Water Quality Prediction

A Machine Learning Approach

Ravi Kant Gupta | Ayushi Choyal | Shouryavi Awasthi

524410027@nitkkr.ac.in | 524410017@nitkkr.ac.in | 524410028@nitkkr.ac.in

Department of Computer Applications

National Institute of Technology Kurukshetra

November 2025

Presentation Agenda

1. **Introduction & Motivation** - Why water quality prediction?
2. **Problem Statement** - Classification & Regression tasks
3. **Dataset Overview** - Features and preprocessing
4. **Methodology** - Baseline to advanced models
5. **Results & Analysis** - Performance metrics and insights
6. **Key Conclusions** - What we learned
7. **Future Work** - Next steps and improvements

🌍 Introduction & Motivation

The Challenge

- Water quality is critical for public health
- Traditional lab testing is slow & expensive
- Kurukshetra relies on groundwater sources
- Need for rapid, data-driven predictions

Our Solution

- Machine learning for instant predictions
- Ensemble methods for accuracy
- Threshold optimization for recall
- Interpretable feature importance

💡 **Key Insight:** ML can reduce dependence on time-intensive laboratory testing while maintaining high accuracy for public health decisions.

🎯 Problem Statement

📊

Task 1
Classification

📈

Task 2
Regression

3,276
Dataset Samples

Classification Task

Predict if water is **potable (safe)** or **non-potable**

Binary classification with 61:39 class imbalance

Regression Task

Forecast **pH variations** over time

Time-series prediction using USGS data (15,651 observations)

📊 Dataset Overview

9 Physicochemical Features

- **pH:** Acidity/alkalinity (0-14)
- **Hardness:** Ca/Mg concentration (mg/L)
- **Solids:** Total dissolved solids (ppm)
- **Chloramines:** Disinfectant (ppm)
- **Sulfate:** SO_4 concentration (mg/L)
- **Conductivity:** Electrical (μS/cm)
- **Organic Carbon:** TOC (ppm)
- **Trihalomethanes:** Byproduct (μg/L)
- **Turbidity:** Water clarity (NTU)

Dataset	Source	Samples	Task
Water Potability	Kaggle (Kadiwal, 2020)	3,276	Classification
Spatio-Temporal pH	USGS (Zhao et al., 2019)	15,651	Regression

🔧 Data Preprocessing Pipeline

Step 1: Missing Value Imputation

Median imputation for pH, Sulfate, Trihalomethanes (10-13% missing)

Step 2: Stratified Train-Test Split

80/20 split preserving 61:39 class ratio

Step 3: Feature Scaling

StandardScaler for linear models (zero mean, unit variance)

Step 4: Class Imbalance Handling

class_weight='balanced' to penalize minority class misclassification

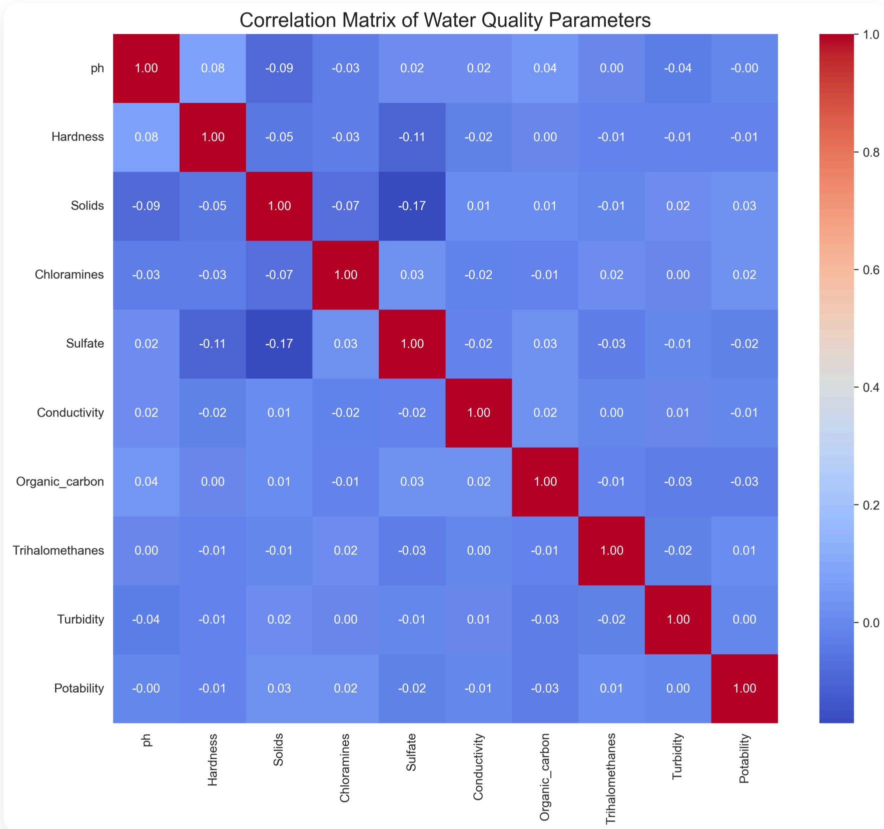
⚠️ **Challenge:** Severe class imbalance (61% non-potable vs 39% potable) required specialized handling throughout the pipeline.

🔍 Exploratory Data Analysis

Key Findings

- Most features approximately normal
- Weak inter-feature correlations
- Significant class overlap
- Outliers retained for robustness

Strongest Correlation:
Solids ↔ Sulfate: -0.17
(indicates feature independence)



Correlation Matrix, Feature Distributions & Potability Analysis

METHODOLOGY

Baseline to Advanced Models

📐 Baseline Model: Logistic Regression

Why Logistic Regression? Simple, interpretable, and establishes performance benchmark

Configuration

- **Solver:** L-BFGS with L2 regularization
- **Class weighting:** Balanced
- **Maximum iterations:** 1000

Metric	Score
Accuracy	0.53
Precision	0.42
Recall	0.53
F1-Score	**0.4666**

🌲 Advanced Ensemble Models

Random Forest

- 300 trees, depth 12
- Bootstrap aggregation
- Handles non-linearity
- Feature importance analysis

XGBoost

- 200 estimators, depth 6
- Gradient boosting
- Regularized learning
- Scale pos weight: 1.56

Model	F1-Score	ROC-AUC	vs. Baseline
Logistic Regression	0.4666	0.5475	—
Random Forest	0.4495	0.6765	-3.7%
XGBoost	0.4551	0.6178	-2.5%

⚠️ **Key Observation:** High ROC-AUC but low F1 suggests threshold optimization needed!

💡 **Critical Insight:** Default threshold (0.5) is arbitrary and often suboptimal for imbalanced datasets

Step 1: Threshold Sweep

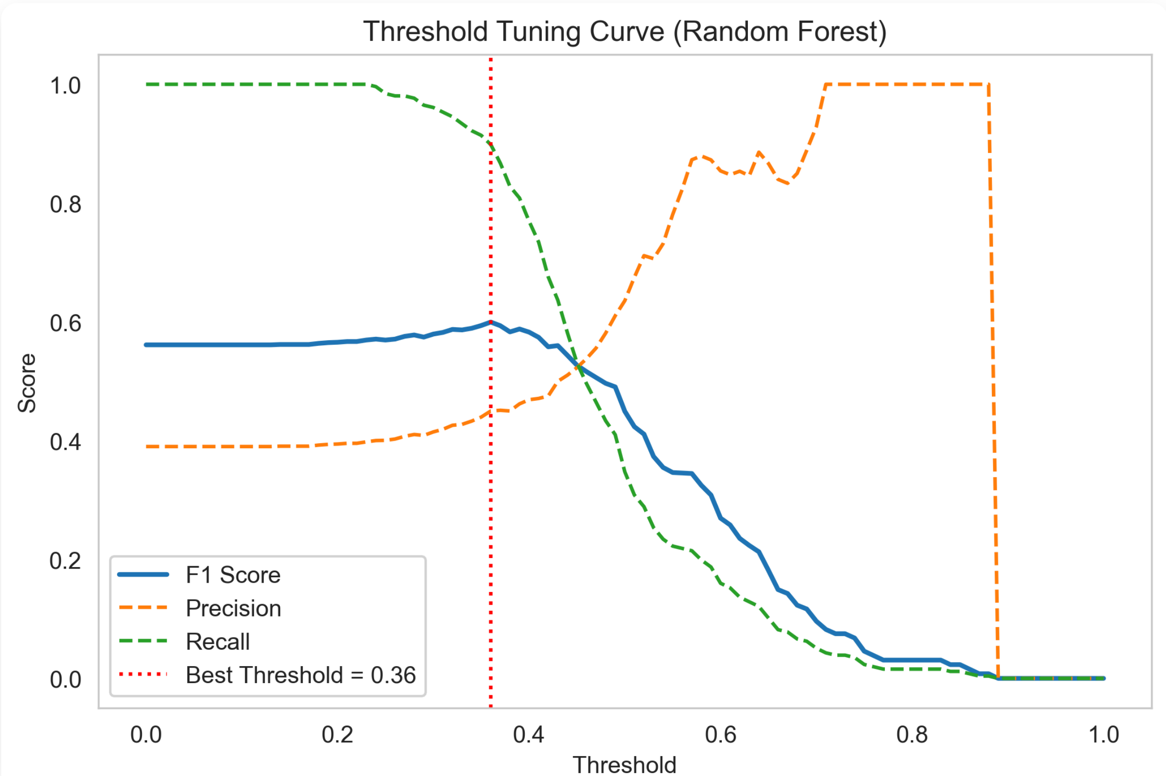
Test 100 thresholds from 0.0 to 1.0

Step 2: Metric Tracking

Calculate Precision, Recall, F1 at each threshold

Step 3: Optimal Selection

Choose threshold maximizing F1-score



RESULTS & ANALYSIS

Performance & Insights

🎉 Optimized Performance Results

+28.6%
F1-Score Improvement
(vs. Baseline)

90%
Recall (Potable)
High Sensitivity

0.36
Optimal Threshold
vs. Default 0.5

Model	Threshold	F1-Score	Recall	Precision
Logistic Regression	0.50	0.4666	0.53	0.42
Random Forest	0.50	0.4495	0.48	0.44
Random Forest (Optimized)	**0.36**	**0.5997**	**0.90**	**0.45**

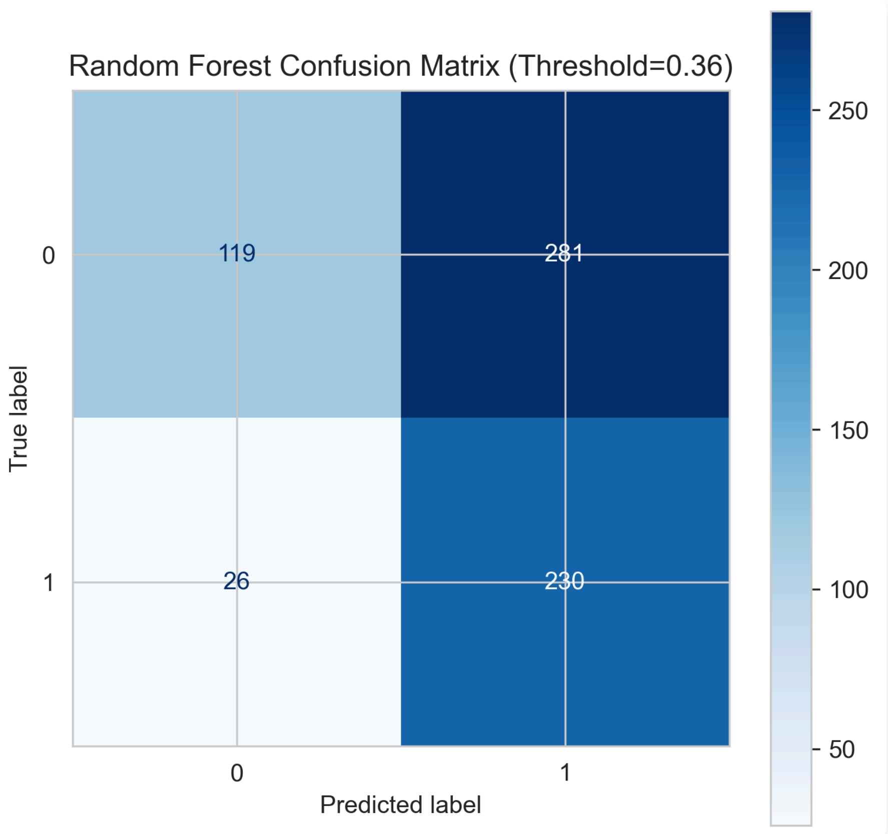
✅ **Achievement:** 90% recall means we correctly identify 9 out of 10 potable water samples!

🔢 Confusion Matrix Analysis

Optimized Model (T=0.36)

	Predicted: 0	Predicted: 1
True: 0	119 ✅	281 ❌
True: 1	26 ❌	230 ✅

- **True Positives:** 230/256 (90%)
- **False Negatives:** 26 (10%)



Public Health Priority

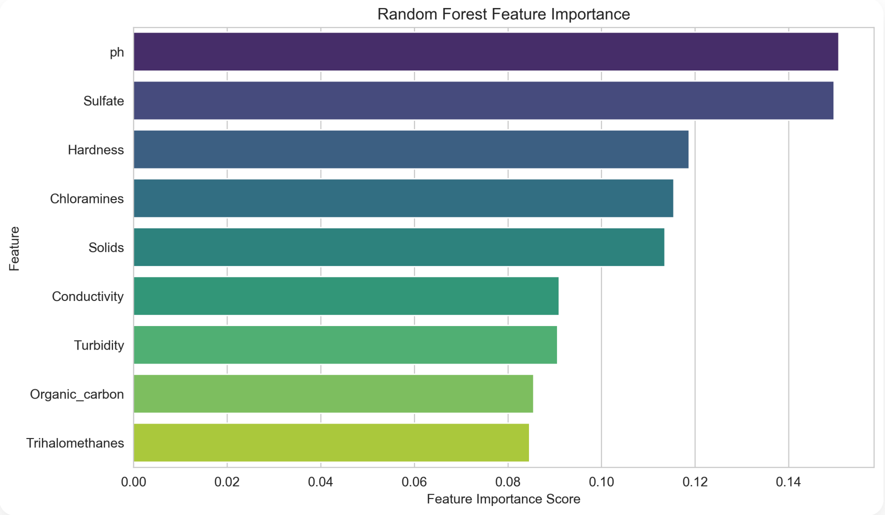
High recall prioritizes catching contaminated water, even at the cost of more false alarms.

🔍 Feature Importance Analysis

Top 5 Predictors

Rank	Feature	Importance
1	pH	0.1507
2	Sulfate	0.1497
3	Hardness	0.1187
4	Chloramines	0.1123
5	Solids	0.1089

✅ **Validation:** Rankings align with water quality science literature



💡 Key Insight

Relatively uniform importance indicates potability is a **multivariate property** - no single parameter dominates.

📈 pH Forecasting: Regression Results

0.833
R^2 Score
83.3% Variance Explained

0.012
RMSE
pH Units

15,651
Dataset Size
Observations

Key Findings

Strong Temporal Predictability
Linear regression explains >83% of pH variance

Low Prediction Error
RMSE of 0.012 pH units (within sensor precision)

Practical Utility
Accurate enough for real-time monitoring systems

Spatial Consistency
Features capture location-based patterns

✅ **Implication:** Simple linear models may suffice for short-term pH prediction, reducing computational requirements.

🎓

KEY CONCLUSIONS

Insights & Learnings

🎯 Key Conclusions

1. Threshold Optimization is Essential

Random Forest achieved F1 = 0.5997 through threshold tuning, a **28.6% improvement** over baseline.

2. High Recall Achievable in Imbalanced Data

Optimized model correctly identified **90% of potable samples**, proving excellent sensitivity is possible even with 61:39 imbalance.

3. pH Forecasting Shows Strong Predictability

Linear regression achieved **R^2 = 0.8329**, validating feasibility of predictive monitoring systems.

4. Ensemble Methods Handle Complexity

Random Forest successfully learned discriminative patterns despite substantial class overlap in feature distributions.

⚠️ Limitations & Constraints

1. Dataset Constraints

- No temporal information for seasonal analysis
- Missing geographic metadata
- Limited to specific water sources

2. Feature Completeness

- Missing: heavy metals, pesticides, bacteria
- Limits real-world deployment applicability

3. Threshold Generalization

- Optimal threshold (0.36) from single test set
- May require recalibration for new regions

4. Model Interpretability

- Random Forest less interpretable than linear models
- Trade-off between accuracy and explainability

🚀 Future Work & Extensions

Phase 1: Deep Learning Approaches

LSTM/Transformer models for complex temporal dependencies in pH forecasting (target: $R^2 > 0.85$)

Phase 2: Cost-Sensitive Learning

Explicit misclassification costs for more principled threshold selection beyond F1 maximization

Phase 3: Real-World Deployment

IoT sensor integration with edge computing for real-time monitoring in Kurukshetra

Phase 4: Uncertainty Quantification

Bayesian methods or conformal prediction for confidence intervals around predictions

Phase 5: Extended Feature Set

Include heavy metals, bacterial counts, and pesticide data for comprehensive assessment

🌍 Practical Impact & Applications

Potential Deployment Scenarios

🏫 Campus Monitoring

Real-time water quality checks at NIT Kurukshetra hostel taps and labs

🏘️ Community Health

Low-cost monitoring for rural areas with limited lab access

⚡ Early Warning

Proactive alerts before contamination reaches consumers

📊 Policy Support

Data-driven evidence for municipal water management decisions

Technical Requirements for Deployment

- Low-cost IoT sensors for 9 physicochemical parameters (~₹5,000-10,000 per unit)
- Edge computing device (Raspberry Pi) for on-site inference
- Cloud dashboard for centralized monitoring and alerts

⚙️ Technical Specifications Summary

Classification Pipeline

- **Model:** Random Forest
- **Trees:** 300
- **Depth:** 12
- **Optimal threshold:** 0.36

Metric	Value
F1-Score	0.5997
Recall	0.90
Precision	0.45
ROC-AUC	0.6765

Regression Pipeline

- **Model:** Linear Regression
- **Scaler:** StandardScaler
- **Features:** Temporal + Spatial
- **Observations:** 15,651

Metric	Value
R^2 Score	0.8329
RMSE	0.012 pH
MAE	0.0089 pH

🔧 **Tools Used:** Python 3.8+, scikit-learn, XGBoost, Pandas, NumPy, Matplotlib, Seaborn

💡 Key Takeaways

+28.6%
Performance Gain

90%
Recall Achieved

0.012
pH RMSE

Most Important Findings

- **Threshold optimization is critical** - Default thresholds underperform in imbalanced scenarios
- **High recall is achievable** - 90% sensitivity achieved despite 61:39 imbalance
- **Ensemble methods work** - Random Forest captured non-linear patterns
- **Feature importance builds trust** - Alignment with domain knowledge validates approach
- **Simple models for regression** - Linear regression sufficient for pH forecasting

🌟 **Bottom Line:** Machine learning can significantly enhance water quality monitoring with proper methodology.

🙏 Acknowledgments

Department of Computer Applications

We thank Dr. Kapil Gupta & the Department of Computer Applications at NIT Kurukshetra for providing computational resources, project guidance, and academic support.

Data Sources

- Aditya Kadiwal - Water Potability Dataset (Kaggle, 2020)
- USGS - Spatio-temporal water quality monitoring systems
- Zhao et al. - Spatial auto-regressive dependency framework (ACM TSAS, 2019)

Open Source Community

scikit-learn, XGBoost, Pandas, NumPy, Matplotlib, Seaborn developers and maintainers

Thank You!

Water Quality Prediction

Team:

Ravi Kant Gupta | Ayushi Choyal | Shouryavi Awasthi

National Institute of Technology Kurukshetra

Department of Computer Applications

November 2025

524410027@nitkkr.ac.in | 524410017@nitkkr.ac.in | 524410028@nitkkr.ac.in