Pinnacle eBook

Big Data in Mechanical Integrity: The Next Generation of Corrosion Models

This e-book will Illustrate how data science and machine learning can be used to predict corrosion rates.

Table of Contents

Overview

The first step in mechanical integrity involves identifying threats to integrity. To do this, we need to estimate corrosion rate and damage type. Corrosion rate estimation is typically performed by subject matter experts who use historical data, process and equipment data, and industry standard tools. Experience and “what has been seen before” is a heavily weighted factor that goes into corrosion estimation.

This e-book is going to present a **better way of utilizing data to get a more accurate and predictable corrosion rate estimation**. Specifically, this book will present a study we did on reformer units within the refinery space. For this study, we used big data to train data science and machine learning algorithms to understand how corrosion works in the field and how it potentially differs from the theoretical corrosion rates that we often use in industry standard tools.

We're going to talk about the types of techniques that we use to cleanse our data before we do this kind of analysis and then how we do the actual analysis so that the machine is enabled to provide corrosion rate estimates that can be more accurate than industry standard tools.

Corrosion Rate Estimation

- Important for reliability programs
- Determines inspection frequency, risk
- Estimation currently done by SMEs using industry standard tools (API, etc.)

The Promise of Big Data

- Data revolution can augment SME ability to estimate corrosion accurately
- Data can have errors that limit utility (anomalies, missing values, etc.)
- Statistical techniques can be used to mitigate many of these problems

Use Big Data to Predict Corrosion Rates for Reformers

- We applied data science models to estimate corrosion rate on a set of reformer units
- The machine was able to outperform industry standard tools by a large margin
- This work can ultimately serve as a tool to assist SMEs in their work

The Difficulties of Data

1. We have unprecedented access to data.

- Inspection history, asset properties, stream information, etc.
- All of this data is useful for making inferences about equipment condition, risk, etc.

2. Data Deluge

- Even though we have access to an unprecedented amount of data, the sheer volume of data makes it almost impossible for a human or a team of humans to adequately analyze

3. Data is often plagued with quality issues

- Can lead to bad inference / decision making
- It is possible for a machine to flag questionable data, and can often correct it outright

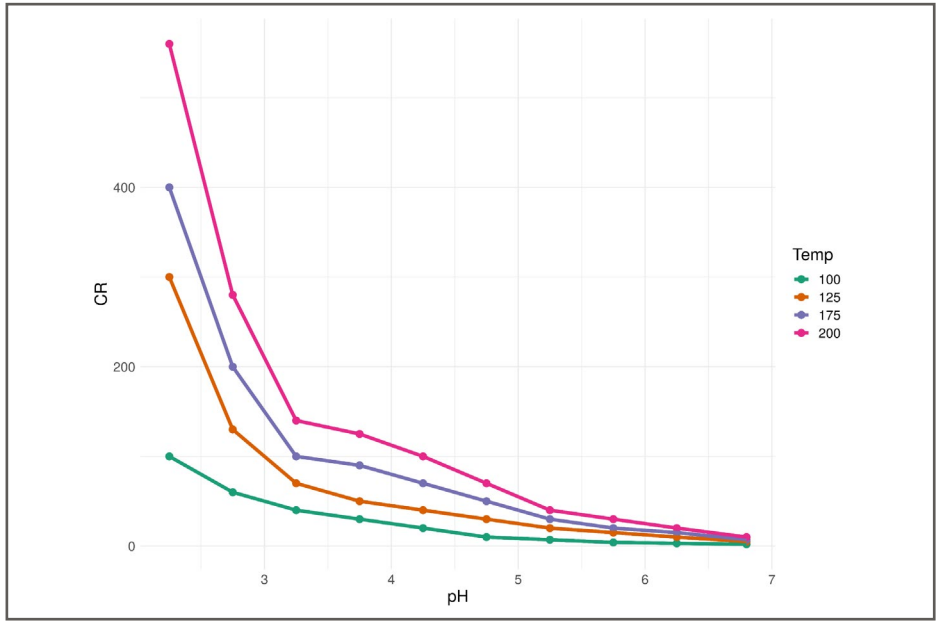
Throughout this book, we'll demonstrate some of the techniques we use to cleanse our data before we run the types of analyses for corrosion rate prediction. By presenting the machine with a cleaner set of data, it will be able to make better predictions.

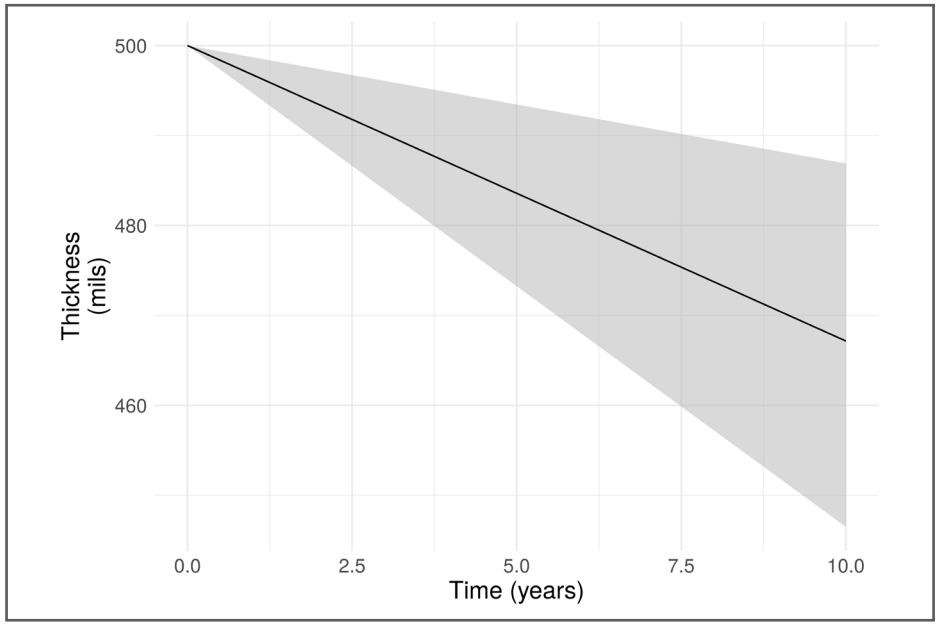
Theory vs Reality

HCL Corrosion – Theoretical Corrosion Rates

Here we'd like to highlight the difference between theory and reality. The first graphic displays the industry standard way of predicting corrosion rates—a lookup table. We're looking at hydrochloric acid corrosion. We have ph on the X-axis and different curves for a variety of temperatures. We all know that as ph decreases, the solution becomes more acidic and we're going to get more corrosion. So lower ph, higher corrosion rate; higher temperature, higher corrosion rate.

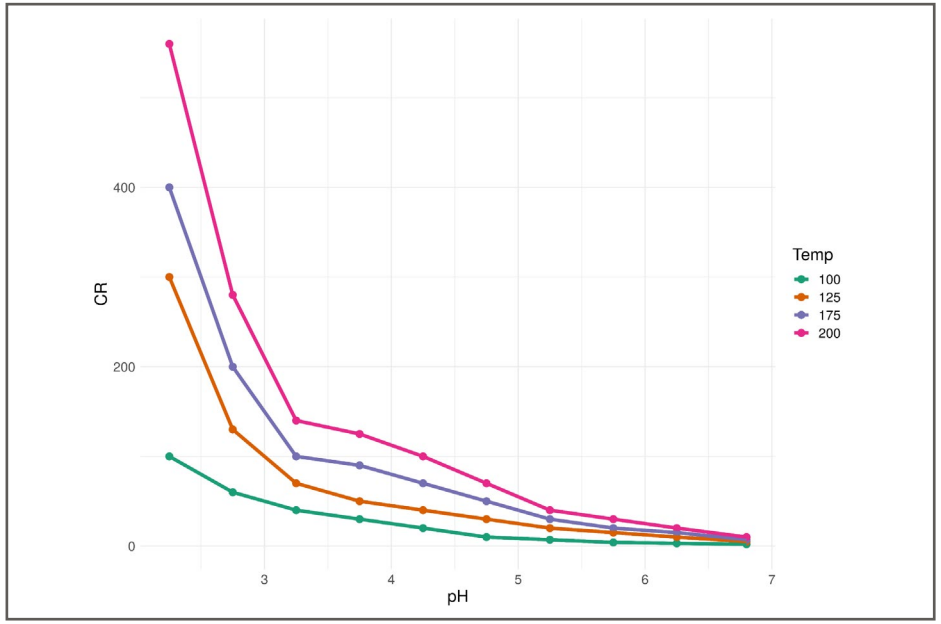
Turning our attention to the second figure, we have thickness in mils, starting at 500 mils. We expect our component to thin out over time at a predictable rate—that's the black line that we're showing. We've also drawn this gray cone around the black line which is indicating some uncertainty that we have with our data. Temperature may not be exactly what we expected—we know it's going to fluctuate a bit—so, the gray region encapsulates the uncertainty that we have with our model.

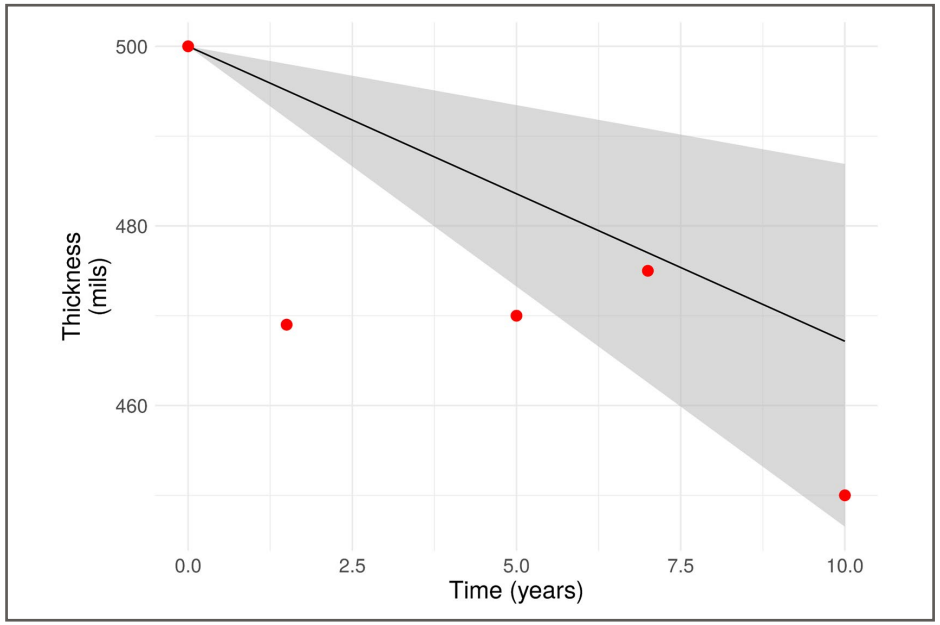




HCL Corrosion – Adding Single CML of Inspection Data

To highlight this a little bit more, we're going to show an example of some real CML data measured on a component undergoing hydrochloric acid corrosion. So, we start at 500 mils (this is time of installation). But after that, the data generally looks different than what we expected. We see very severe corrosion between point one and point two, and then we see things flatten out for a while. We also see a measurement that goes up after a while, which we wouldn't really expect to have happen. We know there's measurement error on these data points. We know there are changes in process that are happening. But the real take-home message is that **the theoretical models are not always what we're going to see in practice**. This is where the machine learning and the data science come in—where we can take real measurement data, real process data, and real asset data to come up with a better model that explains things happening in the field rather than just relying on the theoretical model.





Our Approach

The approach we take is threefold. First, we use statistical tools and methods to cleanse the data. This is crucial if we want a model that actually does a good job. If we don't cleanse the data, it will be garbage in, garbage out. We then use data science to teach machines how corrosion works, using the data that we've cleansed.

Finally, we let the machine provide its estimates back to the subject matter expert for review. This approach marries the subject matter expertise with the data science to provide the best of both worlds and give us something that's going to work better than either could do alone.

Use statistical tools and methods to cleanse data

Use data science to teach machines how corrosion works “in the wild”

Let the machine provide estimates to SME for review

Example 1: Anomaly Detection

The first bit of data cleansing that we're going to do is what we'll call anomaly detection. As an example, let's look at the graph, which displays inspection data taken over time. The trend line here represents a reasonable fit to the data that's going down over time. However, the cone shape here is different than our example from before. That's because the cone here is trying to focus in on the data, so it's going to get wider or thinner based on how much data is informing it. For example, it's thinning out in the area where we have more data—that's providing a better estimate for us. The previous example with the cone did not have data—it was theoretical—therefore, the uncertainty was increasing because there was no information. But here, the cone is responding to the data.

The red dots in the graph are things that we consider inlier valid points. They're following a reasonably good trend. On the other hand, the blue dot is being flagged as an *outlier*. It doesn't look like any of the other data points we have. It represents a huge positive jump in thickness—it's likely a measurement error or a typo from data entry. Either way, these outliers have a negative effect on modeling. It's going to make it very hard for the machine to learn how corrosion rates work if we have a lot of data skewing the results. So, one thing that the machine is very good at is identifying those points automatically. It can recognize a point within a particular set of measurement data that doesn't match expectations.

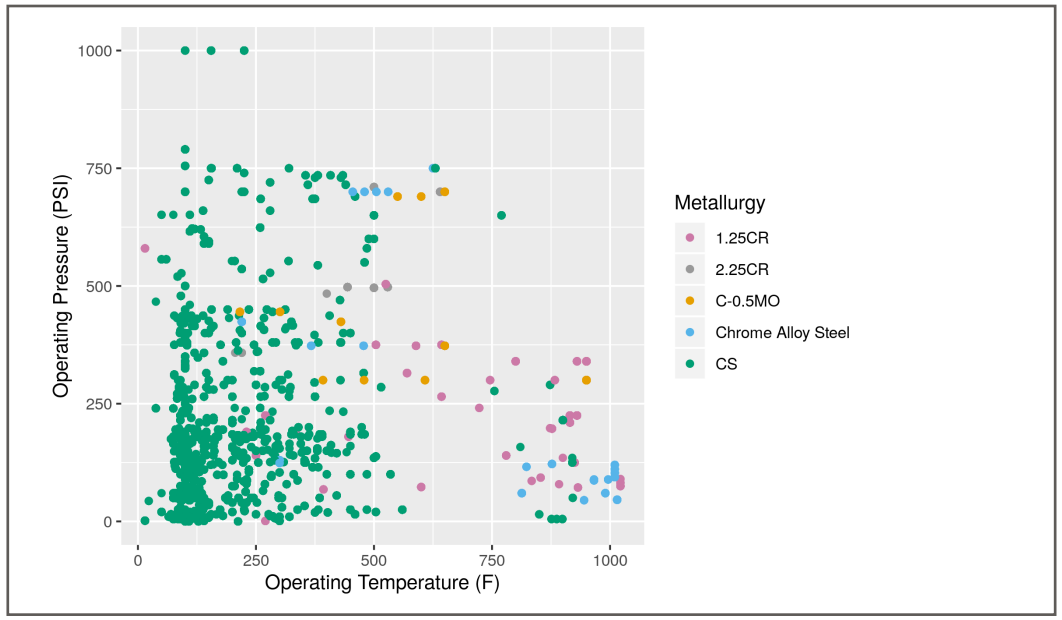
When this occurs, we can flag the reading and send it to an SME for review who can then decide to keep or remove the data point. Another option is to automatically remove these things or discount them from our analysis. To do this, we would tell the machine to flag data points that go past certain values, still giving the SME input and control over this process.

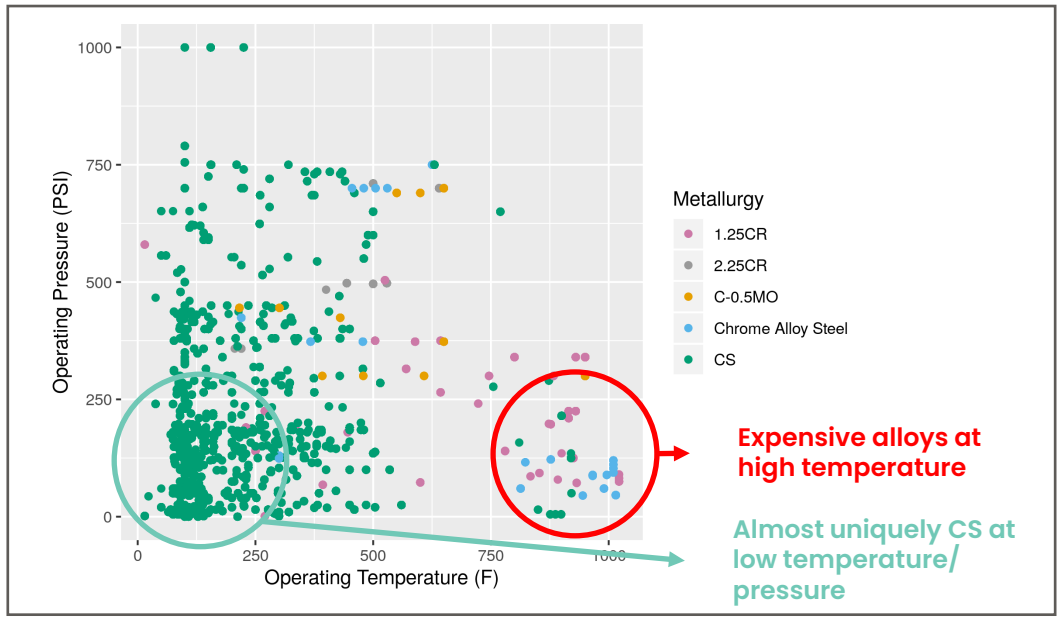
Example 2: Dealing with Missing Data

Missing data is a problem that we run into in our industry. As an example, let's talk about metallurgy. Within our data set used for this study, metallurgy was generally available for all components, but it was missing a good portion of the time.

One way you can deal with missing data is to throw out any observations that have a missing data point—but that would be a tremendous amount of data throw it away and it would severely hamper the power of the machine learning algorithm, which is data hungry and requires a lot of data to be able to get off the ground. In short, we don't want to throw data out.

A lot of times the data that's missing correlates very strongly with other fields. Metallurgy, for example, it's going to correlate a lot with the stream information, the temperature, or other variables that we might have access to in our data set. We can use all that information to enable the machine to make reasonable guesses for what the missing value should be. And again, this can be sent back to an SME for review. The machine can even try out a couple of them if it's uncertain about things.



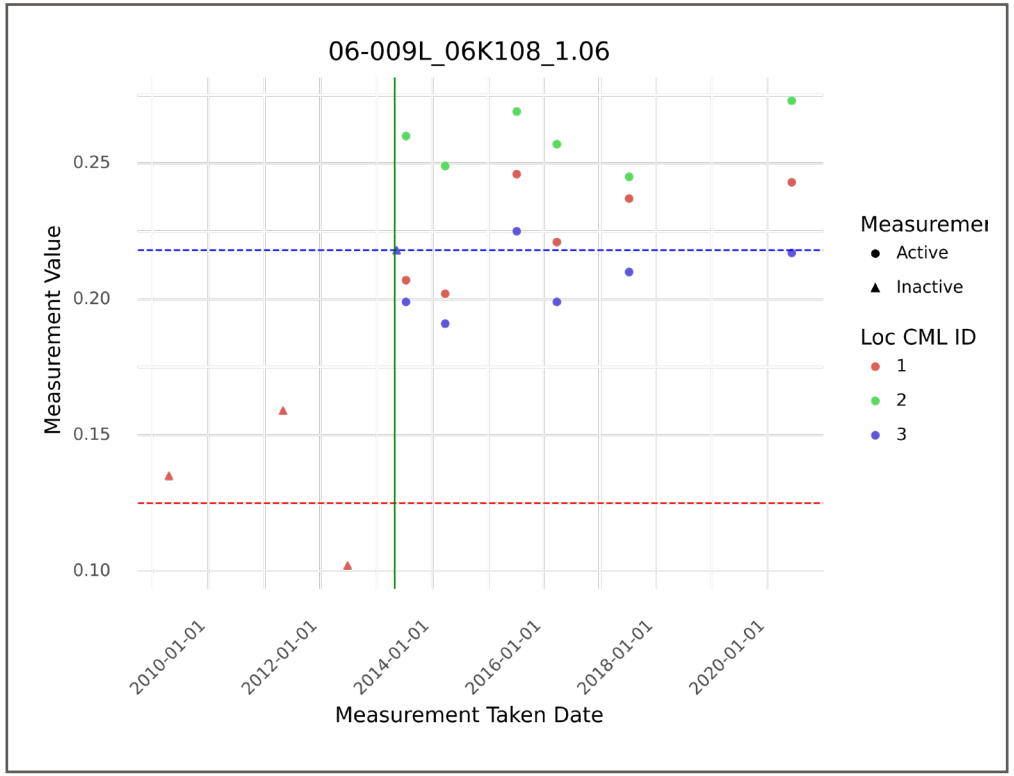


Example 3: Poor Data Quality

We've been discussing missing data but a lot of times we are simply dealing with erroneous data where something has been labeled, but it's probably been labeled incorrectly. In these cases, the machine can flag those types of fields and refer to an SME to confirm or deny.

One example of this that was definitely a problem in this project included undisclosed repairs and resets. In the data set we have a few CMLs on a single piping circuit. When looking at this graph, it clearly looks like a repair has happened. We're starting with data that has a very low thickness and then suddenly, around 2014, there's a huge step upwards. The machine flagged this as a potential undisclosed repair and the client later confirmed this was the case. So, the machine can flag differences in the data which can then be given to an SME for review. Again, the machine is learning based on the data it's exposed to, so you can prepare it for changes—it can self-evaluate and recognize when there is something going on that should be flagged.

And again, this is about enabling the SME to do their job more efficiently. We don't trust the machine entirely to do all of this, but with the SME in tandem, we can make good decisions that are ultimately going to enable better production.



Big Data in Practice: Reformer Study

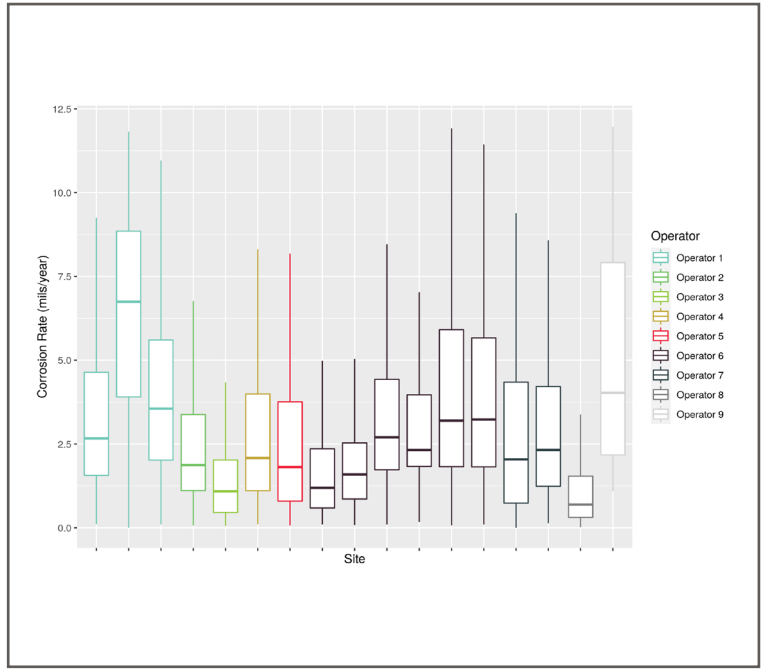
The reformer study took a large body of data that we have access to—over 20 facilities worldwide, representing ~10% of U.S. reformers and ~2% of worldwide reformers. The goals of this study were to show that show that data-driven methods can accurately estimate corrosion rates and to compare a data-driven method with industry standard approaches. The data set included 20 facilities, 36 units, and 9,943 assets.

This study gave us a great opportunity to compare the operations, the performance, and specific information on a variety of units that have a related function, some of which do their job differently than others. We did not distinguish between a fixed bed or continuous regeneration type of a reformer.

So, there's going to be some variety in the data, which the machine will respond to. And that variability gives the machine more power, more insight, and more ability to interpret an individual unit's data.

To demonstrate the variability, the first graphic shows a snapshot of what corrosion rates look like for every different site and location within the dataset. Some of these sites have very high corrosion rates or very wide corrosion rates—some are very low and narrow. And again, it's good for the machine to see diversity. It's going to observe similar operating conditions at these different sites and locations. and knowing that there are differences between the sites is useful—it helps the machine understand what it can predict well and what it can't predict well.

It ultimately enables it to hedge its bets a little bit better and provide more reasonable estimation than what it could do if we try to slice and dice the data in different ways and segregate what the machine gets to see.



Data Used By Model

The data we fed into the model included things like process conditions, temperature pressure, material of the equipment, measured thickness data from the inspection history, operating trends and changes in those trends over time, and stream constituents.

Inspection Data
Inspection date, measured thickness

Location Information:
Operator and Site

Asset Conditions:
Temperature, pressure, metallurgy, insulation, PWHT

Stream information:
H2S, NH3, Water Mole %, etc.

Developing the Model

Once we've prepared our data, we're ready to train the machine to understand how corrosion rates work in the field. We do this through something called *supervised machine learning*. This is where we feed the machine data examples. For example, we give it data surrounding a component such as temperature, pressure, stream information, metallurgy, and observed corrosion rate. We feed the machine more and more data examples like this, and as the machine is exposed to the data it starts to learn relationships between how those pieces of data correlate with the corrosion rate that it's being told happened. As an example, it will learn that as temperature increases, we also tend to see an increase in corrosion rate, or, that as certain stream constituents increase, we're going to see a higher corrosion rate. After we've fed the machine all these examples, it's going to get an idea of how corrosion rate actually works. Then, what's exciting is that we can use this to make predictions on things that the machine has never seen before—different temperatures, different metallurgy configurations, other things that it's seen a little bit of, but never exactly quite the same as what we're feeding it—and it can make reasonable predictions.

Example-Based (Supervised) Machine Learning

- Feed machine data examples
- This is done at the CML level
  - Location, operating conditions, stream info
  - The measured corrosion rate (from the inspection data)
- Machine learns from these examples how data relates to corrosion rate

Model Outputs

- Estimated corrosion rates on new equipment with confidence intervals
- Variable importance

Corrosion Rate Estimates

As an example, we have a plot that shows what a predicted corrosion rate might look like. The blue dashed line indicates the actual corrosion rate that was observed on this particular component. The machine predicts around five mils per year—the actual is about six miles per year. Alongside the corrosion rate prediction, the machine also provides a degree of uncertainty of what the corrosion rate should be. Our machine is relatively confident that the actual corrosion is somewhere between four and six mils per year. It turns out that's within our window. When we look at the 95% confidence interval across the predictions that it made, we found that most of our predictions fell within that. So, the model is doing a good job. It's doing reasonable things. But the fact that it provides a confidence level is good because, one, it's giving us an estimate of what it thinks corrosion will be like, but it also lets us know when it's certain about things and when it's less certain about things.

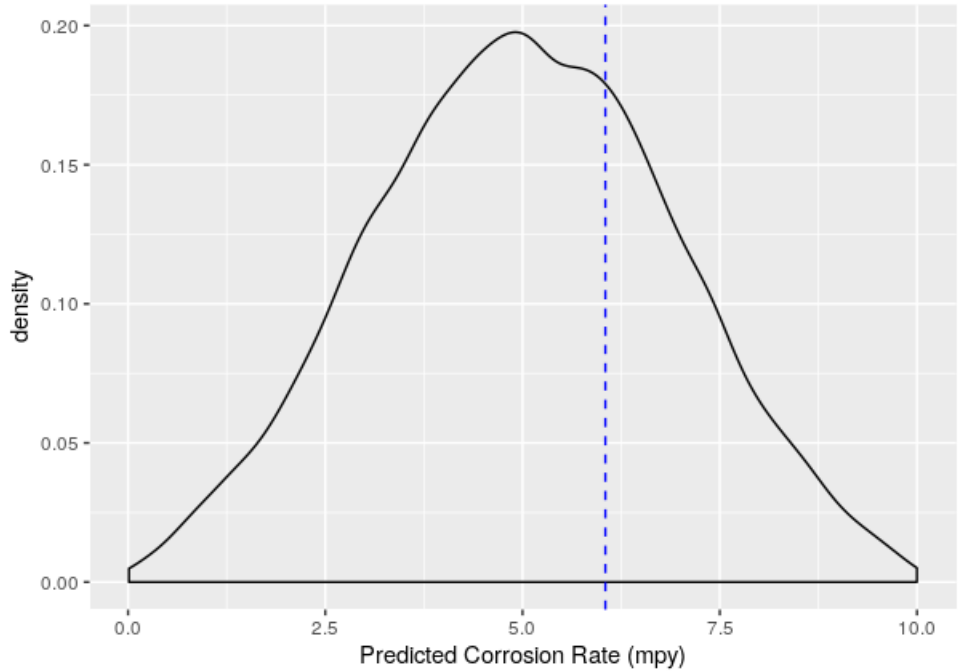
Model produces corrosion rate estimate and confidence *interval*.

Example (single component):

- Actual corrosion rate: 6.05 mpy
- Our model estimates an expected 5.0 mpy
- 80% confidence interval: [3, 8] mpy
- 95% confidence interval: [1.5, 9] mpy

Across the entire dataset:

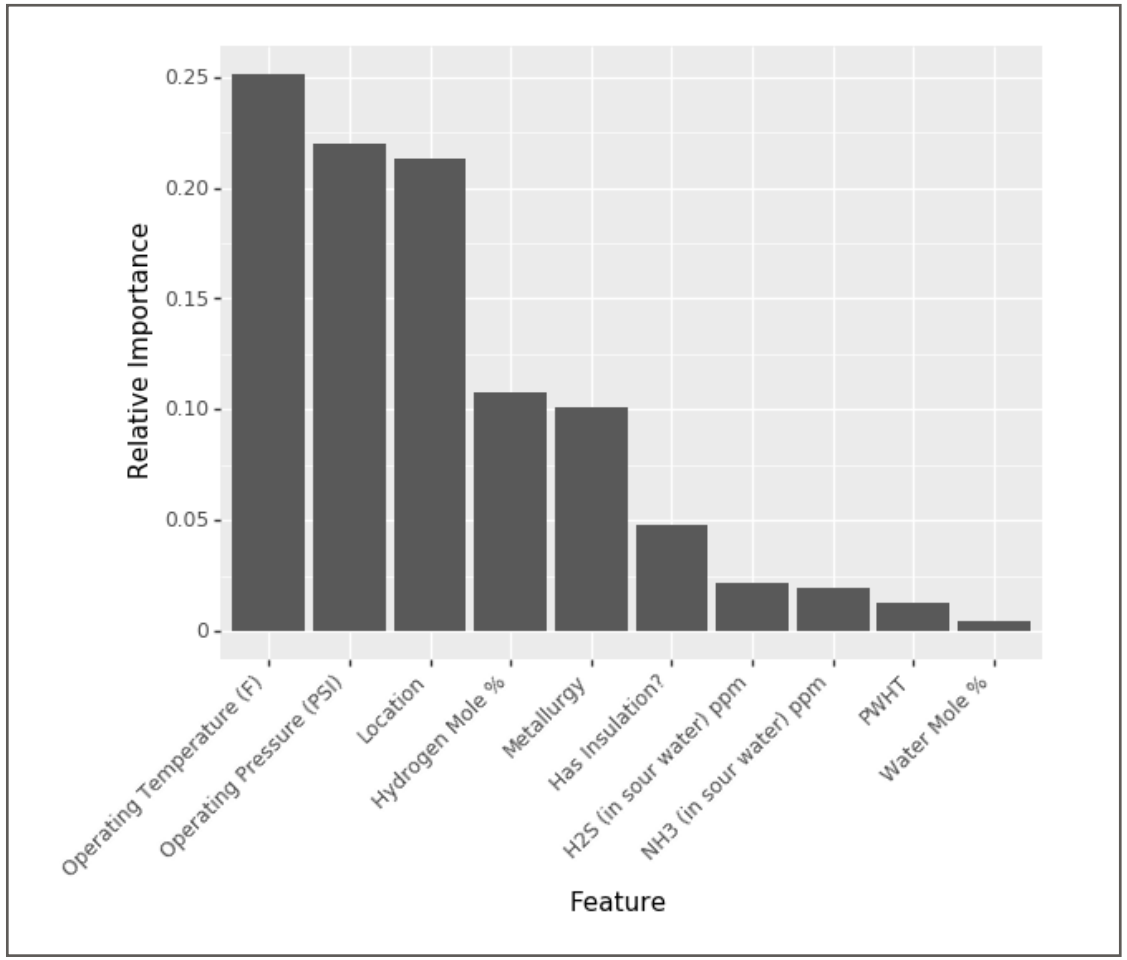
- 93% of our predictions fall within 95% confidence interval



Variable Importance

The machine gets an idea of what things matter and what things don't—things like temperature, pressure, the location where the reformer is operating, hydrogen mole percentage—all those things matter quite a bit. As demonstrated in the graphic, the water mole percentage didn't matter very much to the machine. There is a reason for this. Reformers have to operate dry, therefore, the absolute amount of water is going to be very low. So, in this particular process, unit water should not be a big factor. And again, this is a data point that can be reviewed and verified by an SME.

The model considers a lot of variables. Some were deemed more valuable than others.



Model Validation

We trained a model . . . how do we know if it's any good?
We did two things to validate the model:

1. Internal testing on the data we have (cross validation)
We can't test the model using data it's already seen (that would be cheating). So, we divided our big reformer dataset into a series of chunks, a portion of which would be used for training. We then trained the model using a limited subset of the data and had it make predictions on things it had not seen yet. This testing did fairly well. The model had an average error of 2.2 mpy across all data, which we considered to be very good, especially given the wide range of corrosion rates that that we saw in practice.

Site	Metal	T	P	Water Mole %	H2S ppm	NH3 ppm	H2 PP	True rate (mpy)	Pred Rate (mpy)
Site 1	CS	200	20	0.092	198	61.5	319	4.03	4.05
Site 2	CS	100	130	0	155	0	337	2.07	2.59
Site 1	Nickel St	1050	45	8e-5	5	5	201	4.35	3.76

Metric	Value
Avg. Absolute Error	2.2 mpy

2. Direct comparison to industry standard tools

Here, we enlisted a human subject matter expert to go up against a machine and compare results. We gave the SME access to the same data that the machine had—but we held out 11 different components inside of that data set—and asked them to provide their own estimation of corrosion rates using industry standard tools, along with their own experience and knowledge. We also trained the machine learning model on everything except for those 11 components and asked the model to do the same thing—predict corrosion rates.

When we compared the two, the machine outperformed the human on 8 of the 11 components. There were 3 where the human did slightly better than the machine did, which we would actually expect because, as mentioned earlier, industry standard tools are necessarily generic and therefore also unavoidably conservative. When we did the final comparison, the machine ended up outperforming—the machine error was about 73% while the human error was about 184%, 5 mils per year for the industry standard method, 3.1 for the machine learning model.

Asset ID	Comp ID	Meas. Rate (mpy)	Industry Rate (mpy)	Industry % Error	ML Rate (mpy)	ML % Error
1	A	11.4	1	91%	7.5	34%
2	A	0.8	10	1150%	4.2	425%
3	A	18.9	6	68%	3.2	83%
4	A	6.9	4	42%	4.9	29%
5	A	1.9	6	216%	3.1	63%
5	B	3.3	6	82%	4.6	39%
6	C	3.8	6	58%	2.6	32%
6	C	2.1	6	186%	2.6	24%
7	A	3.1	6	94%	3.3	6%
8	A	8.2	6	27%	3.9	52%
8	B	3.3	3	9%	2.9	12%

Metric	Industry	Model
Mean Abs Error	5 mpy	**3.1 mpy**
Mean % Error	184%	**73%**

Conclusion and Takeaways

This study has demonstrated that we are able to combine subject matter expertise with data driven methods to revolutionize corrosion rate estimation. **The data science model outperformed industry standard tools** in terms of accuracy and we demonstrated that the model made sense and passes the SME smell test. By marrying the strengths of Big Data with subject matter expertise, we end up with the best of both worlds and with quality that exceeds what we're able to do currently in the industry. We'd like to note that we do not at all suggest that data driven methods be allowed to run wild and we are not trying to replace subject matter experts.

While this study covered reformer units, we have also applied this modeling to other types of units—we've done this on hydrocrackers and crude units, and we see very similar results to what we saw with reformers. So, we're just getting started and then we're going to be doing a lot more in the months to come. Visit pinnaclereliability.com to follow along.

Major Takeaways:

- Combining SME experience and expertise with data-driven methods has potential to revolutionize corrosion rate estimation
- Big Data provides attractive possibilities for analysis
  - Machine can analyze patterns in data faster and more efficiently than human.
  - Data quality issues can limit effectiveness of machine methods. Appropriate strategies to deal with missing/poor quality data need to be employed.
- Demonstration of method for reformer units
  - The study showed that the data driven model was accurate, generally providing more accurate corrosion rate estimation than industry standard tools
  - The model can be applied to units other than reformers, e.g., hydrocrackers, crude units, etc.

Contact Us

Headquartered in Pasadena, Texas, Pinnacle is exclusively focused on helping industrial facilities in oil and gas, chemical, mining, and water and wastewater better leverage their data to improve reliability performance, resulting in more production, optimized reliability and maintenance spend, and improved process safety and environmental impact. For more information, visit pinnaclereliability.com

info@pinnaclereliability.com

+1 281.598.1330

pinnaclereliability.com