Coulton Theuer (theuerc@umich.edu)
Bella Karduck (bkarduck@umich.edu)
Haley Johnson (haleyej@umich.edu)
GitHub
Data Inventory

SI 670: Predicting Water Main Breaks

Introduction

A water main is an underground pipe in a municipal water distribution system that delivers potable water from a water treatment plant to homes and businesses. These water mains have the potential to break when there is a crack or rupture in the pipe, which can lead to water leakage, flooding, and property damage.

Two of our group members are graduate student instructors for a project-based class that uses Ann Arbor water infrastructure data. During one of the lectures for that class, a guest speaker named David Wilburn, who is a Senior Applications Specialist at the City of Ann Arbor, said it would be helpful for the city to be able to predict which water mains will break soon. If the city knows a pipe will likely break, they can take immediate preventative measures like decreasing the water pressure on a particular pipe to reduce the likelihood of it rupturing.

The goal for this project is to determine if it is feasible to use machine learning to predict water main breaks for the city of Ann Arbor. These predictions can then potentially be used by the city to mitigate the repair costs, flooding, and property damage associated with water main breaks and help the city better allocate scarce resources for infrastructure improvements.

Related Work

The existing research on this problem ranges from unsupervised clustering of problematic areas for water distribution infrastructure[1] to supervised learning on when future water main breaks might occur.[2] In particular, we based much of our approach off of the paper "Using Machine Learning to Assess the Risk of and Prevent Water Main Breaks", where the authors created a machine learning model to predict if particular water mains were going to break in the next three

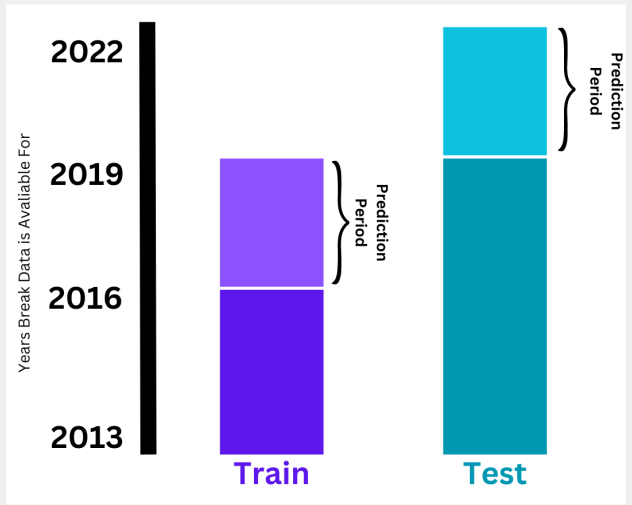
[1] Babak Aslani, Shima Mohebbi, and Hana Axthelm, "Predictive Analytics for Water Main Breaks Using Spatiotemporal Data," *Urban Water Journal* 18, no. 6 (July 3, 2021): 433–48, https://doi.org/10.1080/1573062X.2021.1893363.
[2] Brett Snider and Edward A. McBean, "Combining Machine Learning and Survival Statistics to Predict Remaining Service Life of Watermains," *Journal of Infrastructure Systems* 27, no. 3 (September 1, 2021): 04021019, https://doi.org/10.1061/(ASCE)IS.1943-555X.0000629.

years in Syracuse, New York.[3] However, there are some substantial differences between their approach and our project. For instance, the data is different — they are predicting water main breaks in Syracuse, New York; and we are predicting water main breaks in Ann Arbor, Michigan. Likewise, they predicted if any pipes would break on a given city block, while are predicting if individual pipes would break. Predicting at the block level does have some advantages — cities need to dig up an entire road to replace water main infrastructure, so they often elect to do work on the entire block at once.[3] Unfortunately, we do not have precise enough location data to aggregate breaks to the block level. Predicting individual pipes is a more difficult task; we expect our model to perform worse and a direct comparison of our results isn't meaningful.

In terms of what will be replicated from the Syracuse paper, we will be using a similar metric to evaluate our final chosen model. The metric that we will use is precision at 3%. P@3 is a concept borrowed from information retrieval. It is essentially sorting the predicted probabilities of the positive class from most confident to least confident, filtering down to the top 3% of those predictions, and then finding the precision of those ranked results. This metric makes sense to use because water main breaks are rare, which leads to imbalanced classes. Similarly, this is a practical metric because city officials would only care about the pipes that the model predicts are most likely to break.

Methods

We used two datasets from the City of Ann Arbor: one with information about 876 reported breaks between 2013 and 2023 and another with data about 32,654 water mains. The dataset was shared with us by the city's Information Technology department and is not publicly available. We dropped duplicate rows and combined them to create a final dataset of 27,479 water mains.



Previous work indicated that it's easier to frame this as a classification problem, where we predict if a pipe will break in a given window of time (e.g. the next 6 months), rather than predicting time until the pipe's next break. Due to our small sample size, we

[3] Avishek Kumar et al., "Using Machine Learning to Assess the Risk of and Prevent Water Main Breaks," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London United Kingdom: ACM, 2018), 472–80, https://doi.org/10.1145/3219819.3219835.

decided it was feasible to predict if a pipe would break anytime in the next 3 years. We split the dataset into train and test sets using time cutoffs, so we could train on earlier data and test on the most recent information. We trained all the different models we evaluated to predict if a pipe would break between January 1st, 2016, and January 1st, 2019 using information from before 2016 and tested them on their ability to predict if a pipe would break after January 1st, 2019 based on information from before 2019. We removed any information on pipes broken after 2016 in the training set and on pipes broken after 2019 in the test set to prevent data leakage.

Notably, there was a significant class imbalance once we restricted our positive instances to just breaks in a certain 3-year period. Even though we had data about over 800 breaks across 695 different pipes, only 157 broke between 2016 and 2019. We specifically choose metrics and techniques that work well on imbalanced datasets to mitigate this.

To correct for the large class imbalance, we tried oversampling the majority class and undersampling the minority class. We found that undersampling performed better. Undersampling means that we kept all of the data on the pipes that broke, and randomly sampled from the data on the pipes that didn't break. We decided on a 1:9 ratio of positive to negative instances. After our preprocessing of the data was complete, we moved onto creating the modeling architecture and pipelines supporting that, which is discussed in the evaluation section.

Dataset	Prediction Period	Number of Water Main Breaks
Train	2016-2019	157
Test	2019-2022	36

We evaluated three different model architectures: support vector machines, logistic regressions, and random forest. All three of these architectures work well on imbalance data. Likewise, previous search in Syracuse indicated that tree-based and ensemble models work well on this problem. In particular, random forests allow errors from different decision trees to cancel out in the final estimate.

We used the following variables to predict water main breaks:

Variable	Description
FACILITYID	Unique identifier for a pipe
Install Year	The year the pipe was installed
Subtype	Numeric value that corresponds to what kind of water main a pipe is (1 = distribution main, 2 = transmission main, 3 = hydrant lead, 4 = raw water)
Status	Whether or not the pipe is an active water main (IS = in service, AB = abandoned)

Material	Material the pipe is made of. Possible values include lead, steel, ductile irons, and asbestos cement
Length	Length of the water main in feet
Diameter	Diameter of the water main in inches
Pressure System	Different distribution areas. Each zone works like its own distribution system and is closed from the other zones, except when the value is in a PRZ (pressure-reduced zone).
Previous breaks	Engineered feature, the number of times the pipe had broken between 2013 and the predicted date
Time delta first break	Engineered feature, amount of time between the installation and the pipe's first break. If a pipe has never broken, this is equal to the prediction date - installation date
Time delta most recent break	Engineered feature, amount of time between the installation and the pipe's most recent break. If a pipe has never broken, this is equal to the prediction date - installation date

Additionally, we scaled our data using a standard scaler. Scaling is important for support vector machine and logistic regression; it prevents variables that are on larger scales (length, for example, in our dataset) from dominating the model. We one-hot-encoded categorical variables and added two engineered features to the dataset: one that measured the time difference between when a pipe was installed and it's first break and another that measures the time between installation and th epipe's most recent break.

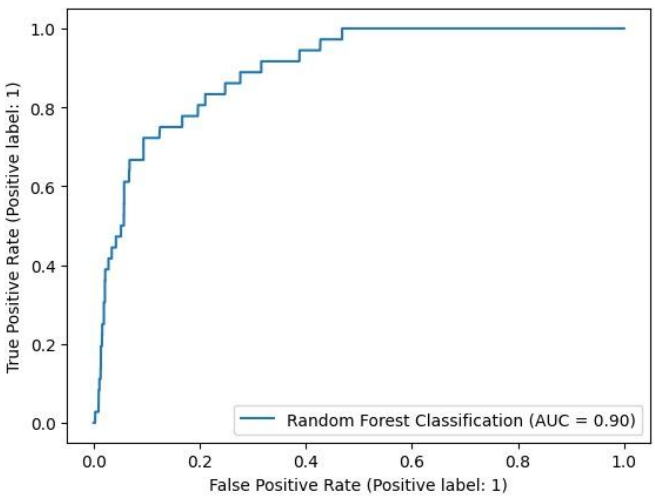
In our project repository we have:

- **Data:** a directory with all of the data we used for this projet.
    - Contained the subdirectory **raw**, which has all the raw data we received from the City of Ann Arbor, and the subdirectory **transformed**, which has the combined dataset we used for model selection and evaluation
- **Notebooks:** This directory contains two important notebooks: one where we combined the data from all of our sources into a final dataset and another where we divided the dataset into training and testing sets
- **Models:** This directory contains code for all the different models we tried. It has a notebook for the simple baselines, the SVM model (which attempted to replace the approach in the Syracuse paper), and the cross validation procedure we used to compare different architectures

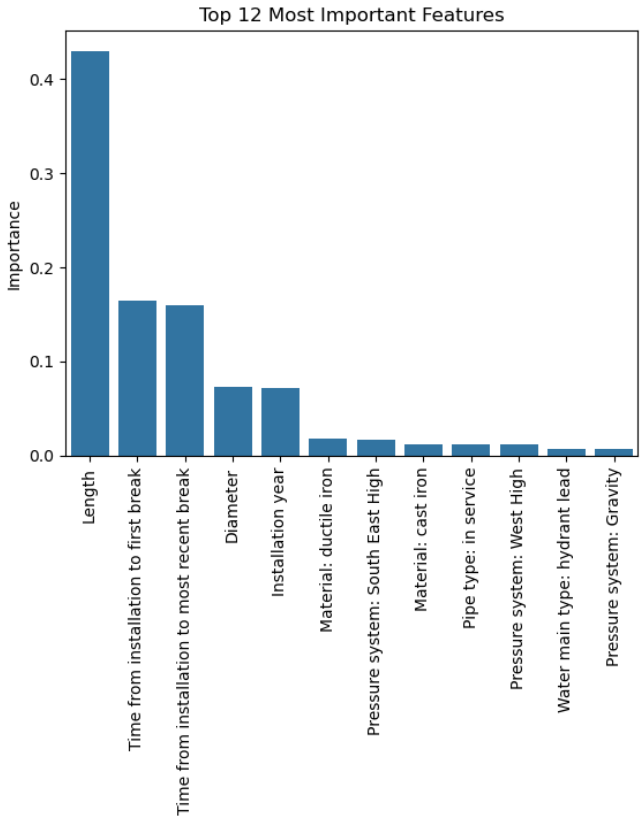
Evaluation & Analysis

We used three-fold nested cross-validation with GridSearchCV to select the best model architecture and hyperparameter configuration. For each architecture, we selected the configuration that produced the highest F1 and precision @ 3% score. Ultimately, we decided to use a random forest classifier because it offered the best performance and is fairly interpretable. We also compared each of these models against a random ranking of pipes to asses if our method was doing better than a naive approach.

Model Type	AUC Score	F1 Score	Precision @ 3%	Parameter Configuration
Random Performance	N/A	N/A	0.01170	Uses random rankings from the entire dataset for the top 3%
Logistic Regression	0.8562	0.0680	0.1182	C = 2 class_weights = balanced
SVC	0.8575	0.0766	0.08182	C = 12 class_weights = balanced gamma = 'auto'
Random Forest	0.8961	0.2080	0.1447	max_depth = 5 max_features = 17 n_estimators = 90 min_samples_split = 3



We trained a new random forest model using the hyperparameter configuration we selected with cross validation. The model used all of our data to predict if a pipe would break between 2019-2022 based on information from before 2019. Finally, we had the retrained model output the probability that each pipe in our dataset would break in the next 3 years. Since we were trying to predict the future, the final set of predictions had access to all information from the entire date range in our dataset.



Top 12 most important features in the random forest model

The most important feature in the random forest model was the length of the pipe, followed by the two engineered features that encoded how long a pipe had gone without breaking.

These are the 10 pipes that the final model was most confident would break within the next three years:

Pipe ID	**Location**	**Installation Date**	**Previous Breaks Between 2013-2023**
00-19129	Glendale Drive	1964-11-09	0
00-19821	Ironwood Drive	1964-09-24	0
00-15141	Lans Way	1964-04-27	0
00-14216	Greenview Drive	1963-04-27	0
00-19393	Archwood Drive	1962-03-20	0
00-19373	Dellwood Drive	1962-03-20	0
00-19392	Archwood Drive	1962-03-20	0
00-19391	Archwood Drive	1962-03-20	0

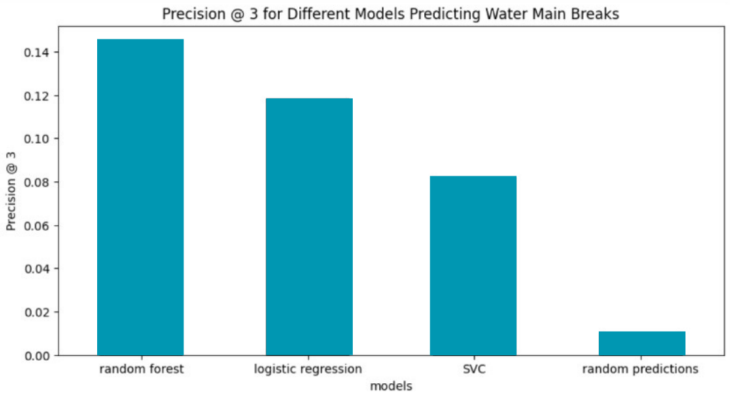
00-19371	Dellwood Drive	1962-03-20	0
00-17608	Vernon Downs Subdivision #4	1961-05-25	0

We visually inspected several of these locations on Google Maps. Municipalities often replace water main infrastructure when they redo a road[3]: if the pavement is in poor condition, the water mains often are too. We found that the roads that the highest-risk pipes run under were in very poor condition. The Syrcause paper used the road condition as an input into their model to predict water main breaks, but this information was not available to us. Still, our model is able to make sure of some spatial information because the different values for pressure system correspond to different parts of town. The majority of high-risk pipes are located on the west side of Ann Arbor (which roughly corresponds to pressure_sys = “wh”). The ‘wh’ pressure area had high feature importance in our final model, which indicates the random forest was able to effectively learn and make use of local information.

Archwood Street (left) and Dellwood Street (right) are neighboring streets on the west side of Ann Arbor. Five of the ten highest-risk water mains are located under these roads.

Discussion & Conclusion

The goal of this project was to evaluate if machine learning techniques are effective for predicting water main breaks in Ann Arbor. We trained many models, and we found the best performing models were random forest classification, logistic regression, and SVC.



In order to evaluate these models, we used precision @ 3%. This metric is only concerned with the most confident predictions of breaks, which is what city officials would be most interested in. We found random forest classifier to outperform all other models using this metric, and to vastly outperform our

model that was returning random rankings of pipes. This suggests that machine learning models may be effective for predicting water main breaks in Ann Arbor.

We learned a lot during this project. One of the more interesting concepts was to choose the appropriate metric to match the application of a machine learning task. In our case, we were interested in our most confident predictions of breaks, as those are the pipes on which city officials would want to take immediate preventative measures. Using precision @ 3% for evaluation allowed us to optimize our model selection and tuning for this specific real-world application. We had discussed using different metrics in class, but creating an entirely new metric was an interesting lesson that we learned from this project.

If we had another 6-12 months to continue working on this project, we would focus more on feature engineering. The Syracuse paper referenced in the related works section had some extensive feature engineering including rolling windows of many different sizes. We tried creating some features like the number of previous breaks and the days since the last break, but we were unable to explore and test more complex features due to time constraints. With more feature engineering, we believe that model performance could be vastly improved.

References

Aslani, Babak, Shima Mohebbi, and Hana Axthelm. “Predictive Analytics for Water Main Breaks Using Spatiotemporal Data.” *Urban Water Journal* 18, no. 6 (July 3, 2021): 433–48. https://doi.org/10.1080/1573062X.2021.1893363.

Kumar, Avishek, Syed Ali Asad Rizvi, Benjamin Brooks, R. Ali Vanderveld, Kevin H. Wilson, Chad Kenney, Sam Edelstein, et al. “Using Machine Learning to Assess the Risk of and Prevent Water Main Breaks.” In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 472–80. London United Kingdom: ACM, 2018. https://doi.org/10.1145/3219819.3219835.

Snider, Brett, and Edward A. McBean. “Combining Machine Learning and Survival Statistics to Predict Remaining Service Life of Watermains.” *Journal of Infrastructure Systems* 27, no. 3 (September 1, 2021): 04021019. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000629.

python = "3.11.2"
pandas = "^2.1.2"
numpy = "^1.26.1"
scikit-learn = "^1.3.2"
matplotlib = "^3.8.1"
seaborn = "^0.13.0"
xgboost = "^2.0.1"
catboost = "^1.2.2"
tqdm = "^4.66.1"
ipykernel = "^6.26.0"
pyarrow = "^14.0.1"
requests = "^2.31.0"
dask = "^2023.10.1"
fastparquet = "^2023.10.1"
lightgbm = "^4.1.0"
pytest = "^7.4.3"
jupyter = "^1.0.0"
kaggle = "^1.5.16"
usaddress = "^0.5.10"
imbalanced-learn = "^0.11.0"