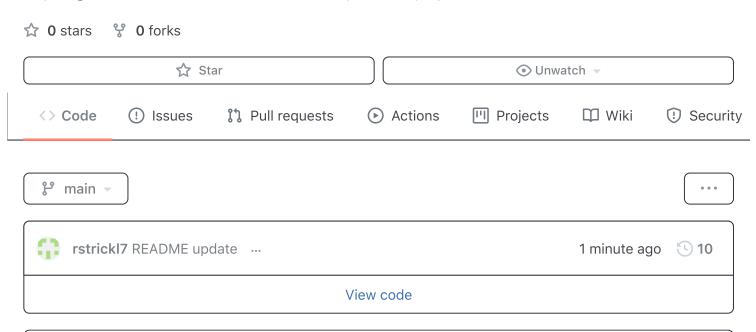
#### ☐ rstrickl7 / Tanzania-well-analysis

Ternary classification to predict the condition of water wells in Tanzania. (Description of project: https://github.com/learn-co-curriculum/dsc-phase-3-project)



# Tanzania-well-analysis

This analysis is based on the competition Driven Data® published about water pumps in Tanzania. The competition information was obtained by the Tanzania Ministry of Water using an open-source platform called Taarifa. Tanzania is the largest country in East Africa, with a population of about 60 million. Half of the population does not have access to clean water. The Tanzanian government is struggling to solve this problem. A significant part of water pumps are entirely out of order or do not function; the others require repair. Tanzania's Ministry of Water Resources agreed with Taarifa, and they launched the DrivenData competition.

Author: [Becky Strickland]

README.md

## Data

0

The data has many characteristics associated with water pumps. Data related to geographical locations, organizations that create and manage them, and some data about the region, local government areas. Also, there is information on the types of checkouts, types and number of payments. The water supply points were divided into functional, non-functional and functional but in need of repair. The goal of the competition is to build a model that predicts the functionality of water supply points.

### **Data Fields**

The following set of information about waterpoints is presented for analysis: amount tsh — Total static head (amount water available to waterpoint) date\_recorded — The date the row was entered funder — Who funded the well gps\_height — Altitude of the well installer — Organization that installed the well longitude — GPS coordinate latitude — GPS coordinate wpt\_name — Name of the waterpoint if there is one num\_private — No information basin — Geographic water basin subvillage — Geographic location region — Geographic location region\_code — Geographic location (coded) district\_code — Geographic location (coded) Iga — Geographic location ward — Geographic location population — Population around the well public\_meeting — True/False recorded\_by — Group entering this row of data scheme\_management — Who operates the waterpoint scheme\_name — Who operates the waterpoint permit — If the waterpoint is permitted construction\_year — Year the waterpoint was constructed extraction\_type — The kind of extraction the waterpoint uses extraction\_type\_group — The kind of extraction the waterpoint uses extraction\_type\_class — The kind of extraction the waterpoint uses management — How the waterpoint is managed management\_group — How the waterpoint is managed payment — What the water costs payment\_type — What the water costs water\_quality — The quality of the water quality\_group — The quality of the water quantity — The quantity of water quantity\_group — The quantity of water (duplicates quality) source — The source of the water source\_type — The source of the water source\_class — The source of the water waterpoint\_type — The kind of waterpoint waterpoint\_type\_group — The kind of waterpoint

## This project was created using the following libraries:

import pandas as pd import matplotlib.pyplot as plt import matplotlib.ticker as mtick import seaborn as sns import numpy as np import scipy.stats as stats import statsmodels.api as sm import catboost import time import warnings warnings.filterwarnings('ignore')

from sklearn.utils import class\_weight from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report from catboost import Pool, sum\_models from catboost import CatBoostClassifier from sklearn.feature\_selection import RFE from sklearn.metrics import mean\_squared\_error, r2\_score, mean\_absolute\_error, balanced\_accuracy\_score from sklearn.model\_selection import KFold, cross\_val\_score, StratifiedKFold from sklearn.preprocessing import LabelEncoder, OneHotEncoder from sklearn.tree import DecisionTreeRegressor from sklearn.ensemble import RandomForestClassifier from sklearn.model\_selection import train\_test\_split from sklearn.preprocessing import StandardScaler from sklearn.preprocessing import MinMaxScaler from sklearn.metrics import classification\_report, confusion\_matrix from sklearn.ensemble import GradientBoostingClassifier from sklearn.linear\_model import LogisticRegression from sklearn.model\_selection import GridSearchCV from sklearn import metrics from sklearn.model\_selection import RandomizedSearchCV from scipy.stats import uniform, truncnorm, randint

### For More Information

See the full analysis in the Jupyter Notebook or review this presentation.

# **Repository Structure**

	<pre>ipynb_checkpoints</pre>
<u></u>	data
<u></u>	README.md
<u></u>	Tanzanian-well-analysis-Jupyter_Notebook.pdf
<u> </u>	Tanzanian-well-analysis.ipynb
<u></u>	Water-Pump-Analysis.pdf

#### Releases

No releases published Create a new release

#### **Packages**

No packages published Publish your first package

### Languages

Jupyter Notebook 100.0%