MACHINE LEARNING ENGINEER NANODEGREE

Predicting machine failure

Capstone Proposal

Marcus Holmgren
June 2021, Stockholm, Sweden

Domain Background

Machine maintenance is usually scheduled to occur with a regular interval where inspection of the functionality, wear and remaining service of life is assessed. In this study I will explore a dataset compiled by Stephan Matzka at the School of Engineering - Technology and Life, Hochschule für Technik und Wirtschaft Berlin.

Successfully predicting machine failure is useful for more efficient maintenance schedules and also improved operational costs of equipment in an end to end lifecycle.

Problem Statement

The problem that I'm going to explore is to see if it is possible to use machine learning algorithms to predict machine failure from sensor measurements with a binary classification algorithm. This problem should be possible explain with a linear model and I will also attempt to use a deep neural network algorithm to predict machine failures.

Datasets and Inputs

The main dataset that I intend to use for this experiment is the *AI4I 2020 Predictive Maintenance Dataset Data Set*[1] published by Stephan Matzka and hosted at the UCI Machine Learning Repository. This data set collects five independent failure modes and several features for each of the machine subjects. This a synthetic dataset that reflects real predictive maintenance data encountered in industry.

The dataset consist of 14 features of different sensors measurements ranging from air temperature to tool wear in minutes

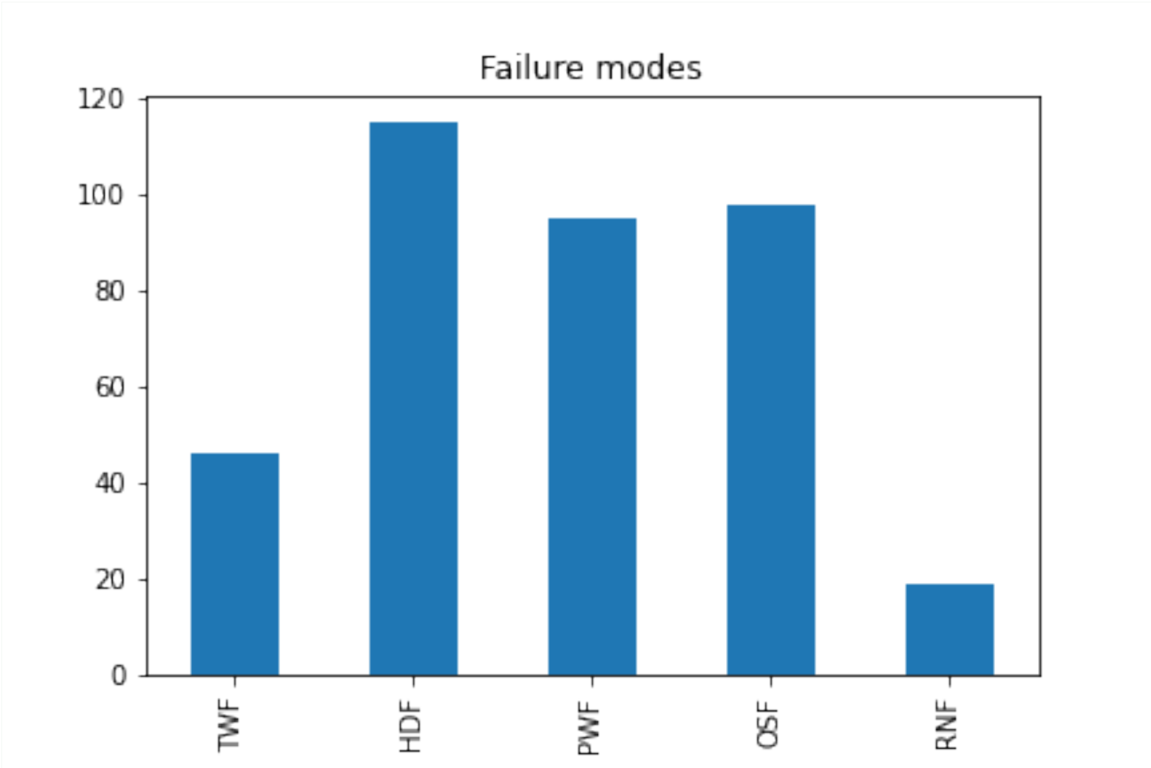
- UDI - unique identifier ranging from 1 to 10000
- Product ID - product identifier
- Type - custom label of product quality
- Air temperature [K] - temp in Kelvin
- Process temperature [K] - temp in Kelvin
- Rotational speed [rpm]
- Torque [Nm]
- Tool wear [min]
- Machine failure - boolean value if the machine have failed

The machine failure consists of five independent failure modes
- TWF - tool wear failure
- HDF - heat dissipation failure
- PWF - power failure
- OSF - overstrain fails due to overstrain
- RNF - random failures

The data set is highly imbalanced where the feature Machine failure consists of 9661 (0.9661) false values and 339 (0.0339) failures according to the five failure modes. This will be important to acknowledge and be cautious about when performing evaluation metrics. This kind of imbalance

[1] http://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset

is also common when analysing fraud detection and I will try and find methods used from that field when training my model and evaluating it's performance.



Solution Statement

The solution I attempt to explore is to see if I can build and use a custom deep neural network model to successfully predict machine failure from the possible input features published in the AI4I 2020 Predictive Maintenance Dataset.

Benchmark Model

I'm going to use a linear learner algorithm as benchmark against my custom deep neural network model. AWS SageMaker Linear Learner Algorithm[2] will be used as a predictive baseline model.

Evaluation Metrics

Since I'm going to evaluate and predict failure the machine learning model is going to be a binary classification. So I am to evaluate on a holdout test set the metrics precision and recall[3]. I will also calculate accuracy. Accuracy can be misleading if the data set is imbalanced and precision and recall can be a little too simplistic when used in isolation. I will have this in mind when evaluation my trained model.

Accuracy[4] is calculated as the weighted mean

$$Accuracy = \frac{T_p + T_n}{T_p + f_n + f_p + T_n}$$

---

[2] https://docs.aws.amazon.com/sagemaker/latest/dg/linear-learner.html

[3] https://en.wikipedia.org/wiki/Precision_and_recall

[4] https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/

Precision gives the fraction of examples of true positive values that belong to the positive predictions.

$$Precision = \frac{T_p}{T_p + f_p}$$

Recall summarise how well true positive values was predicted and is the same calculation as sensitivity.

$$Recall = \frac{T_p}{T_p + f_n}$$

Project Design

This machine learning project will leverage AWS SageMaker as the platform for storing training data and the compute resources needed to train and evaluate a machine learning algorithm. The experiment will be recorded in Jupyter[5] notebooks for exploratory data analysis, training of machine learning algoritm and its evaluation.

The first part of the exploratory data analysis will identify the characteristics and prepare the data set for training on a machine learning algorithm. I will try to use graphs to better illustrate the distribution of the data used for training.

The second part will be to implement a custom deep neural network algorithm that will be trained in AWS SageMaker. This custom algorithm will be implemented in PyTorch[6]. This algorithm will then be evaluated and benchmarked against Linear Learner that is one of the built in SageMaker algorithms.

The third and last part of this project will become a blog post that describes the journey and result from this experiment.

http://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset

[5] https://jupyter.org

[6] https://pytorch.org