Rare event detection and dash-boarding for industrial applications

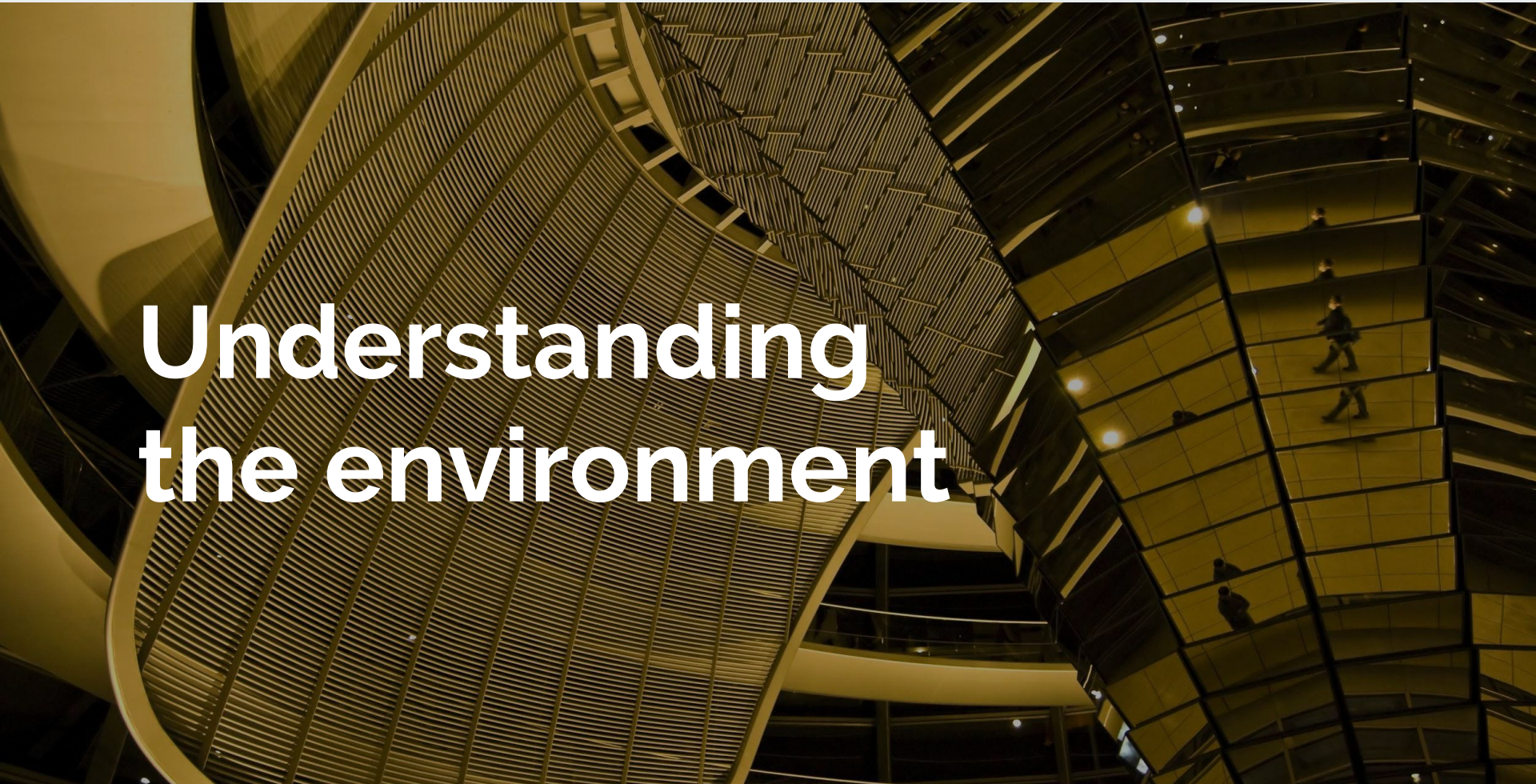
Brandon Bagley
Data Scientist at Cloudreach

Overview

Gauges, sensors, and IOT devices in industrial production facilities such as water treatment have been Key Performance Indicators (KPI's) used to measure and optimize performance. Cloudreach utilized a cloud native technology stack to build a tagging model that predicts rare cleaning events using historical device readings, institutional knowledge, and predictive analytics. Working with subject matter experts within the client's organization we identified the minimum number of sensors at sites within the portfolio to detect events, engineered features, and created an XGBoost logistic regression binary classification model for tagging events with a positive predictive value >= 88% (Note the events have ~0.5% prevalence).

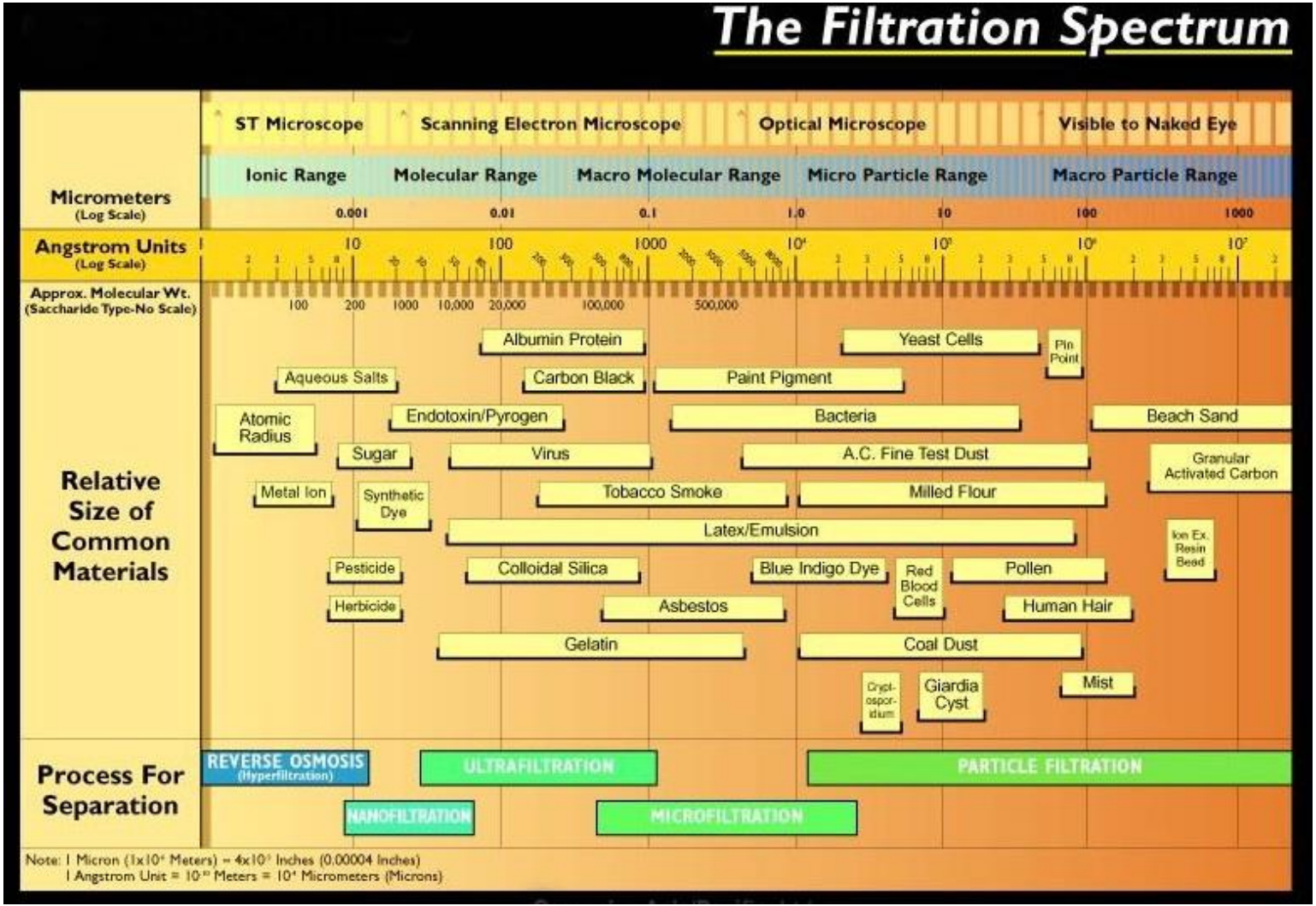
Project objective

Produce a machine learning pipeline to tag rarely occurring cleaning events at Reverse Osmosis (RO) assets and visualize the results



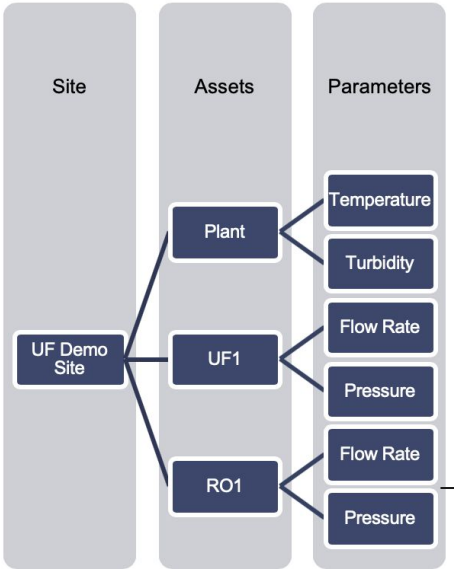
Understanding the environment

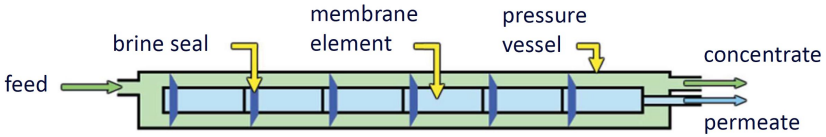
The Filtration Spectrum

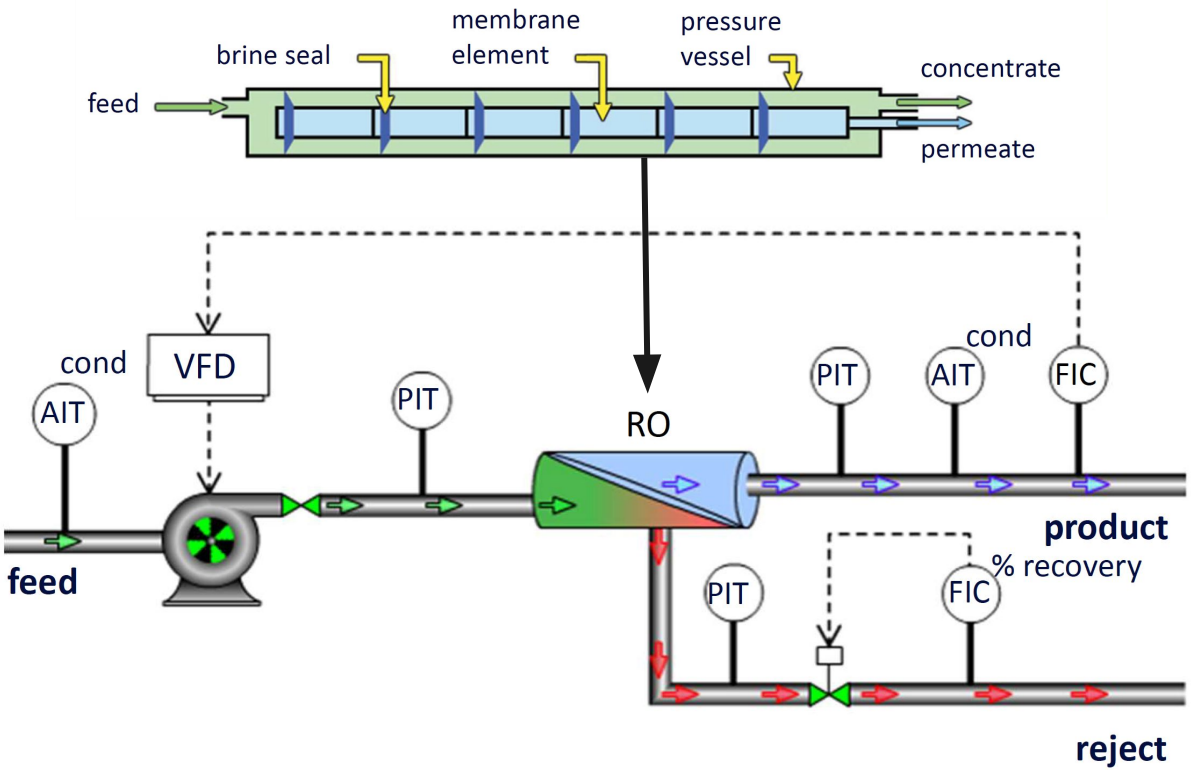


RO Overview

- Sites contain assets
- Assets can be a single RO unit
- A single RO unit can contain multiple RO elements



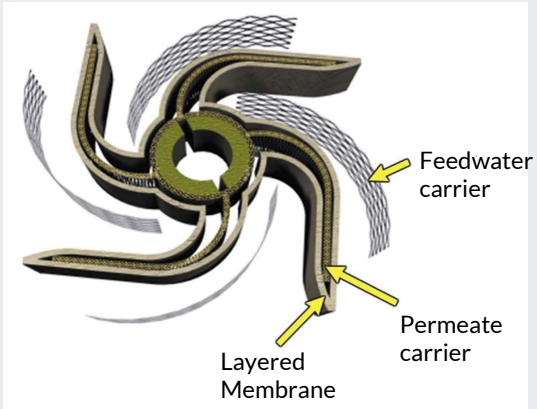




RO Design

Exterior View

Profile View

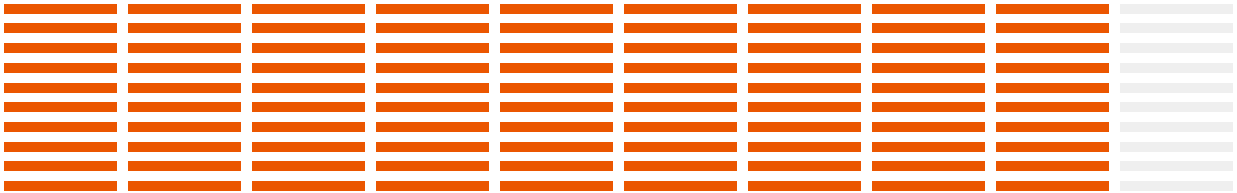


Cleaning Events (*CIP* or Clean in Place)

Cleaning Events at a RO asset occur due to **fouling** or regular maintenance. During cleaning events the RO asset is not operating, requires specialized staff member(s) to identify the fouling type, and application of cleaning fluids of precise makeup and concentrations depending on fouling. In the worse case scenario membranes will have to be replaced, which is the most costly.

- Fouling Types
  - Biofouling
  - Organic fouling
  - Inorganic scaling
  - Colloidal fouling

Reverse Osmosis Assets



~90 %

Reverse Osmosis assets do ***NOT*** have tagged cleaning events

Reverse Osmosis (RO) Data Flow

Data Collection

Collection for each asset, at each site is collected and stored. Very few sites are streaming data directly to the clients data lake in the raw zone

Storage in Data Lake

The current data governance platform stores data in a raw, and refined once new features are computed and raw data is validated

Analyze KPI's and make decisions

Process analysts and clients act on the viz front end, this is where analysts make recommendations for future changes, and predictive suggestions

Feature Computation

The client's data ingestion pipeline includes calculation of new features when there are enough asset related parameters ('steady state' parameters)

Visualize in Client's Custom App

Clients and Process Analysts analyze the data in the custom application where initial settings can be updated, KPI's defined, site "score" identified, and is the basis for the clients offering.

RO Data Tables

Time series (raw) & definitions tables

Transactional Time Series Data

```
|- datetime
|- parameter_key (Primary Key)
|- data quality
|- value
```

Definitions Tables

```
|- parameter_key
|- Parameter Name
|- Site
| | -- Name & ID
|- Asset
| | -- Name & ID
|- Parameter Attributes
| | -- units
| | -- asset_type
| | -- calculation_type (binary)
| | -- comments **(includes cip)**
| | -- etc...
```

RO Data

Step or Mode
&
Cleaning Tags creation

Creating Cleaning Tag (Binary):

1. Map Step/Mode comments to values
2. Validate mappings
3. Create new parameter where mappings contain `CIP` or Clean In Place
    a. 1 = Cleaning Event at datetime
    b. 0 = Not Cleaning Event at datetime
4. On average cleaning events occur at 0.5% prevalence

Issues:

- ➔ RO Train Step/Train Mode is a parameter created during during initial setup
- ➔ Step/Mode is only tracked if the Site Operator uses this feature (hardware or state setting)
- ➔ Step/Mode values are decoded using the parameter comments, which often require validation
- ➔ Only 10% of data contain this parameter, and most require further mapping adjustments

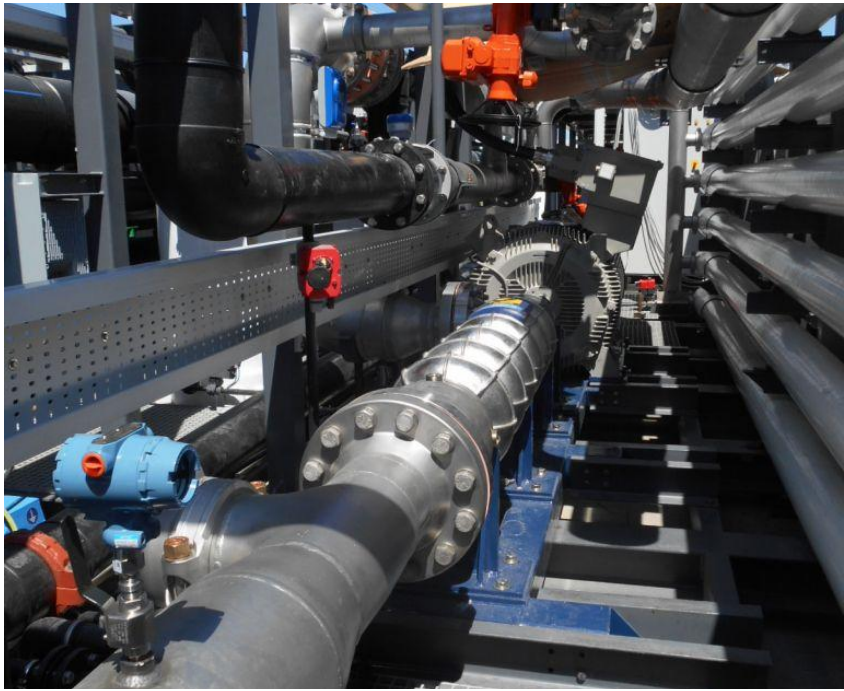
Why predict Reverse Osmosis Cleaning events?

Value Added

01

Reduce RO fouling, extend the life of RO membranes, increase operational excellence

Client Implications:
The main goal of the ML use case is to reduce costs. Completion of the project will reduced the operating costs of RO assets by decreasing the number of membrane replacements and reduce lost production time.



Value Added

02

Introduce services to the market not offered by other analytics services to the public.

Client Implications:
The customer sells service with custom analytics, front end UI, and process engineering after the initial hardware is installed or running. The customer wants to stay ahead of the market on tools for both customer and engineers to use. Correctly predicting cleaning events provides a service that is ahead of the market that integrates into their existing offerings

Value Added

03

More efficient allocation of on-site staff, off-site high billing resources, and lower cost of raw materials raw material for cleans.

Client Implications:
Cleaning at RO sites requires knowledge of the on site hardware, reason for fouling, and correct application of chemical additives or new membranes. Providing a dashboard to both the customer and process analysts can indicate when cleaning events are going to happen based on past occurances and lead to a reduction in cost of the materials for cleaning as well as the staff needed.

Value Added

04

Create a template for further machine learning problems

Client Implications:
The machine learning use case is part of a broader migration from on-site hardware to full cloud integration. By creating a template for the client, this reduces the time from ideas to implementation for future work. In addition by integrating the subject matter experts into the process, decisions can not only come from the IT department but from the business itself

Problems to solve

1. Correctly Identify data sources to use for tagging. Not all sites or machines are created equal and will need to be narrowed down.

2. Work with technical experts to determine which parameters to use, calculated features, and most extreme edge cases that will need to be established.

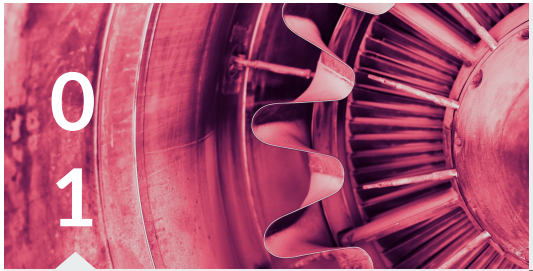
3. Produce Tagging model (binary) with >80% accuracy of cleaning events for a range of sites.

4. Produce a dashboard to visualize the results of both predicted and ground truth values along with KPI's

Proposed solution

From a data lake structure, ingest raw data & ETL to a useable format for an XGBoost Decision Tree model hosted on a deployed endpoint (AWS) and visualize in Tableau

Process



ETL from raw Data Lake

Using raw parquet files from the raw zone, filter useable sites, remove duplicates and poor quality data, extract tags for training, and transform to transactional data in an HDFS

Create engineered features, Train XGBoost model & Deploy

Utilize Amazon Sagemaker for feature engineering, hyperparameter tuning, model development, and versioning before deploying a final model to be integrated into production

Productionalize the process

Create a pipeline with custom modules within AWS for each step for easy re-use and customization. Output all data back to a refined zone for dashboarding in Tableau

ETL from raw Data Lake

- Use raw data (extract for initial development) before any filtration is performed
- Filter
  - Duplicates
  - Non-RO Assets/Sites
  - Lowest quality data by *parameter_key* & *datetime*
- Standardize all data to 15-min medians
- Identify parameter_keys present across most sites
- Create cip_tag for each asset

Data Quality Issues

- Startup data does not reflect running state
- Incorrectly input comments for cip_tag
- Variable parameter_keys across all sites
- Duplicated values
- Rapid cycling on and off creates short duration data

Data Prep & Feature Engineering

Data Preparation for Modeling

- Filter out rare RO parameters
  - All parameters are not available, rare features not helpful
- Convert to wide format
- Scale (Z-score)

Feature Engineering

- RO Calculated Features
  - Created with subject matter experts
- Rolling Mean
  - ½ day, full day, 2 day
- Lag - Used to remove the dependency upon previous data point
  - Reverse and Forward (historical)
- Polynomial Features
  - Up to 3 interactions, to 3rd degree


Model Building
XGBoost

XGBoost
Boosted Decision Trees

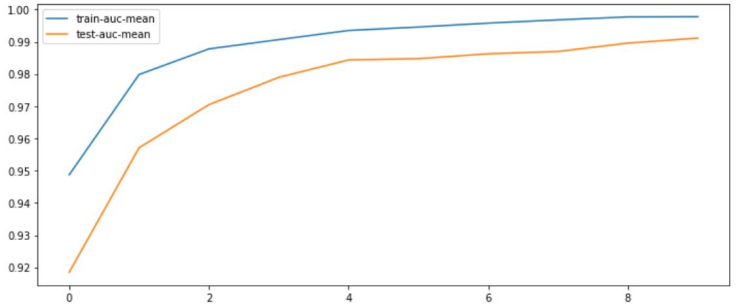
Rare Event Detection with Boosted Decision Trees

- Boosted Trees are computationally efficient for high accuracy
- Accepts NA or missing values
- Rare events can be detected by setting hyperparameters for small output groups
- Easily understood Feature Importances
- Easy hyperparameter tuning

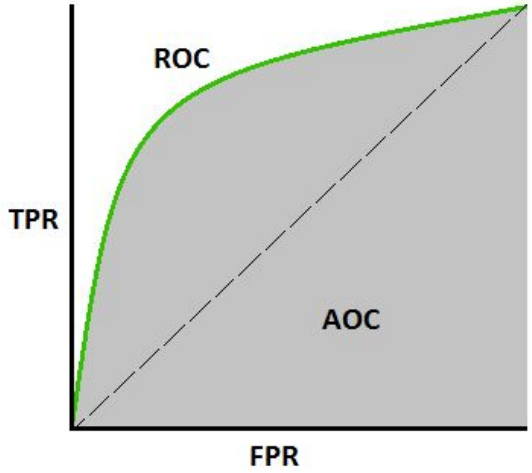
Other Model Descriptors

➔ Logistic Regression (Log Loss Function)
➔ AUC evaluation metric
➔ F-Score to determine Logistic Regression Threshold for binary classification

AUC: Area Under the Curve

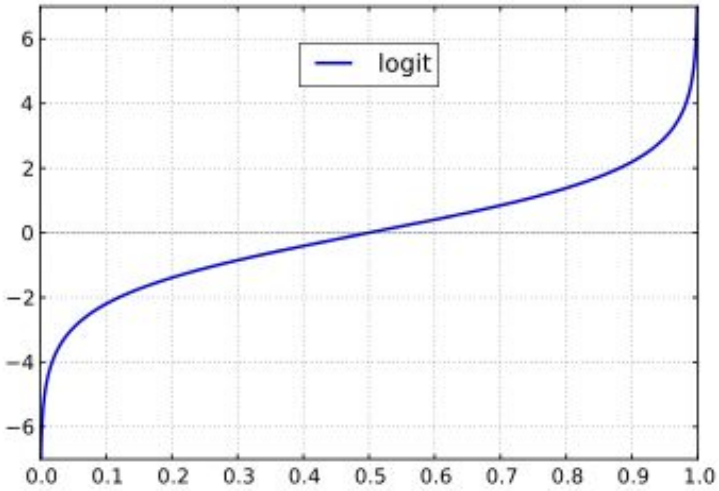


- Refers to the area under a ROC Curve (receiver operating characteristic curve)
  - Larger Values are better, up to 1
- Balances True Positive & False Positive rates
  - Unbalanced classes require different evaluation metric
  - True Positive (Correctly Identified Cleaning Events) occur ~ 0.5% of the time
    - 99.5% Accuracy would be achieved by setting all classifications to False (Not Cleaning)
- Scale Invariant
  - Useful for many sites with varying scales of True Positive Rates (TPR) & False Positive Rates (FPR)



Logistic Regression

- Predicting Binary
  - Clean or Not Cleaning
- Logistic Regression outputs prediction along distribution of values between 0 & 1
  - Logit
- Determine threshold for *Binary Classification*
  - F-Score used based on predictions
- Can be used as a threshold per Asset or globally



Code and Dashboard

Thank you.