Picking the Best Predictive Model: A Case Study in Analytics Operationalization

Craig Adams

craig.adams@elementaryinsights.com

July 5, 2022

INTRODUCTION

Background

Machine Learning (ML) is a powerful approach for converting data into actionable information for faster, better data-driven decisions within an organization. However, ML techniques are not necessarily the best choice for each business situation. A certain amount of expertise is required to design and develop predictive ML models in order to generate the desired analytical insights and these skills can be expensive to acquire and retain. Expertise is also required to "operationalize" the insights by converting them into an effective and sustainable digital product that can be used by the decision makers on demand. Both of these steps can require investments in technology infrastructure, such as third-party cloud services to host the data, run the ML model and publish the actionable information to a web-based digital interface.

There are situations where simpler, perhaps more traditional data analysis techniques would suffice, both in terms of ease of development (i.e. an Excel spreadsheet on a PC) and deployment (i.e. a "business rule" that people use to make the right decision when a known problem arises).

Project Goal and Case Study Objectives

The goal of this project is to explore this space, more specifically the development of different analytical solutions for a particular business problem, starting with the simplest techniques before proceeding to more complex ML models. A case study was chosen from the field of Equipment Reliability, where a major challenge is managing the lifecycle of a piece of equipment as the components wear down and performance degrades. At some point during the lifecycle, there is a failure event where the equipment is no longer operable. From a cost perspective, it is ideal to run the equipment or component to the end of its useful life prior to replacing it BUT, it is usually ideal to replace it just before failure to avoid the increased cost of unplanned failures (Figure 1).

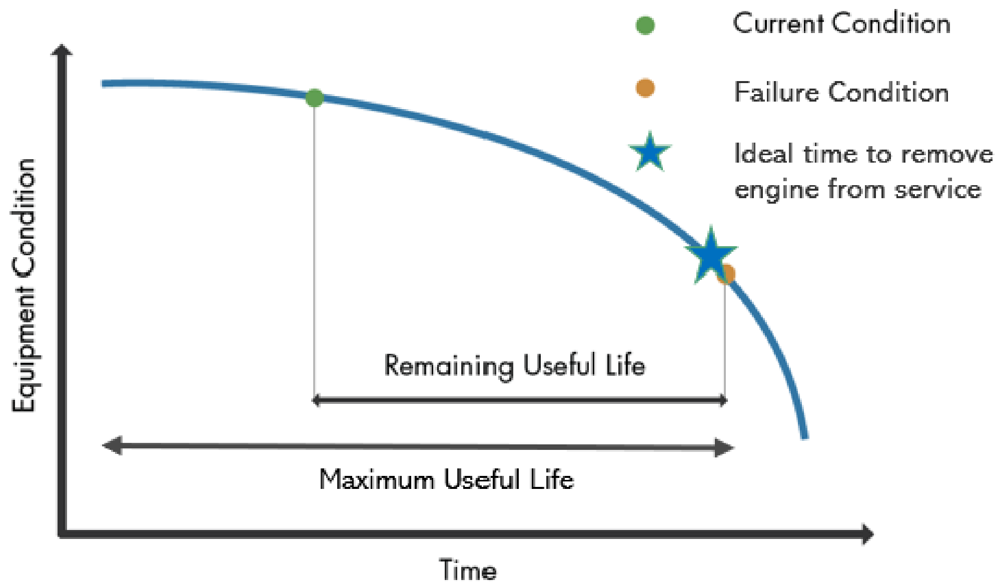


Figure 1. Generic Deterioration Profile and Useful Life Concepts for Equipment that Runs to Failure

The case study uses the FD_001 NASA Turbofan Jet Engine Data Set which describes the degradation in performance then failure of a fleet of 100 turbofan jet engines. The specific objective of the case study was to explore the development of predictive models that would maximize the life of the jet engines while avoiding unplanned failures. In keeping with the project goal, each model was then scored in the context of its technical performance, its potential impact on the competing business KPIs and the relative ease of development and operationalization.

METHODOLOGY

The NASA data set was loaded into Python then analysed and modified for the modeling exercises. The initial data set contained the lifecycle records for each engine, with three operational settings and 21 sensors tracking the engine performance for each cycle. The processed data set was reduced to 12 sensors with useful information about the degradation in engine performance. Six scenarios were considered and five progressively more complex modeling exercises were conducted using known statistical and data science techniques. Full details of the work can be found in a series of Jupyter notebooks.[1]

RESULTS AND DISCUSSION

The results of the study are summarized in Table 1.

Table 1. Case Study Results for the Six Scenarios

Scenario / Model	Model metrics & Goodness of Fit	Relative Impact on Business KPIs	Development Complexity	Relative Operationalization Costs
Do Nothing	-	100% operating efficiency, 100 unplanned failures	not applicable	none
Risk Adverse Weibull Curve	statistical test unconfirmed	64% operating efficiency*, 1 failure in 100	low	low
Sensor Alert	normality and t-tests passed	83% operating efficiency, no failures	moderate	low to moderate
Tangential Regression model with threshold	R2 = 0.60, did not pass other stastistical tests	92% operating efficiency, 1 major unplanned failure	high	not applicable
DT3 Model	F1 score 0.73	91% operating efficiency, no failures	high	moderate
Logistic Regression Model with PCA	F1 score 0.78	94% operating efficiency, no failures	high	high

* full data set

The first scenario was the baseline "Do Nothing" scenario, where theoretically another set of 100 engines from this fleet would be allowed to run to failure. In this case, the maximum useful life expected for the fleet would be 20,531 cycles and there would be 100 costly engine failures anywhere between 128 cycles and 362 cycles of operation.   There would not be any costs however for developing or operationalizing a predictive model.

The second scenario took the opposite approach and used a simple probability curve on the failure data to establish a business rule that would remove all future engines at 131 cycles. In this case there would only be 1 expected failure. However, the operating efficiency, or consumption of available life cycles would be very low at 64%. The model complexity was quite low, and it would be deployable as a simple

[1] The series of Jupyter notebooks are titled "Craig_Adams_Capstone_Part1 through 4"

Excel spreadsheet that would require very infrequent updating. One drawback was that the statistical significance of the Weibull curve could not be confirmed.

The third scenario utilized the concept of a check engine light, where a threshold would be set for one of the sensors. When that value is reached, the engine would be removed from service. This statistical model once developed worked quite well for the given data, increasing the operating efficiency to 83% without a failure being detected. Figure 2 shows the Remaining Useful Life (RUL) for the engines that were removed from service, with a mean value of 35 cycles remaining.

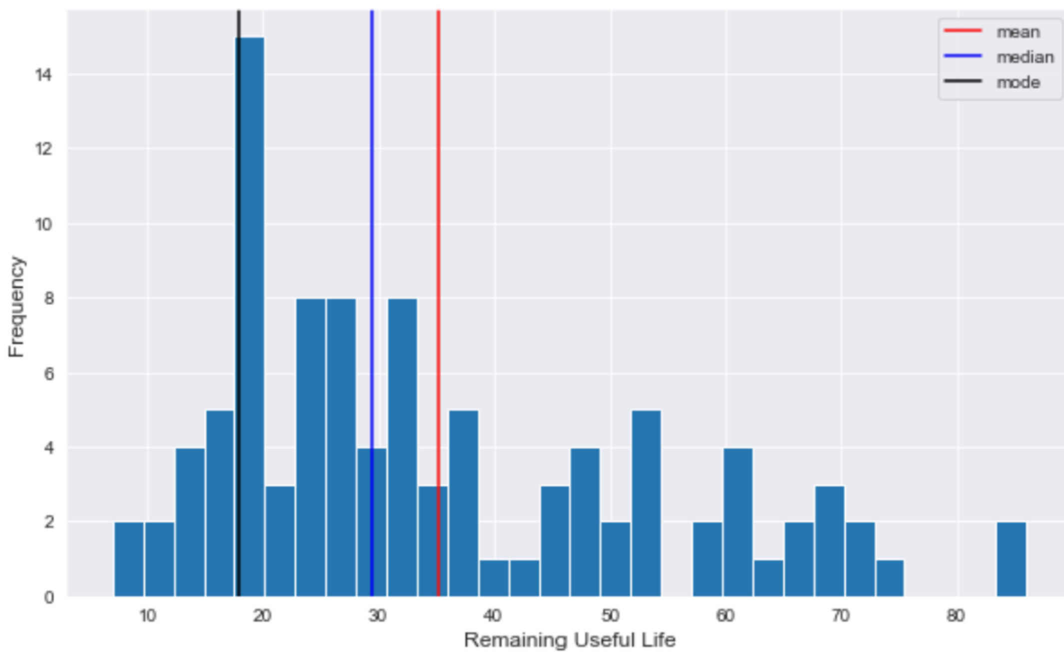


Figure 2. Distribution of the Remaining Useful Life of the 100 Engines Removed from Service by the Sensor Alert model

Deployment of the Sensor Alert could take the form of a rule, or an automated alert to shut the engine down. Gaining the majority of the operating life while avoiding failures with a simple rule is a strong solution in most business situations.

In the fourth scenario, an attempt was made to use linear regression techniques on the sensor data to predict the Remaining Useful Life metric itself. The predicted values would then be used to shut the engines down before failure. Unfortunately, due to the curved nature of the sensor signals with respect to the RUL metric, the modeling was unsuccessful, including an attempt to only model the data close to failure (RUL = 0) with a more linear tangential model. The work however did lead to the concept of setting RUL-based failure threshold values for removing the engines from service, which was used in the two Machine Learning exercises that followed.

The first machine learning exercise leveraged the Decision Tree classifier algorithm with three branches to build a model that could be converted into simple logical if/then statements that could become business rules. The development work was more involved than for the Sensor, but the final model generated a 91% operating efficiency and no engine failures even when tested on fresh data.

The final exercise was an unconstrained approach to ML classification which would require full support from a data scientist for development and deployment as well as the associated IT infrastructure. The approach utilized Principal Component Analysis to blend the sensor data into new vectors for the predictive modeling. The results were impressive with 94% operating efficiency and no failures when tested on the fresh data.

To further explore the relationships between the different alerting systems, a Plotly Dash web application was built. The app visualizes a representative sensor trace as the RUL metric moves to zero. The alerts are overlaid on the chart, and the operating efficiencies for each alert are reported (Figure 3). Both ML models have an Alert threshold metric that can be tuned to further optimize their performances. In most cases, efficiencies of 95+% can be achieved without inducing a failure.

Inputs:

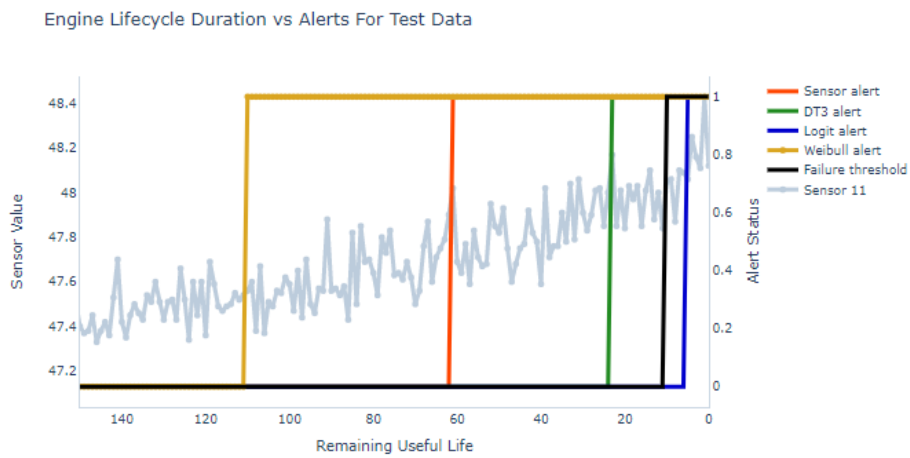
Engine Selector

81

Threshold Selector

0.9

Results:



Operating Efficiency Values:

Weibull Efficiency	54	%
Sensor Efficiency	74.5	%
DT3 Efficiency	90.4	%
Logit Efficiency	97.9	%

Figure 3. The Relationship between Remaining Useful Life, the Sensor Data and the Four Alert Models for One Engine and an Alert Threshold of 0.9

In a real business situation, the next steps would be to review the different options and develop quantitative cost-benefit analyses based on the actual costs of failures versus the actual costs of model development and operationalization and the costs associated with the lost operating cycles. At a minimum, in this scenario, the complex Logit model and the less effective but more straightforward Sensor Alert should be in the mix as solutions to develop and test further with strong potential for operationalization.

SUMMARY AND CONCLUSIONS

The project goal was achieved. Simple modeling techniques for generating predictions that balanced the competing business KPIs were tested along with full Data Science ML algorithms. For the case study that was analysed, there turned out to be room for operationalization of both complex ML and more simple solutions depending on relative value placed on the impact to the business KPIs.

Some additional insights from the work were as follows:

- Progressively more lines of code and data science skills are indeed required to build the ML models versus simple statistical methods
- There is uncertainty with all of the models, simple and complex, and rigor must be applied in each case to fully test and understand the model's sensitivities
- Having or acquiring domain knowledge and an understanding of the business KPIs can help shape and accelerate the development path for a given model. In some cases the business KPIs will drive the direction of development in the opposite direction from what the technical metrics might suggest.
- An unconstrained approach that tries a variety of techniques at the beginning is likely wise rather than forcing a type of solution onto the data

These findings are key lessons to be taken forward into future modeling projects.