PREDICTIVE MAINTENANCE FOR HYDRAULIC SYSTEMS

IVAN CHEREDNIKOV

EXECUTIVE SUMMARY

Through extensive research around predictive maintenance, it is clear that there needs to be a balance between competence and technology in companies. Implementing advanced and complicated technology and systems is of no use if the competence of the workers lacks behind. Companies may benefit from predictive maintenance of their equipment, but it is important to perform maintenance and repairs on the equipment when it is actually needed, to keep downtime of the production to a minimum and to not get into the over-maintenance zone, which can result in the company losing profit.

With the constant technological advance of the 21st century, it is obvious that companies who lack behind lose potential profit, but as the tools and systems get more advanced, the workers need to get more competent in using these.

Predictive maintenance is an important tool to keep downtime and loss of profit to a minimum for a business. This report investigates a hydraulic systems dataset and the importance of predictive maintenance in a hydraulic system to see how PdM could be used to benefit a company. Machine learning is a powerful tool that is used in the report to assist the research and to build a robust ML model, capable of predicting when a system needs maintenance, before it fails.

The overall goal in this project is to conduct research showcasing that predictive maintenance of a company's hydraulic systems is beneficial and more profitable than buying new systems, while also avoiding going into the over-maintenance territory. A company that knows exactly when their tools and machines need to be repaired will logically gain more net profit that a company that performs repairs when their system shuts down, resulting in a halt in production.

Assisting in reaching this goal is a dataset provided by UC Irvine machine learning repository consisting of various data gathered from testing of hydraulic systems. With the use of machine learning and several python libraries, the group expects to find an optimal solution to the problem by developing a predictive data model.

To achieve this, the following objectives and activities as been set:

Research

- Perform a theoretical review of key concepts in order to provide a theoretical framework for understanding the project
- Identify state-of-art dataset through a technical review

Development predictive data Model

- Data investigation
- Suggest an overall system architecture for the system
- Development of a predictive data Model for hydraulics systems

Evaluation and results of data model

- Suggest a "evaluation pipeline" based on the identified dataset
- Perform testing of model
- Provide summary

The results of this project …………………………………………

Based on key findings in the investigation and analysis of predictive maintenance and a hydraulic dataset, the group recommends every business to investigate predictive maintenance and consider investing into it, if possible, because of the huge benefits with few drawbacks. This is especially important for the hydraulic industry, where uptime and optimal utilization of available resources is crucial, minimizing LCC.

If the group could have more quantitative data and an actual, real-life case, with hydaulic data not obtained using a test rig, more work could be done. As certain points in the dataset could only be referred to as cycles, without actual timestamps, it is not possible to compare the results obtained with the model to a real-life hydraulic rig being used in a business. Without having access to costs of maintenance, equipment parts and other cost-related data, such as budgets and accounting information, deeper analysis to gain further insight is not possible.

Model improvement could also be done to generalize the model and streamline the machine learning operations. First things to improve and add to the code would be:

- Functions to quickly train the models, instead of having to go through lengthy code
- Code/Debug testing.
- Functions or even simple “If tests” to detect when and what part needs to be repaired, alerting and informing the user.
- Interactive UI/Streamlit page to quickly import data, preprocess, visualize it and gain insight. Moreover, the model could be trained and adjusted quickly, giving out formatted predictions.
- Implementation of deep neural networks and TensorFlow

There are many areas of improvement and more work to be done, but the dataset and lack of cost data are the main limitations and the points mentioned above are to be focused on first.

CONTENT

INTRODUCTION

BACKGROUND

Through extensive research around predictive maintenance, it is clear that there needs to be a balance between competence and technology in companies. Implementing advanced and complicated technology and systems is of no use if the competence of the workers lacks behind. Companies may benefit from predictive maintenance of their equipment, but it is important to perform maintenance and repairs on the equipment when it is actually needed, to keep downtime of the production to a minimum and to not get into the over-maintenance zone, which can result in the company losing profit.

With the constant technological advance of the 21st century, it is obvious that companies who lack behind lose potential profit, but as the tools and systems get more advanced, the workers need to get more competent in using these.

PROBLEM STATEMENT

How can companies predict the need for maintenance of their hydraulic systems through use of data science and analysis with regards to profit and usefulness?

GOALS AND OBJECTIVES

The overall goal is to conduct research showcasing that predictive maintenance of a company's hydraulic systems is beneficial and more profitable than buying new systems, while also avoiding going into the over-maintenance territory. A company that knows exactly when their tools and machines need to be repaired will logically gain more net profit that a company that performs repairs when their system shuts down, resulting in a halt in production.

Assisting in reaching this goal is a dataset provided by UC Irvine machine learning repository consisting of various data gathered from testing of hydraulic systems. With the use of machine learning and several python libraries, the group expects to find an optimal solution to the problem by developing a predictive data model.

To achieve this, the following objectives and activities as been set:

Research

- Perform a theoretical review of key concepts in order to provide a theoretical framework for understanding the project
    - What are hydraulic systems?
    - What is predictive maintenance?
- Identify state-of-art dataset through a technical review
- Provide summary
    - Added business value

Development predictive data Model

- Data investigation
- Suggest an overall system architecture for the system
- Development of a predictive data Model for hydraulics systems
- Provide summary

Evaluation and results of data model

- Suggest a "evaluation pipeline" based on the identified dataset
- Perform testing of model
- Provide summary

Recommendations and further work

LIMITATIONS

The group consists of amateur data science student, with limited knowledge of machine learning. Things like deep learning and the TensorFlow library are complicated, and the group will try to use these, but the understanding of these will be limited as the group's knowledge around the subject is scarce.

The dataset contains test rig data, done in set cycles, opposing to data gathered from a real hydraulic system at a factory, gathered from a system until it got written off. This sets hard limitations for what could be predicted with the data.

THEORY AND KEY CONCEPTS

HYDRAULICS SYSTEM

The test rig used to collect the date consists of a primary and a secondary cooling-filtration circuit, all connected via the oil tank.

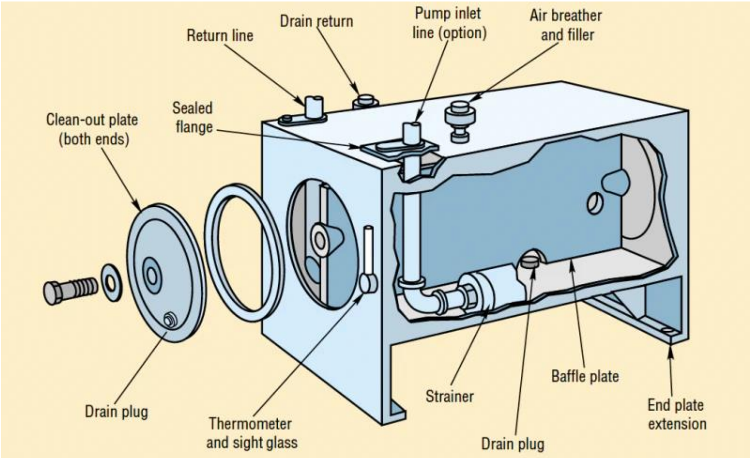


Figure 1: A typical rectangular fluid reservoir.

The figure above illustrates the central parts of a rectangular fluid reservoir. Although the properties of the test rig used to gather the data are unknown, this is a typical reservoir used in hydraulic systems. This provides a hydraulic system with a lot of fluid for a system's needs, as well as giving the workers a surface to mount other component or sensors on. [1]

Pressure sensors are used to convert pressure information into electric signals, these are usually resistant to vibration and shock and widely used in hydraulic systems[2].

Electronic temperature sensors process and convert the recorded temperature into electric signals, these are usually made of two metals which generate current when the temperature changes[3].

Volume flow sensors measure the volume of fluid which passes per a given time. There are different types of these, but the most common ones are variable office, gear type and turbine-type flow meters. A variable office flow works through displacement of a piston or a ring which is then measured. A gear type meter records the displacement, rotation of a gear while the fluid passes outside a pair of gears to generate a pulse which then measured. The rotation of the gears is proportional to the flow rate. In a turbine flow meter, fluid passes through a flow meter mounted with a turbine motor on a shaft. The frequency recorded here is proportional to the flow.[4]

Vibration sensors. There are many different types of vibration sensors, ranging from accelerometer sensor to microphone and laser sensors. These generally utilize the piezoelectric effect, the ability of some materials to

[1] https://www.hydraulicspneumatics.com/technologies/reservoirs-accessories/article/21882642/fundamentals-of-hydraulic-reservoirs
[2] https://www.variohm.com/news-media/technical-blog-archive/hydraulic-pressure-sensors
[3] https://www.hydac.com/de-en/products/sensors/temperature-sensors.html
[4] https://www.fluidpowerworld.com/need-know-hydraulic-flow-meters/

generate an electric charge in response to mechanical pressure applied to them[5]. These are also used for finding certain scents in the air, measuring air quality. [6]

The rest of the sensors used in the hydraulic system to collect the data provided in the dataset work in a similar manner, converting recorded values into electric signals.

PREDICTIVE MAINTENANCE

Proactive maintenance aims toward correcting the source of hardware conditions. Here, the goal is to maximally reduce downtime and any potential failure in the instruments, preventing it from losing function. A typical example of proactive maintenance is PM, preventive maintenance. This could be maintenance performed depending on how much the equipment is used, or it could be date-based, where maintenance is done calendar based.

Predictive maintenance monitors the performance and condition of equipment while it operates normally to reduce the failure ratio. It's been used since the 90s and it can only exist with the existence of condition monitoring which is defined as used. The result of predicted maintenance is prevention of failure through corrective maintenance forecast by continuous monitoring of the equipment used. Actual predictive maintenance takes many steps, and a company needs to go through these to establish a well working predictive maintenance programme. The steps include everything from analysing the history of the equipment and tools used, potential reviewing of available records on the equipment records of the downtime and potential regulations on the equipment. A predictive maintenance system is costly and need to be implemented right for it to be worth it for the company. Starting with simple tasks like analysis of fluids passing through, vibration analysis or analysis of infrared-sensor data are methods that are easy to implement and give quick results for the company. A business needs to consider if the investment in technology for predictive maintenance is worth it, valuing their equipment and the actual necessity of it accordingly. As mentioned above, the history of the equipment needs to be analysed, seeing if the downtime and previous defects need to be addressed to benefit the company.

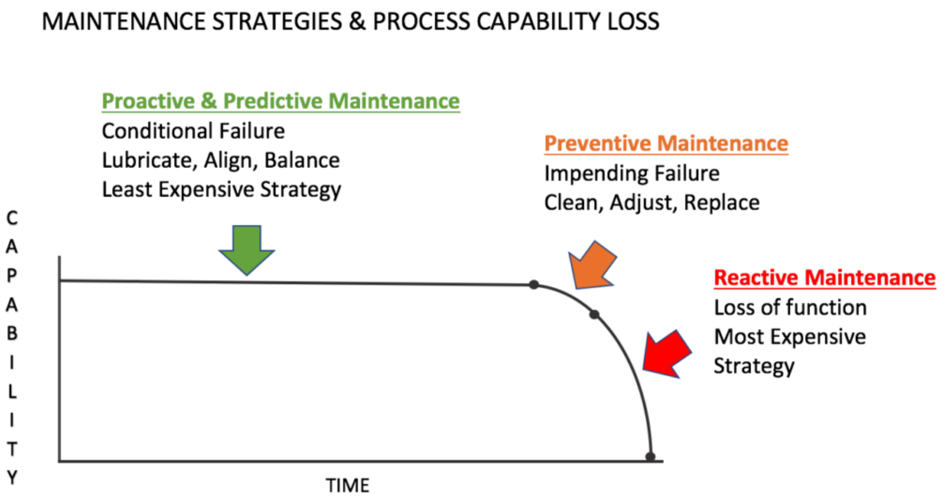


Figure 2: Proactive & predictive, preventive and reactive maintenance curve.

[5] https://www.elprocus.com/what-is-the-piezoelectric-effect-working-and-its-applications/
[6] https://www.elprocus.com/vibration-sensor-working-and-applications/

Integration of predictive maintenance into an already existing scheme of maintenance in a company will generally improve the availability of the said system. While the integration of a PdM system can be doable, the challenges arise with the need of creation and maintenance of the system itself. A company's predictive maintenance system can fail in multiple ways, these include but are not limited to:

- The PdM plan works, but the workers lack competence in the area and fail to keep themselves updated with the advancing of the system. There is a lack in continuity in the overall work.
- The company lacks an analysis and report system or there are deficiencies in the data analysis structure. This could be for both internal and external facilities.
- A lack of documentation of the PdM system and the different programs used leading to a lack in competence in workers trying to understand and use it.

DATASET INFORMATION

Data Basis - condition monitoring of hydraulic systems

The dataset "Condition Monitoring of Hydraulic Systems" - used for training machine learning models in the project is gathered from the UC Irvine machine learning repository. Here is the description of the dataset pasted straight from the repository:

"The dataset was experimentally obtained with a hydraulic test rig. This test rig consists of a primary working and a secondary cooling-filtration circuit which are connected via the oil tank. The system cyclically repeats constant load cycles (duration 60 seconds) and measures process values such as pressures, volume flows and temperatures while the condition of four hydraulic components (cooler, valve, pump and accumulator) is quantitatively varied.

The dataset contains raw process sensor data (i.e. without feature extraction) which are structured as matrices (tab-delimited) with the rows representing the cycles and the columns the data points within a cycle. The sensors involved are:

Sensor Physical quantity Unit Sampling rate

- PS1 Pressure bar 100 Hz
- PS2 Pressure bar 100 Hz
- PS3 Pressure bar 100 Hz
- PS4 Pressure bar 100 Hz
- PS5 Pressure bar 100 Hz
- PS6 Pressure bar 100 Hz
- EPS1 Motor power W 100 Hz
- FS1 Volume flow l/min 10 Hz
- FS2 Volume flow l/min 10 Hz
- TS1 Temperature Â°C 1 Hz
- TS2 Temperature Â°C 1 Hz
- TS3 Temperature Â°C 1 Hz
- TS4 Temperature Â°C 1 Hz
- VS1 Vibration mm/s 1 Hz
- CE Cooling efficiency (virtual) % 1 Hz
- CP Cooling power (virtual) kW 1 Hz
- SE Efficiency factor % 1 Hz

Attribute Information:

Attributes are sensor data (all numeric and continuous) from measurements taken at the same point in time, respectively, of a hydraulic test rig's working cycle.

Total number of Attributes: 43680 (8x60 (1 Hz) + 2x600 (10 Hz) + 7x6000 (100 Hz))

The sensors were read with different sampling rates, leading to different numbers of attributes per sensor despite they were all exposed to the same working cycle.

1. Pressure sensors (PS1-6): 100 Hz, 6000 attributes per sensor (6 sensors)
2. Motor power sensor (EPS1): 100 Hz, 6000 attributes per sensor (1 sensor)
3. Volume flow sensors (FS1/2): 10 Hz, 600 attributes per sensor (2 sensors)
4. Temperature sensors (TS1-4): 1 Hz, 60 attributes per sensor (4 sensors)
5. Vibration sensor (VS1): 1 Hz, 60 attributes per sensor (1 sensor)
6. Efficiency factor (SE): 1 Hz, 60 attributes per sensor (1 sensor)
7. Virtual cooling efficiency sensor (CE): 1 Hz, 60 attributes per sensor (1 sensor)
8. Virtual cooling power sensor (CP): 1 Hz, 60 attributes per sensor (1 sensor)

Class distribution:

The target condition values are cycle-wise annotated in profile.txt (tab-delimited). As before, the row number represents the cycle number. There are 5 different class value vectors provided in profile.txt. All but number 5 (stable flag) describe degradation processes over time and, thus, their values do not represent distinct categories, but continuous values.

The columns are:

1: Cooler condition / %:

- 3: close to total failure
- 20: reduced efficiency
- 100: full efficiency

2: Valve condition / %:

- 100: optimal switching behavior
- 90: small lag
- 80: severe lag
- 73: close to total failure

3: Internal pump leakage:

- 0: no leakage
- 1: weak leakage
- 2: severe leakage

- 4: Hydraulic accumulator / bar:
- 130: optimal pressure
- 115: slightly reduced pressure
- 100: severely reduced pressure
- 90: close to total failure

5: stable flag:

- 0: conditions were stable
- 1: static conditions might not have been reached yet"

The data is presented in multiple text files in the form of time series data, with a "profile.txt" files containing the target values.

COST AND BENEFITS

The four primary costs coming with predictive maintenance are: sensors, software, hardware and software installation and an expert in the field. The cost of sensors depends, ranging from hundreds of dollars to many thousands, with the case data being collected using many different sensors, gathering pressure, motor power, volume flow and other information. PdM needs proper software and hardware, having a PdM and a data analytics program is crucial to process information from sensors. This, together with the need of a salaried maintenance worker or engineer will cost an enterprise anything from tens of thousands of dollars to hundreds of thousands of dollars.[7] In some cases, a dedicated PdM worker might not be needed, depending on the scale of the maintenance needed and worker's competence with the system.

For some business, like manufacturing, power plants or waste management, PdM is crucial. For a relatively large company, an hour of downtime is worth around 250000$. As an average round of downtime lasts about 4 hours, it yields a million-dollar loss. According to a research paper written by the United States Department of Energy (DOE)[8], depending on the company's reliance on proper maintenance, "it could easily recognize savings opportunities exceeding 30% to 40%. ", along with providing the business with other benefits. These are: a return on the investment into the system being 10 times the initial cost, a maintenance cost reduction of 27,5%, an elimination of breakdowns of 72,5%, a downtime reduction of 40% and a production increase of around 22,5%. If a company can afford the initial cost, it would be worth it in the long run.

[7] https://coastapp.com/blog/predictive-maintenance/
[8] https://www1.eere.energy.gov/femp/pdfs/OM_5.pdf

DESIGN DESCRIPTION

The goal of the predictive algorithms presented by the group is to make the classification as effective as possible, with the least computational power. The natural complexity of hydraulic systems and industrial system makes it complicated to predict faults, however the model presented by the group performs very well on the given dataset. A major limitation in a real-life system of this sort would be the handling of incorrect sensor readings. A simple solution for some of the incorrect readings would be a function that removes outliers or values that are not possible in the system the sensor is collecting data from. Although because of the nature of AdaBoost and our forest-based model, the model wouldn't be altered in the same way a linear model would be because of incorrect-readings of extreme values, although these are still important to remove. However, an extensive Fault Detection and Diagnosis (FDD)[9] process would have to be performed to find incorrect readings that are among the regular collected data.

DATA PREPROCESSING

Work on the dataset used was performed using Jupyter Notebook and Python 3.8.5. Since the data is split into 17 .txt files, expectations are that some preprocessing and data cleaning was to be done. Note: n_jobs = -1 was used for most of the models, ensemble classifiers and hyperparameter tuning tools. Setting this to a value of -1 consumes all available resources of the machine one is working with, using all processors available. This speeds up the runtime of the models, but could be adjusted depending on if a company has a computer dedicated to working with the models, or if they only want to use part of the available machine resources and power for the task. Working with very large datasets also requires a lot of Random Access Memory (RAM), so that should be noted as well. An i9 processor was used to run the jupyter notebook file, build and adjust the models and process the data.

The target conditions are split into multiple columns and values, which made it necessary and logical to analyze the file and select the best target value to be used together with the subsets later. A list with the column names was created, and a pandas dataframe was created with the target values from the profile.txt file. The isna() function was used to check the dataframe for any missing values and no such values were found in the file. Simple histograms were plotted to see the distribution of the target values. Seaborn displot() represented the cooler condition values better than a classic histogram and was thus chosen for that column. The target showed to have good spread among the available target values and would work well for training a machine learning model. The histograms and profile.txt description file showed that this was a classification problem, with distinct classes representing hydraulic conditions.

---

9 https://www.researchgate.net/publication/343497868_A_Hybrid_Approach_Dynamic_Diagnostic_Rules_for_Sensor_Systems_in_Industry_40_Generated_by_Online_Hyperparameter_Tuned_Random_Forest

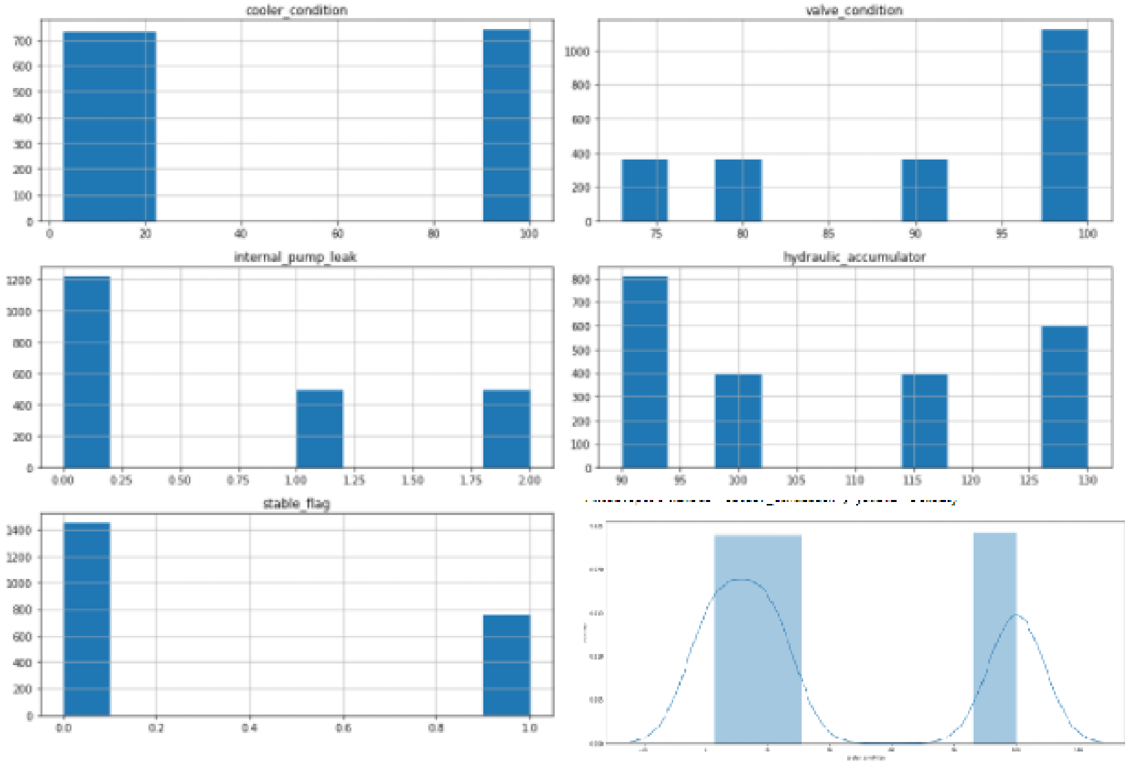


Figure 3: histograms and distplot for target values.

The hydraulic test data was to be read in. The 17 files together totalled 2205 rows and a whopping 43680 columns, which meant a lot of features to work with. Working with this data directly would most likely lead to overfitting and a model like that would show poor performance on test data. There are different ways to handle multidimensional data like this with a lot of features. Feature extraction could be used here and would work well since we are presented with time series data. However, it is computationally expensive and since one would expect operations like this to be done on a regular basis in a company, going for simpler data preprocessing methods is most likely the reasonable choice here. Another method could be partioning the data and picking out samples to be split into validation sets. However, using Pandas .describe(), we see that the standard deviation for all the datafiles is low, and the difference between the min and max values is not significant. Thus, using mean values is reasonable and .mean() was used on all the files to form a single column with 2205 rows from each file. Therefore, after meaning the data and using pandas .concat(), the group was able to compact the data to a dataframe with 2205 rows and 17 columns, presenting 17 features to work with.

```
df_full.describe()
```

	PS1	PS2	PS3	PS4	PS5
count	2205.000000	2205.000000	2205.000000	2205.000000	2205.000000
mean	160.485315	109.379906	1.753227	2.600266	9.163320
std	4.699425	4.986585	0.251902	4.279355	0.576296
min	155.391547	104.406307	0.840252	0.000000	8.365800
25%	158.100195	106.962382	1.729733	0.000000	8.547239
50%	158.960895	107.730169	1.779631	0.000000	9.115781
75%	161.000735	109.421612	1.932047	3.503266	9.844351
max	180.922708	131.589089	2.023398	10.207068	9.978510

Figure 4: Part of the descriptive statistics provided by pandas.describe().

2205 rows × 17 columns

Figure 5: Dataset size after reduction and preprocessing

Plotting of the data showed a fair bit of outliers in the pressure data, as well as the motor power, vibration and efficiency factor data. Since the data was meaned and the standard deviation was low for most of the features, the outlier data would not be problematic in the training of the models. However, the efficiency factor column (SE) had a fair bit of extreme outlier. For the bigger picture, these were ignored for now. If the models would underfit, overfit or otherwise show poor performance, the data in this feature would be worked with, and outliers would be removed/reduced. However, doing minimal preprocessing and data manipulation on the presented data and getting optimal results if of importance for the group, as this would show the ease of streamlining the predictive maintenance workflow for future, real hydraulic data with a low amount of computational power needed.

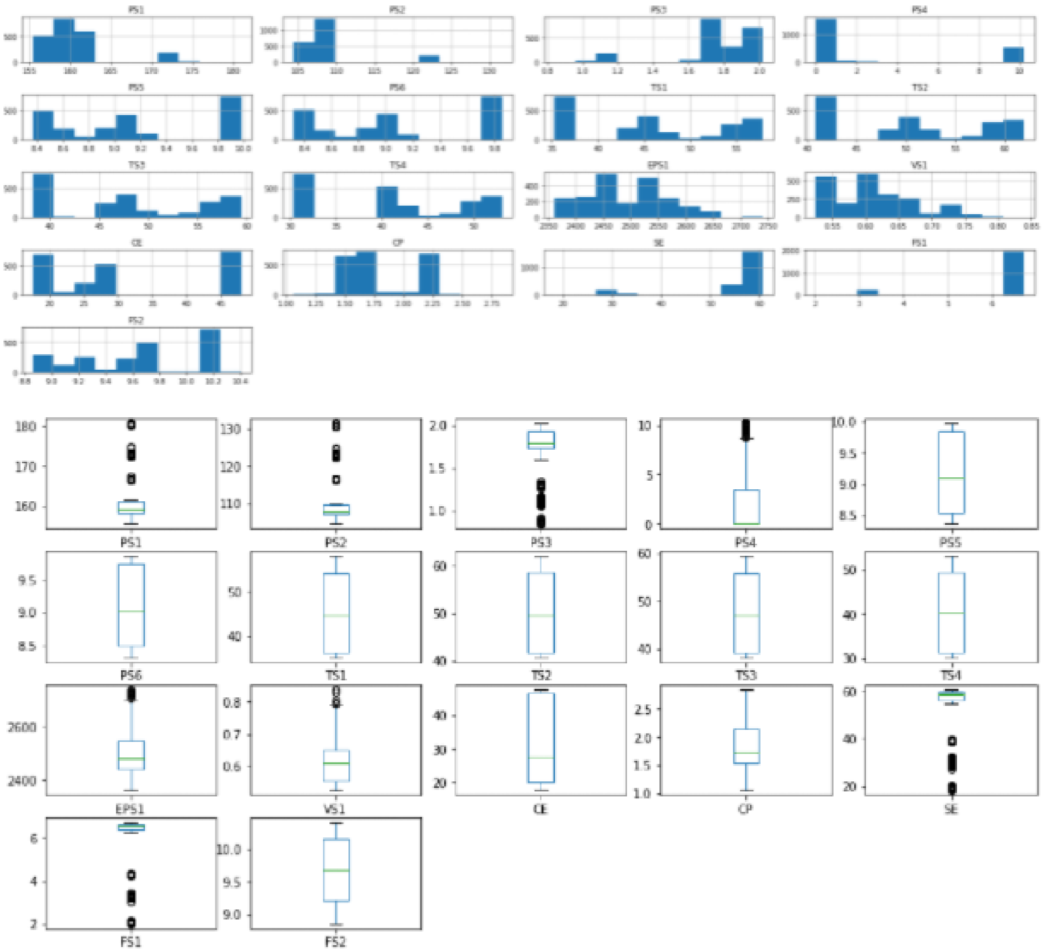


Figure 6: Simple histograms and boxplots on the data

The group proceeded with making a correlation matrix with a heatmap to see how the different feature are correlated. This could be very useful if the dataset had missing values which could be then imputed, or while replacing the outlier. The relationship in this case is obvious, as our data comes from a test rig with timed, constant load cycles with target conditions being varied. However, the correlation matrix also gives us insight in whether we could use some of the model techniques needed for a robust model. Some strong negative correlation is observed for some of the features, which could lead to multicollinearity. This could impact some models negatively and give skewed and biased results. However, tree-based models and boosting algorithms are not affected by this, which is why it is a good idea to use such models for this data. If a company would require the use of models that suffer from multicollinearity, dimension reduction using LDA, PCA or others could be used.

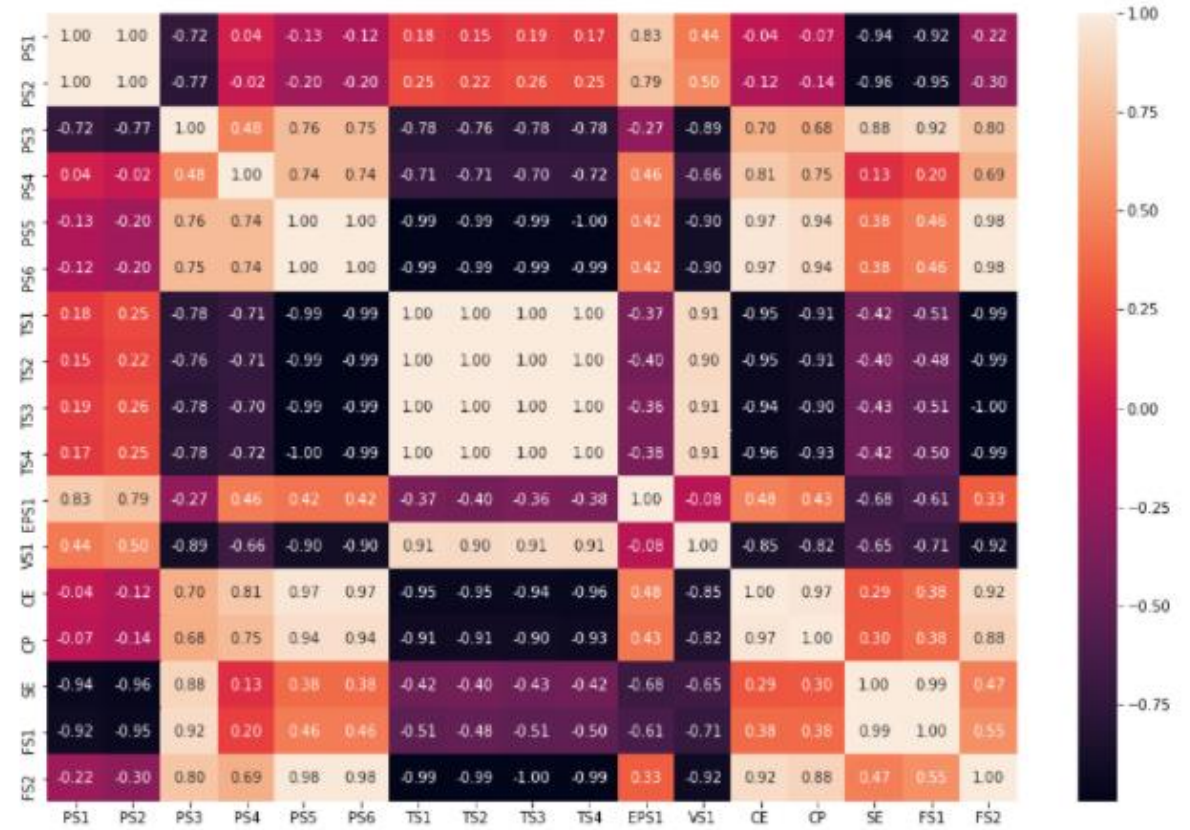


Figure 7: correlation matrix with heatmap

With the data ready for fitting and prediction, different classification models were put together and tested on the data. Valve condition was chosen as the target (y) value because of the even distribution, as seen through the histograms. A train_test_split was then used to split the data into a training and testing test, with the testing data being 20% of the total dataset.

```
X = df_full
y = target.valve_condition

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
                                                    random_state=1, stratify=y)
```

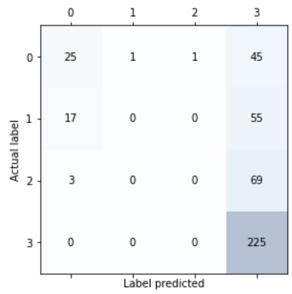
Figure 8: Splitting into training and testing data

Support Vector Machine's SVC was tested and gave poor results with default values, most likely because of correlation problems and multicollinearity. A tweaked version of this model with regularization and kernel functions could be effective if it was fit on the high dimensional space data which was the case with the dataset before feature reduction and preprocessing. A simple logistic regression model was tested on the data, using a pipeline with a standardscaler to scale the data and PCA for dimensionality reduction. The model performed poorly, suffering from the same problems as SVC. A confusion matrix and classification report for both models showed that they mislabel a lot and half the labels had no correctly predicted samples.

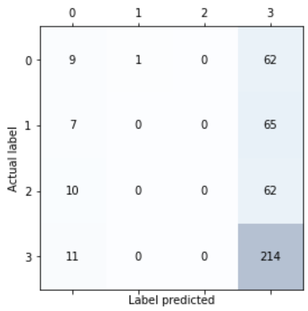
```
Test Accuracy: 0.567
```

```
Test Accuracy: 0.506
```

Figure 9: SVC and LogisticRegression test accuracies



	precision	recall	f1-score	support
close to total failure	0.56	0.35	0.43	72
severe lag	0.00	0.00	0.00	72
small lag	0.00	0.00	0.00	72
optimal switching behavior	0.57	1.00	0.73	225
accuracy			0.57	441
macro avg	0.28	0.34	0.29	441
weighted avg	0.38	0.57	0.44	441



	precision	recall	f1-score	support
close to total failure	0.24	0.12	0.17	72
severe lag	0.00	0.00	0.00	72
small lag	0.00	0.00	0.00	72
optimal switching behavior	0.53	0.95	0.68	225
accuracy			0.51	441
macro avg	0.19	0.27	0.21	441
weighted avg	0.31	0.51	0.37	441

Figure 10: Confusion matrices and classification reports for SVC and LogisticRegression models, both suffering from problems with multicollinearity

A pipeline with a simple random forest classifier with default values performed very well on the data. A test accuracy of 98,4% was achieved, without touching hyperparameters. A confusion matrix and a classification report show that the few mislabeled samples were for labels 1 and 2, which in this case were “small lag” and “severe lag”, with a 100% precision for the “optimal switching behaviour” and “close to total failure” labels. A RandomizedSearchCV with triple cross validation was performed for find the best potential hyperparameter values for the random forest model. The search was made to not be too extensive, as to once again save on computational power and time. After fitting a random forest model with the tuned hyperparameters, the model got slightly more accurate, going from 98,4 to 98,6%.

```
Test Accuracy: 0.984
```

Figure: Test Accuracy with simple randomforest model

```
Fitting 3 folds for each of 80 candidates, totalling 240 fits
[Parallel(n_jobs=-1)]: Done  10 tasks      | elapsed:    6.1s
[Parallel(n_jobs=-1)]: Done 160 tasks      | elapsed:   57.1s
[Parallel(n_jobs=-1)]: Done 240 out of 240 | elapsed:  1.3min finished
```

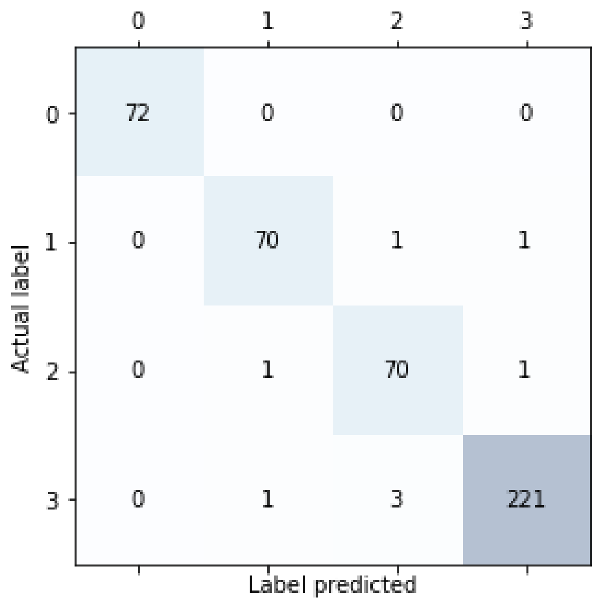
```
print(randomRFC.best_score_)
print(randomRFC.best_params_)
```

```
0.9665532879818595
{'n_estimators': 2600, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_feat
ures': 'sqrt', 'max_depth': 65, 'bootstrap': True}
```

Figure 11: RandomSearchCV with n_jobs=-1 and 6 hyperparameters being tuned

```
Test Accuracy: 0.986
```

Figure 12: Test Accuracy with tuned RandomForestClassifier model



	precision	recall	f1-score	support
close to total failure	1.00	1.00	1.00	72
severe lag	0.97	0.97	0.97	72
small lag	0.95	0.97	0.96	72
optimal switching behavior	0.99	0.98	0.99	225
accuracy			0.98	441
macro avg	0.98	0.98	0.98	441
weighted avg	0.98	0.98	0.98	441

Figure 13: Confusion matrix and classification report for the tuned model

Having trained and made a robust RandomForestClassifier model, next goal was to try out majority voting and see if bagging would reduce overfitting, as well as apply boosting to build very powerful models from weak learners. Even if a forest-based model shows great results, doesn't mean that is will have great general performance on future datasets, where predictions are to be made. Applying ensemble methods allows us to combine predictions to deliver a very accurate result and a strong model. The group proceeded with setting up a simple DecisionTree classifier to be used together with bootstrap aggregation, setting max_depth to "None" to try and overfit the model. Even the overfit tree classifier together with a very simple bagging classifier produced a good accuracy score, performing on par with the hyperparameter-tuned RandomForest classifier.

```
Decision tree train/test accuracies: 1.000/0.914
Bagging train/test accuracies: 1.000/0.984
```

Figure 14: Simple decision tree vs bagging accuracies

Next, the group proceeded with trying out weak learners to build an even more robust and generalized problem. An AdaBoostClassifier was fit and trainer with the simple DecisionTree classifier, previously used for bagging. The results showed a 0.916 accuracy, which was surprising for a model this simple. The final step was combining the tweaked, robust RandomForestClassifier with AdaBoost for the final model and using it to predict all the target values from “profile.txt”. The model predicted all labels of the training dataset correctly and showed an improvement of the RandomForest model. The model is assumed to have great generalization performance for any future hydraulic system-related prediction tasks. The train/test accuracies for all target values with the final model are:

```
AdaBoost train/test accuracies for valve condition: 1.000/0.998
AdaBoost train/test accuracies for cooler condition: 1.000/0.998
AdaBoost train/test accuracies for stable flag: 1.000/0.975
AdaBoost train/test accuracies for stable flag: 1.000/0.975
AdaBoost train/test accuracies for hydraulic accumulator/bar: 1.000/0.984
```

Figure 15: train/test accuracies

SYSTEM ARCHITECTURE

The flowchart shows the flowchart fow a typical machine learning model which could be applied to a PdM problem like the one the group worked with. Most of the processes could be automated, but data cleaning, feature selection and visualization of data has to be done manually to avoid costly mistakes and over-maintenance. Visualizing the data and checking the statistics is important to get an insight into the dataset and the techniques that could be used to build a robust and effective model.

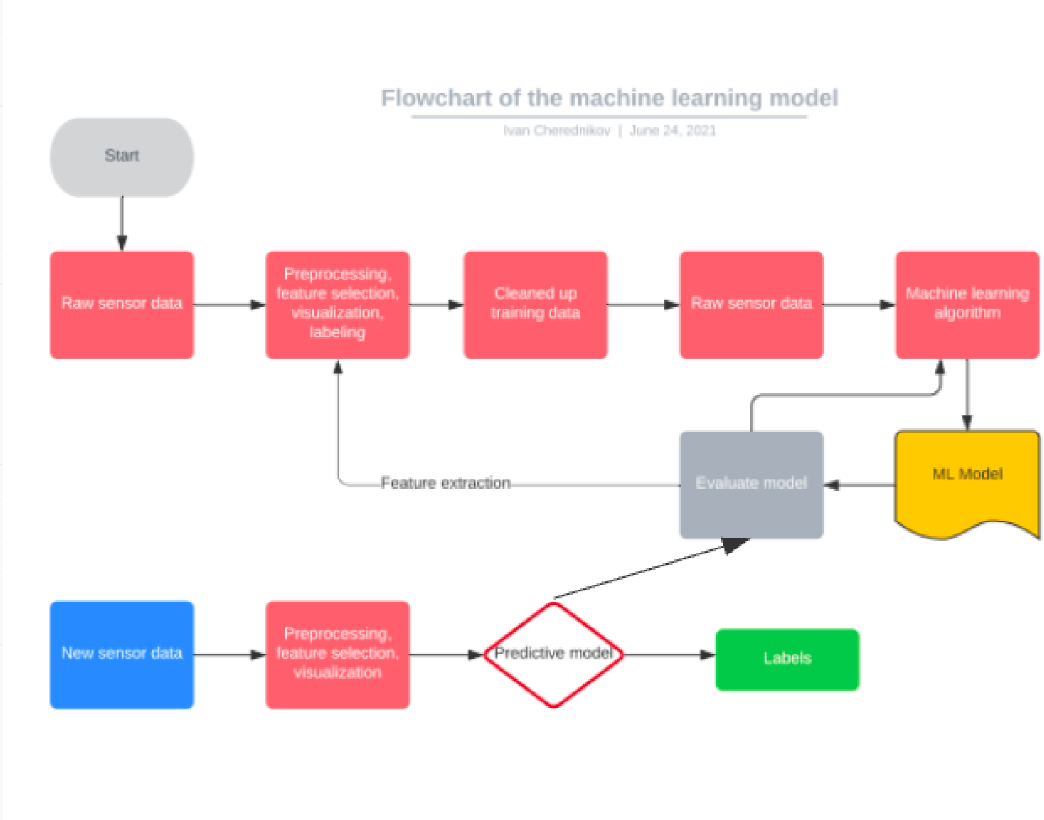


Figure 16: A system architecture flowchart for the PdM system:

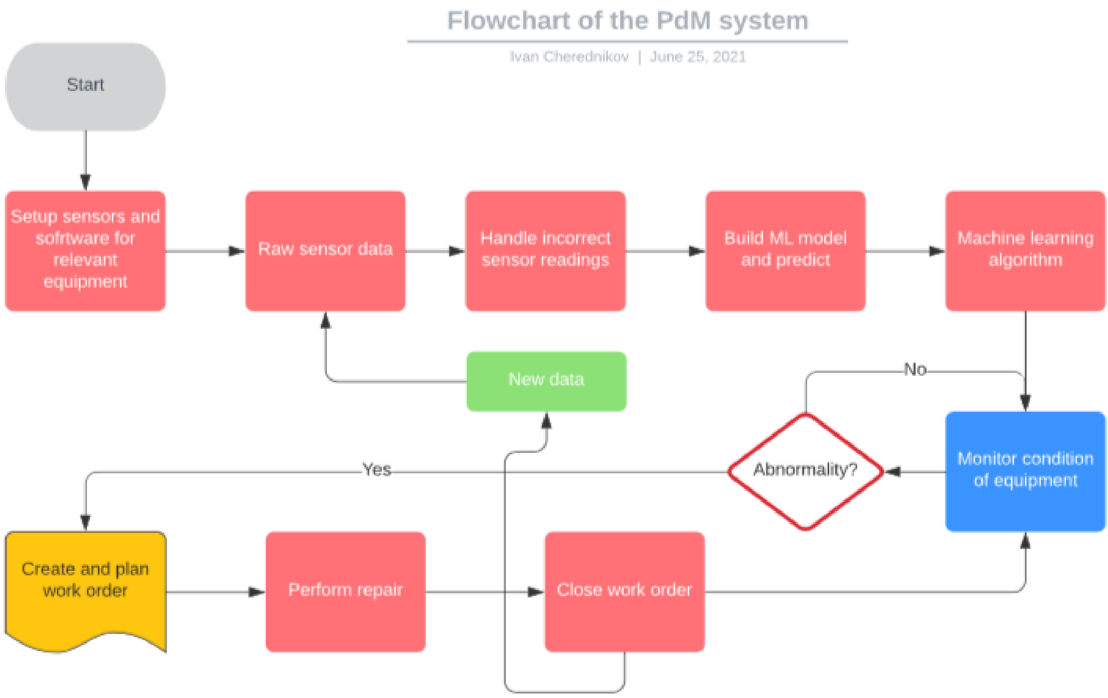


Figure 17: PdM flowchart

Here, the system is dependant on functioning data collection and working sensors that gather the data. In the case of poor data collection, a general model could be applied instead, without adding new data and retraining a model all the time. However, that might lead to poor generalization and a less dynamic workflow. A static model wouldn't be able to handle sudden faults and sensor malfunction. Although one could write a script in a few functions, using statistical methods to detect anomalies in data from a particular sensor compared to previous data. This wouldn't be hard to perform, but it would requiere a competent worked managing these operations. Of course, an automatic system would be able to do that too, but systems of that sort are beyond the group's knowledge.

EVALUATION AND RESULTS

MODEL EVALUATION

The model provided us with some unbelievable accuracy, with the best model coming out with an accuracy of around 98-99% on the test data for all the available labels. Thus, in theory giving us a very robust model. One could question the actual robustness of the model, as results like this usually tend to mean that the model is not very generalizable and would not show good results with new data, being very specialized on this data. This is easily explained by the nature of the data and the preprocessing done. As mentioned before, the data had 2205 rows and 43680 columns, which is a lot of features to work with. One could argue that time series data of this sort should not be reduced into 17 feature like the group has done, using mean values to reduce the amount of features. Instead, one could argue that extensive feature selection and partioning should be done for a more robust model. However, it is much more computationally expensive and would just give a small edge for the robustness of the model on new data. Since one could combine data-driven PdM with model-driven PdM, running hyperparameter tuners and preprocessing when new sensor data gets would be needed to be done occasionally, it is important for the process to not use a lot of time and resources.

Combining AdaBoost and a forest-based model generally makes for a robust model, as it avoids multicollinearity which is common in time series data, where tightly related data is often unavoidable, as well as being naturally relatively resistant to overfitting.

SUMMARY

POTENTIAL BUSINESS BENEFITS

Without quantitative, the group can only theorize about the commercialization potential of the said model and method. However, as mentioned before, predictive maintenance is very important for hydraulic equipment, as the drawbacks of reactive and preventive maintenance are too costly for a company relying on such equipment. Catastrophic failures, labor intensity and inefficient use of company resources add up to large sums, which could be fully avoided with a PdM program. Assuming a company uses clean oil, hydraulic equipment can last tens of thousands of hours[10], so a proper maintenance system is crucial. Furthermore, according to the Office of Energy Efficiency & Renewable Energy (energy.gov/eere/), "more than 55% of maintenance resources and activities of an average facility are still reactive."[11]. Although theoretically a company won't have any associated maintenance cost, in reality this approach is heavily outclassed by predictive maintenance, especially one done with a data-driven and model-driven hybrid ML approach. As previously mentioned, PdM gives an average returnf on investment of 10 times, as well as providing saving of 8% to 12% over a preventive maintenance program or 30% to 40% over a reactive maintenance program (factoring in the condition of the equipment, as a consequence of reactive maintenance).

Looking at Hydac's Predictive Maintenance strategy report from 2016, one could observe that the total costs of maintenance and costs assosiated with production failure were significantly reduced when using a predictive model[12]. This was true for many different fields that Hydac operates in, including marine, mobile and steel industries, wind energy and aviation. PdM in the wind energy case helped the company detect and repair a damaged bearing, saving them from unplanned downtime and costs of a new gearbox, costing "(roughly €360,000)".

In the case of this particular dataset and the information given, a remaining life depreciation calculation hard to make, since one would need thing like costs, salvage values and accumulated deprecations, as well as timestamps. However, if timestamps were acquired, a RUL (Remaining Useful Life) prediction and graph could be made. This would give the company an early warning when a machine or part needs to be repaired, which would be predictive maintenance. In the case of other datasets, like the NASA Turbofan Engine Degradation Simulation Data Set[13], a RUL could be made because of the nature of the data provided. Here, the train set gives us information about the condition of the system when it actually fails, allowing us to make a prediction on the test dataset, as well as containing vector RUL values. To make a Remaining Useful Life prediction and graph, the group would add a separate column after preprocessing and feature reduction, named "RUL", used as the target value for ML models. Noise reduction is often requiered when visualizing RUL, as sensor data tends to be rowdy. A combination of a forest-based model with an ensemble technique with weak learners as done in this report for predicting valve conditions would make a great RUL prediction and graph. A RUL graph would be very benificial for a business, especially one that is centered around hydraulic systems.

[10] https://askinglot.com/how-long-do-hydraulic-pumps-last , https://www.hydraproducts.co.uk/Hydraulic-Power-Units/Standard-Power-Units

[11] https://www1.eere.energy.gov/femp/pdfs/OM_5.pdf

[12] https://www.hydac.com/uploads/media/E_Handout_Predictive_Maintenance_6Seiten_Ausgabe1_HMI2016_LQ.pdf

[13] https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/ , #6

CONCLUSION AND RECOMMENDATIONS

Based on key findings in the investigation and analysis of predictive maintenance and a hydraulic dataset, the group recommends every business to investigate predictive maintenance and consider investing into it, if possible, because of the huge benefits with few drawbacks. This is especially important for the hydraulic industry, where uptime and optimal utilization of available resources is crucial, minimizing LCC.

FURTHER WORK

If the group could have more quantitative data and an actual, real-life case, with hydaulic data not obtained using a test rig, more work could be done. As certain points in the dataset could only be referred to as cycles, without actual timestamps, it is not possible to compare the results obtained with the model to a real-life hydraulic rig being used in a business. Without having access to costs of maintenance, equipment parts and other cost-related data, such as budgets and accounting information, deeper analysis to gain further insight is not possible.

Model improvement could also be done to generalize the model and streamline the machine learning operations. First things to improve and add to the code would be:

- Functions to quickly train the models, instead of having to go through lengthy code
- Code/Debug testing.
- Functions or even simple “If tests” to detect when and what part needs to be repaired, alerting and informing the user.
- Interactive UI/Streamlit page to quickly import data, preprocess, visualize it and gain insight. Moreover, the model could be trained and adjusted quickly, giving out formatted predictions.
- Implementation of deep neural networks and TensorFlow

There are many areas of improvement and more work to be done, but the dataset and lack of cost data are the main limitations and the points mentioned above are to be focused on first.

REFERENCES

Noria. (2018). Predictive Maintenance Explained. Published on Reliable Plant: https://www.reliableplant.com/Read/12495/preventive-predictive-maintenance

Espset, T. (2016). Predikitivt Vedlikehold. NTNU Master Thesis. https://ntnuopen.ntnu.no/ntnu-xmlui/bitstream/handle/11250/2399080/15198_FULLTEXT.pdf?sequence=1

UpKeep. (n.d.). Proactive Maintenance. https://www.onupkeep.com/learning/maintenance-types/proactive-maintenance

Fiix. (n.d.). Predictive Maintenance. https://www.fiixsoftware.com/maintenance-strategies/predictive-maintenance/

Endeavor Business Media. (2012, 01 01). Fundamentals of Hydraulic Reservoirs. Gathered from Hydraulics & Pneumatics. https://www.hydraulicspneumatics.com/technologies/reservoirs-accessories/article/21882642/fundamentals-of-hydraulic-reservoirs

Variohm. (2019, 04 15). Hydraulic Pressure Sensors. https://www.variohm.com/news-media/technical-blog-archive/hydraulic-pressure-sensors

Hydac. (n.d.). Temperature sensors. https://www.hydac.com/de-en/products/sensors/temperature-sensors.html

Fluid Power World. (2017, 10 03). What you need to know about hydraulic flow meters. https://www.fluidpowerworld.com/need-know-hydraulic-flow-meters/

ElProCus. (n.d.). What is the Piezoelectric Effect – Working and its Applications. https://www.elprocus.com/what-is-the-piezoelectric-effect-working-and-its-applications/

ElProCus. (n.d.). Vibration Sensor Working and Applications. https://www.elprocus.com/vibration-sensor-working-and-applications/

Coast. (2020, 12 02). What is Predictive Maintenance? (+Benefits, Cost, & Examples). https://coastapp.com/blog/predictive-maintenance/

US Office of Energy Efficiency & Renewable Energy. (2010, 08). Operations & Maintenance Best Practices, Chapter

5. https://www1.eere.energy.gov/femp/pdfs/OM_5.pdf

ResearchGate. (2020, 08). A Hybrid Approach: Dynamic Diagnostic Rules for Sensor Systems in Industry 4.0 Generated by Online Hyperparameter Tuned Random Forest. https://www.researchgate.net/publication/343497868_A_Hybrid_Approach_Dynamic_Diagnostic_Rules_for_Sensor_Systems_in_Industry_40_Generated_by_Online_Hyperparameter_Tuned_Random_Forest

Askinglot. (2020, 06 14). How long do hydraulic pumps last?. Unreliable source, only used for an approximation value and not for any valuable information. https://askinglot.com/how-long-do-hydraulic-pumps-last

Hydra Products. (n.d.). Standard Hydraulic Power Units. https://www.hydraproducts.co.uk/Hydraulic-Power-Units/Standard-Power-Units

Hydac International. (2016, 04). HYDAC Predictive Maintenance. https://www.hydac.com/uploads/media/E_Handout_Predictive_Maintenance_6Seiten_Ausgabe1_HMI2016_LQ.pdf

ATTACHMENTS

SOURCE CODE

https://github.com/inoplanetka/Hydraulic-System-Monitoring-ML