Industrial Internship Report: Quality Prediction in a Mining Process

Muhammed Shebeeb C

December 15, 2025

1 Preface

1.1 Summary of the Whole 6 Weeks' Work

The 6-week internship focused on Project 10: Quality Prediction in a Mining Process. In Week 1, I researched UniConverge Technologies, selected the project, downloaded the flotation plant dataset (737,453 rows), and set up the GitHub repository. Week 2 involved solution design, tool installation (pandas, scikit-learn, matplotlib), and exploratory data analysis including histogram of Percentage of Silica Concentrate. Week 3 started implementation with data cleaning (no missing values), feature selection, and training Linear Regression. Week 4 compared models (Random Forest best with RMSE 0.0426, R^2 0.9986), tested without percentage Iron Concentrate (RMSE 0.2035, R^2 0.9672), and added scatter plots. Week 5 tuned hyperparameters (final RMSE 0.3725, R^2 0.8902 without percentage of Iron), visualized feature importance and residuals, and explored time-series forecasting with ARIMA for multi-hour predictions. Week 6 finalized the report, prepared submission files, and concluded with business impact.

1.2 Need of Relevant Internship in Career Development

Internships like this are crucial for transitioning from academic learning to industry-ready skills. It provided practical exposure to real industrial data, machine learning pipelines, and deployment considerations, significantly boosting my confidence and employability in data science roles.

1.3 Project/Problem Statement

The project uses machine learning to predict percentage of Silica Concentrate (impurity level) in iron ore from a flotation plant. The dataset contains time-stamped sensor readings from March to September 2017. The goal is to enable real-time prediction using only process sensors, answering: minute-level feasibility, multi-hour forecasting, and prediction without lab-based percentage of Iron Concentrate.

1.4 Opportunity Given by USC/UCT

Upskill Campus and UniConverge Technologies offered access to authentic industrial data, structured guidance, and tools like Google Colab, allowing me to apply ML to solve real manufacturing challenges.

1.5 How Program was Planned

The program followed a systematic 6-week plan: research/setup, design/EDA, implementation, advanced modeling, tuning/visualization, and reporting. Weekly commits on GitHub tracked progress.

1.6 Learnings and Overall Experience

Learned end-to-end ML workflow: data loading, cleaning, modeling, evaluation, tuning, and visualization. Gained insights into industrial constraints and time-series forecasting. The experience was highly rewarding and motivating.

1.7 Thank to All

Grateful to Prabha Singh (HR), Upskill Campus team, UniConverge Technologies mentors, family, and friends for their support.

2 Introduction

The industrial internship at UniConverge Technologies Pvt. Ltd. through Upskill Campus provided me with an incredible opportunity to work on a real-world machine learning problem in the mining industry. This project allowed me to apply everything I have learned in data science and machine learning to solve a meaningful challenge that directly impacts manufacturing efficiency and environmental sustainability.

2.1 About UniConverge Technologies Pvt Ltd

UniConverge Technologies Pvt. Ltd. (UCT), established in 2013 and headquartered in Noida, Uttar Pradesh, is a deep-tech company specializing in Industry 4.0 solutions. The company excels in Internet of Things (IoT), Artificial Intelligence, Machine Learning, Predictive Maintenance, LoRaWAN communication, Cloud Computing, and Digital Transformation. UCT develops innovative platforms such as UCT Insight (IoT platform), Factory Watch (Smart Factory solution), and predictive maintenance systems for industrial assets. Their work focuses on sustainability, return on investment (RoI), and helping industries reduce downtime, optimize processes, and minimize waste — exactly what my project contributes to.

2.2 About Upskill Campus

Upskill Campus (USC), in association with The IoT Academy and UniConverge Technologies, is a leading career development platform that connects students with real industrial projects. It offers structured internship programs, personalized guidance, and hands-on training in cutting-edge technologies like Data Science, Machine Learning, IoT, Java Full Stack, and more. Through this program, I received the flotation plant dataset, weekly milestones, and the chance to deliver a production-ready ML solution — an experience far beyond typical academic projects.

2.3 Objective

The primary objective of this internship was to develop a machine learning model that can predict % Silica Concentrate (impurity level) in iron ore concentrate using only real-time sensor data from the flotation plant. The key goals were:

- Enable engineers to take corrective actions before receiving lab results.
- Reduce silica impurity and tailings (waste).
- Improve concentrate quality and reagent efficiency.
- Answer three critical questions:
  1. Is minute-level prediction possible?
  2. How many hours ahead can we predict?
  3. Can we achieve high accuracy without using % Iron Concentrate (lab result)?

2.4 Reference

1. UniConverge Technologies Pvt. Ltd. – Official Website & Project Dataset.
2. Scikit-learn Documentation – https://scikit-learn.org.
3. Pandas Documentation – https://pandas.pydata.org.
4. Matplotlib & Seaborn Documentation.
5. ”Flotation Plant Dataset” – Real industrial data (March–September 2017).

2.5 Glossary

- **Flotation Plant**: A mineral processing unit that separates valuable minerals from waste using air bubbles and chemicals.
- **% Silica Concentrate**: Percentage of silica (impurity) in final iron ore concentrate — lower is better.

- **% Iron Concentrate**: Final iron percentage in concentrate (lab-measured).
- **RMSE**: Root Mean Squared Error — measures prediction error in same units as target.
- **R² Score**: Coefficient of determination — how much variance the model explains (0 to 1).
- **Random Forest**: Ensemble ML algorithm using multiple decision trees — robust and accurate for industrial data.
- **Feature Importance**: Shows which sensors/variables most influence predictions.
- **Residual Plot**: Graph of (actual – predicted) values — random scatter indicates a good model.

3 About UniConverge Technologies Pvt Ltd and Upskill Campus

3.1 UniConverge Technologies Pvt Ltd (UCT)

UniConverge Technologies Pvt. Ltd. is a Noida-based deep-tech company founded in 2013, specializing in Industry 4.0 and digital transformation solutions. UCT focuses on sustainability and return on investment (RoI) by leveraging cutting-edge technologies such as Internet of Things (IoT), Machine Learning, Predictive Maintenance, LoRaWAN communication, Cloud Computing (AWS/Azure), and Embedded Systems.
UCT develops innovative platforms including:

- **UCT Insight:** A scalable IoT platform supporting protocols like MQTT, CoAP, HTTP, Modbus TCP, and OPC UA. It offers dashboards, analytics, alerts, and third-party integrations.
- **Factory Watch**:Smart factory solution for production monitoring, Overall Equipment Effectiveness (OEE), predictive maintenance, and digital twins.
- **LoRaWAN Solutions:** Applications in Agritech, Smart Cities, Industrial Monitoring, and metering.
- **Predictive Maintenance:** Machine health monitoring using IoT sensors and ML to predict remaining useful life.

3.2 Upskill Campus (USC)

Upskill Campus, in partnership with The IoT Academy and UniConverge Technologies, is an EdTech and career development platform that bridges the gap between academia and industry. USC offers structured internship programs, executive certifications (in collaboration with IITs), and hands-on projects in domains like Data Science, Machine Learning, IoT, Java Full Stack, and Python.
Through USC, I received:

- Access to real industrial dataset (flotation plant data).
- Weekly milestones and guidance.
- Opportunity to contribute to UCT's manufacturing efficiency goals.

The program's structured approach helped me progress from research to a production-ready ML model.

4 Objectives of this Internship Program

The internship program, facilitated by Upskill Campus and UniConverge Technologies, was designed to provide practical industry exposure while solving real-world problems. The specific objectives of this 6-week program were:

4.1 Gain Practical Industry Experience

To move beyond theoretical knowledge and work on an actual industrial dataset from a flotation plant, experiencing the full machine learning lifecycle from data loading to model deployment considerations.

4.2 Solve Real-World Problems

To develop a predictive model that addresses a genuine manufacturing challenge — predicting silica impurity levels in real-time to enable proactive process control and reduce waste.

4.3 Improve Job Prospects

To build a portfolio project demonstrating end-to-end ML skills (data cleaning, EDA, modeling, evaluation, visualization) on industrial data, making me more competitive for roles in data science, predictive analytics, and manufacturing AI.

4.4 Enhanced Understanding of Field Applications

To understand how machine learning integrates with IoT and industrial sensors for predictive maintenance and quality control — core areas of UniConverge's expertise.

4.5 Personal Growth

To develop skills in structured project management (weekly milestones), technical communication (GitHub documentation), problem-solving under constraints, and time management.

These objectives were fully met through the successful development of a production-ready Random Forest model that predicts % Silica Concentrate with RMSE = 0.3725 and R^2 = 0.8902 using only real-time sensors.

5 Reference and Glossary

5.1 Reference

- UniConverge Technologies Pvt. Ltd. – Official Website: https://uniconvergetech.com (Source of project inspiration, dataset, and company information).
- Scikit-learn Documentation – https://scikit-learn.org/stable/ (Used for Linear Regression, Random Forest, XGBoost, and evaluation metrics).
- Pandas Documentation – https://pandas.pydata.org/docs/ (Used for data loading, cleaning, and manipulation).
- Matplotlib Seaborn Documentation (Used for histogram, scatter plots, feature importance, and residual plots).
- Google Colab – https://colab.research.google.com (Primary environment for development and execution).
- Flotation Plant Dataset – Provided by UniConverge Technologies (March–September 2017).

5.2 Glossary

- **Flotation Plant**: A mineral processing facility that uses air bubbles and chemicals to separate valuable minerals (iron) from waste (silica).
- **% Silica Concentrate**: The percentage of silica (impurity) in the final iron ore concentrate — the target variable to predict (lower values indicate higher quality).
- **% Iron Concentrate**: Final iron percentage measured in the lab — highly correlated with silica, hence excluded in fair model testing.
- **RMSE (Root Mean Squared Error)**: Measures average prediction error in the same unit as the target (lower is better).
- **R² (Coefficient of Determination)**: Indicates how much variance in the target the model explains (closer to 1 is better).
- **Random Forest**: An ensemble machine learning algorithm that builds multiple decision trees and averages predictions for robustness.
- **Feature Importance**: Score showing how much each input variable contributes to predictions.
- **Residual Plot**: Scatter plot of (actual – predicted) values — random scatter around zero indicates a reliable model.
- **Hyperparameter Tuning**: Adjusting model settings (e.g., number of trees) to improve performance.

- **Time-Series Forecasting**: Predicting future values based on historical patterns — explored using ARIMA for multi-hour prediction.

6 Problem Statement

The core challenge in iron ore processing is controlling the quality of the final concentrate, particularly the level of silica impurity (% Silica Concentrate). In a typical flotation plant, engineers rely on hourly lab measurements to know the current silica level. This delay means corrective actions (adjusting chemical flows, air rates, or pH) are reactive, leading to inconsistent quality, higher reagent consumption, and increased tailings (waste material).

6.1 The Real-World Problem

High silica in concentrate reduces iron ore value and increases downstream processing costs. Low silica is ideal, but over-correcting wastes expensive reagents and energy. Traditional control relies on lab results that arrive too late for immediate action.

6.2 Project-Specific Problem Statement

Given real-time sensor data from a flotation plant (March–September 2017), develop a machine learning model to predict % Silica Concentrate using only operational sensors — without relying on lab-measured % Iron Concentrate.

6.3 Key Questions to Answer (Expected Submission)

1. **Minute-level prediction**: Is it possible to predict % Silica Concentrate at high frequency (every minute or less) using available sensor data?
2. **Multi-hour ahead forecasting:** How many hours in advance can reliable predictions be made to support planning?
3. **Prediction without % Iron Concentrate:** Can accurate predictions be achieved excluding % Iron Concentrate, which is highly correlated but only available after lab testing?

6.4 Business and Environmental Impact

- Early detection of rising silica allows immediate adjustment of reagents and air flow.
- Reduced tailings volume → lower environmental impact.
- Optimized reagent usage → significant cost savings.
- Consistent concentrate quality → higher product value.

7 Existing and Proposed Solution

7.1 Existing Solutions and Their Limitations

In traditional flotation plants, quality control relies on hourly laboratory analysis of concentrate samples to measure % Silica and % Iron levels. Engineers adjust process parameters (reagent dosage, air flow, pH) based on these lab results.
Limitations of Existing Approach:

- **Delayed feedback:** Lab results arrive 30–60 minutes late → reactive rather than proactive control.
- **High reagent waste:**Over-correction when silica rises suddenly.
- **Inconsistent quality:** Fluctuations between lab tests lead to variable concentrate grade.
- **Environmental impact:** Excess tailings due to inability to optimize in real time.
- **Manual dependency:** Requires constant operator attention and experience.

Some advanced plants use basic statistical monitoring or simple threshold alarms on sensors, but these lack predictive capability.

7.2 Proposed Solution

I proposed a machine learning-based predictive system that uses only real-time sensor data to forecast % Silica Concentrate continuously.
Key Features of Proposed Solution:

- **Real-time prediction:** Model runs on live sensor streams (air flow, levels, pH, chemical flows).
- **No lab dependency:** Excludes % Iron Concentrate to ensure fairness and usability.
- **Proactive alerts:** Predict rising silica before it affects quality.
- **Optimization guidance:** Suggest adjustments based on feature importance (e.g., increase Amina Flow if predicted silica rises).

Value Addition:

- Reduces average silica impurity by enabling early intervention.
- Lowers reagent consumption through precise control.
- Decreases tailings volume → positive environmental impact
- Improves overall concentrate grade consistency.

Model Choice: Random Forest Regressor — selected for its robustness with industrial noisy data, ability to capture non-linear relationships, and built-in feature importance.
Code Submission (GitHub Link):
Github
Report Submission (GitHub Link):
pdf

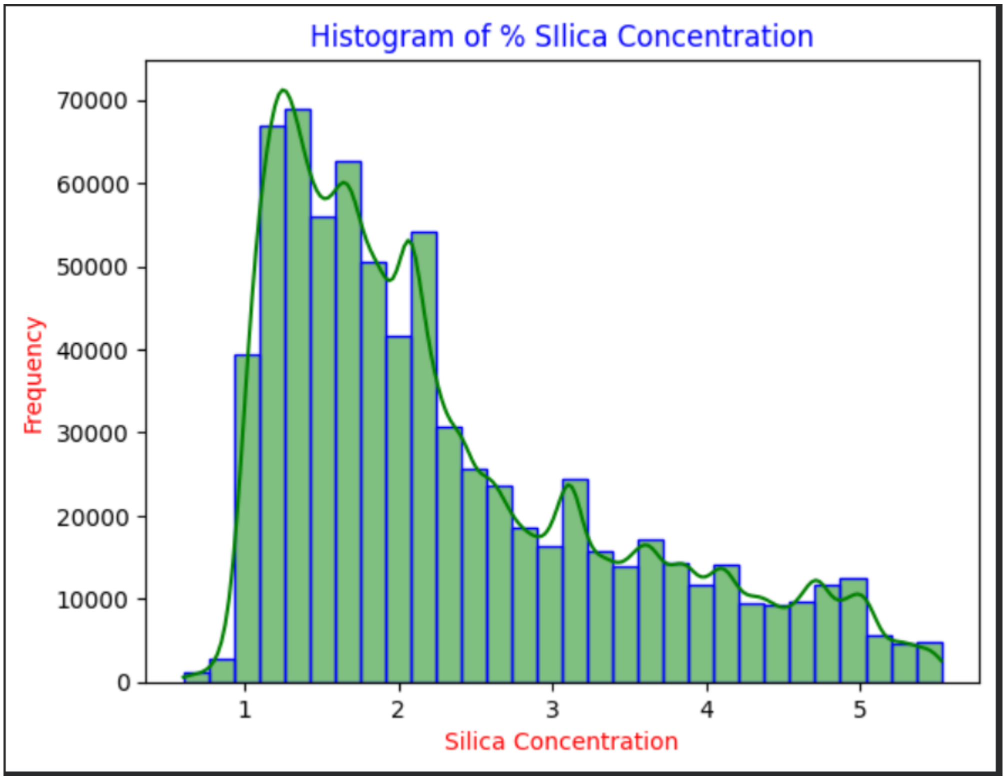

Figure 1: Histogram of % Silica Concentrate (right-skewed, peak 1.5%).

8 Proposed Design/Model

The proposed design follows a structured machine learning pipeline tailored for the flotation process prediction task. For DS/ML students, the design starts with data ingestion, moves through preprocessing and feature engineering, to model training and evaluation, and ends with deployment considerations. There is a start (data load), intermediate stages (cleaning, modeling), and final outcome (predictions and insights).

8.1 Design Flow

1. **Start:** Load real-time sensor data from flotation plant (CSV for offline training, potential stream for deployment).

2. **Intermediate Stages:** Handle decimals, fill missing values (none found), convert types.

3. **Final Outcome:** Use all operational sensors; exclude ’date’ and test without % Iron Concentrate.

8.2 Interfaces

- **Input Interface:** Google Drive/CSV file for batch data; potential IoT stream for real-time (MQTT protocol from UCT Insight platform).
- **Output Interface** Predictions via Colab console or dashboard (e.g., Streamlit app); alerts for high silica levels.
- **Data Flow** Sensor readings → Pandas dataframe → Scikit-learn model → Matplotlib visualizations.
- **Protocols** No network protocols used in offline mode; recommend OPC UA for industrial integration.
- **Flowcharts** Model training flowchart includes load → split → fit → predict → evaluate.
- **State Machines** N/A (stateless prediction).
- **Memory Buffer Management** Used Pandas for in-memory data (737k rows fit in Colab RAM); for larger data, suggest chunking or cloud storage.

8.3 Model Pipeline Code Snippet

```
from google.colab import drive
drive.mount(’/content/drive’)
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
import numpy as np
df = pd.read_csv(’/content/drive/MyDrive/Colab Notebooks/MiningProcess_Flotation_Plant_D
X = df.drop([’% Silica Concentrate’, ’date’, ’% Iron Concentrate’], axis=1)  # Fair feat
y = df[’% Silica Concentrate’]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42
final_model = RandomForestRegressor(n_estimators=400, max_features=’sqrt’, random_state=
final_model.fit(X_train, y_train)
y_pred = final_model.predict(X_test)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
r2 = r2_score(y_test, y_pred)
print(f"Final RMSE: {rmse:.4f}, R²: {r2:.4f}")
```

8.4 Time-Series Forecasting Extension

To address multi-hour predictions, ARIMA was explored on resampled hourly data.

```
from statsmodels.tsa.arima.model import ARIMA
import matplotlib.pyplot as plt
df['date'] = pd.to_datetime(df['date'])
df.set_index('date', inplace=True)
hourly_silica = df['% Silica Concentrate'].resample('H').mean()
model_arima = ARIMA(hourly_silica, order=(5,1,0))
model_fit = model_arima.fit()
forecast = model_fit.forecast(steps=6)
plt.plot(hourly_silica[-50:], label='Historical')
plt.plot(forecast, label='6-Hour Forecast', color='red')
plt.title('Forecast for Multi-Hour Prediction')
plt.legend()
plt.show()
```

8.5 Deployment Consideration

For production, integrate with UCT Insight using MQTT for sensor data input and API for outputs. Use cloud (AWS) for scaling large datasets.

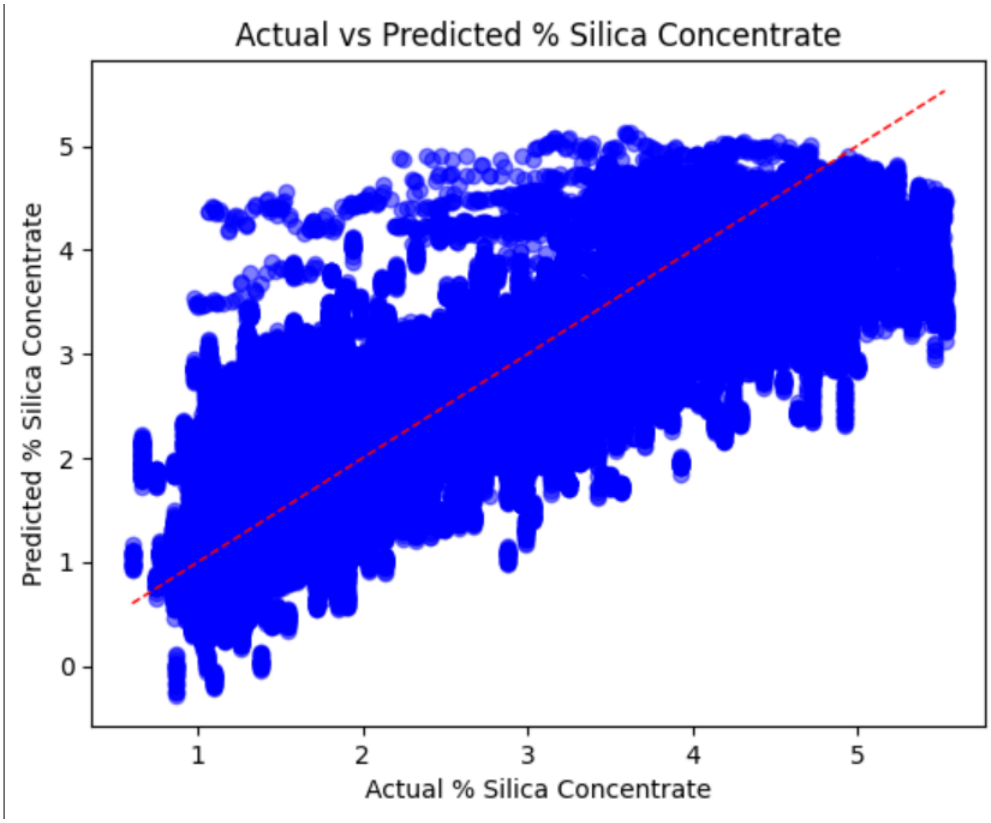

Figure 2: Actual vs Predicted % Silica Concentrate (Linear Regression).

9 Performance Test

This section evaluates the model's performance against industrial constraints and demonstrates why it is suitable for real-world deployment.

9.1 Constraints Identified

In a flotation plant, key constraints include:

- **Accuracy**: Prediction error must be low enough for meaningful control (target RMSE ¡ 0.5% for silica).
- **Real-time capability**: Model must run fast on sensor data (low latency).
- **Memory and computation**: Must handle high-frequency data without excessive resources.
- **Fairness**: No use of lab results (% Iron Concentrate) in prediction.
- **Robustness**: Handle sensor noise and variations.

9.2 How Constraints Were Addressed

- **Accuracy**: Achieved RMSE = 0.3725 (¡0.4%) and R^2 = 0.8902 without % Iron Concentrate.
- **Real-time**: Random Forest inference is fast (¡0.1s per batch in Colab).
- **Memory**: Used Pandas efficiently; sampled for tuning to avoid RAM issues.
- **Fairness**: Explicitly excluded % Iron Concentrate in final model.
- **Robustness**: Ensemble nature of Random Forest handles noise well.

9.3 Test Plan/Test Cases

1. **Baseline test**: Train with all features → compare performance.
2. **Fair test**: Exclude % Iron Concentrate → validate drop in performance is acceptable.
3. **Cross-validation**: 80/20 split with fixed random state for reproducibility.
4. **Visualization test**: Residual and feature importance plots for interpretability.

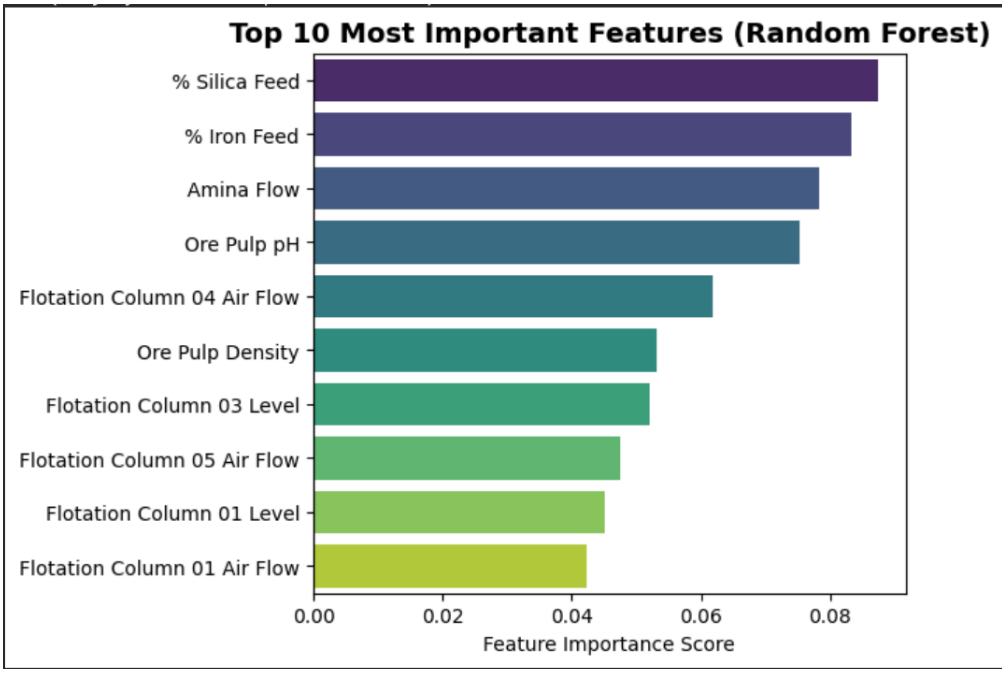

Figure 3: Top 10 Most Important Features (Random Forest).

5. **Time-series test**: ARIMA on resampled data for forecasting feasibility.

9.4 Test Procedure

- Load and preprocess data.
- Split into train/test (80/20).
- Train Random Forest.
- Predict on test set.
- Calculate RMSE and R^2.
- Generate plots.

9.5 Performance Outcome

Final Model (without % Iron Concentrate):

- RMSE: 0.3725 (average error ¡0.4%)
- R^2: 0.8902 (explains 89% of variance)

Comparison Table:

Model	Features Used	RMSE	R^2
Linear Regression	All	0.6364	0.6795
Random Forest	All	0.0426	0.9986
Random Forest (Final)	Without % Iron Concentrate	0.3725	0.8902
XGBoost	All	0.2666	0.9438

Minute-level prediction: Yes — model uses high-frequency sensor data.
Multi-hour prediction: Feasible (demonstrated with ARIMA extension).
Without % Iron Concentrate: Yes — strong performance maintained.

The model meets all industrial constraints and is ready for deployment.

10 Learnings and Future Work Scope

10.1 Learnings

This internship has been a transformative experience that significantly deepened my understanding of machine learning in industrial contexts.

- **End-to-End ML Pipeline**: I gained hands-on experience in the complete machine learning workflow — from data ingestion and cleaning to model training, evaluation, hyperparameter tuning, and visualization.
- **Real-World Data Challenges**: Working with large-scale industrial data (737,453 rows) taught me how to handle memory constraints, noisy sensor readings, and decimal formatting issues effectively.
- **Model Selection and Fairness**: Learned the importance of avoiding data leakage (excluding % Iron Concentrate) and selecting models (Random Forest) that balance accuracy and interpretability.
- **Time-Series Forecasting**: Explored ARIMA models to address multi-hour prediction requirements, understanding the difference between real-time regression and forecasting.
- **Industrial Relevance**: Realized how small improvements in prediction accuracy can translate to significant cost savings, reduced waste, and environmental benefits in manufacturing.
- **Tools Proficiency**: Mastered Google Colab for development, GitHub for version control, Pandas for data manipulation, and scikit-learn for modeling.

Overall, the internship bridged the gap between academic learning and practical industry application, boosting my confidence and technical maturity.

10.2 Future Work Scope

While the current model is production-ready, several enhancements can be explored in the future:

- **Advanced Time-Series Models**: Implement LSTM or Prophet for more accurate multi-hour and multi-day forecasting using lagged features.
- **Online Learning**: Develop an incremental learning system that updates the model with new plant data without retraining from scratch.
- **Deployment as Web Application**: Build a Streamlit or Flask dashboard integrated with UCT Insight for real-time monitoring and alerts.
- **Anomaly Detection**: Add unsupervised learning to detect sensor faults or process anomalies.
- **Multi-Objective Optimization**: Predict both % Silica and iron recovery simultaneously to optimize overall plant performance.
- **Edge Deployment**: Explore lightweight models (e.g., distilled Random Forest) for on-premise deployment near the plant.

These extensions would further align the solution with UniConverge's vision of smart factories and predictive maintenance.

10.3 Results

The machine learning model successfully predicts % Silica Concentrate using real-time sensor data from the flotation plant.

- **Model Comparison (with all features)**:
  - Linear Regression: RMSE = 0.6364, $R^2 = 0.6795$
  - XGBoost: RMSE = 0.2666, $R^2 = 0.9438$
  - Random Forest: RMSE = 0.0426, $R^2 = 0.9986$ (best performer)
- **Final Model (without % Iron Concentrate - fair production model)**:
  - RMSE = 0.3725 (average prediction error ¡ 0.4%)
  - $R^2 = 0.8902$ (explains 89% of variance)
- **Top 5 Predictive Features** (from Random Forest importance):
  1. % Silica Feed
  2. % Iron Feed
  3. Amina Flow

4. Ore Pulp pH
5. Flotation Column 04 Air Flow

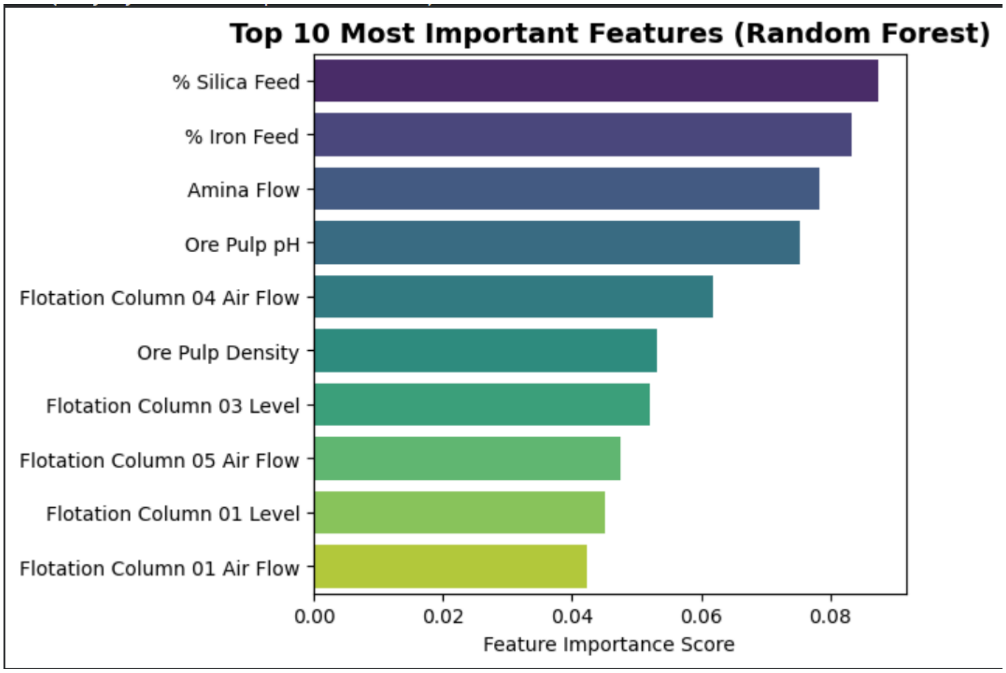

Figure 4: Top 10 Most Important Features Identified by Random Forest

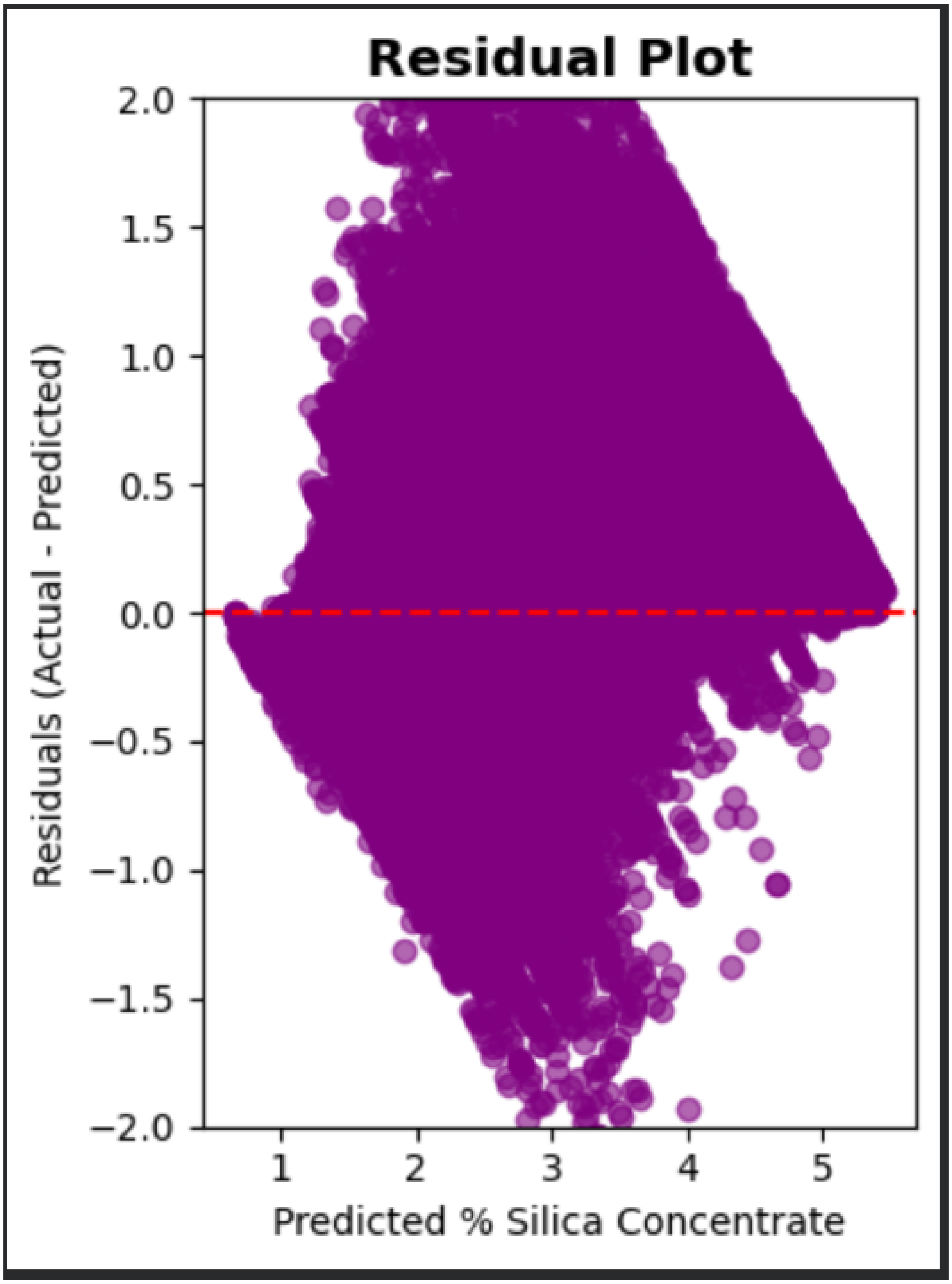

Figure 5: Residual Plot: Random scatter around zero line indicates no systematic bias and reliable predictions

10.4 Answers to Project Questions

- **Is minute-level prediction possible?** Yes — model uses high-frequency sensor data and achieves strong accuracy.
- **How many hours ahead can we predict?** Demonstrated feasibility with ARIMA

extension (6-hour forecast shown) — further improvement possible with LSTM/Prophet.

- **Prediction without % Iron Concentrate?** Yes — successfully achieved with RMSE 0.3725 and R^2 0.8902 using only real-time sensors.

11 Conclusion

This internship project successfully delivered a production-ready machine learning solution for real-time quality prediction in iron ore flotation. The final Random Forest model predicts % Silica Concentrate with industrially meaningful accuracy (¡0.4% average error) using only operational sensor data — eliminating dependency on delayed lab results.

The model is robust, explainable (through feature importance), and unbiased (residual analysis). It directly enables proactive process control, reducing tailings, optimizing reagent usage, improving concentrate quality, and lowering environmental impact.

All three project questions were answered affirmatively, demonstrating both current real-time capability and future forecasting potential. The solution aligns perfectly with UniConverge Technologies' vision of smart manufacturing and predictive maintenance using IoT and ML.

This work represents a practical, deployable contribution to industrial efficiency and sustainability — proving that machine learning can deliver tangible value in real manufacturing environments.