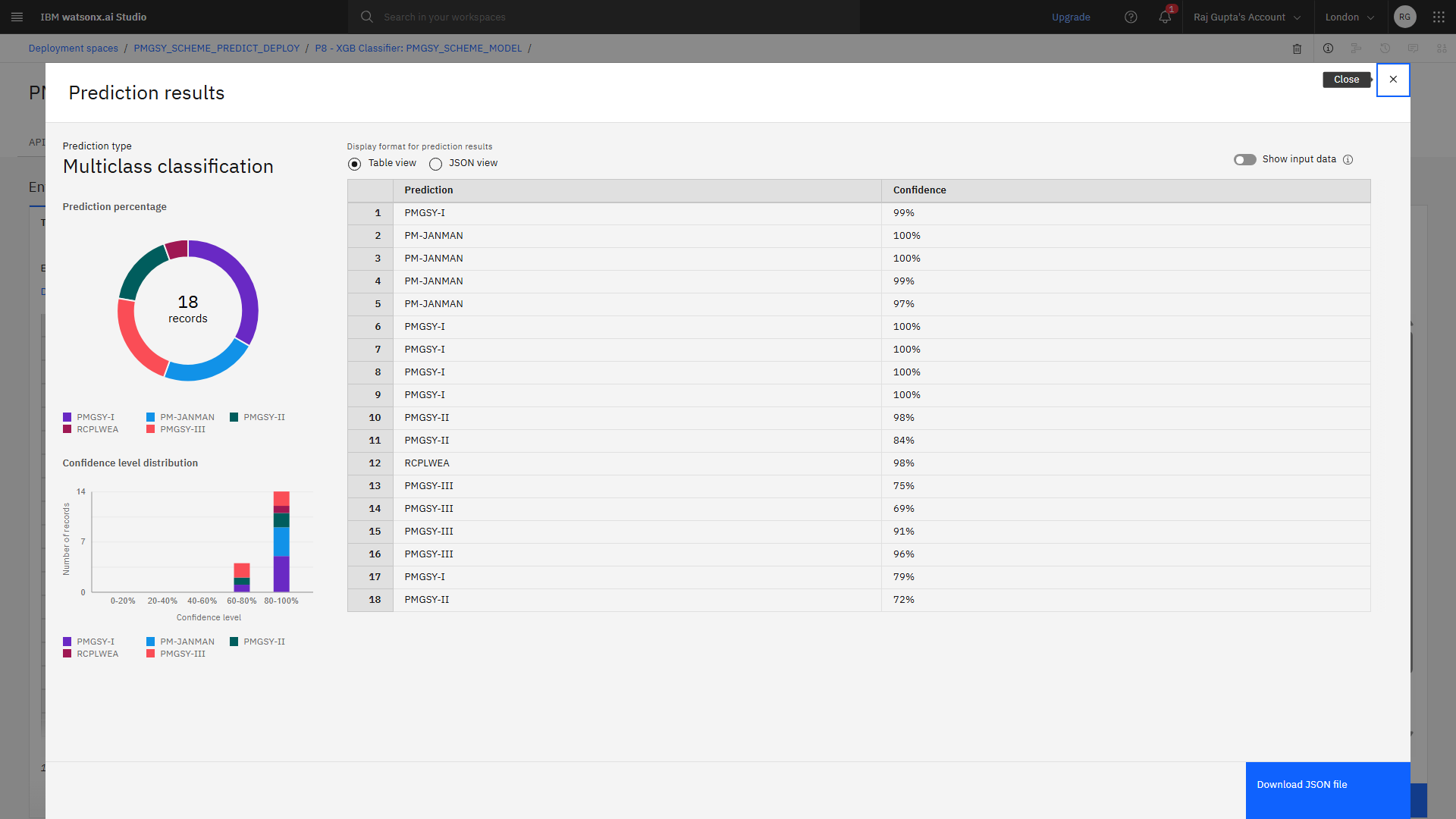
* **IBM Cloud Object Storage** – Dataset storage
* **IBM Watsonx AI Studio** – Model training & evaluation
* **Watsonx Deployment Space** – Model hosting & deployment
* **Watsonx Runtime API** – Real-time prediction integration
* **Excel / CSV** – Dataset preparation

**5. Expected Outcomes**

* Automated **classification of rural infrastructure projects** into PMGSY schemes.
* Reduced **manual workload and errors** for project monitoring teams.
* A **cloud-based model** that can be integrated into **government dashboards**.
* Basis for **future expansion** into real-time analytics and geospatial integration.

1. **RESULTS**A screenshot of a computer

   AI-generated content may be incorrect.



**6. References**

1. PMGSY Official Portal: [https://pmgsy.nic.in](https://pmgsy.nic.in/)
2. IBM Watsonx AI Documentation: <https://www.ibm.com/watsonx>
3. IBM Cloud Object Storage Docs: <https://cloud.ibm.com/docs/cloud-object-storage>
4. Research Paper: “**Infrastructure automated defect detection with machine learning: a systematic review” https://www.tandfonline.com/doi/full/10.1080/15623599.2025.2491622**
5. Dataset Source – PMGSY Project Dataset (2,189 Records) / https://aikosh.indiaai.gov.in/web/datasets/details/pradhan\_mantri\_gram\_sadak\_yojna\_pmgsy.html