Pump it Up: Mining the Water Table

Alex Pham apham7@stanford.edu
Benjamin Backus mbackus@stanford.edu
Lauren Zhu laurenz@stanford.edu

Introduction

Millions of people still use unreliable water sources, including wells and pumps that are in questionable states.

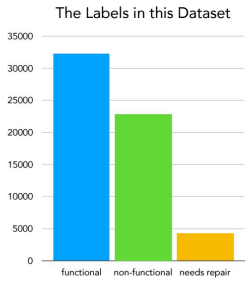
Our goal is to predict which pumps are functional, non-functional, or need repair based on characteristics of the pump. We can then improve maintenance operations and ensure that potable water is available to communities across Tanzania.

Data

Our dataset is from Taarifa and the Tanzanian Ministry of Water.

It has 59,400 rows (examples) and 40 columns (features). Of the features, there are 31 categorical vars, 7 numerical vars, and 2 date vars. Some are redundant or missing data.

Examples have corresponding labels with ground truth (functional, functional needs repair, non-functional).



Features

We experimented extensively with dataset pre-processing by removing near-duplicate features, replacing certain missing data with means, and one-hot encoding all other categorical features (capped). After this process, our input data had 432 features.

Sample feature preprocessing:

- **altitude**: 1390
- **installer**: DWE
- **region**: Kilimanjaro
- **population**: 250
- **construct_year**: 2007
- **source_type**: spring

→

- **altitude**: 1390
- **installer**: 1, 0, 0, … , 0
- **region**: 0, 0, 1, … , 0
- **population**: 250
- **construct_year**: 2007
- **source_type**: 0, 0, … , 1

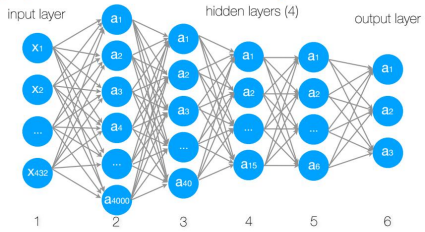
Project Summary

Our dataset consists of 59,400 total examples, which we split into a 93:7 train to test ratio. We preprocessed the data so each example contains 432 feature values (most of which are one-hot encodings), and a corresponding label of one-hot size 3.

We built a neural network with 6 fully connected layers, including the input layer and output layer (softmax to 3 classes). Our final training accuracy reaches 77.39% and test accuracy 78.36%.

Model

After testing different model architectures, our final design is a fully connected neural network with the input layer (432 neurons), 4 hidden layers (4000, 40, 15, 6 neurons), and an softmax output layer (3 neurons). We use the Adam Optimizer, which implements RMSprop and momentum for faster learning.



Discussion

We tested many different NN model architectures by varying the encoding of our data and numbers of layers and neurons. This entailed extensive hyperparameter tuning and repeated trainings to juxtapose discrepancies among achieved accuracies.

The challenge for this project was preprocessing the data as best as we could to maximize accuracy. In several instances we found that more data processing actually lowered our accuracy. But overall, we thought the results were very good and that with this classifier, the Tanzanian Ministry of Water can set appropriate priority for their maintenance operations.

Results

Learning Rate	No. Epoch	Batch Size	Train Accuracy	Test Accuracy	Hidden Layer Shapes	No. Layer
10^{-5}	100	250	71.66	71.88	100/40/15/6	6
10^{-5}	40	250	73.23	74.27	1000/40/15/6	6
10^{-5}	60	250	75.46	76.41	4000/40/15/6	6
10^{-5}	49	250	76.19	76.88	4000/40/15/6	6
10^{-5}, 2000, .9	300	250	77.01	77.47	4000/40/15/6	6
10^{-5}, 2000, .8	150	250	72.91	73.067	1000/400/100/30/15/6	8
10^{-5}, 2000, .89	181	920	77.39	78.36	4000/40/15/6	6

*Training and test data consisted of 55,000 and 4,400 examples respectively.

During hyperparameter tuning, we tested nearly 100 different combinations of model architectures and hyperparameters using three different representations of our dataset. We found our best results when using a fully connected neural network using a decaying learning rate, four hidden layers, and more epochs.

Future Work

1. **Identify Important Features**
   - Better data, faster training, higher accuracy
2. **Estimate Lifetime of a Pump**
   - Using construction year, identify decay rate, efficient maintenance
3. **Improve Success Metrics**
   - Calculate lives improved per pump