Application of Machine Learning to Define Instrumentation

Background

Lack of data, particularly relating to unhealthy or faulty states of an equipment, persists to be a common problem in machine learning for industrial applications. Availability of such data presents a range of opportunities: early detection in future occurrences for operators; better insight on system response for manufacturers; visibility into the effectiveness of the instrumentation layout; a reliable data-driven fault model, etc.

Controlled tests under unhealthy/faulty conditions are used to collect such data and such was the method used by Helwig et al.[1], hereafter referred to as OW (Original Work), on a hydraulic test rig (schematics in Figure 1 below) with a redundant set of instrumentation. UCI Machine Learning Repository (https://archive.ics.uci.edu/) hosts the data under the title "Condition Monitoring of Hydraulic Systems".

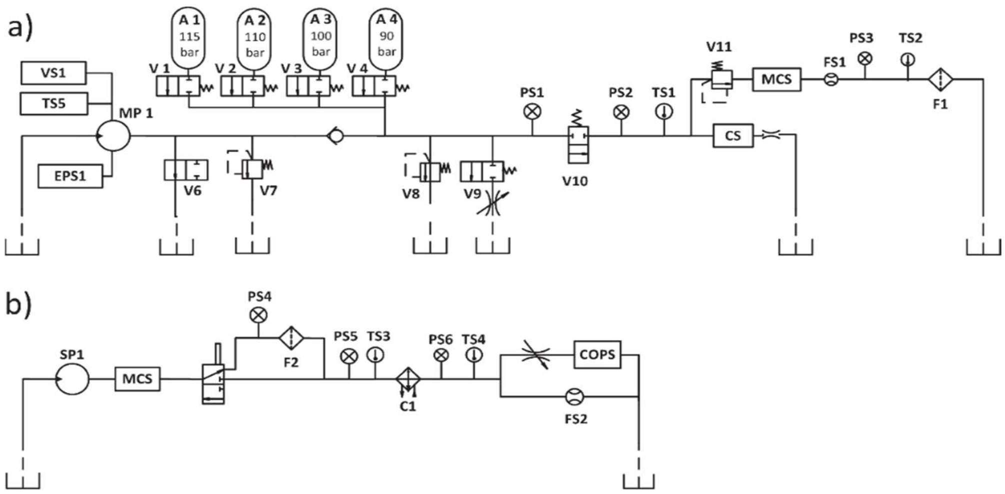

Figure 1. Schematic of hydraulic rig. a) Primary   b) Secondary filtration/cooling circuit.[1]

Components: MP1, SP1: Pumps; A1-A4: Accumulators; C1: Cooler; V11: Load simulating relief valve;V10:Switching valve
Sensors: FSx: volume flow rate; TSx: temperature; PSx: pressure; EPSx: power; VSx: vibration;
Others: CS,MCS,COPS: Oil particle monitoring; Vx: various types of valves; Oil tank

Helwig et al. run the hydraulic rig under cyclic load conditions (Figure 2) while at the same time introducing various levels of faults, in different combinational scenarios, to the four components of the rig, i.e. pump, accumulator, switching valve and cooler. The purpose of the OW was, based on the collected data, to develop and evaluate a classification algorithm that would identify component(s) with faults, and the severity of the fault(s).

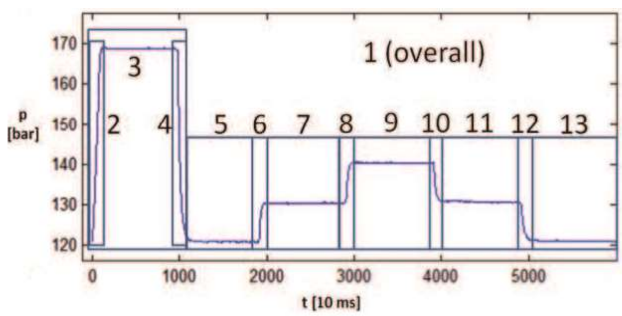

Figure 2. A profile of one cycle of loading.[1]
(Note: 1 cycle: 60 seconds; Total Run: 2202 cycles.)

Table 1. Summary of Faults and Introduction Methods

Component	Fault	Fault Introduction Method	Fault Levels
(C1)	Lower Cooling	Fan duty cycle (100/20/3% duty cycle)	3:20;100 (% Efficiency)
(V10)	Degraded switching	Control of current to V10 (0…100% nominal current)	73;80;90;100 %
(P1)	Leakage	Switchable Bypass orifices(V9) 3 X 0.2 mm; 3 X 0.25 mm orifices	Heavy, weak, no leakage (converted to numerical data)
(A1-A4)	Leakage	Accumulators with different press.	90;100;115;130 Bar

The Objective

This exercise is exploratory. It repurposes the collected data from the OW to search for, and possibly identify instrument configuration/layout that is capable of quantifying /predicting the cooler efficiency with as minimal prediction error as possible.

It is also an evaluation whether instrumentation(s) on the primary circuit could predict cooler efficiency. The OW, it must be noted, introduces faults on multiple components simultaneously, which as a consequence complicates the search for an instrumentation solution because of the cross-fault effect on readings, particularly those of instrumentations on the primary circuit.

From a machine learning perspective, the readings from the instrumentations (and the derived parameters) represent features and hence the feature selection process equates to instrument selection (with location). The exercise extends to finding a prediction model that is capable of quantifying the efficiency of the cooler with minimal error.

The Motivations

Cooler efficiency is computed in the OW, as is commonly done, using an all-temperature instrumentation:

$$= (TS3-TS4) / (TS3-Ta) \quad \text{where Ta=atmospheric temperature}$$

The fact that the volume flow rate, pumping power, pressure, etc. of a viscous fluid (in a circuit) such as the actuating hydraulic oil in this test rig, change as temperature varies, renders the opportunity to use other non-temperature instrumentations to quantify/predict cooler efficiency - and as a consequence the potential to optimize the instrumentation.

A quantified degradation level provides leading information to decide whether and/or when to remove the cooler from service in order to maintain optimal function to the overall system as well as to reduce maintenance cost of the cooler – an implementation of predictive maintenance.

The Approach

The exercise constitutes of three major steps: data extraction, preparation and evaluation; feature selection; regression and final evaluation. All scripting was done using the Python programming language along with the Sklearn data science library.

The initial data extraction required a script as pressure, flow and temperature readings were sampled at 100 Hz, 10 Hz and 1 Hz respectively. Initial review of extracted data showed pressure readings of the secondary circuit having significant jitter that a rolling average was applied to smoothen the data. Unusual load settings/readings were discovered and hence removed at this stage. Plots of temperature readings showed lags in the range of 3-4 seconds from the time of load change. The resulting variations in temperature were in the range of 0.3-0.4-degree c which was considered negligible (versus 30-degree c of oil temperature) for the exercise.

Data at one specific load setting, load schedule 11(randomly selected), was further extracted to preclude the effect of load variation. Scripts filtered the stable points (criteria set by the OW). Coolereff, TS3 and TS4 allowed the backward computation of atmospheric temperature.

In order to gain transparency and better understanding, the features were grouped into three categories as detailed in Table 2 below based on the number of instrumentations. Maximum number of instrumentations is capped to three.

Most derived features were defined such that a feature has a physics/ engineering/statistics-based interpretation; e.g. dP-Cooler *FS2 translates to the work done (rate of) across the cooler. Sklearn's Sequential Feature Selection (SFS) algorithm with the cross validated Ridge regression as an estimator, was used to select candidate features. The SFS, in this exercise, adds one feature at a time and measures the predictive effect of the addition on the aforementioned estimator, using an evaluation metric, the maximum prediction error.

Table 2. Grouping of Features.

Feature Category	List of Features
Single Instrumented (Raw)	PS1, Pw, TS1, TS3, TS4, FS2, Ta, PS5_SMTH, PS6_SMTH
Dual Instrumented	TS4/Ta, FS2/Ta, PS5_SMTH/Ta, dT-Cooler, dP-Cooler, Pw/TS4, Pw/Ta, Pw/FS2, Pw/PS5_SMTH, TS4/dT-Cooler
Triple Instrumented	CoolerEff, dPCooler_SMTH/Ta, dPCooler_SMTH*FS2, TS4*FS2/Ta, PS6_SMTH*FS2/Ta, dT-Cooler/Ta, dT-Cooler * FS2, Pw/dT-Cooler, Pw/(TS4-Ta), Pw/(TS1-Ta), Pw/dP-Cooler

OLS (Ordinary Least Square) regression was finally performed on those selected from each group to finally make a comparative review based on r-squared, mean absolute error (MAE) and the error distribution. OLS regression provides a transparent modelling unlike the black box AI of neural networks.

The data provided from the OW consisted of three cooler health states: 3%, 20%, and 100% efficiency levels. The drawback is that from the operational standpoint these points are extreme points of operation and may not be representative. Computationally, modelling close to operational extremes may not result in a practical solution.

Intermediate Results

An illustration of the search on a grouping is as follows: Given a list of error improvement thresholds i.e. first columns in Tables 3 to 5, the script, for each threshold in the list, uses the SFS (and the ridge regressor) to find features that improve prediction error by, at minimum, the selected threshold. As could be expected, as the threshold is lowered more and more features are selected.

A second run is then performed, on the same list of improvement thresholds, but after removing those selected in the first run. And so on. Tables 3 to 5 summarize the feature selection for the three groupings. As a complement to this approach, all groupings are searched in one go - Table 6 summarizes the results for the two runs.

Table 3. **Results on Single Instrumented Features**

(PS6_SMTH & TS3 removed @ second run; PS6_SMTH, TS1, PS5_SMTH, and TS3 removed @ third run)

Max-error improvement	**Selected Features/Instrumentation (Single)**		
	First Run	**Second Run**	**Third Run**
0.3	PS6_SMTH	PS5_SMTH	TS4
0.2	PS6_SMTH	PS5_SMTH	TS4
0.1	PS6_SMTH, TS3	PS5_SMTH, TS1	TS4
0.08	PS6_SMTH, TS3	PS5_SMTH, TS1	TS4
0.06	PS6_SMTH, TS3	PS5_SMTH, TS1	TS4
0.03	PS6_SMTH, TS3	PS5_SMTH, TS1, TS4	TS4
0.01	PS6_SMTH, TS3, TS4	PS5_SMTH, TS1, TS4	TS4

Table 4. **Results on Dual Instrumented Features**

(Pw/TS4 removed @ second run; Pw/TS4 and dT-Cooler removed for third run.)

Max-error improvement	**Selected Features/Instrumentation (Dual)**		
	First Run	**Second Run**	**Third Run**
0.3	Pw/TS4	dT- Cooler	TS4/Ta
0.2	Pw/TS4	dT- Cooler	TS4/Ta
0.1	Pw/TS4, FS2/Ta	dT- Cooler	TS4/Ta
0.08	Pw/TS4, FS2/Ta	dT- Cooler	TS4/Ta
0.06	Pw/TS4, FS2/Ta	dT- Cooler	TS4/Ta
0.03	NA	NA	TS4/Ta

Table 5. **Results on Triple Instrumented Features**

(Second run with Pw/(TS4-Ta) removed; Third Run with Pw/(TS4-Ta) & CoolerEff removed; etc.)

Max-error Improv-ement	**Selected Features/Instrumentation (Triple)**			
	First Run	**Second Run**	**Third Run**	**Fourth Run**
0.3	Pw/(TS4-Ta)	CoolerEff	Pw/(TS1-Ta)	TS4*FS2/Ta
0.2	Pw/(TS4-Ta)	CoolerEff	Pw/(TS1-Ta)	TS4*FS2/Ta
0.1	Pw/(TS4-Ta)	CoolerEff	Pw/(TS1-Ta)	TS4*FS2/Ta, Pw/PS6*FS2
0.08	Pw/(TS4-Ta)	CoolerEff	Pw/(TS1-Ta)	TS4*FS2/Ta, Pw/PS6*FS2
0.06	Pw/(TS4-Ta)	CoolerEff	Pw/(TS1-Ta)	TS4*FS2/Ta, Pw/PS6*FS2

Table 6. **Results on All Features**

(Pw/(TS4-Ta) removed @ second run)

Max-error Improvement	**Selected Features/All Groupings**	
	First Run	**Second Run**
0.3	Pw/(TS4-Ta)	CoolerEff
0.2	Pw/(TS4-Ta)	CoolerEff
0.1	Pw/(TS4-Ta)	CoolerEff, TS4/dT-Cooler
0.08	Pw/(TS4-Ta)	CoolerEff, TS4/dT-Cooler
0.06	Pw/(TS4-Ta)	CoolerEff, TS4/dT-Cooler
0.03	Pw/(TS4-Ta)	CoolerEff, TS4/dT-Cooler
0.01	Pw/(TS4-Ta), Pw	CoolerEff, TS4/dT-Cooler, PS1 * Ta/PS6SMTH

Comparison

Figures 3,4 and 5 provide self-explanatory visualizations for the prediction metrics for the OLS regression on the selected features.

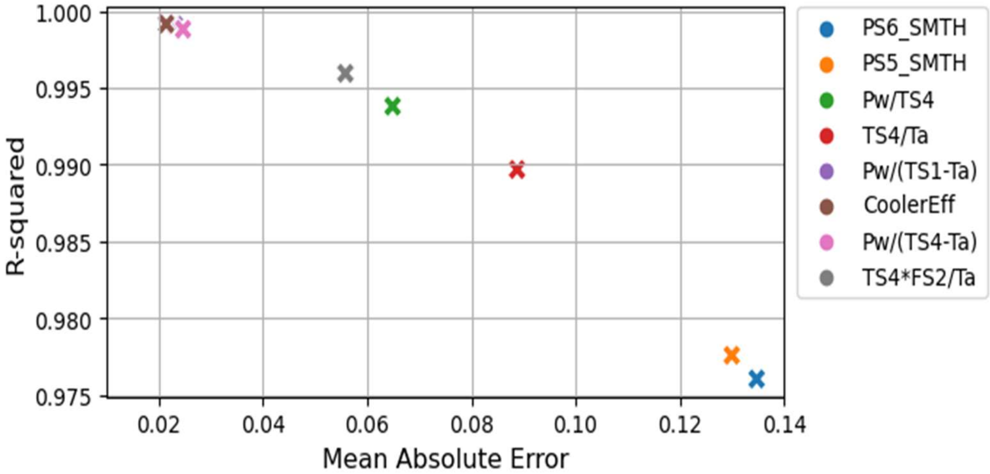

Figure 3. R-Squared vs MAE

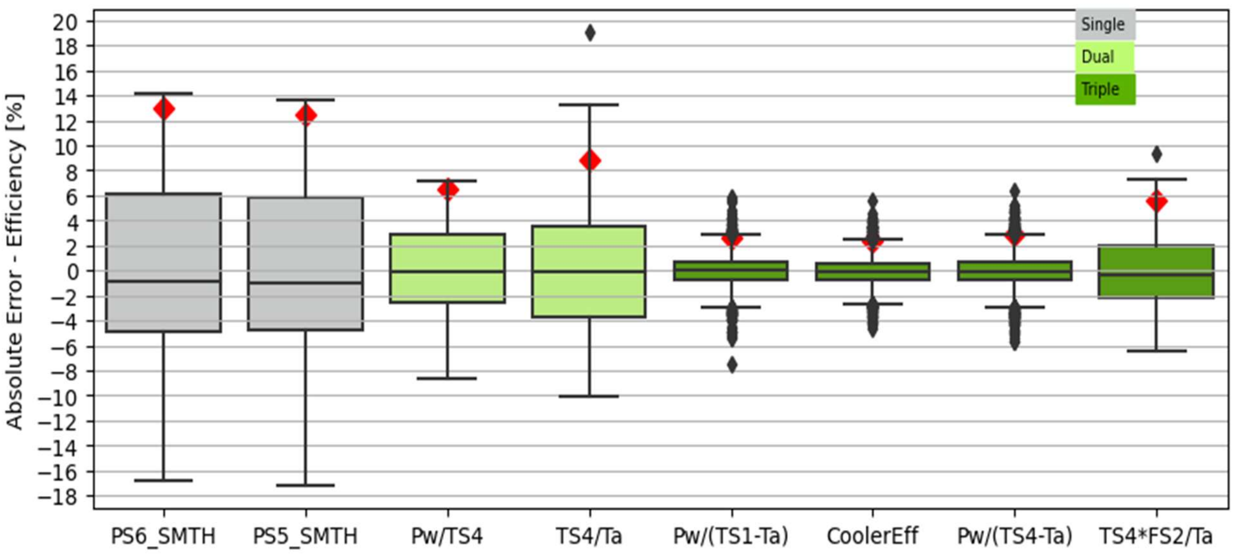

Figure 4. Distribution of Absolute Error
(Note: Red diamonds represent prediction intervals.)

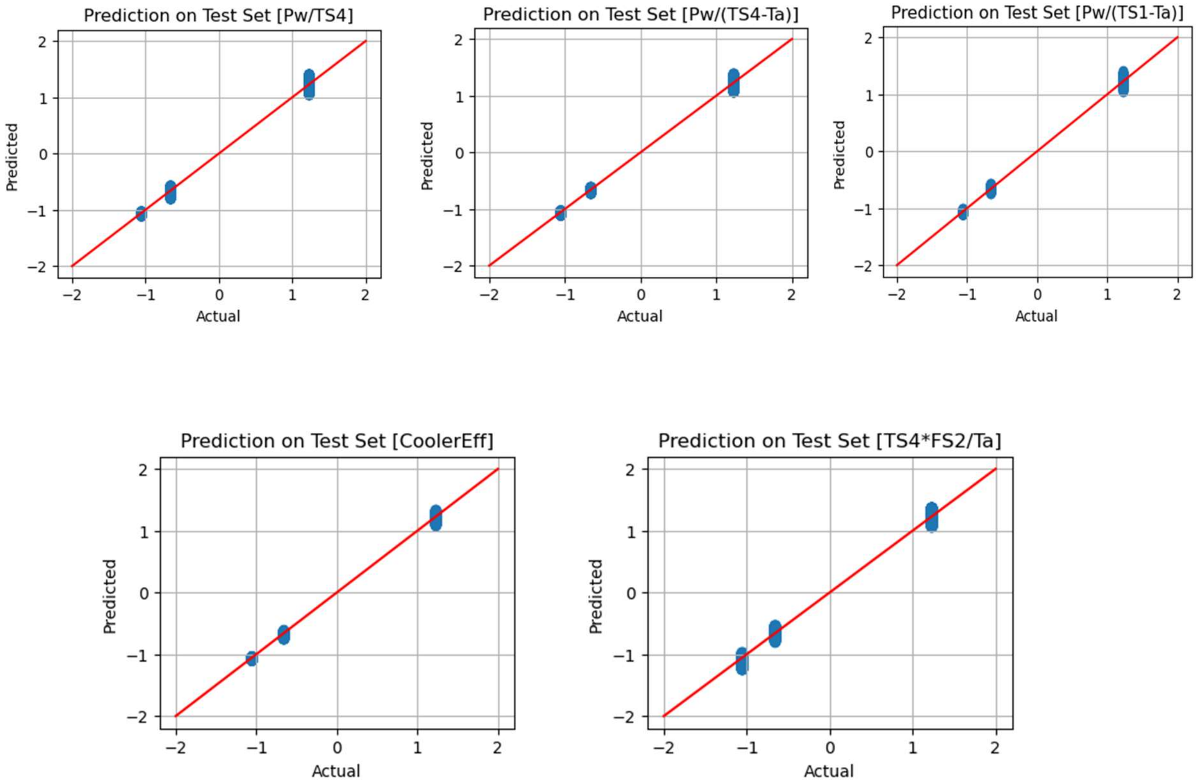

Figure 5: Predicted vs Actual Efficiency (Scaled units) for select features
(Note: The reference red line is line of equality.)

Final Observations

1. Temperature sensors dominate the selected features, however, Power (Pw) instrumentation, as a constituent of the features Pw/(TS1-Ta) and Pw/(TS4-Ta), has influential contribution in predicting cooler efficiency – while installed on the primary circuit, under the effect of cross-faults. None of the other non-temperature readings, i.e. pressure and flow, appear as influential as Pw does in the prediction.

2. Triple-instrumented features, CoolerEff, Pw/(TS4-Ta), Pw/(TS1-Ta) are better predicting features based on Figures 3, 4 and 5 than the dual instrumentation. This should not be misunderstood to mean a dual instrumentation will not predict better. The knowledge gained on the effect of Pw on the prediction provides reasonable ground to test further; a Pw instrumentation on the secondary circuit merits an evaluation.

3. Pw/(TS1-Ta) is a feature all of whose constituent sensors are on the primary circuit with a comparable predictive performance. It is a demonstration of robustness while under the effects of cross faults.

4. The first four plots of Figure 5 show the magnitude of error increasing as the actual efficiency increases, particularly at the 100% efficiency. Could the error reduce at less efficiency points? More data at intermediate points would have helped.

Reference

[1] Nikolai Helwig, Eliseo Pignanelli, Andreas Schütze, "Condition Monitoring of a Complex Hydraulic System Using Multivariate Statistics", in Proc. I2MTC-2015 - 2015 IEEE International Instrumentation and Measurement Technology Conference, paper PPS1-39, Pisa, Italy, May 11-14, 2015, doi: 10.1109/I2MTC.2015.7151267