

ORACLE®

UTILITIES

Using Iterative Automation in Utility Analytics

A utility use case for identifying orphaned meters

ORACLE WHITE PAPER | OCTOBER 2015



ORACLE®



Introduction

Adoption of operational analytics can increase an organization's ability to respond effectively to issues. This has a direct impact on the organization's top-line growth. Within the realm of operational analytics, a quick response to an occurrence of an event of interest is critical to the insight's relevancy and also its financial impact on the organization. Many skills and resources are required to create an analytic solution that produces both accurate and timely insights. The analytic solution needs to be iterative in nature, or have opportunities to have integrated feedback. To achieve an optimized solution, these iterations need to happen quickly. This can be achieved through native automation, wherein response or feedback to the previous analytic insight is fed back into the algorithm so that it can fine-tune itself for subsequent insights. In this paper we examine, through a use case for orphaned meters, why the techniques used for iterative automation are critical to successful analytics deployment within the enterprise.

Orphaned Meters

The term "orphaned meters" describes a common phenomenon whereby meter installations are misconfigured in the field such that the new meters cannot be identified by either the head-end system or the customer information system. An orphaned meter is a meter lost to the system of record. This scenario typically occurs during meter swaps—AMI rollouts, broken device replacement, etc.—and results in both a new device that is physically "lost" and an old device that appears to have stopped reporting reads, even though energy continues to be consumed.

Orphaned meters, which may number in the thousands at any given utility, are associated with heavy costs not only from the misplaced assets themselves, but from the resulting unbilled energy consumption. The timeliness of identifying and recovering the meter is crucial to its financial impact on the organization, i.e. the longer it takes to identify the lost meter, the longer the consumed commodity goes unbilled. Discovering meters lost in this particular way is analytically challenging, and many utilities without the right analytic solution forgo trying to find these meters altogether. Once these orphaned meters have been swapped again, from an analytics standpoint, they can never be found. Not only is this an analytically challenging issue, it is time sensitive as well, as the orphaned meter must be found before the second swap because the system of record is not getting the reads necessary for the given premises. This adds a secondary timeliness component to identifying orphaned meters.

The orphaned meter analytics project is just one example out of hundreds of operational scenarios that utilize analytic insights from Oracle DataRaker's library of pre-built algorithms. There are many developmental pieces that need to come together to produce continuously reliable insights that balance the utility customer's requirements for both high hit rate and comprehensiveness. In this orphaned meters example, we narrate the detailed work in methodology selection, integration of client feedback in algorithm refinement and, finally, presentation of the insight in an intuitive format that seamlessly fits into the client's operational processes. While each piece of this development process may seem trivial, they become critical in combination for producing timely results not only for identifying orphaned meters, but for all other algorithms in this pre-built analytics suite.

Defining the process

Oracle DataRaker's initial approach used a manual process of filtering and sorting "candidate" locations for each orphan, based on a number of summary statistics. These statistics were derived from the only two pieces of information Oracle DataRaker was given about the orphans: the relationship of a candidate location to the orphan's date of installation and approximate geographic coordinates. The candidates themselves were culled from the list of meters on the 50 closest premises to each approximate orphan location. This process resulted in many successful orphan-to-candidate matches. Oracle DataRaker was able to confirm the accuracy of many of our predicted matches, although we discovered that the list of orphaned meters provided by the utility was out of date and contained a number of orphans that had since been located. This initial approach was extremely time intensive and was not beneficial for many of the candidate meters, as they had already been recovered.

However, instead of discarding these "found" or "matched" orphans as redundant or duplicate work, we realized we could use the summary statistics of each of the matched orphans to train a generic statistical model of what a successful candidate match should look like. This, ultimately, led to our being able to successfully automate orphaned meter discovery. An initial plot of the data (Figure 1) shows verified orphaned meters (blue) among potential candidates (red) based on various metrics available in our system.

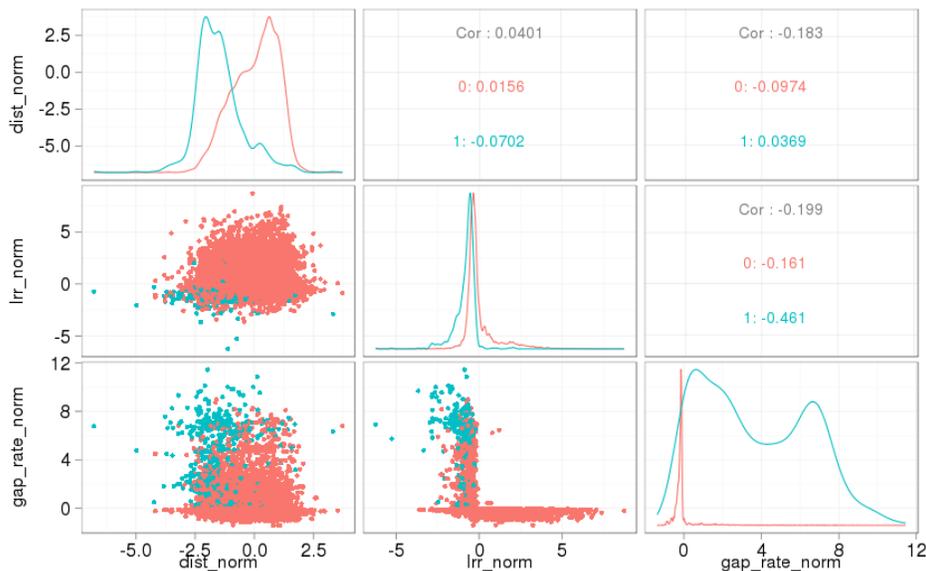


Figure 1: Scatter Pairs: A comparison of three of the metrics used to generate our statistical model, demonstrating how logistic regression curves could be used to identify good candidates (blue) from false positives (red).

As a result of this data exploration process, we implemented two models: a logistic regression and a gradient boosting classifier (GBC). The logistic regression model aims to classify the data into orphans and non-orphans based on the logistic function. On the other hand, the GBC methodology is iterative in nature, as it continuously builds additive decision trees to effectively identify classes that are easily delineated by horizontal and vertical lines. The boosting component is performed by iteratively fitting small trees to subsequent sets of residuals (fitting error), thereby forcing the model to learn more slowly, which reduces the potential for over-fitting. GBC uses a "deviance" loss function to find the optimal parameter thresholds for each tree, equivalent to a logistic regression loss function.

We implemented both models by training and testing on a subset of our data, and selected the model that returned the most accurate results. Based on Figures 2 and 3, the GBC methodology returned more candidates that were considered high priority, which translated to those with a predictive probability greater than 90 percent. We concluded that the boosting component was crucial to providing more accuracy for identifying orphaned meters than a single-iteration of logistic regression.

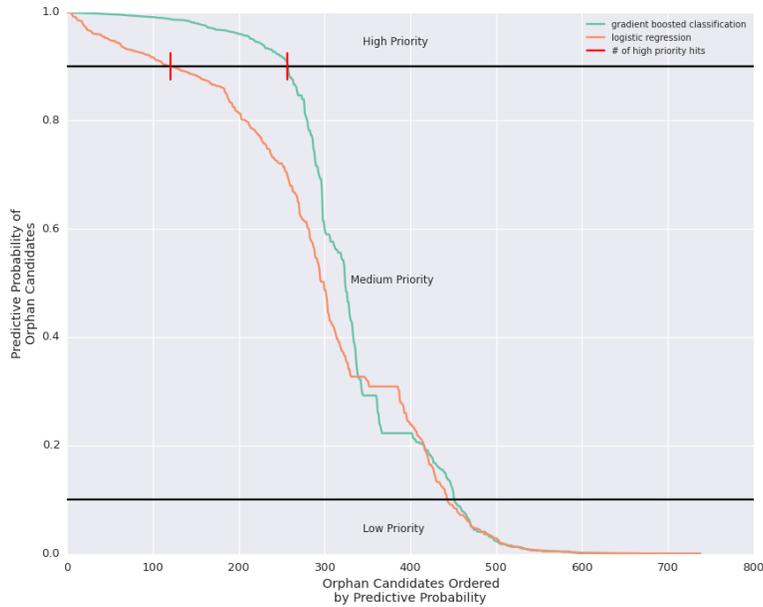


Figure 2: Logit vs. Boosted Line Graph: Although the distributions of predictive probabilities are similar for both algorithms, gradient boosted classification clearly produces more high priority candidate matches, defined by a predictive probability of 90 percent or higher.

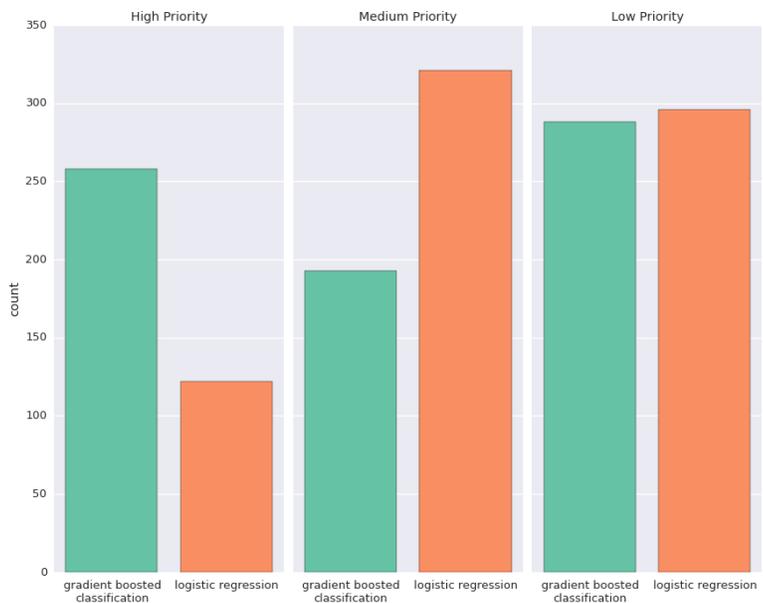


Figure 3: Prioritized Hits Bar Plot: Gradient boosted classification produces more than twice the number of high priority candidate matches than simple logistic regression. These are the candidates that will most likely be fielded by utility personnel.



Refining the approach

The success of our gradient boosted algorithm was encouraging, but the rate of success was still less than expected. After retracing the steps of our analysis, we concluded that the algorithm itself was working properly, and therefore, not the source of the underperforming results. The major obstacle, as is typical when attempting to synthesize results from massive amounts of data, was the data itself. In data science, this principle is often referred to as “garbage in, garbage out,” and implies that any statistical model can only be as good as the data upon which it is based.

As it turned out, the data we were using to train our model actually included many candidates that would not otherwise meet our criteria for candidacy in a real-world scenario: the orphaned meters that had been located by our utility client, contained erroneous data and other atypical characteristics. By incorporating these outliers into our generalized model, we were limiting the model’s ability to discern good matches from false positives. After cleaning up our training data, we also imposed stricter requirements for potential match candidacy, such that any candidate match could not have registered any reads following the installation date of the orphaned meter in question. These two modifications resulted in an increase of nearly 30 percent to the hit rate on our test data, from 67 percent to 96 percent.

Having achieved 96 percent accuracy on the test data, we felt confident in applying our model to real-world orphaned meters to generate a list of predicted orphan accounts. A preliminary subset of these predictions were fielded by our utility client, and evaluated according to two metrics: hit rate (exact match between orphan and candidate location) and find rate (any other orphan is found at the candidate location). After fielding this subset of our orphan location predictions, we achieved an unprecedented 72 percent hit rate, and a 114 percent find rate. Our find rate in this case actually topped 100 percent because multiple orphaned meters were found at locations where we expected to find only one. Although the find rate essentially guarantees productive field reconnaissance for meter technicians, a highly accurate hit rate was our ultimate goal, as this would allow orphaned meter identification to be completely resolved without a field visit for corroboration.

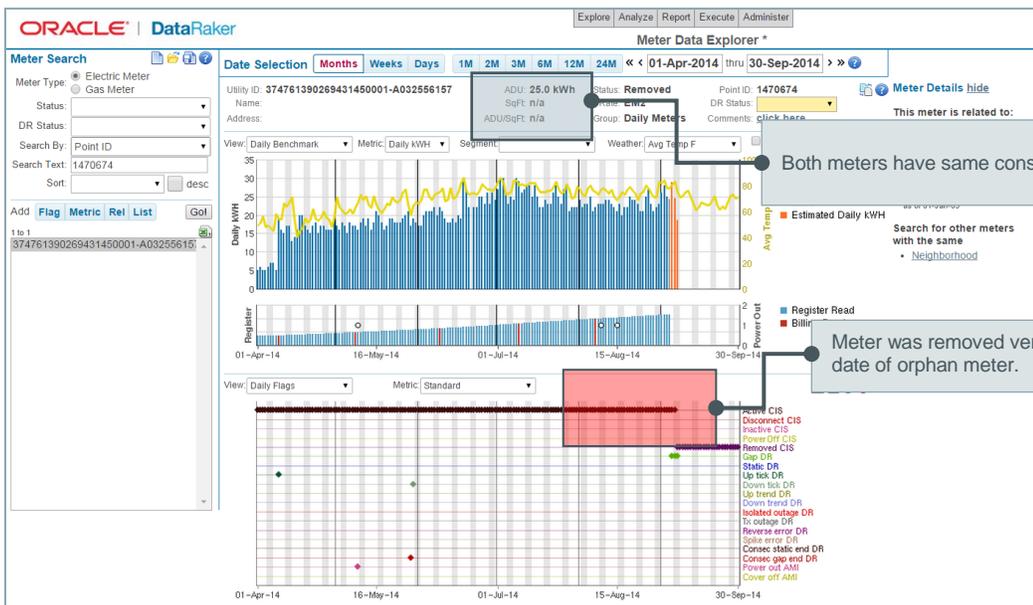


Figure 4: This is an example of an orphan meter pair combination identified by the algorithm. The top meter is the meter that was swapped in but misconfigured in the system of record because the meter does not have an associated account. Fortunately, unbilled consumption was avoided at this account because the original meter prior to the swap (bottom meter) and the correct meter-to-account association, were identified in the system of record before the billing period ended.

Conclusion

In summary, both an iterative approach to algorithm design and evaluation led to the successful implementation of advanced analytics for identifying orphan meters in a timely and efficient manner. Using this approach, we were able to not only find unaccounted-for commodities (gas or electricity) and equipment for the utility, but we were able to do



so in an ongoing, automated fashion such that all future scenarios were discovered within a reasonable amount of time. The net benefits to the business are relatively straightforward; without this analytic solution, these orphaned meters would otherwise be lost and part of the unbilled revenue may never be recovered. This solution enabled data reconciliation for back bill recovery, reduced back billing in general, and reinstated meter assets into the working inventory.

Orphaned meters are just one of many scenarios within meter operations and services that impact revenue collection and asset management. Good operational analytics can also be applied to many other issues, both within the meter-to-bill area and others within the utility organization. There were many contributing factors that led to the successful implementation of analytics for revenue assurance: the right algorithms, the platform that enabled the selection and optimization of statistical methods used, the industry knowledge of what data to pair with what methods and, finally, the efficiency of the platform that enables the algorithm to be performed daily, so that this becomes an ongoing evaluation for timely results. To truly make a quantifiable impact with operational analytics, you must have the surrounding technology and talent to ensure repeatable, comprehensive, and accurate insights.

Contact Us

For more information about Oracle Utilities solutions, visit oracle.com/industries/utilities or call +1.800.ORACLE1 to speak to an Oracle representative.



Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries

Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Integrated Cloud Applications & Platform Services

Copyright © 2015, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 1015

Using Iterative Automation in Utility Analytics
October 2015
Author: Oracle Utilities
Contributing Authors: Ilyssa Norda, Kate Rowland