Paris Dauphine University

Challenge Data ENS [1]

Failure risk prediction on pipeline network.

Edouard Pineau | Adrien Savary

September 25, 2017

[1] https://challengedata.ens.fr/en/home

Contents

1 Introduction

1.1 Context and Objective

1.1.1 Context

Veolia manages different types of fluid networks: drinking water distribution network, wastewater collection network, district heating network. Network management requires an effective and relevant maintenance on pipes. More and more operationnal data are available. These data can be used to build new models for improving performance of processes and network management.

1.1.2 Goal of the Challenge

In order to optimize maintenance operations on pipeline network (renewal plan) the objective is to predict which pipes have the highest failure risk in the two next years.

1.1.3 Metric

The metric used to evaluate the prediction performance is a weighted average AUC defined by:

$$AUC_{challenge} = 0.6 \times AUC_{2014} + 0.4 \times AUC_{2015}$$

The AUC is the Area Under the Curve ROC (Receiver Operating Characteristic). The ROC curve represents the True Positive Rate (TPR or Recall) against the False Positive Rate (FPR).

$$TPR = \frac{TP}{TP + FN},$$

$$FPR = \frac{FP}{FP + TN},$$

with TP the True Positive, FP the False Positive, FN the False Negative and FP the False Positive. A useful interpretation of ROC-AUC is the following:
If we pick a random positive and a random negative, the AUC gives the probability that a classifier assigns a higher score to the positive example.

2 Description of the Data and Exploratory Analysis

2.1 Imbalanced Data

We are in the context of imbalanced data. Training set is composed of around 20000 pipes. The proportion of failures for 2014 is 0.27% and 0.19% for 2015. Anomaly detection literature references many strategies to deal with such dataset [3] [4] [5]. We can cite under-sampling and over-sampling methods. An other approach would be to give more weights to samples in the minority class. Some popular algorithm are built to deal with it too, like Isolation Forest for continuous variable or One-Class SVM.

2.2 Features

2.2.1 *YearConstruction*

YearConstruction represents the year of construction of the pipe. We remark Figure 2.2.1 a pick construction period just before 1980. Then a slow increase during the 90's.

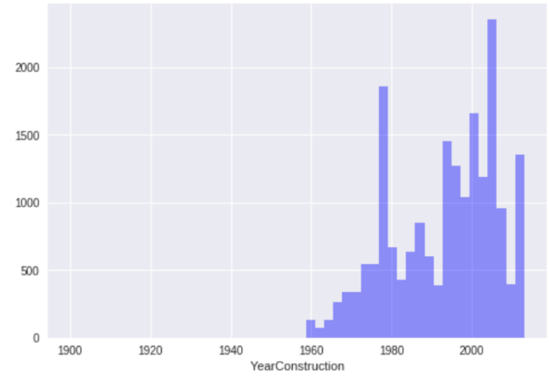

Figure 1: Feature YearConstruction

2.2.2 *Length*

Length represents the length of the pipes, there are smaller than 50 for the most part. We don't know the metric used to measure them.

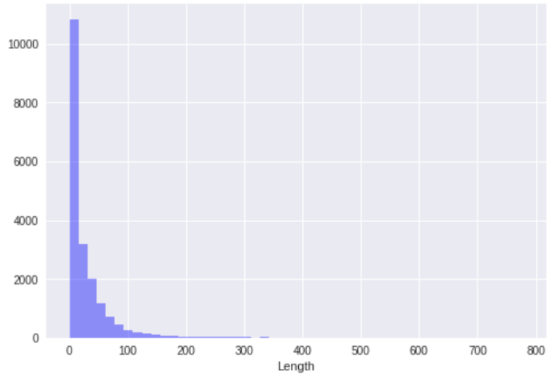

Figure 2: Feature Length

2.2.3 *YearLastFailureObserved*

YearLastFailureObserved is mostly composed of NaNs. From this feature we created the feature *TimeToFailure* which is the time to last failure observed for pipes that already failed. We asked ourselves if these were real missing values or if it represented the absence of failure. With our experimentations, we found out that it represented the absence of failure. There is a peak around 30 years.

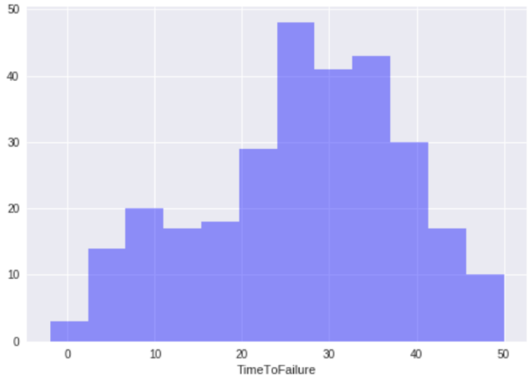

Figure 3: Feature TimeToFailure

2.2.4 *Feature1*

Feature1,*Feature2*,*Feature3* and *Feature4* are "characteristic features of the pipes" and we don't know about there precise meanings. *Feature1* has two categories T and P.

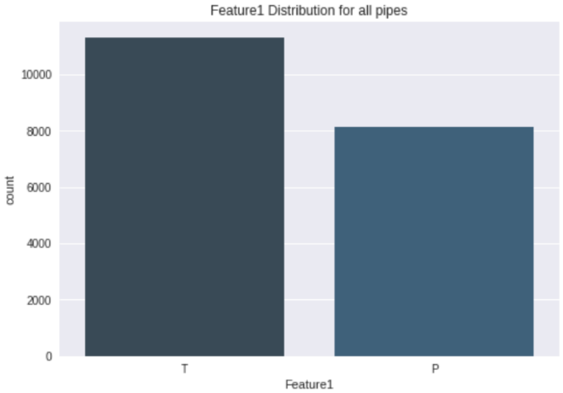

Figure 4: Feature Feature1

2.2.5 *Feature2*

Feature2 has 3 categories IAB, U and O with O a rare value.

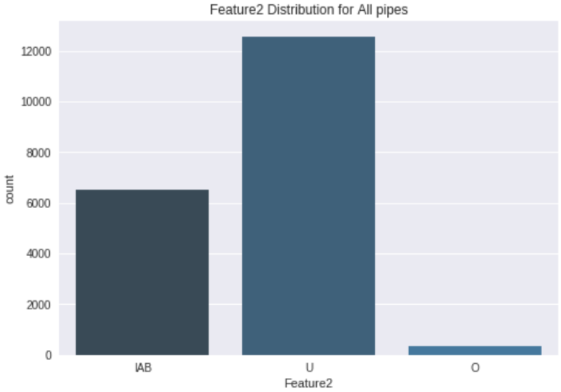

Figure 5: Feature Feature2

2.2.6 *Feature3*

Feature3 is a categorical Feature. It has approximately 20 categories.

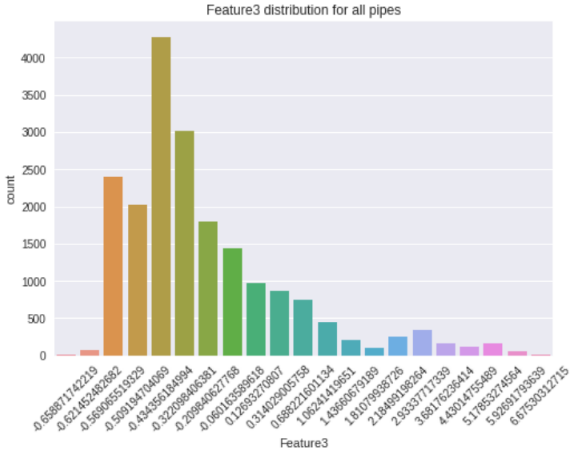

Figure 6: Feature Feature3

2.2.7 *Feature4*

Feature4 has 4 categories C, M, D and Dr with Dr under represented.

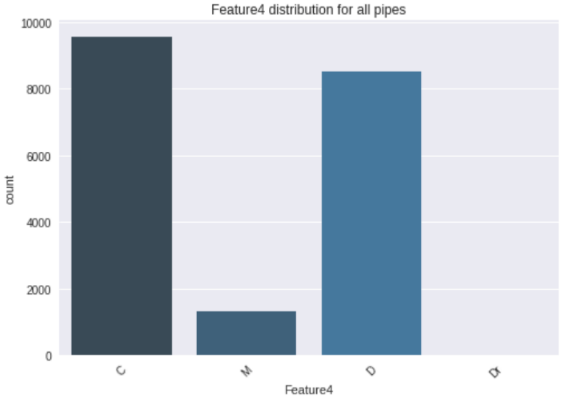

Figure 7: Feature Feature4

2.3 Comparing pipes that will fail in 2014 or 2015 with safe pipes.

A good idea would be to plot these same graphs but separating the two classes. We got more insights about *YearConstruction*,*Length* and *Feature2*.

2.3.1 *YearConstruction* comparison

We notice that fail pipes tend to be older than safe ones. Although it remains uncertain because we don't have much data on fail pipes

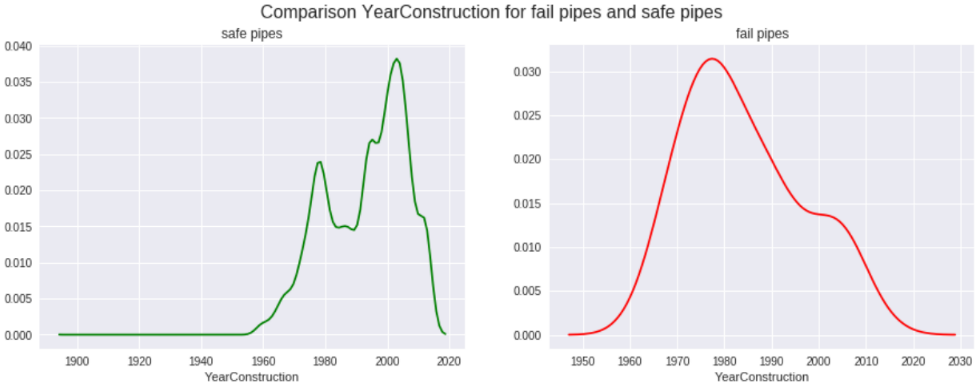

Figure 8: Comparison YearConstruction for safe and fail pipes

2.3.2 *Length* comparison

Again, fail pipes seems to be larger but not significantly.

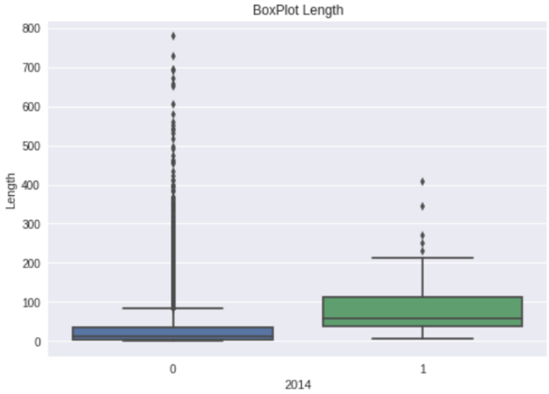

Figure 9: Comparison Length for safe and fail pipes

2.3.3 *Feature2* comparison

There we have an interesting pattern. All the pipes that will fail in 2014 or 2015 have U value for Feature 2 (Figure 2.3.3). We checked this pattern looking for Feature 2 value of *AlreadyFailed* (new feature we created from *YearLastFailureObserved*) pipes and it is very interesting (Figure 2.3.3). Feature2 will probably be very important in our prediction model.

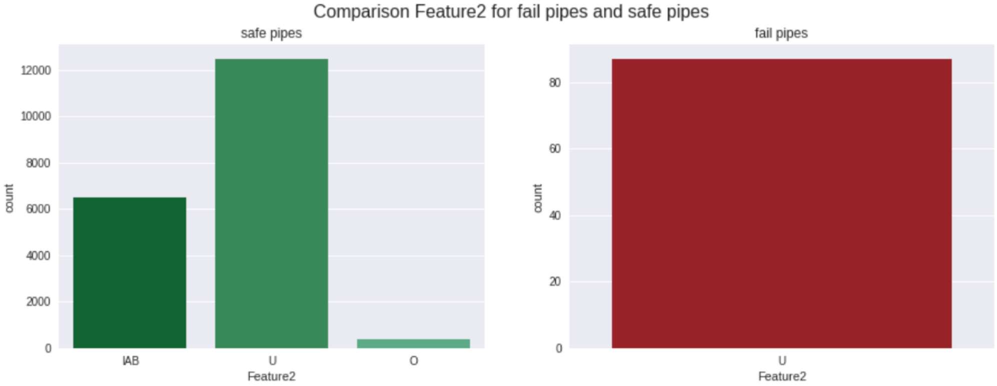

Figure 10: Comparison Feature2 for safe and fail pipes

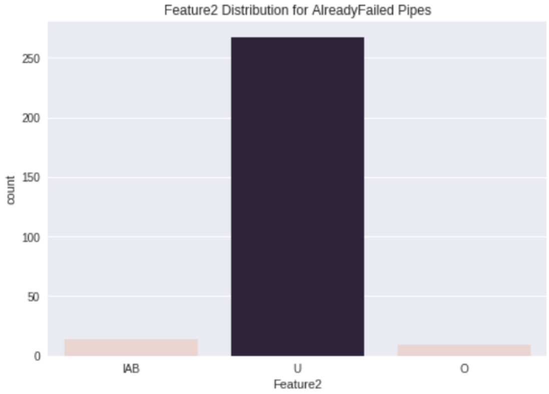

Figure 11: Feature2 for AlreadyFailed pipes

3 Prediction

3.1 General approach

Observing the data, we determined that we would face two main problems.

- The very low dimensionality of the data might make the distinction between good and bad pipes very difficult.
- The very small amount of outliers.

We decided to use different prediction methods and to propose a final prediction aggregated from several classifiers. To tackle cleverly the problem without testing the all panel of methods proposed by skicit-learn, we based our first tries over the sentences of Chandola and alumni (2009):
"Normal points occur in dense regions while anomalies occur in sparse regions. Normal point is close to its neighbuors ans anomaly is far from its neighbours."

3.2 Simple Anomaly detection

We first began to build a simple anomaly detection model to have a main idea of the troubles encountered when dealing with imbalanced data: fraud detection, cancer diagnosis. We can cite two categories of methods to deal with this problem:

- For continuous Data: probabilistic, distance based methods.
- For categorical Data: tree methods.

The first point takes advantage of the idea that normal observations are dense, the second that tree methods could be a good way to discriminate unusual observations.

3.3 Density estimation

The objective is to evaluate the density of the data and find a threshold to discriminate highly unlikely points. Those points would be anomalies within the dataset. We tried:

- Parametric approach: GMM density estimation using GaussianMixture from mixture library.
- Nonparametric approach: Kernel density estimation using KernelDensity from neighbors library.

We then selected for each the 10% less likely points, and found an empty intersection between failed pipes and predicted failures. We rapidly concluded that probabilistic approach might be inappropriate for our problem.

3.4 Distance based/clustering-based methods

3.4.1 K-nearest neighbour

A natural distance based method for anomaly detection is the nearest neighbour approach. It is based on the assumption that normal data points have close neighbours in the normal training set while novel points are located far from those points. A point is declared outlier if located far from its neighbours. But as we saw before, the high level of similitude between failed and normal pipes makes a spatial discrimination with KNN very hard.

A test with the function KNeighborsClassifier from sklearn.neighbors confirmed the difficulty to discriminate by spatial proximity. We see that thinking too micro does not work with our problem.

3.5 Domain based methods

3.5.1 One class SVM

A classical approach of novelty detection is the consideration of only one class because of the highly imbalanced dataset. Intuitively for our problem, considering the high similitude of the different classes over the continuous feature space, the opportunity to increase the dimensionality of our space by using kernel trick might be salutary.

Despite this ability to distort and over dimension our space, the method is very unstable and we find back the troubles we encountered with any spatial approach of the problem. The high level of imbrication of failed pipes within the normal ones makes one class SVM inappropriate for our problem.

3.6 Tree Methods

Very used and known procedures, for example in credit default prediction for financial institutions, are the random forests and extra tree classifiers. Those methods facilitate the treatment of unbalanced datasets, in particular for categorical features. Unlike the previously presented methods, this one does not suffer robustness problems. No matter the under samples selected, the cross validation score stays relatively constant.

3.6.1 Isolation Forest

Isolation Forest [4] is an model that tries to isolate anomalies. It works only for continuous variables. Isolation forest builds an ensemble of trees and anomalies are instances which have short average path length. It works well if the two conditions are checked :

1. anomalies are the minority consisting in fewer instances.
2. anomalies have attribute values that are very different from those of normal instances.

Our dataset fulfils condition 1 but not really condition 2. Which is the main reason why we had problems with this methods. Along with the fact that we just have two continuous features.

3.6.2 Adaboost

The main idea of Adaboost is to train multiple classifiers (trees), every tree is computed using information of the previous tree. Taking a closer look at the algorithm we see that every example is weighted according to its difficulty with the normal classifier. Finally, the final classifier is obtained with a weighted average of all trees to reduce the variance. In our case, this method looked very promising because of its capacity to deal with outliers.

This algorithm is working very well with the SMOTE oversampling method (described later). When we write this report, it is the method that gave us the best results so far in the test set. (87%)

3.7 Oversampling tools

3.7.1 SMOTE

The idea of Smote is to create synthetics samples of the minority class in order to introduce diversity in the training set and fix the imbalanced property.The idea developed in the [1] is the following: we consider a vector of the minority class, and then we find its nearest neighbour and compute the difference between the two. Then we just multiply it by a number between 0 and 1 and add it to the original vector (Figure 3.7.1). Like this, we created a new data point along the data set. The synthetic examples cause the classifier to create larger and less specific decision regions: it reduces over fitting which is one the major issue when working on imbalanced data sets. The strategy of just duplicating samples is limited because doesnt prevent over fitting.

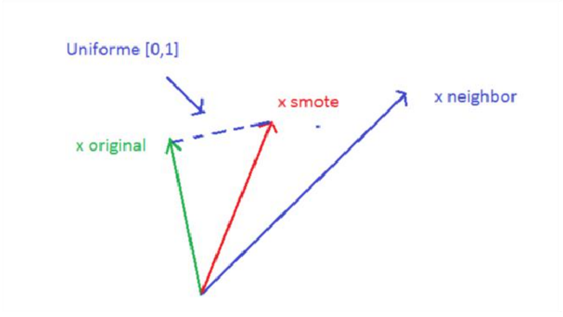

Figure 12: SMOTE idea

Using an aggregation of the algorithms described above, with different classifiers for the years 2014 and 2015, we achieved the score of 0.88 of AUC in validation set (Figure 3.7.1).

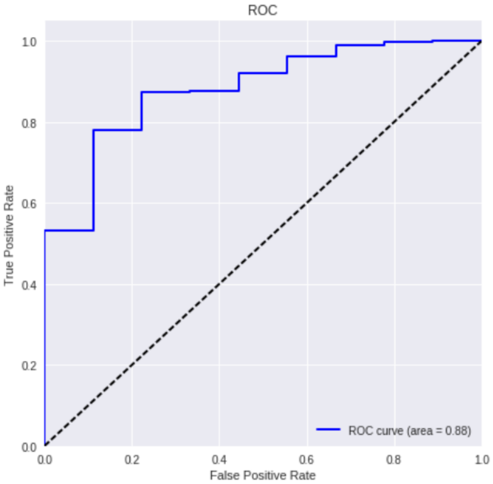

Figure 13: ROC curve

4 Is the choice of ROC-AUC metric wise?

4.1 Disadvantages of ROC-AUC for imbalanced dataset

With such an imbalanced dataset, the use of ROC-AUC may not be very convenient. Large number change in the False Positive [2] can just lead to a small change in the false positive rate in ROC. A classifier that overestimates the risk of failure for apparently safe pipes may do better than a more precise classifier only targeting risky pipes. We think that it is important for Veolia to target most risky pipes to save money for maintenance.

4.2 Another possible choice of metric: Precision-Recall AUC

Precision-Recall curve represents the Recall against the Precision. Precision Recall doesn't consider true negatives so are not affected by imbalance.
Example:
We have 10.000 pipes and only 100 will fail next year.

- Classifier 1 finds 100 risky pipes with 90 True Positive.
- Classifier 2 finds 500 risky pipes with 90 True Positive.

ROC:

- Classifier 1: 0.9 TPR and $\frac{10}{10000} = 0.001$ FPR
- CLassifier 2: 0.9 TPR and $\frac{410}{10000} = 0.041$ FPR

That's difference of 0.040 between FPR of classifiers, it is very small.

Precision-Recall:

- Classifier 1: 0.9 TPR and $\frac{90}{100} = 0.9$ precision
- CLassifier 2: 0.9 TPR and $\frac{90}{500} = 0.18$ precision

That's a difference of 0.72 between the two classifiers precision. That's significant.
This small example shows that Precision-Recall AUC could be a better metric for the challenge (if we keep the idea of maintenance costs). We calculate the precision-recall AUC for some of our best models (for ROC-AUC metric) and got some pretty bad results at it.

5 Conclusion

The most difficult part of the challenge was to identify ways to deal with the imbalanced feature of the data set. Small mistakes slowed us down but we tried to come up with simple ideas and improved slowly but surely our score. We went from a made by hand model to a more sophisticated Gaussian mixture averaged with some tree methods to finally get the best result with boosting methods using Smote. The most important part of our work is not represented by this last result but more by all the steps we put together to understand the problem.

6 References

References

[1] Kevin W. Bowyer et al. "SMOTE: Synthetic Minority Over-sampling Technique". In: *CoRR* abs/1106.1813 (2011). URL: http://arxiv.org/abs/1106.1813.

[2] Jesse Davis and Mark Goadrich. "The Relationship Between Precision-Recall and ROC Curves". In: *Proceedings of the 23rd International Conference on Machine Learning.* ICML '06. Pittsburgh, Pennsylvania, USA: ACM, 2006, pp. 233–240. ISBN: 1-59593-383-2. DOI: 10.1145/1143844.1143874. URL: http://doi.acm.org/10.1145/1143844.1143874.

[3] Tom Fawcett. "An Introduction to ROC Analysis". In: *Pattern Recogn. Lett.* 27.8 (June 2006), pp. 861–874. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2005.10.010. URL: http://dx.doi.org/10.1016/j.patrec.2005.10.010.

[4] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation-Based Anomaly Detection". In: *ACM Trans. Knowl. Discov. Data* 6.1 (Mar. 2012), 3:1–3:39. ISSN: 1556-4681. DOI: 10.1145/2133360.2133363. URL: http://doi.acm.org/10.1145/2133360.2133363.

[5] Marco A. F. Pimentel et al. “Review: A Review of Novelty Detection”. In: *Signal Process.* 99 (June 2014), pp. 215–249. ISSN: 0165-1684. DOI: 10.1016/j.sigpro.2013.12.026. URL: http://dx.doi.org/10.1016/j.sigpro.2013.12.026.