**🌾 Crop Recommendation using IBM Watsonx.ai (AutoAI)**

**📌 Project Summary**

This project leverages IBM Cloud’s **Watsonx.ai AutoAI** to build and deploy a machine learning model that recommends the most suitable crop to grow based on soil and climate parameters. The objective is to support **precision agriculture** by helping farmers make data-driven decisions that maximize yield and efficiency.

**🧠 Objective**

To predict the ideal crop to cultivate based on:

* **N, P, K** – Nutrient content in the soil
* **Temperature (°C)**
* **Humidity (%)**
* **pH value** of the soil
* **Rainfall (mm)**

**🔧 Tools & Technologies Used**

* **IBM Watsonx.ai Studio (AutoAI)**
* **IBM Cloud Deployment Space**
* **Crop Recommendation Dataset** from [Kaggle](https://www.kaggle.com/datasets/siddharthss/crop-recommendation-dataset)

**🗂️ Workflow Summary**

1. Loaded and cleaned the dataset in Watsonx.ai
2. Ran an AutoAI experiment to generate and compare ML models
3. Selected the best-performing pipeline from the leaderboard
4. Saved and deployed the model to a deployment space
5. Tested the model using custom input values for prediction
6. Generated a **public API endpoint** for external integration

**✅ Outcome**

* Deployed a machine learning model with **~99% prediction confidence**
* Created a fully functional **no-code AI solution** for crop recommendation
* Enabled external access via a secure endpoint for integration into other platforms

**🌐 Public API Endpoint (for IBM Cloud users)**

If you have access to IBM Cloud and proper credentials, you can use the deployed model via the following **public endpoint**:

<https://au-syd.ml.cloud.ibm.com/ml/v4/deployments/c9919cd9-7478-4600-98a8-eed5189c94e8/predictions?version=2021-05-01>

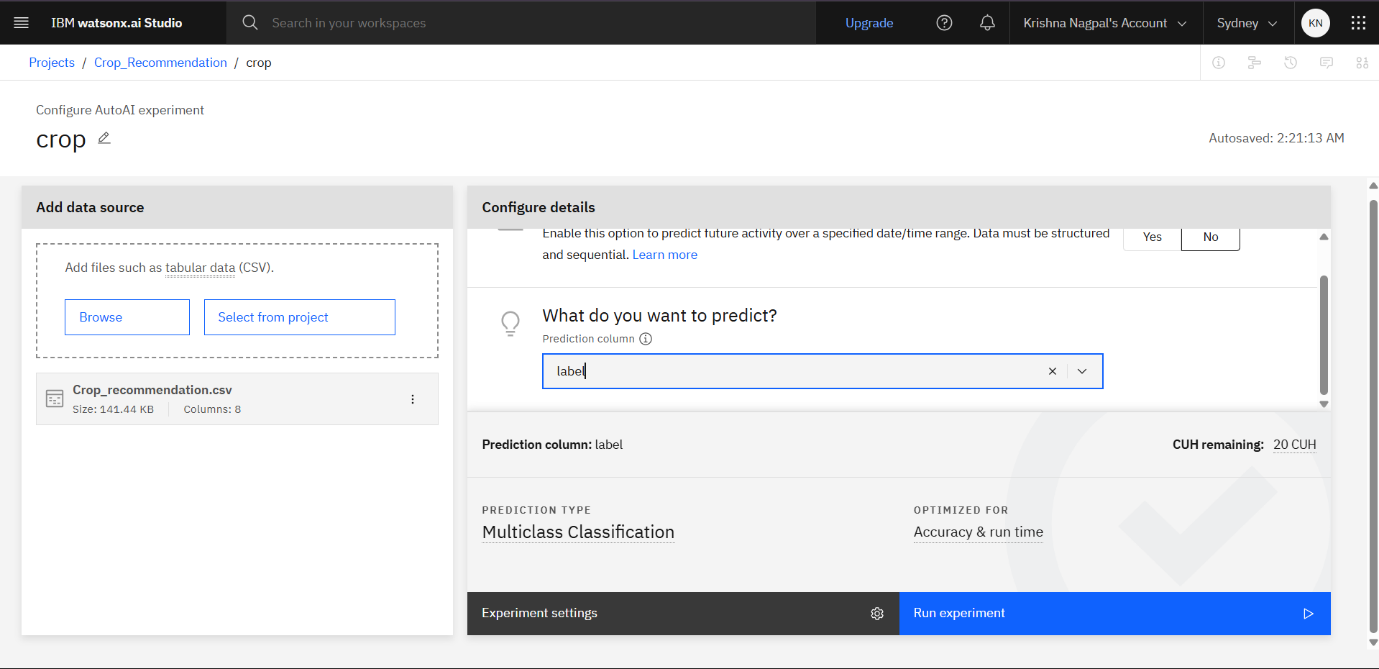
🔐 **Note**: To use this endpoint, you need:

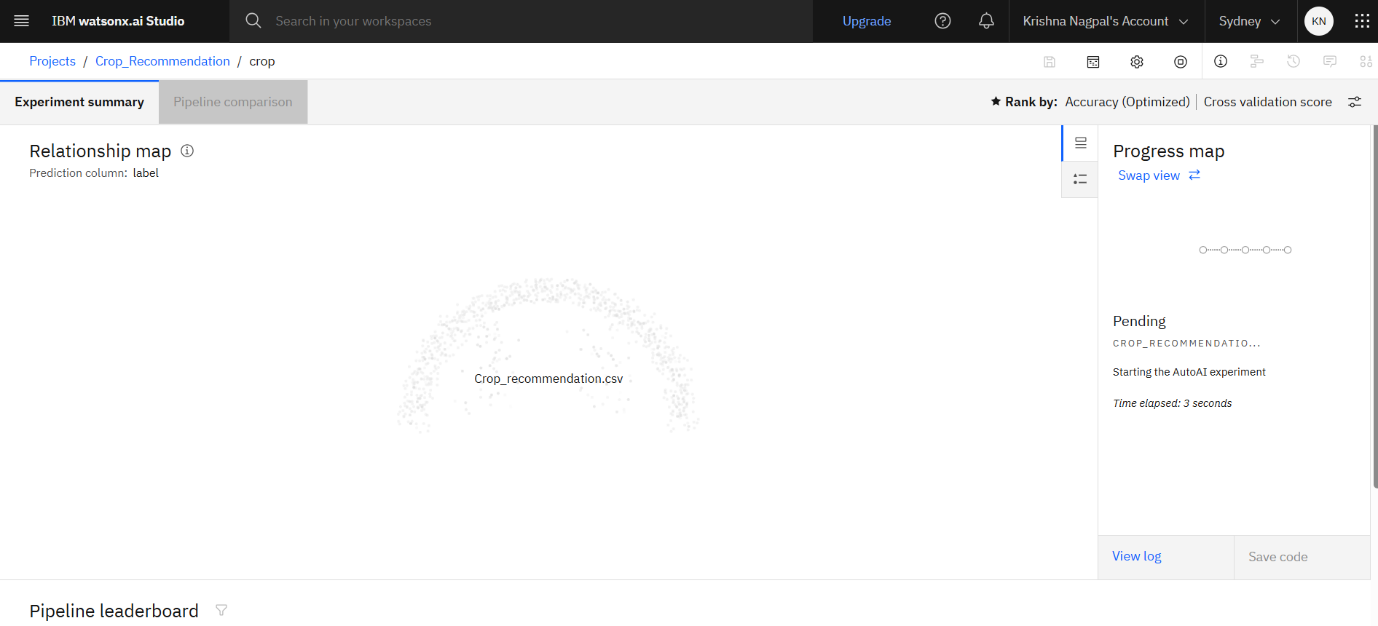
* An **IBM Cloud account**
* A valid **API key** or bearer token
* The correct **payload structure** in JSON format

If you're testing in Python or Postman, make sure to include the authentication token in your headers and provide input data matching the model schema (e.g., N, P, K, temperature, humidity, pH, rainfall).

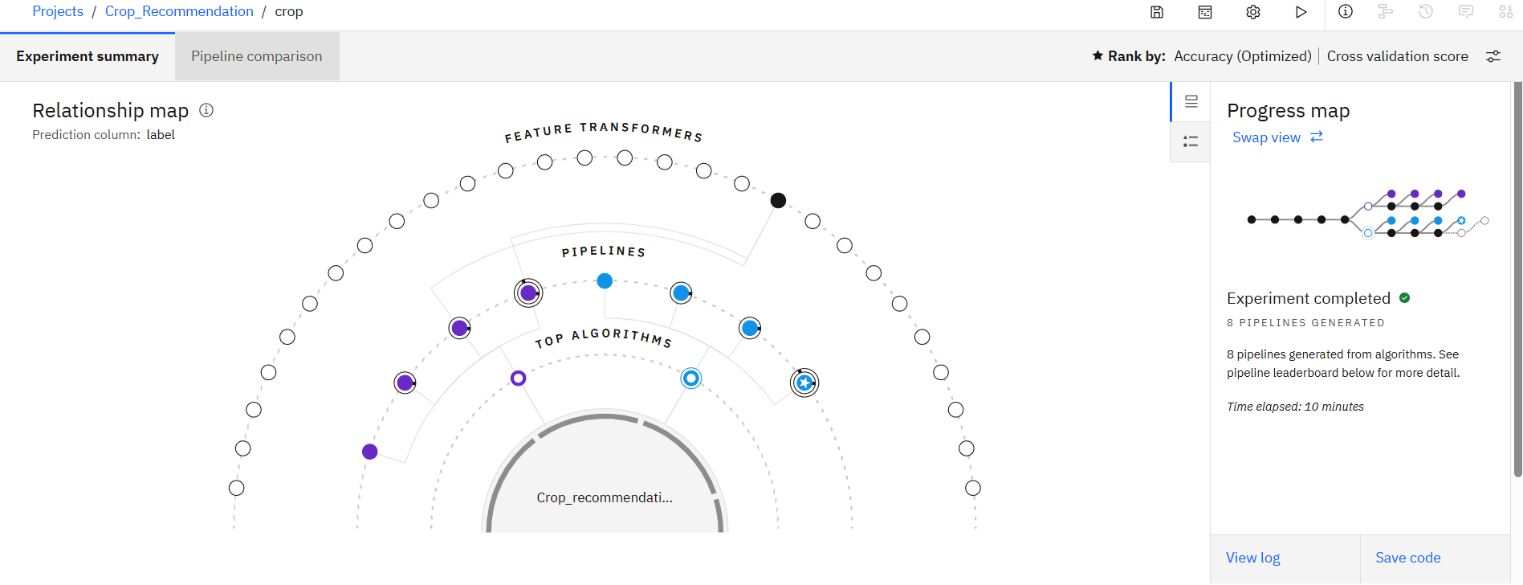
**📸 Model Deployment & Prediction – Screenshots**

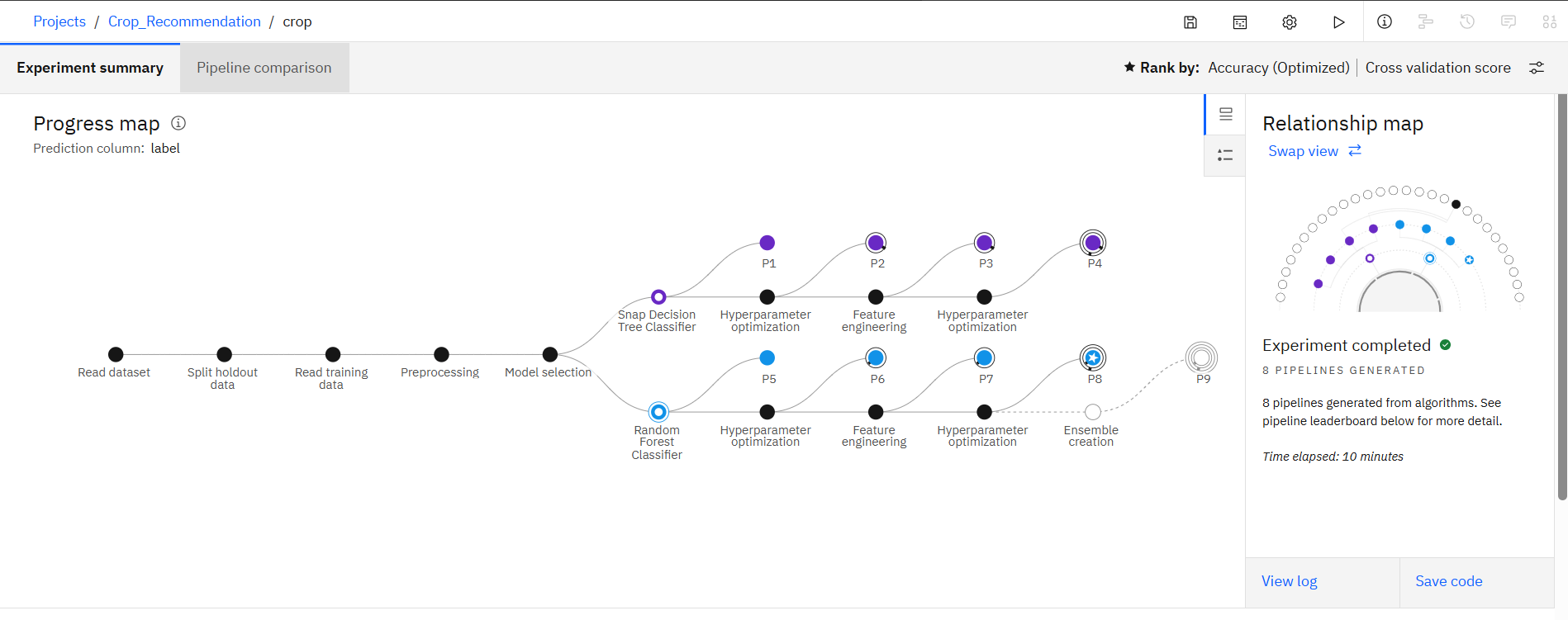
* **Upload the csv file**

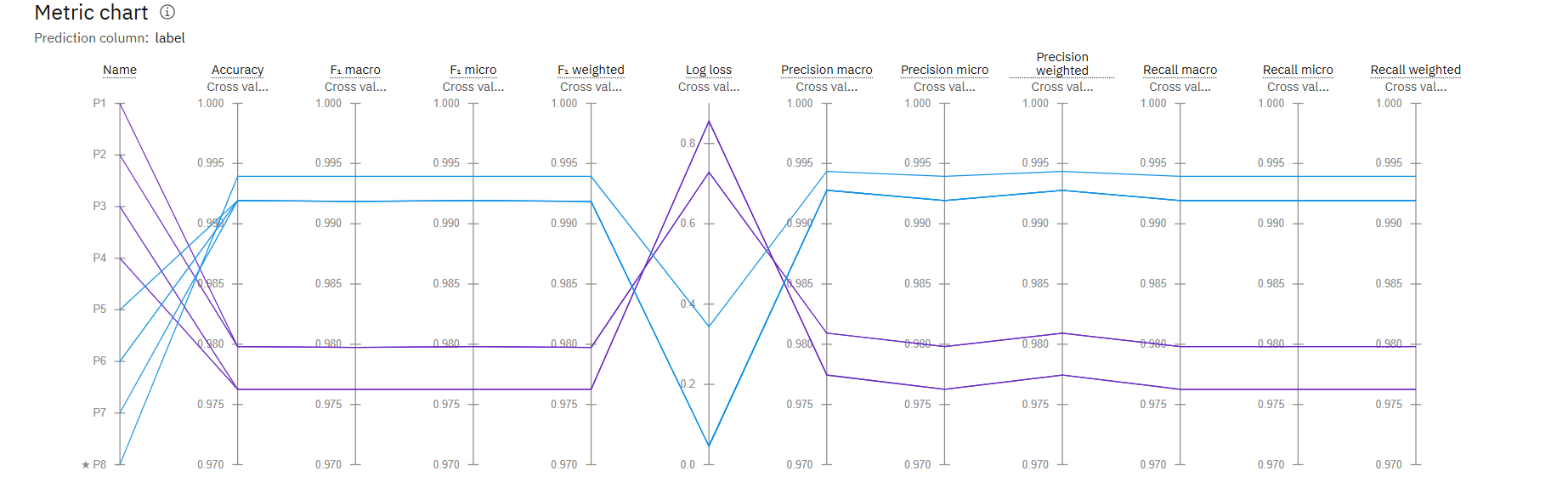


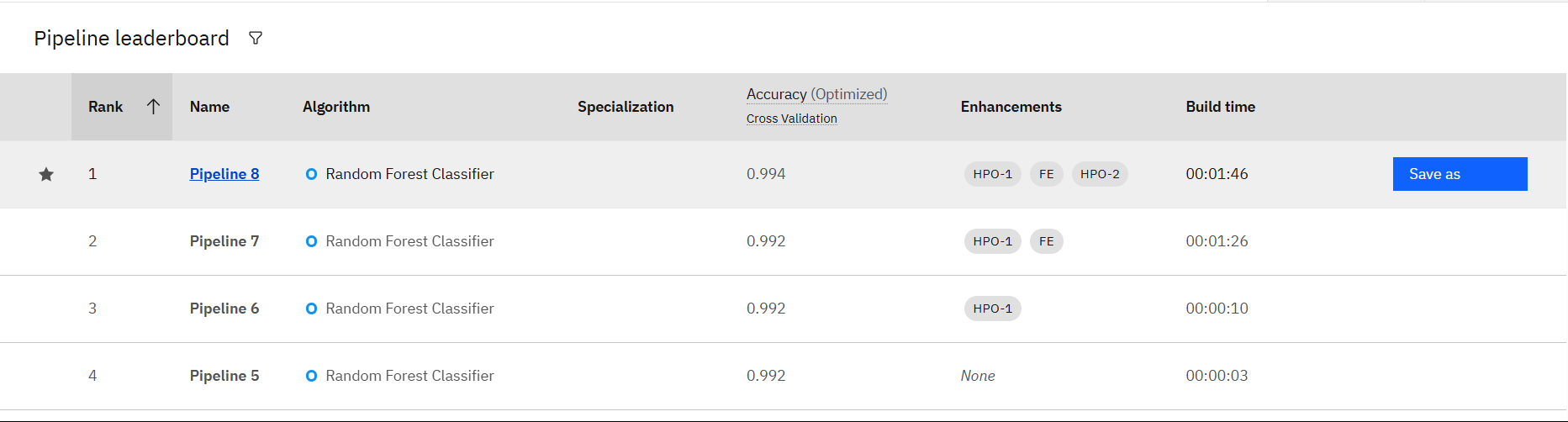


* Different charts for Comparison of Model









* Testing (Input details from csv file)

