Dahir-ke / dsc_phase3_project

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Watch 0

0 stars 0 forks 0 watching Branches Activity
Tags

Public repository

main 1 Branch 0 Tags Go to file t Go to file Add file Code

Dahir-ke updated projected 2fb6ab8 · 6 minutes ago

Images	Second commit	2 days ago
Presentation.pdf	Project updated	17 minutes ago
README.md	updated README.md	14 minutes ago
Test_set_values.csv	first commit	3 days ago
Training_set_labels.csv	combined data sets	2 days ago
phase3_project.ipynb	updated projected	6 minutes ago
phase3_project.pdf	updated projected	6 minutes ago

README

Water Well Status Prediction in Tanzania

Project Overview

Access to clean and reliable water is a cornerstone of public health and economic development in Tanzania. However, thousands of water wells across the country suffer from breakdowns, poor maintenance, or complete failure. This project uses data from the **Tanzanian Water Point Mapping** initiative to build a predictive model that determines the operational status of a water well.

Primary Stakeholders:

- Ministry of Water (Tanzania)
- Water-focused NGOs (e.g., WaterAid, World Vision)

Goal:
To maximise the number of functional water wells by identifying which water points are most likely to fail, enabling proactive maintenance and smarter investment decisions.

Dataset

- **Source:** Tanzanian Water Point Mapping
- **Size:** 59,400 rows, 41 columns
- **Target variable:** `status_group` – three classes:
  - `functional` (encoded as 2)
  - `functional needs repair` (encoded as 1)

- `non functional` (encoded as 0)

- **Features include:**
  - **Technical:** pump type, extraction type, construction year, amount_tsh (water yield)
  - **Environmental:** GPS coordinates, elevation, basin, region
  - **Management:** funder, installer, payment type, permit, public meeting, scheme management

- **Missing data & invalid values:**
  - Categorical columns with low missing percentages imputed with mode or `"unknown"`.
  - Numeric columns contained invalid zeros (e.g., `gps_height = 0`, `longitude = 0`, `construction_year = 0`). These were replaced using **domain-aware imputation** (non-zero medians, regional means for coordinates).
  - `scheme_name` (~49% missing) was dropped.

Methodology

1. Data Preprocessing & Feature Engineering

Step	Description
Imputation	Zero values in `amount_tsh`, `gps_height`, `population`, `construction_year` replaced with the median of non-zero entries. Geographic coordinates imputed by regional means.
Noise addition	Added random noise (±5 years) to `construction_year` to break an artificial spike at 2000 (caused by mass imputation of zeros).
Feature engineering	`well_age` = 2013 – `construction_year` `amount_tsh_log` = log1p(`amount_tsh`) `population_log` = log1p(`population`) `gps_height_bin` – binned elevation into `low`, `medium`, `high`.
Encoding	**High-cardinality** columns (`funder`, `installer`, `subvillage`, `lga`, `ward`) → Label Encoding. **Low/medium-cardinality** columns → One-Hot Encoding with `drop_first=True`. Target encoded ordinally: `non functional` = 0, `needs repair` = 1, `functional` = 2.
Train/Test split	75% training, 25% testing, stratified by target.

2. Modeling Approach

Model	Rationale	Key Outcome
Logistic Regression	Interpretable baseline.	ROC-AUC = 0.6846; recall for class 1 = 0.00 → unusable for business goal.
Decision Tree	Captures non-linear patterns; human-readable rules.	ROC-AUC = 0.7771; recall for class 1 = 0.07 → still misses most at-risk wells.
Random Forest	Ensemble reduces variance; handles complex interactions.	ROC-AUC = 0.8644; recall for class 1 = 0.05 → excellent overall, but fails on minority class.

3. Addressing Class Imbalance

Three Random Forest variants were compared:

Model	ROC-AUC (OvR)	Accuracy	Weighted F1	Recall (class 1)
Base RF (tuned)	0.8654	0.75	0.72	0.05
RF + `class_weight='balanced'`	**0.8733**	0.73	**0.75**	**0.66**
RF + SMOTE	0.8651	0.76	0.76	0.40

Hyperparameter tuning: `RandomizedSearchCV` on a 30% subset, then final model trained on full dataset.

Final model recommendation: Random Forest with `class_weight='balanced'` – it balances strong overall performance with a **66% recall** of wells that need repair (compared to 5% in the base model).

4. Feature Importance

Top predictors of well failure (from the final Random Forest):

1. **Water quantity (`quantity`)** – dry or insufficient wells are 79%/52% likely to be non-functional.
2. **Well age (`well_age`)** – failure risk doubles after 20 years.
3. **Payment type (`payment_type`)** – “never pay” wells fail twice as often.
4. **Elevation (`gps_height`)** – higher elevation correlates with better functionality.
5. **Pump mechanics (`extraction_type`)** – some designs (gravity, Afridev) are significantly more reliable.

Results & Business Impact

- **Final ROC-AUC: 0.8733** – surpasses the project target of 0.85.
- **Minority class recall (Needs Repair): 0.66** – the model now identifies **two out of every three** at-risk wells, enabling **proactive maintenance** before complete failure.
- **Functional wells** are predicted with **94% recall** – authorities can trust the model when it declares a well operational.

Before this model: maintenance was purely reactive – wells were fixed only after communities called with a crisis.
Now: the Ministry receives a monthly risk-score list and can schedule inspections for the highest-risk wells during routine trips – **no extra travel costs, just smarter routing.**

368 wells correctly identified as “needs repair” – 368 communities that will **not** lose water access. That is the real impact.

Actionable Recommendations (Evidence-Based)

Based on the model’s insights, we propose three immediate, no-regret moves:

1. **Target “Dry” & “Insufficient” Wells First**
   - Wells flagged as `dry` are **79% non-functional**; `insufficient` are **52% non-functional**.
   - **Action:** Deploy rapid response teams to every such well within 7 days.
2. **Phase Out Wells Older Than 25 Years**
   - Wells built **before 1995** are **2.3× more likely** to be non-functional.
   - **Action:** Create a 5-year phased replacement plan for all pre-1995 wells (≈12,000 wells).
3. **Replace “Never Pay” with Community Contributions**
   - Wells with **no payment system** fail **2× more often** than those with any fee.
   - **Action:** Convert the **25,000+ wells currently marked “never pay”** to a low-cost community maintenance model (e.g., pay-per-bucket or monthly fee).

How to Run the Notebook

1. **Clone the repository**

```
git clone https://github.com/Dahir-ke/dsc_phase3_project.git
cd <repository-files.csv>
```

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● **Jupyter Notebook** 100.0%